# Table of Contents

Welcome  .................................................................................................................. 1
Actions ....................................................................................................................... 2

AddCustomAttributes ................................................................................................. 5
   Request Syntax ........................................................................................................ 5
   Request Parameters ............................................................................................... 5
   Response Elements ............................................................................................... 6
   Errors .................................................................................................................... 6
   See Also ................................................................................................................ 6
AdminAddUserToGroup ........................................................................................... 8
   Request Syntax ........................................................................................................ 8
   Request Parameters ............................................................................................... 8
   Response Elements ............................................................................................... 9
   Errors .................................................................................................................... 9
   See Also ................................................................................................................ 9
AdminConfirmSignUp .............................................................................................. 11
   Request Syntax ...................................................................................................... 11
   Request Parameters ............................................................................................. 11
   Response Elements ............................................................................................. 12
   Errors ................................................................................................................... 12
   See Also ............................................................................................................... 13
AdminCreateUser .................................................................................................... 15
   Request Syntax ...................................................................................................... 15
   Request Parameters ............................................................................................. 16
   Response Syntax .................................................................................................. 19
   Response Elements ............................................................................................. 19
   Errors ................................................................................................................... 19
   Examples ............................................................................................................... 21
   See Also ............................................................................................................... 22
AdminDeleteUser ..................................................................................................... 23
   Request Syntax ...................................................................................................... 23
   Request Parameters ............................................................................................. 23
   Response Elements ............................................................................................. 23
   Errors ................................................................................................................... 24
   See Also ............................................................................................................... 24
AdminDeleteUserAttributes ..................................................................................... 25
   Request Syntax ...................................................................................................... 25
   Request Parameters ............................................................................................. 25
   Response Elements ............................................................................................. 26
   Errors ................................................................................................................... 26
   See Also ............................................................................................................... 26
AdminDisableProviderForUser ................................................................................. 28
   Request Syntax ...................................................................................................... 28
   Request Parameters ............................................................................................. 28
   Response Elements ............................................................................................. 28
   Errors ................................................................................................................... 29
   See Also ............................................................................................................... 29
AdminDisableUser .................................................................................................... 31
   Request Syntax ...................................................................................................... 31
   Request Parameters ............................................................................................. 31
   Response Elements ............................................................................................. 32
   Errors ................................................................................................................... 32
   See Also ............................................................................................................... 32
AdminEnableUser ...................................................................................................... 33
   Request Syntax ...................................................................................................... 33

Amazon Cognito User Pools API Reference  

API Version 2016-04-18  

iii
<table>
<thead>
<tr>
<th>Request Syntax</th>
<th>Response Elements</th>
<th>Errors</th>
<th>See Also</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Cognito User Pools API Reference</td>
<td>33</td>
<td>33</td>
<td>34</td>
</tr>
<tr>
<td>AdminForgotDevice</td>
<td>35</td>
<td>35</td>
<td>35</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>35</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>35</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>Response Elements</td>
<td>36</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>See Also</td>
<td>36</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>AdminGetDevice</td>
<td>38</td>
<td>38</td>
<td>38</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>38</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>38</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>39</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>Response Elements</td>
<td>39</td>
<td>39</td>
<td>39</td>
</tr>
<tr>
<td>Errors</td>
<td>39</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>See Also</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>AdminInitiateAuth</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>41</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>42</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>Response Elements</td>
<td>42</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>Errors</td>
<td>42</td>
<td>43</td>
<td>43</td>
</tr>
<tr>
<td>See Also</td>
<td>43</td>
<td>44</td>
<td>44</td>
</tr>
<tr>
<td>AdminLinkProviderForUser</td>
<td>45</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>45</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>45</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>48</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>Response Elements</td>
<td>48</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>Errors</td>
<td>48</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>See Also</td>
<td>50</td>
<td>51</td>
<td>51</td>
</tr>
<tr>
<td>AdminListDevices</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>53</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>53</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>Response Elements</td>
<td>55</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>Errors</td>
<td>55</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>See Also</td>
<td>55</td>
<td>55</td>
<td>55</td>
</tr>
<tr>
<td>AdminListGroupsForUser</td>
<td>57</td>
<td>57</td>
<td>57</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>57</td>
<td>58</td>
<td>58</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>57</td>
<td>58</td>
<td>58</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>58</td>
<td>58</td>
<td>58</td>
</tr>
<tr>
<td>Response Elements</td>
<td>58</td>
<td>58</td>
<td>58</td>
</tr>
<tr>
<td>Errors</td>
<td>58</td>
<td>59</td>
<td>59</td>
</tr>
<tr>
<td>See Also</td>
<td>59</td>
<td>59</td>
<td>59</td>
</tr>
<tr>
<td>AdminListUserAuthEvents</td>
<td>60</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>60</td>
<td>61</td>
<td>61</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>60</td>
<td>61</td>
<td>61</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>61</td>
<td>61</td>
<td>61</td>
</tr>
<tr>
<td>Response Elements</td>
<td>61</td>
<td>61</td>
<td>61</td>
</tr>
<tr>
<td>Errors</td>
<td>61</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>See Also</td>
<td>62</td>
<td>62</td>
<td>62</td>
</tr>
<tr>
<td>Amazon Cognito User Pools API Reference</td>
<td>63</td>
<td>63</td>
<td>63</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>63</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>63</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>64</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Response Elements</td>
<td>64</td>
<td>64</td>
<td>64</td>
</tr>
<tr>
<td>Errors</td>
<td>64</td>
<td>65</td>
<td>65</td>
</tr>
</tbody>
</table>

API Version 2016-04-18
iv
<table>
<thead>
<tr>
<th>Request Syntax</th>
<th>Request Parameters</th>
<th>Response Elements</th>
<th>Errors</th>
<th>See Also</th>
<th>AssociateSoftwareToken</th>
</tr>
</thead>
<tbody>
<tr>
<td>102</td>
<td>102</td>
<td>103</td>
<td>103</td>
<td>103</td>
<td>107</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>ChangePassword</td>
</tr>
<tr>
<td>108</td>
<td>108</td>
<td>109</td>
<td>109</td>
<td>110</td>
<td>111</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>ConfirmDevice</td>
</tr>
<tr>
<td>111</td>
<td>111</td>
<td>112</td>
<td>112</td>
<td>113</td>
<td>115</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>ConfirmForgotPassword</td>
</tr>
<tr>
<td>117</td>
<td>117</td>
<td>119</td>
<td>119</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>ConfirmSignUp</td>
</tr>
<tr>
<td>122</td>
<td>122</td>
<td>124</td>
<td>124</td>
<td>125</td>
<td>125</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>CreateGroup</td>
</tr>
<tr>
<td>127</td>
<td>127</td>
<td>127</td>
<td>127</td>
<td>129</td>
<td>129</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>CreateIdentityProvider</td>
</tr>
<tr>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>133</td>
<td>133</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Elements</td>
<td>Errors</td>
<td>See Also</td>
<td>CreateResourceServer</td>
</tr>
<tr>
<td>134</td>
<td>134</td>
<td>134</td>
<td>134</td>
<td>135</td>
<td>135</td>
</tr>
<tr>
<td>Method</td>
<td>See Also</td>
<td>Request Syntax</td>
<td>Request Parameters</td>
<td>Response Syntax</td>
<td>Response Elements</td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------</td>
<td>----------------</td>
<td>--------------------</td>
<td>----------------</td>
<td>------------------</td>
</tr>
<tr>
<td>CreateUserImportJob</td>
<td>135</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CreateUserPool</td>
<td>139</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CreateUserPoolClient</td>
<td>157</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CreateUserPoolDomain</td>
<td>169</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeleteGroup</td>
<td>172</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeleteIdentityProvider</td>
<td>174</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeleteResourceServer</td>
<td>176</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeleteUser</td>
<td>178</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DeleteUserAttributes</td>
<td>180</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

API Version 2016-04-18
Amazon Cognito User Pools API Reference

Request Parameters ......................................................... 180
Response Elements ......................................................... 180
Errors ................................................................. 180
See Also ................................................................. 181

DeleteUserPool .............................................................. 183
Request Syntax .............................................................. 183
Request Parameters ........................................................ 183
Response Elements ........................................................ 183
Errors ................................................................. 183
See Also ................................................................. 184

DeleteUserPoolClient ......................................................... 185
Request Syntax .............................................................. 185
Request Parameters ........................................................ 185
Response Elements ........................................................ 185
Errors ................................................................. 185
See Also ................................................................. 186

DeleteUserPoolDomain ......................................................... 187
Request Syntax .............................................................. 187
Request Parameters ........................................................ 187
Response Elements ........................................................ 187
Errors ................................................................. 187
See Also ................................................................. 188

DescribeIdentityProvider ....................................................... 189
Request Syntax .............................................................. 189
Request Parameters ........................................................ 189
Response Syntax ............................................................ 189
Response Elements ........................................................ 190
Errors ................................................................. 190
See Also ................................................................. 190

DescribeResourceServer ....................................................... 192
Request Syntax .............................................................. 192
Request Parameters ........................................................ 192
Response Syntax ............................................................ 192
Response Elements ........................................................ 193
Errors ................................................................. 193
See Also ................................................................. 193

DescribeRiskConfiguration .................................................... 195
Request Syntax .............................................................. 195
Request Parameters ........................................................ 195
Response Syntax ............................................................ 195
Response Elements ........................................................ 196
Errors ................................................................. 196
See Also ................................................................. 197

DescribeUserImportJob ......................................................... 198
Request Syntax .............................................................. 198
Request Parameters ........................................................ 198
Response Syntax ............................................................ 198
Response Elements ........................................................ 199
Errors ................................................................. 199
See Also ................................................................. 199

DescribeUserPool .............................................................. 201
Request Syntax .............................................................. 201
Request Parameters ........................................................ 201
Response Syntax ............................................................ 201
Response Elements ........................................................ 203
Errors ................................................................. 203
See Also ................................................................. 204
Request Parameters ................................................................. 229
Response Syntax ........................................................................ 229
Response Elements ................................................................. 229
Errors .................................................................................. 230
See Also ................................................................................ 230
GetSigningCertificate .............................................................. 231
Request Syntax ........................................................................ 231
Request Parameters ............................................................... 231
Response Syntax ...................................................................... 231
Response Elements ............................................................... 231
Errors .................................................................................. 232
See Also ................................................................................ 232
GetUiCustomization ................................................................. 233
Request Syntax ........................................................................ 233
Request Parameters ............................................................... 233
Response Syntax ...................................................................... 233
Response Elements ............................................................... 234
Errors .................................................................................. 234
See Also ................................................................................ 234
GetUser .................................................................................. 236
Request Syntax ........................................................................ 236
Request Parameters ............................................................... 236
Response Syntax ...................................................................... 236
Response Elements ............................................................... 236
Errors .................................................................................. 237
See Also ................................................................................ 238
GetUserAttributeVerificationCode ............................................. 239
Request Syntax ........................................................................ 239
Request Parameters ............................................................... 239
Response Syntax ...................................................................... 240
Response Elements ............................................................... 240
Errors .................................................................................. 241
See Also ................................................................................ 242
GetUserPoolMfaConfig ............................................................. 244
Request Syntax ........................................................................ 244
Request Parameters ............................................................... 244
Response Syntax ...................................................................... 244
Response Elements ............................................................... 244
Errors .................................................................................. 245
See Also ................................................................................ 245
GlobalSignOut ......................................................................... 247
Request Syntax ........................................................................ 247
Request Parameters ............................................................... 247
Response Elements ............................................................... 247
Errors .................................................................................. 248
See Also ................................................................................ 248
InitiateAuth ............................................................................. 250
Request Syntax ........................................................................ 250
Request Parameters ............................................................... 250
Response Syntax ...................................................................... 253
Response Elements ............................................................... 253
Errors .................................................................................. 254
Examples ............................................................................... 256
See Also ................................................................................ 258
ListDevices .............................................................................. 259
Request Syntax ........................................................................ 259
Request Parameters ............................................................... 259
Response Syntax ................................................................. 260
Response Elements .......................................................... 260
Errors ................................................................................. 260
See Also ............................................................................... 261
ListUserPoolClients ............................................................. 267
Request Syntax ................................................................... 267
Request Parameters ............................................................ 267
Response Syntax .................................................................. 267
Response Elements ............................................................. 267
Errors ................................................................................. 267
See Also ............................................................................... 268
ListIdentityProviders .......................................................... 268
Request Syntax ................................................................... 268
Request Parameters ............................................................ 268
Response Syntax .................................................................. 268
Response Elements ............................................................. 268
Errors ................................................................................. 268
See Also ............................................................................... 269
ListResourceServers ........................................................... 269
Request Syntax ................................................................... 269
Request Parameters ............................................................ 269
Response Syntax .................................................................. 269
Response Elements ............................................................. 269
Errors ................................................................................. 269
See Also ............................................................................... 270
ListTagsForResource ............................................................ 271
Request Syntax ................................................................... 271
Request Parameters ............................................................ 271
Response Syntax .................................................................. 271
Response Elements ............................................................. 271
Errors ................................................................................. 271
See Also ............................................................................... 272
ListUserImportJobs ............................................................. 272
Request Syntax ................................................................... 272
Request Parameters ............................................................ 272
Response Syntax .................................................................. 272
Response Elements ............................................................. 272
Errors ................................................................................. 272
See Also ............................................................................... 273
ListUserPoolClients ............................................................. 273
Request Syntax ................................................................... 273
Request Parameters ............................................................ 273
Response Syntax .................................................................. 273
Response Elements ............................................................. 273
Errors ................................................................................. 273
See Also ............................................................................... 274
ListUserPools ......................................................................... 274
Request Syntax ................................................................... 274
Request Parameters ............................................................ 274
Response Syntax .................................................................. 274
Response Elements ............................................................. 275
Errors ................................................................................. 275
See Also ............................................................................... 276
ListUsers .............................................................................. 276
Request Syntax ................................................................... 276
Request Parameters ............................................................ 276
Response Syntax .................................................................. 276
Response Elements ............................................................. 276
Errors ................................................................................. 276
See Also ............................................................................... 277

API Version 2016-04-18
xi
<table>
<thead>
<tr>
<th>Method Name</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>See Also</td>
<td>320</td>
</tr>
<tr>
<td>SetUserPoolMfaConfig</td>
<td>322</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>322</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>322</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>323</td>
</tr>
<tr>
<td>Response Elements</td>
<td>323</td>
</tr>
<tr>
<td>Errors</td>
<td>324</td>
</tr>
<tr>
<td>See Also</td>
<td>324</td>
</tr>
<tr>
<td>SetUserSettings</td>
<td>326</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>326</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>326</td>
</tr>
<tr>
<td>Response Elements</td>
<td>326</td>
</tr>
<tr>
<td>Errors</td>
<td>326</td>
</tr>
<tr>
<td>See Also</td>
<td>327</td>
</tr>
<tr>
<td>SignUp</td>
<td>329</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>329</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>329</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>332</td>
</tr>
<tr>
<td>Response Elements</td>
<td>332</td>
</tr>
<tr>
<td>Errors</td>
<td>332</td>
</tr>
<tr>
<td>See Also</td>
<td>332</td>
</tr>
<tr>
<td>StartUserImportJob</td>
<td>335</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>335</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>335</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>335</td>
</tr>
<tr>
<td>Response Elements</td>
<td>336</td>
</tr>
<tr>
<td>Errors</td>
<td>336</td>
</tr>
<tr>
<td>See Also</td>
<td>336</td>
</tr>
<tr>
<td>StopUserImportJob</td>
<td>338</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>338</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>338</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>338</td>
</tr>
<tr>
<td>Response Elements</td>
<td>339</td>
</tr>
<tr>
<td>Errors</td>
<td>339</td>
</tr>
<tr>
<td>See Also</td>
<td>339</td>
</tr>
<tr>
<td>TagResource</td>
<td>341</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>341</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>341</td>
</tr>
<tr>
<td>Response Elements</td>
<td>342</td>
</tr>
<tr>
<td>Errors</td>
<td>342</td>
</tr>
<tr>
<td>See Also</td>
<td>342</td>
</tr>
<tr>
<td>UntagResource</td>
<td>343</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>343</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>343</td>
</tr>
<tr>
<td>Response Elements</td>
<td>343</td>
</tr>
<tr>
<td>Errors</td>
<td>343</td>
</tr>
<tr>
<td>See Also</td>
<td>343</td>
</tr>
<tr>
<td>UpdateAuthEventFeedback</td>
<td>345</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>345</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>345</td>
</tr>
<tr>
<td>Response Elements</td>
<td>346</td>
</tr>
<tr>
<td>Errors</td>
<td>346</td>
</tr>
<tr>
<td>See Also</td>
<td>346</td>
</tr>
<tr>
<td>UpdateDeviceStatus</td>
<td>348</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>348</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>348</td>
</tr>
<tr>
<td>Response Elements</td>
<td>349</td>
</tr>
</tbody>
</table>

API Version 2016-04-18

xiii
See Also ................................................................. 415
CustomEmailLambdaVersionConfigType .............................................. 416
Contents .................................................................. 416
See Also .................................................................. 416
CustomSMSLambdaVersionConfigType .............................................. 417
Contents .................................................................. 417
See Also .................................................................. 417
DeviceConfigurationType ...................................................... 418
Contents .................................................................. 418
See Also .................................................................. 418
DeviceSecretVerifierConfigType ............................................... 419
Contents .................................................................. 419
See Also .................................................................. 419
DeviceType ................................................................. 420
Contents .................................................................. 420
See Also .................................................................. 420
DomainDescriptionType ..................................................... 421
Contents .................................................................. 421
See Also .................................................................. 422
EmailConfigurationType ...................................................... 423
Contents .................................................................. 423
See Also .................................................................. 425
EventContextDataType ...................................................... 426
Contents .................................................................. 426
See Also .................................................................. 426
EventFeedbackType .......................................................... 428
Contents .................................................................. 428
See Also .................................................................. 428
EventRiskType ............................................................... 429
Contents .................................................................. 429
See Also .................................................................. 429
GroupType .................................................................. 430
Contents .................................................................. 430
See Also .................................................................. 431
HttpHeader ................................................................. 432
Contents .................................................................. 432
See Also .................................................................. 432
IdentityProviderType ......................................................... 433
Contents .................................................................. 433
See Also .................................................................. 435
LambdaConfigType ........................................................... 436
Contents .................................................................. 436
See Also .................................................................. 436
LogConfigurationType ......................................................... 439
Contents .................................................................. 439
See Also .................................................................. 439
LogDeliveryConfigurationType .................................................. 440
Contents .................................................................. 440
See Also .................................................................. 440
MessageTemplateType ......................................................... 441
Contents .................................................................. 441
See Also .................................................................. 441
MFAOptionType ............................................................. 442
Contents .................................................................. 442
See Also .................................................................. 442
NewDeviceMetadataType ..................................................... 443
Contents .................................................................. 443
<table>
<thead>
<tr>
<th>See Also</th>
<th>NotifyConfigurationType</th>
<th>443</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Contents</td>
<td>444</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>445</td>
</tr>
<tr>
<td>NotifyEmailType</td>
<td></td>
<td>446</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>446</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>446</td>
</tr>
<tr>
<td>NumberAttributeConstraintsType</td>
<td></td>
<td>447</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>447</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>447</td>
</tr>
<tr>
<td>PasswordPolicyType</td>
<td></td>
<td>448</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>448</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>449</td>
</tr>
<tr>
<td>ProviderDescription</td>
<td></td>
<td>450</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>450</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>450</td>
</tr>
<tr>
<td>ProviderUserIdentifierType</td>
<td></td>
<td>451</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>451</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>451</td>
</tr>
<tr>
<td>RecoveryOptionType</td>
<td></td>
<td>452</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>452</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>452</td>
</tr>
<tr>
<td>ResourceServerScopeType</td>
<td></td>
<td>453</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>453</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>453</td>
</tr>
<tr>
<td>ResourceServerType</td>
<td></td>
<td>454</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>454</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>454</td>
</tr>
<tr>
<td>RiskConfigurationType</td>
<td></td>
<td>456</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>456</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>457</td>
</tr>
<tr>
<td>RiskExceptionConfigurationType</td>
<td></td>
<td>458</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>458</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>458</td>
</tr>
<tr>
<td>SchemaAttributeType</td>
<td></td>
<td>459</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>459</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>460</td>
</tr>
<tr>
<td>SmsConfigurationType</td>
<td></td>
<td>461</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>461</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>462</td>
</tr>
<tr>
<td>SmsMfaConfigType</td>
<td></td>
<td>463</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>463</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>463</td>
</tr>
<tr>
<td>SMSMfaSettingsType</td>
<td></td>
<td>464</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>464</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>464</td>
</tr>
<tr>
<td>SoftwareTokenMfaConfigType</td>
<td></td>
<td>465</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>465</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>465</td>
</tr>
<tr>
<td>SoftwareTokenMfaSettingsType</td>
<td></td>
<td>466</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>466</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>466</td>
</tr>
<tr>
<td>StringAttributeConstraintsType</td>
<td></td>
<td>467</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>467</td>
</tr>
<tr>
<td>See Also</td>
<td></td>
<td>467</td>
</tr>
<tr>
<td>TokenValidityUnitsType</td>
<td></td>
<td>468</td>
</tr>
<tr>
<td>Contents</td>
<td></td>
<td>468</td>
</tr>
</tbody>
</table>
Welcome

With the Amazon Cognito user pools API, you can configure user pools and authenticate users. To authenticate users from third-party identity providers (IdPs) in this API, you can link IdP users to native user profiles. Learn more about the authentication and authorization of federated users at Adding user pool sign-in through a third party and in the User pool federation endpoints and hosted UI reference.

This API reference provides detailed information about API operations and object types in Amazon Cognito.

Along with resource management operations, the Amazon Cognito user pools API includes classes of operations and authorization models for client-side and server-side authentication of users. You can interact with operations in the Amazon Cognito user pools API as any of the following subjects:

1. An administrator who wants to configure user pools, app clients, users, groups, or other user pool functions.
2. A server-side app, like a web application, that wants to use its AWS privileges to manage, authenticate, or authorize a user.
3. A client-side app, like a mobile app, that wants to make unauthenticated requests to manage, authenticate, or authorize a user.

For more information, see Using the Amazon Cognito user pools API and user pool endpoints in the Amazon Cognito Developer Guide.

With your AWS SDK, you can build the logic to support operational flows in every use case for this API. You can also make direct REST API requests to Amazon Cognito user pools service endpoints. The following links can get you started with the CognitoIdentityProvider client in other supported AWS SDKs.

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3

To get started with an AWS SDK, see Tools to Build on AWS. For example actions and scenarios, see Code examples for Amazon Cognito Identity Provider using AWS SDKs.

This document was last published on November 21, 2023.
The following actions are supported:

- AddCustomAttributes (p. 5)
- AdminAddUserToGroup (p. 8)
- AdminConfirmSignUp (p. 11)
- AdminCreateUser (p. 15)
- AdminDeleteUser (p. 23)
- AdminDeleteUserAttributes (p. 25)
- AdminDisableProviderForUser (p. 28)
- AdminDisableUser (p. 31)
- AdminEnableUser (p. 33)
- AdminForgetDevice (p. 35)
- AdminGetDevice (p. 38)
- AdminGetUser (p. 41)
- AdminInitiateAuth (p. 45)
- AdminLinkProviderForUser (p. 53)
- AdminListDevices (p. 57)
- AdminListGroupsForUser (p. 60)
- AdminListUserAuthEvents (p. 63)
- AdminRemoveUserFromGroup (p. 67)
- AdminResetUserPassword (p. 70)
- AdminRespondToAuthChallenge (p. 74)
- AdminSetUserMFAPreference (p. 82)
- AdminSetUserPassword (p. 85)
- AdminSetUserSettings (p. 88)
- AdminUpdateAuthEventFeedback (p. 91)
- AdminUpdateDeviceStatus (p. 94)
- AdminUpdateUserAttributes (p. 97)
- AdminUserGlobalSignOut (p. 102)
- AssociateSoftwareToken (p. 105)
- ChangePassword (p. 108)
- ConfirmDevice (p. 111)
- ConfirmForgotPassword (p. 115)
- ConfirmSignUp (p. 120)
- CreateGroup (p. 125)
- CreateIdentityProvider (p. 129)
- CreateResourceServer (p. 133)
- CreateUserImportJob (p. 136)
- CreateUserPool (p. 139)
- CreateUserPoolClient (p. 157)
- CreateUserPoolDomain (p. 169)
- DeleteGroup (p. 172)
• DeleteIdentityProvider (p. 174)
• DeleteResourceServer (p. 176)
• DeleteUser (p. 178)
• DeleteUserAttributes (p. 180)
• DeleteUserPool (p. 183)
• DeleteUserPoolClient (p. 185)
• DeleteUserPoolDomain (p. 187)
• DescribeIdentityProvider (p. 189)
• DescribeResourceServer (p. 192)
• DescribeRiskConfiguration (p. 195)
• DescribeUserImportJob (p. 198)
• DescribeUserPool (p. 201)
• DescribeUserPoolClient (p. 205)
• DescribeUserPoolDomain (p. 208)
• ForgetDevice (p. 210)
• ForgotPassword (p. 213)
• GetCSVHeader (p. 218)
• GetDevice (p. 220)
• GetGroup (p. 223)
• GetIdentityProviderByIdentifier (p. 226)
• GetLogDeliveryConfiguration (p. 229)
• GetSigningCertificate (p. 231)
• GetUICustomization (p. 233)
• GetUser (p. 236)
• GetUserAttributeVerificationCode (p. 239)
• GetUserPoolMfaConfig (p. 244)
• GlobalSignOut (p. 247)
• InitiateAuth (p. 250)
• ListDevices (p. 259)
• ListGroups (p. 263)
• ListIdentityProviders (p. 266)
• ListResourceServers (p. 269)
• ListTagsForResource (p. 272)
• ListUserImportJobs (p. 274)
• ListUserPoolClients (p. 277)
• ListUserPools (p. 280)
• ListUsers (p. 283)
• ListUsersInGroup (p. 289)
• ResendConfirmationCode (p. 292)
• RespondToAuthChallenge (p. 297)
• RevokeToken (p. 305)
• SetLogDeliveryConfiguration (p. 308)
• SetRiskConfiguration (p. 311)
• SetUICustomization (p. 316)
• SetUserMFAPreference (p. 319)
• SetUserPoolMfaConfig (p. 322)
• SetUserSettings (p. 326)
• SignUp (p. 329)
• StartUserImportJob (p. 335)
• StopUserImportJob (p. 338)
• TagResource (p. 341)
• UntagResource (p. 343)
• UpdateAuthEventFeedback (p. 345)
• UpdateDeviceStatus (p. 348)
• UpdateGroup (p. 351)
• UpdateIdentityProvider (p. 354)
• UpdateResourceServer (p. 358)
• UpdateUserAttributes (p. 361)
• UpdateUserPool (p. 366)
• UpdateUserPoolClient (p. 373)
• UpdateUserPoolDomain (p. 383)
• VerifySoftwareToken (p. 386)
• VerifyUserAttribute (p. 390)
AddCustomAttributes

Adds additional user attributes to the user pool schema.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
    "CustomAttributes": [
        {
            "AttributeDataType": "string",
            "DeveloperOnlyAttribute": boolean,
            "Mutable": boolean,
            "Name": "string",
            "NumberAttributeConstraints": {
                "MaxValue": "string",
                "MinValue": "string"
            },
            "Required": boolean,
            "StringAttributeConstraints": {
                "MaxLength": "string",
                "MinLength": "string"
            }
        }
    ],
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**CustomAttributes (p. 5)**

An array of custom attributes, such as Mutable and Name.

Type: Array of [SchemaAttributeType (p. 459)] objects

Array Members: Minimum number of 1 item. Maximum number of 25 items.

Required: Yes

**UserPoolId (p. 5)**

The user pool ID for the user pool where you want to add custom attributes.

Type: String
Pattern: [\w-]+_[0-9a-zA-Z]+  
Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors](#).  

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.  
HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.  
HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.  
HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.  
HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.  
HTTP Status Code: 400

**UserImportInProgressException**

This exception is thrown when you're trying to modify a user pool while a user import job is in progress for that pool.  
HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminAddUserToGroup

Adds a user to a group. A user who is in a group can present a preferred-role claim to an identity pool, and populates a cognito:groups claim to their access and identity tokens.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

### Request Syntax

```json
{
    "GroupName": "string",
    "Username": "string",
    "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**GroupName (p. 8)**

The name of the group that you want to add your user to.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]*

Required: Yes

**Username (p. 8)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]*

Required: Yes

**UserPoolId (p. 8)**

The user pool ID for the user pool.
Type: String
Pattern: [\\w-]+_[0-9a-zA-Z]+
Required: Yes

Response Elements
If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors
For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

UserNotFoundException
This exception is thrown when a user isn't found.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminConfirmSignUp

This IAM-authenticated API operation provides a code that Amazon Cognito sent to your user when they signed up in your user pool. After your user enters their code, they confirm ownership of the email address or phone number that they provided, and their user account becomes active. Depending on your user pool configuration, your users will receive their confirmation code in an email or SMS message.

Local users who signed up in your user pool are the only type of user who can confirm sign-up with a code. Users who federate through an external identity provider (IdP) have already been confirmed by their IdP. Administrator-created users confirm their accounts when they respond to their invitation email message and choose a password.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```json
{
    "ClientMetadata": {
        "string": "string"
    },
    "Username": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientMetadata (p. 11)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

If your user pool configuration includes triggers, the AdminConfirmSignUp API action invokes the AWS Lambda function that is specified for the post confirmation trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. In this payload, the clientMetadata attribute provides the data that you assigned to the ClientMetadata parameter in your AdminConfirmSignUp request. In your function code in Lambda, you can process the ClientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

Note
When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
• Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
• Validate the ClientMetadata value.
• Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map
Key Length Constraints: Minimum length of 0. Maximum length of 131072.
Value Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

**Username (p. 11)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn’t an alias attribute in your user pool, you can also use their sub in this request.

Type: String
Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]*
Required: Yes

**UserPoolId (p. 11)**

The user pool ID for which you want to confirm user registration.

Type: String
Pattern: [\w-]+_[0-9a-zA-Z]+
Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyFailedAttemptsException

This exception is thrown when the user has made too many failed attempts for a given action, such as sign-in.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
See Also

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminCreateUser

Creates a new user in the specified user pool.

If MessageAction isn't set, the default is to send a welcome message via email or phone (SMS).

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](https://docs.aws.amazon.com/cognito/latest/developerguide/sms-settings.html) in the *Amazon Cognito Developer Guide*.

This message is based on a template that you configured in your call to create or update a user pool. This template includes your custom sign-up instructions and placeholders for user name and temporary password.

Alternatively, you can call AdminCreateUser with SUPPRESS for the MessageAction parameter, and Amazon Cognito won't send any email.

In either case, the user will be in the FORCE_CHANGE_PASSWORD state until they sign in and change their password.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](https://docs.aws.amazon.com/cognito/latest/developerguide/signing-api-requests.html)
- [Using the Amazon Cognito user pools API and user pool endpoints](https://docs.aws.amazon.com/cognito/latest/developerguide/using-apis.html)

**Request Syntax**

```json
{
    "ClientMetadata": {
        "string": "string"
    },
    "DesiredDeliveryMediums": [ "string" ],
    "ForceAliasCreation": boolean,
    "MessageAction": "string",
    "TemporaryPassword": "string",
    "UserAttributes": [
        {
            "Name": "string",
            "Value": "string"
        }
    ],
    "Username": "string",
    "UserPoolId": "string"
}
```

*API Version 2016-04-18*
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientMetadata (p. 15)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminCreateUser API action, Amazon Cognito invokes the function that is assigned to the pre sign-up trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminCreateUser request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**DesiredDeliveryMediums (p. 15)**

Specify "EMAIL" if email will be used to send the welcome message. Specify "SMS" if the phone number will be used. The default value is "SMS". You can specify more than one value.

Type: Array of strings

Valid Values: SMS | EMAIL

Required: No
**ForceAliasCreation (p. 15)**

This parameter is used only if the phone_number_verified or email_verified attribute is set to True. Otherwise, it is ignored.

If this parameter is set to True and the phone number or email address specified in the UserAttributes parameter already exists as an alias with a different user, the API call will migrate the alias from the previous user to the newly created user. The previous user will no longer be able to log in using that alias.

If this parameter is set to False, the API throws an AliasExistsException error if the alias already exists. The default value is False.

Type: Boolean

Required: No

**MessageAction (p. 15)**

Set to RESEND to resend the invitation message to a user that already exists and reset the expiration limit on the user's account. Set to SUPPRESS to suppress sending the message. You can specify only one value.

Type: String

Valid Values: RESEND | SUPPRESS

Required: No

**TemporaryPassword (p. 15)**

The user's temporary password. This password must conform to the password policy that you specified when you created the user pool.

The temporary password is valid only once. To complete the Admin Create User flow, the user must enter the temporary password in the sign-in page, along with a new password to be used in all future sign-ins.

This parameter isn't required. If you don't specify a value, Amazon Cognito generates one for you.

The temporary password can only be used until the user account expiration limit that you set for your user pool. To reset the account after that time limit, you must call AdminCreateUser again and specify RESEND for the MessageAction parameter.

Type: String

Length Constraints: Maximum length of 256.

Pattern: [\S]*

Required: No

**UserAttributes (p. 15)**

An array of name-value pairs that contain user attributes and attribute values to be set for the user to be created. You can create a user without specifying any attributes other than Username. However, any attributes that you specify as required (when creating a user pool or in the Attributes tab of the console) either you should supply (in your call to AdminCreateUser) or the user should supply (when they sign up in response to your welcome message).

For custom attributes, you must prepend the custom: prefix to the attribute name.
To send a message inviting the user to sign up, you must specify the user's email address or phone number. You can do this in your call to AdminCreateUser or in the Users tab of the Amazon Cognito console for managing your user pools.

In your call to AdminCreateUser, you can set the email_verified attribute to True, and you can set the phone_number_verified attribute to True. You can also do this by calling AdminUpdateUserAttributes.

- **email**: The email address of the user to whom the message that contains the code and username will be sent. Required if the email_verified attribute is set to True, or if "EMAIL" is specified in the DesiredDeliveryMediums parameter.
- **phone_number**: The phone number of the user to whom the message that contains the code and username will be sent. Required if the phone_number_verified attribute is set to True, or if "SMS" is specified in the DesiredDeliveryMediums parameter.

Type: Array of AttributeType (p. 403) objects

Required: No

**Username (p. 15)**

The value that you want to set as the username sign-in attribute. The following conditions apply to the username parameter.

- The username can't be a duplicate of another username in the same user pool.
- You can't change the value of a username after you create it.
- You can only provide a value if usernames are a valid sign-in attribute for your user pool. If your user pool only supports phone numbers or email addresses as sign-in attributes, Amazon Cognito automatically generates a username value. For more information, see Customizing sign-in attributes.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**UserPoolId (p. 15)**

The user pool ID for the user pool where the user will be created.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

**ValidationData (p. 15)**

Temporary user attributes that contribute to the outcomes of your pre sign-up Lambda trigger. This set of key-value pairs are for custom validation of information that you collect from your users but don't need to retain.

Your Lambda function can analyze this additional data and act on it. Your function might perform external API operations like logging user attributes and validation data to Amazon CloudWatch Logs. Validation data might also affect the response that your function returns to Amazon Cognito, like automatically confirming the user if they sign up from within your network.

For more information about the pre sign-up Lambda trigger, see Pre sign-up Lambda trigger.
Type: Array of \texttt{AttributeType (p. 403)} objects

Required: No

### Response Syntax

\[
\{ \\
\quad \text{"User": \{} \\
\quad \quad \text{"Attributes": [} \\
\quad \quad \quad \{ \\
\quad \quad \quad \quad \text{"Name": "string",} \\
\quad \quad \quad \quad \text{"Value": "string"} \\
\quad \quad \}\}, \\
\quad \text{"Enabled": boolean,} \\
\quad \text{"MFAOptions": [} \\
\quad \quad \{ \\
\quad \quad \quad \text{"AttributeName": "string",} \\
\quad \quad \quad \text{"DeliveryMedium": "string"} \\
\quad \quad \}\} \\
\quad \text{"UserCreateDate": number,} \\
\quad \text{"UserLastModifiedDate": number,} \\
\quad \text{"Username": "string",} \\
\quad \text{"UserStatus": "string"} \\
\quad \}\}
\]

### Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**User (p. 19)**

The newly created user.

Type: \texttt{UserType (p. 497)} object

### Errors

For information about the errors that are common to all actions, see \texttt{Common Errors (p. 503)}.

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

InvalidPasswordException
This exception is thrown when Amazon Cognito encounters an invalid password.

InvalidSmsRoleAccessPolicyException
This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

InvalidSmsRoleTrustRelationshipException
This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

NotAuthorizedException
This exception is thrown when a user isn't authorized.

PreconditionNotMetException
This exception is thrown when a precondition is not met.

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

UnexpectedLambdaException
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

UnsupportedUserStateException
The request failed because the user is in an unsupported state.

UserLambdaValidationException
This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.
HTTP Status Code: 400

**UsernameExistsException**

This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

Examples

**Example**

An AdminCreateUser request for a test user named John.

**Sample Request**

```plaintext
POST HTTP/1.1
Host: cognito-idp.us-east-1.amazonaws.com
X-Amz-Date: 20230613T200059Z
Accept-Encoding: identity
X-Amz-Target: AWSCognitoIdentityProviderService.AdminCreateUser
User-Agent: <UserAgentString>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>, Signature=<Signature>
Content-Length: <PayloadSizeBytes>

{
    "UserPoolId": "us-east-1_EXAMPLE",
    "Username": "testuser",
    "DesiredDeliveryMediums": ["SMS"],
    "MessageAction": "SUPPRESS",
    "TemporaryPassword": "This-is-my-test-99!",
    "UserAttributes": [
        {
            "Name": "name",
            "Value": "John"
        },
        {
            "Name": "phone_number",
            "Value": "+12065551212"
        },
        {
            "Name": "email",
            "Value": "testuser@example.com"
        }
    ]
}
```

**Sample Response**

```
HTTP/1.1 200 OK
Date: Tue, 13 Jun 2023 20:00:59 GMT
```

API Version 2016-04-18
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminDeleteUser

Deletes a user as an administrator. Works on any user.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
    "Username": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

Username (p. 23)

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String
Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+
Required: Yes

UserPoolId (p. 23)

The user pool ID for the user pool where you want to delete the user.

Type: String
Pattern: [\w-]+_[0-9a-zA-Z]+
Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalServerErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminDeleteUserAttributes

Deletes the user attributes in a user pool as an administrator. Works on any user.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```json
{
    "UserAttributeNames": [ "string" ],
    "Username": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserAttributeNames (p. 25)**

An array of strings representing the user attribute names you want to delete.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

Type: Array of strings


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]`+

Required: Yes

**Username (p. 25)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

Type: String


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]`+

Required: Yes
**UserPoolId (p. 25)**

The user pool ID for the user pool where you want to delete user attributes.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)](https://docs.aws.amazon.com/cognito-user-pools/latest/APIReference/AmznCognitoIdentityUserPool.html).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](https://docs.aws.amazon.com/cli/latest/reference/cognito-idp/delete-user-attributes.html)
See Also

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminDisableProviderForUser

Prevents the user from signing in with the specified external (SAML or social) identity provider (IdP). If the user that you want to deactivate is an Amazon Cognito user pools native username + password user, they can't use their password to sign in. If the user to deactivate is a linked external IdP user, any link between that user and an existing user is removed. When the external user signs in again, and the user is no longer attached to the previously linked DestinationUser, the user must create a new user account. See AdminLinkProviderForUser.

The ProviderName must match the value specified when creating an IdP for the pool.

To deactivate a native username + password user, the ProviderName value must be Cognito and the ProviderAttributeName must be Cognito_Subject. The ProviderAttributeValue must be the name that is used in the user pool for the user.

The ProviderAttributeName must always be Cognito_Subject for social IdPs. The ProviderAttributeValue must always be the exact subject that was used when the user was originally linked as a source user.

For de-linking a SAML identity, there are two scenarios. If the linked identity has not yet been used to sign in, the ProviderAttributeName and ProviderAttributeValue must be the same values that were used for the SourceUser when the identities were originally linked using AdminLinkProviderForUser call. (If the linking was done with ProviderAttributeName set to Cognito_Subject, the same applies here). However, if the user has already signed in, the ProviderAttributeName must be Cognito_Subject and ProviderAttributeValue must be the subject of the SAML assertion.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
  "User": {
    "ProviderAttributeName": "string",
    "ProviderAttributeValue": "string",
    "ProviderName": "string"
  },
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**AliasExistsException**

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
AdminDisableUser

Deactivates a user and revokes all access tokens for the user. A deactivated user can’t sign in, but still appears in the responses to GetUser and ListUsers API requests.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```
{
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](# p. 501).

The request accepts the following data in JSON format.

**Username (p. 31)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]`+

Required: Yes

**UserPoolId (p. 31)**

The user pool ID for the user pool where you want to disable the user.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`+

Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminEnableUser

Enables the specified user as an administrator. Works on any user.

**Note**

Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**

- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```json
{
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**Username (p. 33)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

- **Type:** String
- **Length Constraints:** Minimum length of 1. Maximum length of 128.
- **Pattern:** `\[\p{L}\p{M}\p{S}\p{N}\p{P}\]+`
- **Required:** Yes

**UserPoolId (p. 33)**

The user pool ID for the user pool where you want to enable the user.

- **Type:** String
- **Length Constraints:** Minimum length of 1. Maximum length of 55.
- **Pattern:** `\[\w-]+\[0-9a-zA-Z]`
- **Required:** Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminForgetDevice

Forgets the device, as an administrator.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
    "DeviceKey": "string",
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**DeviceKey (p. 35)**

The device key.

Type: String


Pattern: \[\w-.]+_[0-9a-f-]+

Required: Yes

**Username (p. 35)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\]+

Required: Yes

**UserPoolId (p. 35)**

The user pool ID.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

API Version 2016-04-18
36
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminGetDevice

Gets the device, as an administrator.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "DeviceKey": "string",
  "Username": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**DeviceKey (p. 38)**

The device key.

Type: String


Pattern: [\w-]+_[0-9a-f-]+

Required: Yes

**Username (p. 38)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**UserPoolId (p. 38)**

The user pool ID.
Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+\]

Required: Yes

Response Syntax

```json
{
  "Device": {
    "DeviceAttributes": [
      {
        "Name": "string",
        "Value": "string"
      }
    ],
    "DeviceCreateDate": number,
    "DeviceKey": "string",
    "DeviceLastAuthenticatedDate": number,
    "DeviceLastModifiedDate": number
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Device (p. 39)**

The device.

Type: DeviceType (p. 420) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidUserPoolConfigurationException**

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

API Version 2016-04-18

39
**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
AdminGetUser

Gets the specified user by user name in a user pool as an administrator. Works on any user.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```
{
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Username (p. 41)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

Type: String


Pattern: `/p{L}\p{M}\p{S}\p{N}\p{P}+/`

Required: Yes

**UserPoolId (p. 41)**

The user pool ID for the user pool where you want to get information about the user.

Type: String


Pattern: `/\w-\+[0-9a-zA-Z]+`

Required: Yes

**Response Syntax**

```
{
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Enabled (p. 41)**

Indicates that the status is enabled.

Type: Boolean

**MFAOptions (p. 41)**

This response parameter is no longer supported. It provides information only about SMS MFA configurations. It doesn't provide information about time-based one-time password (TOTP) software token MFA configurations. To look up information about either type of MFA configuration, use UserMFASettingList instead.

Type: Array of **MFAOptionType (p. 442)** objects

**PreferredMfaSetting (p. 41)**

The user's preferred MFA setting.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

**UserAttributes (p. 41)**

An array of name-value pairs representing user attributes.

Type: Array of **AttributeType (p. 403)** objects

**UserCreateDate (p. 41)**

The date the user was created.

Type: Timestamp

**UserLastModifiedDate (p. 41)**

The date and time, in ISO 8601 format, when the item was modified.
Errors

For information about the errors that are common to all actions, see [Common Errors](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminInitiateAuth

Initiates the authentication flow, as an administrator.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](https://docs.aws.amazon.com/cognito/latest/developerguide/sms-configuration.html) in the Amazon Cognito Developer Guide.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](https://docs.aws.amazon.com/AmazonCognito/latest/developerguide/signing-api-requests.html)
- [Using the Amazon Cognito user pools API and user pool endpoints](https://docs.aws.amazon.com/cognito/latest/developerguide/user-pool-api-ref.html)

### Request Syntax

```
{
  "AnalyticsMetadata": {
    "AnalyticsEndpointId": "string"
  },
  "AuthFlow": "string",
  "AuthParameters": {
    "string": "string"
  },
  "ClientId": "string",
  "ClientMetadata": {
    "string": "string"
  },
  "ContextData": {
    "EncodedData": "string",
    "HttpHeaders": [
      {
        "headerName": "string",
        "headerValue": "string"
      }
    ],
    "IpAddress": "string",
    "ServerName": "string",
    "ServerPath": "string"
  },
  "UserPoolId": "string"
}
```
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 45)**

The analytics metadata for collecting Amazon Pinpoint metrics for AdminInitiateAuth calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No

**AuthFlow (p. 45)**

The authentication flow for this call to run. The API action will depend on this value. For example:

- REFRESH_TOKEN_AUTH will take in a valid refresh token and return new tokens.
- USER_SRP_AUTH will take in USERNAME and SRP_A and return the Secure Remote Password (SRP) protocol variables to be used for next challenge execution.
- ADMIN_USER_PASSWORD_AUTH will take in USERNAME and PASSWORD and return the next challenge or tokens.

Valid values include:

- REFRESH_TOKEN_AUTH/REFRESH_TOKEN: Authentication flow for refreshing the access token and ID token by supplying a valid refresh token.
- CUSTOM_AUTH: Custom authentication flow.
- ADMIN_NO_SRP_AUTH: Non-SRP authentication flow; you can pass in the USERNAME and PASSWORD directly if the flow is enabled for calling the app client.
- ADMIN_USER_PASSWORD_AUTH: Admin-based user password authentication. This replaces the ADMIN_NO_SRP_AUTH authentication flow. In this flow, Amazon Cognito receives the password in the request instead of using the SRP process to verify passwords.

Type: String

Valid Values: USER_SRP_AUTH | REFRESH_TOKEN_AUTH | REFRESH_TOKEN | CUSTOM_AUTH | ADMIN_NO_SRP_AUTH | USER_PASSWORD_AUTH | ADMIN_USER_PASSWORD_AUTH

Required: Yes

**AuthParameters (p. 45)**

The authentication parameters. These are inputs corresponding to the AuthFlow that you're invoking. The required values depend on the value of AuthFlow:

- For USER_SRP_AUTH: USERNAME (required), SRP_A (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For ADMIN_USER_PASSWORD_AUTH: USERNAME (required), PASSWORD (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For REFRESH_TOKEN_AUTH/REFRESH_TOKEN: REFRESH_TOKEN (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For CUSTOM_AUTH: USERNAME (required), SECRET_HASH (if app client is configured with client secret), DEVICE_KEY. To start the authentication flow with password verification, include ChallengeName: SRP_A and SRP_A: (The SRP_A Value).
For more information about SECRET_HASH, see Computing secret hash values. For information about DEVICE_KEY, see Working with user devices in your user pool.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**Clientid (p. 45)**

The app client ID.

Type: String


Pattern: \[\w+]+

Required: Yes

**ClientMetadata (p. 45)**

A map of custom key-value pairs that you can provide as input for certain custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminInitiateAuth API action, Amazon Cognito invokes the Lambda functions that are specified for various triggers. The ClientMetadata value is passed as input to the functions for only the following triggers:

- Pre signup
- Pre authentication
- User migration

When Amazon Cognito invokes the functions for these triggers, it passes a JSON payload, which the function receives as input. This payload contains a validationData attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminInitiateAuth request. In your function code in AWS Lambda, you can process the validationData value to enhance your workflow for your specific needs.

When you use the AdminInitiateAuth API action, Amazon Cognito also invokes the functions for the following triggers, but it doesn't provide the ClientMetadata value as input:

- Post authentication
- Custom message
- Pre token generation
- Create auth challenge
- Define auth challenge

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ContextData (p. 45)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: [ContextDataType (p. 413)] object

Required: No

**UserPoolId (p. 45)**

The ID of the Amazon Cognito user pool.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

**Response Syntax**

```json
{
    "AuthenticationResult": {
        "AccessToken": "string",
        "ExpiresIn": number,
        "IdToken": "string",
        "NewDeviceMetadata": {
            "DeviceGroupKey": "string",
            "DeviceKey": "string"
        },
        "RefreshToken": "string",
        "TokenType": "string"
    },
    "ChallengeName": "string",
    "ChallengeParameters": {
        "string": "string"
    },
    "Session": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**AuthenticationResult (p. 48)**

The result of the authentication response. This is only returned if the caller doesn't need to pass another challenge. If the caller does need to pass another challenge before it gets tokens, ChallengeName, ChallengeParameters, and Session are returned.

Type: `AuthenticationResultType (p. 404)` object

**ChallengeName (p. 48)**

The name of the challenge that you're responding to with this call. This is returned in the AdminInitiateAuth response if you must pass another challenge.

- **MFA_SETUP**: If MFA is required, users who don't have at least one of the MFA methods set up are presented with an MFA_SETUP challenge. The user must set up at least one MFA type to continue to authenticate.
- **SELECT_MFA_TYPE**: Selects the MFA type. Valid MFA options are SMS_MFA for text SMS MFA, and SOFTWARE_TOKEN_MFA for time-based one-time password (TOTP) software token MFA.
- **SMS_MFA**: Next challenge is to supply an SMS_MFA_CODE, delivered via SMS.
- **PASSWORD_VERIFIER**: Next challenge is to supply PASSWORD_CLAIM_SIGNATURE, PASSWORD_CLAIM_SECRET_BLOCK, and TIMESTAMP after the client-side SRP calculations.
- **CUSTOM_CHALLENGE**: This is returned if your custom authentication flow determines that the user should pass another challenge before tokens are issued.
- **DEVICE_SRP_AUTH**: If device tracking was activated in your user pool and the previous challenges were passed, this challenge is returned so that Amazon Cognito can start tracking this device.
- **DEVICE_PASSWORD_VERIFIER**: Similar to PASSWORD_VERIFIER, but for devices only.
- **ADMIN_NO_SRP_AUTH**: This is returned if you must authenticate with USERNAME and PASSWORD directly. An app client must be enabled to use this flow.
- **NEW_PASSWORD_REQUIRED**: For users who are required to change their passwords after successful first login. Respond to this challenge with NEW_PASSWORD and any required attributes that Amazon Cognito returned in the requiredAttributes parameter. You can also set values for attributes that aren't required by your user pool and that your app client can write. For more information, see AdminRespondToAuthChallenge.

**Note**

In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In AdminRespondToAuthChallenge, set a value for any keys that Amazon Cognito returned in the requiredAttributes parameter, then use the AdminUpdateUserAttributes API operation to modify the value of any additional attributes.

- **MFA_SETUP**: For users who are required to set up an MFA factor before they can sign in. The MFA types activated for the user pool will be listed in the challenge parameters MFAS_CAN_SETUP value.

To set up software token MFA, use the session returned here from InitiateAuth as an input to AssociateSoftwareToken, and use the session returned by VerifySoftwareToken as an input to RespondToAuthChallenge with challenge name MFA_SETUP to complete sign-in. To set up SMS MFA, users will need help from an administrator to add a phone number to their account and then call InitiateAuth again to restart sign-in.

Type: String

Valid Values: SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED
**ChallengeParameters (p. 48)**

The challenge parameters. These are returned to you in the AdminInitiateAuth response if you must pass another challenge. The responses in this parameter should be used to compute inputs to the next call (AdminRespondToAuthChallenge).

All challenges require USERNAME and SECRET_HASH (if applicable).

The value of the USER_ID_FOR_SRP attribute is the user's actual username, not an alias (such as email address or phone number), even if you specified an alias in your call to AdminInitiateAuth. This happens because, in the AdminRespondToAuthChallenge API ChallengeResponses, the USERNAME attribute can't be an alias.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

**Session (p. 48)**

The session that should be passed both ways in challenge-response calls to the service. If AdminInitiateAuth or AdminRespondToAuthChallenge API call determines that the caller must pass another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next AdminRespondToAuthChallenge API call.

Type: String


**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.
HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

MFAMethodNotFoundException

This exception is thrown when Amazon Cognito can’t find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn’t authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can’t find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn’t confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn’t found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
AdminLinkProviderForUser

Links an existing user account in a user pool (DestinationUser) to an identity from an external IdP (SourceUser) based on a specified attribute name and value from the external IdP. This allows you to create a link from the existing user account to an external federated user identity that has not yet been used to sign in. You can then use the federated user identity to sign in as the existing user account.

For example, if there is an existing user with a username and password, this API links that user to a federated user identity. When the user signs in with a federated user identity, they sign in as the existing user account.

**Note**
The maximum number of federated identities linked to a user is five.

**Important**
Because this API allows a user with an external federated identity to sign in as an existing user in the user pool, it is critical that it only be used with external IdPs and provider attributes that have been trusted by the application owner.

See also [AdminDisableProviderForUser](#) (p. 28).

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more

- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

### Request Syntax

```json
{
    "DestinationUser": {
        "ProviderAttributeName": "string",
        "ProviderAttributeValue": "string",
        "ProviderName": "string"
    },
    "SourceUser": {
        "ProviderAttributeName": "string",
        "ProviderAttributeValue": "string",
        "ProviderName": "string"
    },
    "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#) (p. 501).

The request accepts the following data in JSON format.

**DestinationUser** (p. 53)

The existing user in the user pool that you want to assign to the external IdP user account. This user can be a local (Username + Password) Amazon Cognito user pools user or a federated user.
(for example, a SAML or Facebook user). If the user doesn't exist, Amazon Cognito generates an exception. Amazon Cognito returns this user when the new user (with the linked IdP attribute) signs in.

For a native username + password user, the ProviderAttributeValue for the DestinationUser should be the username in the user pool. For a federated user, it should be the provider-specific user_id.

The ProviderAttributeName of the DestinationUser is ignored.

The ProviderName should be set to Cognito for users in Cognito user pools.

**Important**
All attributes in the DestinationUser profile must be mutable. If you have assigned the user any immutable custom attributes, the operation won't succeed.

Type: ProviderUserIdentifierType (p. 451) object

Required: Yes

**SourceUser (p. 53)**

An external IdP account for a user who doesn’t exist yet in the user pool. This user must be a federated user (for example, a SAML or Facebook user), not another native user.

If the SourceUser is using a federated social IdP, such as Facebook, Google, or Login with Amazon, you must set the ProviderAttributeName to Cognito_Subject. For social IdPs, the ProviderName will be Facebook, Google, or LoginWithAmazon, and Amazon Cognito will automatically parse the Facebook, Google, and Login with Amazon tokens for id, sub, and user_id, respectively. The ProviderAttributeValue for the user must be the same value as the id, sub, or user_id value found in the social IdP token.

For OIDC, the ProviderAttributeName can be any value that matches a claim in the ID token, or that your app retrieves from the userInfo endpoint. You must map the claim to a user pool attribute in your IdP configuration, and set the user pool attribute name as the value of ProviderAttributeName in your AdminLinkProviderForUser request.

For SAML, the ProviderAttributeName can be any value that matches a claim in the SAML assertion. To link SAML users based on the subject of the SAML assertion, map the subject to a claim through the SAML IdP and set that claim name as the value of ProviderAttributeName in your AdminLinkProviderForUser request.

For both OIDC and SAML users, when you set ProviderAttributeName to Cognito_Subject, Amazon Cognito will automatically parse the default unique identifier found in the subject from the IdP token.

Type: ProviderUserIdentifierType (p. 451) object

Required: Yes

**UserPoolId (p. 53)**

The user pool ID for the user pool.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminListDevices

Lists devices, as an administrator.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
    "Limit": number,
    "PaginationToken": "string",
    "Username": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**Limit (p. 57)**

The limit of the devices request.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

**PaginationToken (p. 57)**

This API operation returns a limited number of results. The pagination token is an identifier that you can present in an additional API request with the same parameters. When you include the pagination token, Amazon Cognito returns the next set of items after the current list. Subsequent requests return a new pagination token. By use of this token, you can paginate through the full list of items.

Type: String

Length Constraints: Minimum length of 1.

Pattern: \S+

Required: No

**Username (p. 57)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.
Response Syntax

```json
{
  "Devices": [
    {
      "DeviceAttributes": [
        {
          "Name": "string",
          "Value": "string"
        }
      ],
      "DeviceCreateDate": number,
      "DeviceKey": "string",
      "DeviceLastAuthenticatedDate": number,
      "DeviceLastModifiedDate": number
    }
  ],
  "PaginationToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Devices (p. 58)**

The devices in the list of devices response.

Type: Array of DeviceType (p. 420) objects

**PaginationToken (p. 58)**

The identifier that Amazon Cognito returned with the previous request to this operation. When you include a pagination token in your request, Amazon Cognito returns the next set of items in the list. By use of this token, you can paginate through the full list of items.

Type: String

Length Constraints: Minimum length of 1.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminListGroupsForUser

Lists the groups that a user belongs to.

**Note**

Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**

- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

---

## Request Syntax

```json
{
  "Limit": number,
  "NextToken": "string",
  "Username": "string",
  "UserPoolId": "string"
}
```

---

## Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#). The request accepts the following data in JSON format.

**Limit (p. 60)**

The limit of the request to list groups.

- Type: Integer
- Valid Range: Minimum value of 0. Maximum value of 60.
- Required: No

**NextToken (p. 60)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

- Type: String
- Pattern: [\S]+
- Required: No

**Username (p. 60)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.
Type: String
Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\]+
Required: Yes

**UserPoolId (p. 60)**

The user pool ID for the user pool.
Type: String
Pattern: \[\w-]+\_\[0-9a-zA-Z]+
Required: Yes

### Response Syntax

```
{
    "Groups": [
        {
            "CreationDate": number,
            "Description": "string",
            "GroupName": "string",
            "LastModifiedDate": number,
            "Precedence": number,
            "RoleArn": "string",
            "UserPoolId": "string"
        }
    ],
    "NextToken": "string"
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Groups (p. 61)**

The groups that the user belongs to.

Type: Array of **GroupType (p. 430)** objects

**NextToken (p. 61)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String


Pattern: \[\S]+\]
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminListUserAuthEvents

A history of user activity and any risks detected as part of Amazon Cognito advanced security.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more

• Signing AWS API Requests
• Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```json
{
  "MaxResults": number,
  "NextToken": "string",
  "Username": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

MaxResults (p. 63)

The maximum number of authentication events to return. Returns 60 events if you set MaxResults to 0, or if you don't include a MaxResults parameter.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 63)

A pagination token.

Type: String


Pattern: \S+

Required: No

Username (p. 63)

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.
UserPoolId (p. 63)

The user pool ID.

Response Syntax

```json
{
    "AuthEvents": [
        {
            "ChallengeResponses": [
                {
                    "ChallengeName": "string",
                    "ChallengeResponse": "string"
                }
            ],
            "CreationDate": number,
            "EventContextData": {
                "City": "string",
                "Country": "string",
                "DeviceName": "string",
                "IpAddress": "string",
                "Timezone": "string"
            },
            "EventFeedback": {
                "FeedbackDate": number,
                "FeedbackValue": "string",
                "Provider": "string"
            },
            "EventId": "string",
            "EventResponse": "string",
            "EventRisk": {
                "CompromisedCredentialsDetected": boolean,
                "RiskDecision": "string",
                "RiskLevel": "string"
            },
            "EventType": "string"
        }
    ],
    "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**AuthEvents (p. 64)**

The response object. It includes the EventID, EventType, CreationDate, EventRisk, and EventResponse.

Type: Array of **AuthEventType (p. 406)** objects

**NextToken (p. 64)**

A pagination token.

Type: String


Pattern: \[\S\]+

**Errors**

For information about the errors that are common to all actions, see **Common Errors (p. 503)**.

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

**UserPoolAddOnNotEnabledException**

This exception is thrown when user pool add-ons aren't enabled.

HTTP Status Code: 400
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminRemoveUserFromGroup

Removes the specified user from the specified group.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```
{
  "GroupName": "string",
  "Username": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**GroupName (p. 67)**

The group name.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**Username (p. 67)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**UserPoolId (p. 67)**

The user pool ID for the user pool.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminResetUserPassword

Resets the specified user's password in a user pool as an administrator. Works on any user.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](https://docs.aws.amazon.com/cognito/latest/developerguide/sms-settings.html) in the *Amazon Cognito Developer Guide*.

Deactivates a user's password, requiring them to change it. If a user tries to sign in after the API is called, Amazon Cognito responds with a PasswordResetRequiredException error. Your app must then perform the actions that reset your user's password: the forgot-password flow. In addition, if the user pool has phone verification selected and a verified phone number exists for the user, or if email verification is selected and a verified email exists for the user, calling this API will also result in sending a message to the end user with the code to change their password.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

### Request Syntax

```
{
   "ClientMetadata": {
   "string": "string"
 },
   "Username": "string",
   "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**ClientMetadata (p. 70)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.
You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminResetUserPassword API action, Amazon Cognito invokes the function that is assigned to the custom message trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminResetUserPassword request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**Username (p. 70)**

The username of the user that you want to query or modify. The value of this parameter is typically your user’s username, but it can be any of their alias attributes. If username isn’t an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]+`

Required: Yes

**UserPoolId (p. 70)**

The user pool ID for the user pool where you want to reset the user’s password.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

---

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidEmailRoleAccessPolicyException**

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

**LimitExceededException**

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminRespondToAuthChallenge

Some API operations in a user pool generate a challenge, like a prompt for an MFA code, for device authentication that bypasses MFA, or for a custom authentication challenge. An AdminRespondToAuthChallenge API request provides the answer to that challenge, like a code or a secure remote password (SRP). The parameters of a response to an authentication challenge vary with the type of challenge.

For more information about custom authentication challenges, see Custom authentication challenge Lambda triggers.

Note
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more

- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ChallengeName": "string",
   "ChallengeResponses": {
      "string": "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string": "string"
   },
   "ContextData": {
      "EncodedData": "string",
      "HttpHeaders": [
      {
         "headerName": "string",
         "headerValue": "string"
      }
      ],
      "IpAddress": "string",
```

API Version 2016-04-18
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 74)**

The analytics metadata for collecting Amazon Pinpoint metrics for AdminRespondToAuthChallenge calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No

**ChallengeName (p. 74)**

The challenge name. For more information, see AdminInitiateAuth.

Type: String

Valid Values: SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED

Required: Yes

**ChallengeResponses (p. 74)**

The responses to the challenge that you received in the previous request. Each challenge has its own required response parameters. The following examples are partial JSON request bodies that highlight challenge-response parameters.

**Important**

You must provide a SECRET_HASH parameter in all challenge responses to an app client that has a client secret.

**SMS_MFA**

"ChallengeName": "SMS_MFA", "ChallengeResponses": {"SMS_MFA_CODE": 
"[SMS_code]", "USERNAME": "[username]"}

**PASSWORD_VERIFIER**

"ChallengeName": "PASSWORD_VERIFIER", "ChallengeResponses": {
"PASSWORD_CLAIM_SIGNATURE": 
"[claim_signature]", 
"PASSWORD_CLAIM_SECRET_BLOCK": 
"[secret_block]", 
"TIMESTAMP": 
[timestamp], 
"USERNAME": 
"[username]"

Add "DEVICE_KEY" when you sign in with a remembered device.

**CUSTOM_CHALLENGE**

"ChallengeName": "CUSTOM_CHALLENGE", "ChallengeResponses": {
"USERNAME": 
"[username]", 
"ANSWER": 
"[challenge_answer]"}
Add "DEVICE_KEY" when you sign in with a remembered device.

NEW_PASSWORD_REQUIRED

"ChallengeName": "NEW_PASSWORD_REQUIRED", "ChallengeResponses":
{"NEW_PASSWORD": "[new_password]", "USERNAME": "[username]"}

To set any required attributes that InitiateAuth returned in an requiredAttributes parameter, add "userAttributes.ATTRIBUTE_NAME": "ATTRIBUTE_VALUE". This parameter can also set values for writable attributes that aren't required by your user pool.

**Note**
In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In RespondToAuthChallenge, set a value for any keys that Amazon Cognito returned in the requiredAttributes parameter, then use the UpdateUserAttributes API operation to modify the value of any additional attributes.

SOFTWARE_TOKEN_MFA

"ChallengeName": "SOFTWARE_TOKEN_MFA", "ChallengeResponses":
{"USERNAME": "[username]", "SOFTWARE_TOKEN_MFA_CODE":
[authenticator_code]}

DEVICE_SRP_AUTH

"ChallengeName": "DEVICE_SRP_AUTH", "ChallengeResponses":
{"USERNAME": "[username]", "DEVICE_KEY": "[device_key]", "SRP_A": "[srp_a]"}

DEVICE_PASSWORD_VERIFIER

"ChallengeName": "DEVICE_PASSWORD_VERIFIER", "ChallengeResponses":
{"DEVICE_KEY": "[device_key]", "PASSWORD_CLAIM_SIGNATURE":
[claim_signature]", "PASSWORD_CLAIM_SECRET_BLOCK": "[secret_block]",
"TIMESTAMP": [timestamp], "USERNAME": "[username]"}

MFA_SETUP

"ChallengeName": "MFA_SETUP", "ChallengeResponses":
{"USERNAME": "[username]", "SESSION": "[Session ID from VerifySoftwareToken]"

SELECT_MFA_TYPE

"ChallengeName": "SELECT_MFA_TYPE", "ChallengeResponses":
{"USERNAME": "[username]", "ANSWER": "[SMS_MFA or SOFTWARE_TOKEN_MFA]"}

For more information about SECRET_HASH, see Computing secret hash values. For information about DEVICE_KEY, see Working with user devices in your user pool.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ClientId (p. 74)**

The app client ID.

Type: String

Request Parameters

Pattern: \[\w+\]+  
Required: Yes

**ClientMetadata (p. 74)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminRespondToAuthChallenge API action, Amazon Cognito invokes any functions that you have assigned to the following triggers:
- pre sign-up
- custom message
- post authentication
- user migration
- pre token generation
- define auth challenge
- create auth challenge
- verify auth challenge response

When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute that provides the data that you assigned to the ClientMetadata parameter in your AdminRespondToAuthChallenge request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:
- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ContextData (p. 74)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: `ContextDataType (p. 413)` object

Required: No
Session (p. 74)

The session that should be passed both ways in challenge-response calls to the service. If an InitiateAuth or RespondToAuthChallenge API call determines that the caller must pass another challenge, it returns a session with other challenge parameters. This session should be passed as it is to the next RespondToAuthChallenge API call.

Type: String


Required: No

UserPoolId (p. 74)

The ID of the Amazon Cognito user pool.

Type: String


Pattern: \[\w-]+\_\[0-9a-zA-Z]+

Required: Yes

Response Syntax

```
{
  "AuthenticationResult": {
    "AccessToken": "string",
    "ExpiresIn": number,
    "IdToken": "string",
    "NewDeviceMetadata": {
      "DeviceGroupKey": "string",
      "DeviceKey": "string"
    },
    "RefreshToken": "string",
    "TokenType": "string"
  },
  "ChallengeName": "string",
  "ChallengeParameters": {
    "string": "string"
  },
  "Session": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

AuthenticationResult (p. 78)

The result returned by the server in response to the authentication request.

Type: AuthenticationResultType (p. 404) object

ChallengeName (p. 78)

The name of the challenge. For more information, see AdminInitiateAuth.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

CodeMismatchException

This exception is thrown if the provided code doesn't match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

MFAMethodNotFoundException

This exception is thrown when Amazon Cognito can't find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

**UnexpectedLambdaException**

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

**UserLambdaValidationException**

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

**UserNotConfirmedException**

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- [AWS SDK for Python](https://aws-sdk-docs.amazon.com/python/latest/reference/index.html)
AdminSetUserMFAPreference

The user's multi-factor authentication (MFA) preference, including which MFA options are activated, and if any are preferred. Only one factor can be set as preferred. The preferred MFA factor will be used to authenticate a user if multiple factors are activated. If multiple options are activated and no preference is set, a challenge to choose an MFA option will be returned during sign-in.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "SMSMfaSettings": {
    "Enabled": boolean,
    "PreferredMfa": boolean
  },
  "SoftwareTokenMfaSettings": {
    "Enabled": boolean,
    "PreferredMfa": boolean
  },
  "Username": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**SMSMfaSettings (p. 82)**

The SMS text message MFA settings.

Type: SMSMfaSettingsType (p. 464) object

Required: No

**SoftwareTokenMfaSettings (p. 82)**

The time-based one-time password software token MFA settings.

Type: SoftwareTokenMfaSettingsType (p. 466) object

Required: No

**Username (p. 82)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminSetUserPassword

Sets the specified user's password in a user pool as an administrator. Works on any user.

The password can be temporary or permanent. If it is temporary, the user status enters the FORCE_CHANGE_PASSWORD state. When the user next tries to sign in, the InitiateAuth/AdminInitiateAuth response will contain the NEW_PASSWORD_REQUIRED challenge. If the user doesn't sign in before it expires, the user won't be able to sign in, and an administrator must reset their password.

Once the user has set a new password, or the password is permanent, the user status is set to Confirmed.

AdminSetUserPassword can set a password for the user profile that Amazon Cognito creates for third-party federated users. When you set a password, the federated user's status changes from EXTERNAL_PROVIDER to CONFIRMED. A user in this state can sign in as a federated user, and initiate authentication flows in the API like a linked native user. They can also modify their password and attributes in token-authenticated API requests like ChangePassword and UpdateUserAttributes.

As a best security practice and to keep users in sync with your external IdP, don't set passwords on federated user profiles. To set up a federated user for native sign-in with a linked native user, refer to Linking federated users to an existing user profile.

**Note**

Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**

- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

### Request Syntax

```json
{
  "Password": "string",
  "Permanent": boolean,
  "Username": "string",
  "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Password (p. 85)**

The password for the user.

Type: String

Length Constraints: Maximum length of 256.

Pattern: [\S]+
Required: Yes

**Permanet (p. 85)**

True if the password is permanent, False if it is temporary.

Type: Boolean

Required: No

**Username (p. 85)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**UserPoolId (p. 85)**

The user pool ID for the user pool where you want to set the user's password.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

---

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidPasswordException**

This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

---

API Version 2016-04-18

86
**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
AdminSetUserSettings

*This action is no longer supported.* You can use it to configure only SMS MFA. You can't use it to configure time-based one-time password (TOTP) software token MFA. To configure either type of MFA, use AdminSetUserMFAPreference instead.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
   "MFAOptions": [
      {
         "AttributeName": "string",
         "DeliveryMedium": "string"
      }
   ],
   "Username": "string",
   "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**MFAOptions (p. 88)**

You can use this parameter only to set an SMS configuration that uses SMS for delivery.

*Type:* Array of [MFAOptionType](p. 442) objects

*Required:* Yes

**Username (p. 88)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

*Type:* String


*Pattern:* `[^\{L}\{M}\{S}\{N}\{P}\{\}]+`

*Required:* Yes
UserPoolId (p. 88)

The ID of the user pool that contains the user whose options you're setting.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
AdminUpdateAuthEventFeedback

Provides feedback for an authentication event indicating if it was from a valid user. This feedback is used for improving the risk evaluation decision for the user pool as part of Amazon Cognito advanced security.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```json
{
    "EventId": "string",
    "FeedbackValue": "string",
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**EventId (p. 91)**

The authentication event ID.

Type: String


Pattern: \[\w+\-]+

Required: Yes

**FeedbackValue (p. 91)**

The authentication event feedback value. When you provide a FeedbackValue value of valid, you tell Amazon Cognito that you trust a user session where Amazon Cognito has evaluated some level of risk. When you provide a FeedbackValue value of invalid, you tell Amazon Cognito that you don't trust a user session, or you don't believe that Amazon Cognito evaluated a high-enough risk level.

Type: String

Valid Values: Valid | Invalid

Required: Yes
Username (p. 91)

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.

Type: String


Pattern: `\p{L}\p{M}\p{S}\p{N}\p{P}]+`

Required: Yes

UserPoolId (p. 91)

The user pool ID.

Type: String


Pattern: `[\w-]+_([0-9a-zA-Z]+)`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons aren't enabled.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminUpdateDeviceStatus

Updates the device status as an administrator.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

### Request Syntax

```json
{
  "DeviceKey": "string",
  "DeviceRememberedStatus": "string",
  "Username": "string",
  "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**DeviceKey (p. 94)**

The device key.

- Type: String
- Pattern: `\[w-]+\[0-9a-f-]+`
- Required: Yes

**DeviceRememberedStatus (p. 94)**

The status indicating whether a device has been remembered or not.

- Type: String
- Valid Values: remembered | not_remembered
- Required: No

**Username (p. 94)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If `username` isn't an alias attribute in your user pool, you can also use their `sub` in this request.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException
This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AdminUpdateUserAttributes

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Updates the specified user's attributes, including developer attributes, as an administrator. Works on any user. To delete an attribute from your user, submit the attribute in your API request with a blank value.

For custom attributes, you must prepend the custom: prefix to the attribute name.

In addition to updating user attributes, this API can also be used to mark phone and email as verified.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```
{
    "ClientMetadata": {
        "string" : "string"
    },
    "UserAttributes": [
        {
            "Name": "string",
            "Value": "string"
        }
    ],
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.
ClientMetadata (p. 97)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the AdminUpdateUserAttributes API action, Amazon Cognito invokes the function that is assigned to the custom message trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your AdminUpdateUserAttributes request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**
When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

UserAttributes (p. 97)

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the custom: prefix to the attribute name.

If your user pool requires verification before Amazon Cognito updates an attribute value that you specify in this request, Amazon Cognito doesn't immediately update the value of that attribute. After your user receives and responds to a verification message to verify the new value, Amazon Cognito updates the attribute value. Your user can sign in and receive messages with the original attribute value until they verify the new value.

To update the value of an attribute that requires verification in the same API request, include the email_verified or phone_number_verified attribute, with a value of true. If you set the email_verified or phone_number_verified value for an email or phone_number attribute that requires verification to true, Amazon Cognito doesn't send a verification message to your user.

Type: Array of AttributeType (p. 403) objects

Required: Yes

Username (p. 97)

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

AliasExistsException

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you’ve configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
AdminUserGlobalSignOut

Invalidate the identity, access, and refresh tokens that Amazon Cognito issued to a user. Call this operation with your administrative credentials when your user signs out of your app. This results in the following behavior.

- Amazon Cognito no longer accepts token-authorized user operations that you authorize with a signed-out user's access tokens. For more information, see Using the Amazon Cognito user pools API and user pool endpoints.

Amazon Cognito returns an Access Token has been revoked error when your app attempts to authorize a user pools API request with a revoked access token that contains the scope aws.cognito.signin.user.admin.

- Amazon Cognito no longer accepts a signed-out user's ID token in a GetId request to an identity pool with ServerSideTokenCheck enabled for its user pool IdP configuration in CognitoIdentityProvider.

- Amazon Cognito no longer accepts a signed-out user's refresh tokens in refresh requests.

Other requests might be valid until your user's token expires.

**Note**

Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**

- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "Username": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Username (p. 102)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+
Required: Yes

### UserPoolId (p. 102)

The user pool ID.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

### Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

### Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

- **InternalErrorException**
  - This exception is thrown when Amazon Cognito encounters an internal error.
  - HTTP Status Code: 500

- **InvalidParameterException**
  - This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
  - HTTP Status Code: 400

- **NotAuthorizedException**
  - This exception is thrown when a user isn't authorized.
  - HTTP Status Code: 400

- **ResourceNotFoundException**
  - This exception is thrown when the Amazon Cognito service can't find the requested resource.
  - HTTP Status Code: 400

- **TooManyRequestsException**
  - This exception is thrown when the user has made too many requests for a given operation.
  - HTTP Status Code: 400

- **UserNotFoundException**
  - This exception is thrown when a user isn't found.
  - HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
AssociateSoftwareToken

Begins setup of time-based one-time password (TOTP) multi-factor authentication (MFA) for a user, with a unique private key that Amazon Cognito generates and returns in the API response. You can authorize an AssociateSoftwareToken request with either the user's access token, or a session string from a challenge response that you received from Amazon Cognito.

**Note**
Amazon Cognito disassociates an existing software token when you verify the new token in a VerifySoftwareToken API request. If you don't verify the software token and your user pool doesn't require MFA, the user can then authenticate with user name and password credentials alone. If your user pool requires TOTP MFA, Amazon Cognito generates an MFA_SETUP or SOFTWARE_TOKEN_SETUP challenge each time your user signs. Complete setup with AssociateSoftwareToken and VerifySoftwareToken.

After you set up software token MFA for your user, Amazon Cognito generates a SOFTWARE_TOKEN_MFA challenge when they authenticate. Respond to this challenge with your user's TOTP.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Request Syntax**

```json
{
    "AccessToken": "string",
    "Session": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 105)**

A valid access token that Amazon Cognito issued to the user whose software token you want to generate.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: No

**Session (p. 105)**

The session that should be passed both ways in challenge-response calls to the service. This allows authentication of the user as part of the MFA setup process.

Type: String

Response Syntax

```json
{
  "SecretCode": "string",
  "Session": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**SecretCode (p. 106)**

A unique generated shared secret code that is used in the TOTP algorithm to generate a one-time code.

- Type: String
- Length Constraints: Minimum length of 16.
- Pattern: `[A-Za-z0-9]+`

**Session (p. 106)**

The session that should be passed both ways in challenge-response calls to the service. This allows authentication of the user as part of the MFA setup process.

- Type: String

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**ConcurrentModificationException**

This exception is thrown if two or more modifications are happening concurrently.

- HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

- HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

- HTTP Status Code: 500
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated for the user pool.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ChangePassword

Changes the password for a specified user in a user pool.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```json
{
  "AccessToken": "string",
  "PreviousPassword": "string",
  "ProposedPassword": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AccessToken (p. 108)**

A valid access token that Amazon Cognito issued to the user whose password you want to change.

Type: String

Pattern: `[A-Za-z0-9-_=.]`+

Required: Yes

**PreviousPassword (p. 108)**

The old password.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[^\S]`+

Required: Yes

**ProposedPassword (p. 108)**

The new password.

Type: String

Length Constraints: Maximum length of 256.

Pattern: `[^\S]`+

Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalServerError

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.
HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
ConfirmDevice

Confirms tracking of the device. This API call is the call that begins device tracking.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Request Syntax**

```json
{
   "AccessToken": "string",
   "DeviceKey": "string",
   "DeviceName": "string",
   "DeviceSecretVerifierConfig": {
      "PasswordVerifier": "string",
      "Salt": "string"
   }
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**AccessToken (p. 111)**

A valid access token that Amazon Cognito issued to the user whose device you want to confirm.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

**DeviceKey (p. 111)**

The device key.

Type: String


Pattern: [\w-]+_[0-9a-f-]+

Required: Yes

**DeviceName (p. 111)**

The device name.

Type: String


Required: No
DeviceSecretVerifierConfig (p. 111)

The configuration of the device secret verifier.

Type: DeviceSecretVerifierConfigType (p. 419) object

Required: No

Response Syntax

```
{
  "UserConfirmationNecessary": boolean
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserConfirmationNecessary (p. 112)

Indicates whether the user confirmation must confirm the device response.

Type: Boolean

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException

This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400
InvalidUserPoolConfigurationException
This exception is thrown when the user pool configuration is not valid.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

PasswordResetRequiredException
This exception is thrown when a password reset is required.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

UsernameExistsException
This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.
HTTP Status Code: 400

UserNotConfirmedException
This exception is thrown when a user isn't confirmed successfully.
HTTP Status Code: 400

UserNotFoundException
This exception is thrown when a user isn't found.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ConfirmForgotPassword

Allows a user to enter a confirmation code to reset a forgotten password.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```json
{
   "AnalyticsMetadata": {
      "AnalyticsEndpointId": "string"
   },
   "ClientId": "string",
   "ClientMetadata": {
      "string": "string"
   },
   "ConfirmationCode": "string",
   "Password": "string",
   "SecretHash": "string",
   "UserContextData": {
      "EncodedData": "string",
      "IpAddress": "string"
   },
   "Username": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

- **AnalyticsMetadata (p. 115)**
  
The Amazon Pinpoint analytics metadata for collecting metrics for ConfirmForgotPassword calls.

  Type: [AnalyticsMetadataType (p. 402)](#) object

  Required: No

- **ClientId (p. 115)**
  
The app client ID of the app associated with the user pool.

  Type: String


  Pattern: `[\w]+`

  Required: Yes

- **ClientMetadata (p. 115)**
  
  A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.
You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ConfirmForgotPassword API action, Amazon Cognito invokes the function that is assigned to the post confirmation trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a ClientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your ConfirmForgotPassword request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

**Note**
When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:
- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ConfirmationCode (p. 115)**

The confirmation code from your user’s request to reset their password. For more information, see ForgotPassword.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: \[\S]+

Required: Yes

**Password (p. 115)**

The new password that your user wants to set.

Type: String

Length Constraints: Maximum length of 256.

Pattern: \[\S]+

Required: Yes

**SecretHash (p. 115)**

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message. For more information about SecretHash, see Computing secret hash values.

Type: String
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

CodeMismatchException

This exception is thrown if the provided code doesn't match what the server was expecting.

HTTP Status Code: 400

ExpiredCodeException

This exception is thrown if a code has expired.

HTTP Status Code: 400

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidLambdaResponseException
This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.
HTTP Status Code: 400

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

InvalidPasswordException
This exception is thrown when Amazon Cognito encounters an invalid password.
HTTP Status Code: 400

LimitExceededException
This exception is thrown when a user exceeds the limit for a requested AWS resource.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyFailedAttemptsException
This exception is thrown when the user has made too many failed attempts for a given action, such as sign-in.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

UnexpectedLambdaException
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.
HTTP Status Code: 400

UserLambdaValidationException
This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.
HTTP Status Code: 400

UserNotConfirmedException
This exception is thrown when a user isn't confirmed successfully.
HTTP Status Code: 400
UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ConfirmSignUp

This public API operation provides a code that Amazon Cognito sent to your user when they signed up in your user pool via the SignUp API operation. After your user enters their code, they confirm ownership of the email address or phone number that they provided, and their user account becomes active. Depending on your user pool configuration, your users will receive their confirmation code in an email or SMS message.

Local users who signed up in your user pool are the only type of user who can confirm sign-up with a code. Users who federate through an external identity provider (IdP) have already been confirmed by their IdP. Administrator-created users, users created with the AdminCreateUser API operation, confirm their accounts when they respond to their invitation email message and choose a password. They do not receive a confirmation code. Instead, they receive a temporary password.

Note

Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```json
{
  "AnalyticsMetadata": {
    "AnalyticsEndpointId": "string"
  },
  "ClientId": "string",
  "ClientMetadata": {
    "string": "string"
  },
  "ConfirmationCode": "string",
  "ForceAliasCreation": boolean,
  "SecretHash": "string",
  "UserContextData": {
    "EncodedData": "string",
    "IpAddress": "string"
  },
  "Username": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 120)**

The Amazon Pinpoint analytics metadata for collecting metrics for ConfirmSignUp calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No

**ClientId (p. 120)**

The ID of the app client associated with the user pool.
Request Parameters

Type: String


Pattern: \[\w+\]+

Required: Yes

ClientMetadata (p. 120)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ConfirmSignUp API action, Amazon Cognito invokes the function that is assigned to the post confirmation trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your ConfirmSignUp request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

Note
When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

ConfirmationCode (p. 120)

The confirmation code sent by a user’s request to confirm registration.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Pattern: [\S]+

Required: Yes

ForceAliasCreation (p. 120)

Boolean to be specified to force user confirmation irrespective of existing alias. By default set to False. If this parameter is set to True and the phone number/email used for sign up confirmation already exists as an alias with a different user, the API call will migrate the alias from the previous user to the newly created user being confirmed. If set to False, the API will throw an AliasExistsException error.
Type: Boolean
Required: No

**SecretHash (p. 120)**
A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String
Pattern: \[\w+=/]+
Required: No

**UserContextData (p. 120)**
Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 472) object
Required: No

**Username (p. 120)**
The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String
Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+
Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 503).

**AliasExistsException**
This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

**CodeMismatchException**
This exception is thrown if the provided code doesn't match what the server was expecting.
HTTP Status Code: 400
ExpiredCodeException
This exception is thrown if a code has expired.

HTTP Status Code: 400
ForbiddenException
This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 500
InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 400
InvalidLambdaResponseException
This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
LimitExceededException
This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400
NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
TooManyFailedAttemptsException
This exception is thrown when the user has made too many failed attempts for a given action, such as sign-in.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
UnexpectedLambdaException
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.
UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateGroup

Creates a new group in the specified user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```
{
  "Description": "string",
  "GroupName": "string",
  "Precedence": number,
  "RoleArn": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Description (p. 125)**

A string containing the description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

**GroupName (p. 125)**

The name of the group. Must be unique.

Type: String


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}]`

Required: Yes

**Precedence (p. 125)**

A non-negative integer value that specifies the precedence of this group relative to the other groups that a user can belong to in the user pool. Zero is the highest precedence value. Groups with lower Precedence values take precedence over groups with higher or null Precedence values. If a user
belongs to two or more groups, it is the group with the lowest precedence value whose role ARN is given in the user's tokens for the cognito:roles and cognito:preferred_role claims.

Two groups can have the same Precedence value. If this happens, neither group takes precedence over the other. If two groups with the same Precedence have the same role ARN, that role is used in the cognito:preferred_role claim in tokens for users in each group. If the two groups have different role ARNs, the cognito:preferred_role claim isn't set in users' tokens.

The default Precedence value is null. The maximum Precedence value is \(2^{31}-1\).

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**RoleArn (p. 125)**

The role Amazon Resource Name (ARN) for the group.

Type: String


Pattern: \[arn:\[\w+=/,.@-]+:\[\w+=/,.@-]+(:\[\w+=/,.@-\]*)?:[0-9]+:\[\w+=/,.@-\]+(:\[\w+=/,.@-\]*)?\]

Required: No

**UserPoolId (p. 125)**

The user pool ID for the user pool.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+\]

Required: Yes

---

**Response Syntax**

```json
{
    "Group": {
        "CreationDate": "number",
        "Description": "string",
        "GroupName": "string",
        "LastModifiedDate": "number",
        "Precedence": "number",
        "RoleArn": "string",
        "UserPoolId": "string"
    }
}
```

---

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
Group (p. 126)

The group object for the group.

Type: GroupType (p. 430) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

GroupExistsException

This exception is thrown when Amazon Cognito encounters a group that already exists in the user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
CreateIdentityProvider

Creates an IdP for a user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
    "AttributeMapping": {
        "string": "string"
    },
    "IdpIdentifiers": [ "string" ],
    "ProviderDetails": {
        "string": "string"
    },
    "ProviderName": "string",
    "ProviderType": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**AttributeMapping (p. 129)**

A mapping of IdP attributes to standard and custom user pool attributes.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 32.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**IdpIdentifiers (p. 129)**

A list of IdP identifiers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Request Parameters

**ProviderDetails (p. 129)**

The IdP details. The following list describes the provider detail keys for each IdP type.

- For Google and Login with Amazon:
  - client_id
  - client_secret
  - authorize_scopes

- For Facebook:
  - client_id
  - client_secret
  - authorize_scopes
  - api_version

- For Sign in with Apple:
  - client_id
  - team_id
  - key_id
  - private_key
  - authorize_scopes

- For OpenID Connect (OIDC) providers:
  - client_id
  - client_secret
  - attributes_request_method
  - oidc_issuer
  - authorize_scopes
  - The following keys are only present if Amazon Cognito didn't discover them at the `oidc_issuer` URL.
    - authorize_url
    - token_url
    - attributes_url
    - jwks_uri
  - Amazon Cognito sets the value of the following keys automatically. They are read-only.
    - attributes_url_add_attributes

- For SAML providers:
  - MetadataFile or MetadataURL
  - IDPSignout *optional*

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes

**ProviderName (p. 129)**

The IdP name.

Type: String
Pattern: [^\p{Z}][\p{L}\p{M}\p{S}\p{N}\p{P}][^\p{Z}]+
Required: Yes
ProviderType (p. 129)
The IdP type.
Type: String
Valid Values: SAML | Facebook | Google | LoginWithAmazon | SignInWithApple | OIDC
Required: Yes
UserPoolId (p. 129)
The user pool ID.
Type: String
Pattern: [\w-]+[0-9a-zA-Z]+
Required: Yes

Response Syntax

```
{
  "IdentityProvider": {
    "AttributeMapping": {
      "string": "string"
    },
    "CreationDate": number,
    "IdpIdentifiers": [ "string" ],
    "LastModifiedDate": number,
    "ProviderDetails": {
      "string": "string"
    },
    "ProviderName": "string",
    "ProviderType": "string",
    "UserPoolId": "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

IdentityProvider (p. 131)
The newly created IdP object.
Type: IdentityProviderType (p. 433) object
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

DuplicateProviderException

This exception is thrown when the provider is already supported by the user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateResourceServer

Creates a new OAuth2.0 resource server and defines custom scopes within it.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
    "Identifier": "string",
    "Name": "string",
    "Scopes": [
        {
            "ScopeDescription": "string",
            "ScopeName": "string"
        }
    ],
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Identifier (p. 133)**

A unique resource server identifier for the resource server. This could be an HTTPS endpoint where the resource server is located, such as `https://my-weather-api.example.com`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[\x21\x23-\x5B\x5D-\x7E]+$`

Required: Yes

**Name (p. 133)**

A friendly name for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `^[\w\s+=,.@-]+$`
Required: Yes

**Scopes (p. 133)**

A list of scopes. Each scope is a key-value map with the keys name and description.

Type: Array of `ResourceServerScopeType (p. 453)` objects

Array Members: Maximum number of 100 items.

Required: No

**UserPoolId (p. 133)**

The user pool ID for the user pool.

Type: String


Pattern: `\w-]+_[0-9a-zA-Z]+`

Required: Yes

### Response Syntax

```json
{
  "ResourceServer": {
    "Identifier": "string",
    "Name": "string",
    "Scopes": [
      {
        "ScopeDescription": "string",
        "ScopeName": "string"
      }
    ],
    "UserPoolId": "string"
  }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**ResourceServer (p. 134)**

The newly created resource server.

Type: `ResourceServerType (p. 454)` object

### Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
LimitExceededException
This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400
NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateUserImportJob

Creates a user import job.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "CloudWatchLogsRoleArn": "string",
  "JobName": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](https://docs.aws.amazon.com/cognito-user-pools/latest/APIReference/Overview.html#Common-Parameters).

The request accepts the following data in JSON format.

- **CloudWatchLogsRoleArn (p. 136)**
  - The role ARN for the Amazon CloudWatch Logs Logging role for the user import job.
  - Type: String
  - Pattern: `arn:[:/,.@-]+[:/,.@-]+`:([/w+=/,.@-]+)+(:[\w+=/,.@-]+)?:[0-9]+([/w+=/,.@-]+)+
  - Required: Yes

- **JobName (p. 136)**
  - The job name for the user import job.
  - Type: String
  - Pattern: `[^\s]+,.@-]+`
  - Required: Yes

- **UserPoolId (p. 136)**
  - The user pool ID for the user pool that the users are being imported into.
Type: String
Pattern: [\w-]+_[0-9a-zA-Z]+
Required: Yes

Response Syntax

```json
{
   "UserImportJob": {
      "CloudWatchLogsRoleArn": "string",
      "CompletionDate": number,
      "CompletionMessage": "string",
      "CreationDate": number,
      "FailedUsers": number,
      "ImportedUsers": number,
      "JobId": "string",
      "JobName": "string",
      "PreSignedUrl": "string",
      "SkippedUsers": number,
      "StartDate": number,
      "Status": "string",
      "UserPoolId": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserImportJob (p. 137)

The job object that represents the user import job.

Type: UserImportJobType (p. 473) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalServerErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.
HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**PreconditionNotMetException**

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

---

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateUserPool

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Creates a new Amazon Cognito user pool and sets the password policy for the pool.

**Important**
If you don't provide a value for an attribute, Amazon Cognito sets it to its default value.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
   "AccountRecoverySetting": {
      "RecoveryMechanisms": [
         {
            "Name": "string",
            "Priority": number
         }
      ]
   },
   "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
         "EmailMessage": "string",
         "EmailSubject": "string",
         "SMSMessage": "string"
      },
      "UnusedAccountValidityDays": number
   },
   "AliasAttributes": [ "string" ],
   "AutoVerifiedAttributes": [ "string" ],
   "DeletionProtection": "string",
   "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": boolean,
      "DeviceOnlyRememberedOnUserPrompt": boolean
   },
   "EmailConfiguration": {
```

API Version 2016-04-18
139
"ConfigurationSet": "string",
"EmailSendingAccount": "string",
"From": "string",
"ReplyToEmailAddress": "string",
"SourceArn": "string"
},
"EmailVerificationMessage": "string",
"EmailVerificationSubject": "string",
"LambdaConfig": {
  "CreateAuthChallenge": "string",
  "CustomEmailSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "CustomMessage": "string",
  "CustomSMSSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "DefineAuthChallenge": "string",
  "KMSKeyID": "string",
  "PostAuthentication": "string",
  "PostConfirmation": "string",
  "PreAuthentication": "string",
  "PreSignUp": "string",
  "PreTokenGeneration": "string",
  "UserMigration": "string",
  "VerifyAuthChallengeResponse": "string"
},
"MfaConfiguration": "string",
"Policies": {
  "PasswordPolicy": {
    "MinimumLength": number,
    "RequireLowercase": boolean,
    "RequireNumbers": boolean,
    "RequireSymbols": boolean,
    "RequireUppercase": boolean,
    "TemporaryPasswordValidityDays": number
  }
},
"PoolName": "string",
"Schema": [
  {
    "AttributeDataType": "string",
    "DeveloperOnlyAttribute": boolean,
    "Mutable": boolean,
    "Name": "string",
    "NumberAttributeConstraints": {
      "MaxValue": "string",
      "MinValue": "string"
    },
    "Required": boolean,
    "StringAttributeConstraints": {
      "MaxLength": "string",
      "MinLength": "string"
    }
  }
],
"SmsAuthenticationMessage": "string",
"SmsConfiguration": {
  "ExternalId": "string",
  "SnsCallerArn": "string",
  "SnsRegion": "string"
},
"SmsVerificationMessage": "string",
"UserAttributeUpdateSettings": {

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

AccountRecoverySetting (p. 139)

The available verified method a user can use to recover their password when they call ForgotPassword. You can use this setting to define a preferred method when a user has more than one method available. With this setting, SMS doesn't qualify for a valid password recovery mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS is preferred through email.

Type: AccountRecoverySettingType (p. 395) object

Required: No

AdminCreateUserConfig (p. 139)

The configuration for AdminCreateUser requests.

Type: AdminCreateUserConfigType (p. 399) object

Required: No

AliasAttributes (p. 139)

Attributes supported as an alias for this user pool. Possible values: phone_number, email, or preferred_username.

Type: Array of strings

Valid Values: phone_number | email | preferred_username

Required: No

AutoVerifiedAttributes (p. 139)

The attributes to be auto-verified. Possible values: email, phone_number.
Type: Array of strings

Valid Values: phone_number | email

Required: No

**DeletionProtection (p. 139)**

When active, DeletionProtection prevents accidental deletion of your user pool. Before you can delete a user pool that you have protected against deletion, you must deactivate this feature.

When you try to delete a protected user pool in a DeleteUserPool API request, Amazon Cognito returns an InvalidParameterException error. To delete a protected user pool, send a new DeleteUserPool request after you deactivate deletion protection in an UpdateUserPool API request.

Type: String

Valid Values: ACTIVE | INACTIVE

Required: No

**DeviceConfiguration (p. 139)**

The device-remembering configuration for a user pool. A null value indicates that you have deactivated device remembering in your user pool.

**Note**

When you provide a value for any DeviceConfiguration field, you activate the Amazon Cognito device-remembering feature.

Type: DeviceConfigurationType (p. 418) object

Required: No

**EmailConfiguration (p. 139)**

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages from your user pool.

Type: EmailConfigurationType (p. 423) object

Required: No

**EmailVerificationMessage (p. 139)**

This parameter is no longer used. See VerificationMessageTemplateType.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}\[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}

Required: No

**EmailVerificationSubject (p. 139)**

This parameter is no longer used. See VerificationMessageTemplateType.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.
LambdaConfig (p. 139)

The Lambda trigger configuration information for the new user pool.

Note
In a push model, event sources (such as Amazon S3 and custom applications) need permission to invoke a function. So you must make an extra call to add permission for these event sources to invoke your Lambda function.
For more information on using the Lambda API to add permission, see AddPermission.
For adding permission using the AWS CLI, see add-permission.

Type: LambdaConfigType (p. 436) object

Required: No

MfaConfiguration (p. 139)

Specifies MFA configuration details.

Type: String

Valid Values: OFF | ON | OPTIONAL

Required: No

Policies (p. 139)

The policies associated with the new user pool.

Type: UserPoolPolicyType (p. 489) object

Required: No

PoolName (p. 139)

A string used to name the user pool.

Type: String


Pattern: [\w\s+=,.@-]+

Required: Yes

Schema (p. 139)

An array of schema attributes for the new user pool. These attributes can be standard or custom attributes.

Type: Array of SchemaAttributeType (p. 459) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: No

SmsAuthenticationMessage (p. 139)

A string representing the SMS authentication message.

Type: String
Request Parameters


Pattern: .*\{####\}.*

Required: No

**SmsConfiguration (p. 139)**

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Type: [SmsConfigurationType](p. 461) object

Required: No

**SmsVerificationMessage (p. 139)**

This parameter is no longer used. See [VerificationMessageTemplateType](#).

Type: String


Pattern: .*\{####\}.*

Required: No

**UserAttributeUpdateSettings (p. 139)**

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see [Verifying updates to email addresses and phone numbers](#).

Type: [UserAttributeUpdateSettingsType](p. 471) object

Required: No

**UsernameAttributes (p. 139)**

Specifies whether a user can use an email address or phone number as a username when they sign up.

Type: Array of strings

Valid Values: phone_number | email

Required: No

**UsernameConfiguration (p. 139)**

Case sensitivity on the username input for the selected sign-in option. When case sensitivity is set to False (case insensitive), users can sign in with any combination of capital and lowercase letters. For example, `username`, `USERNAME`, or `UserName`, or for email, `email@example.com` or `EMail@eXamplE.Com`. For most use cases, set case sensitivity to False (case insensitive) as a best practice. When usernames and email addresses are case insensitive, Amazon Cognito treats any variation in case as the same user, and prevents a case variation from being assigned to the same attribute for a different user.

This configuration is immutable after you set it. For more information, see [UsernameConfigurationType](#).

Type: [UsernameConfigurationType](p. 476) object
User pool add-ons. Contains settings for activation of advanced security features. To log user security information but take no action, set to AUDIT. To configure automatic security responses to risky traffic to your user pool, set to ENFORCED.

For more information, see Adding advanced security to a user pool.

Type: UserPoolAddOnsType (p. 477) object

UserPoolTags (p. 139)

The tag keys and values to assign to the user pool. A tag is a label that you can use to categorize and manage user pools in different ways, such as by purpose, owner, environment, or other criteria.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

VerificationMessageTemplate (p. 139)

The template for the verification message that the user sees when the app requests permission to access the user's information.

Type: VerificationMessageTemplateType (p. 499) object

Response Syntax

```
{
    "UserPool": {
        "AccountRecoverySetting": {
            "RecoveryMechanisms": [
                {
                    "Name": "string",
                    "Priority": number
                }
            ],
        },
        "AdminCreateUserConfig": {
            "AllowAdminCreateUserOnly": boolean,
            "InviteMessageTemplate": {
                "EmailMessage": "string",
                "EmailSubject": "string",
                "SMSMessage": "string"
            },
            "UnusedAccountValidityDays": number
        },
        "AliasAttributes": [ "string" ],
        "Arn": "string",
        "AutoVerifiedAttributes": [ "string" ],
        "CreationDate": number,
        "CustomDomain": "string",
        "DeletionProtection": "string"
    }
}
```
"DeviceConfiguration": {
  "ChallengeRequiredOnNewDevice": boolean,
  "DeviceOnlyRememberedOnUserPrompt": boolean
},
"Domain": "string",
"EmailConfiguration": {
  "ConfigurationSet": "string",
  "EmailSendingAccount": "string",
  "From": "string",
  "ReplyToEmailAddress": "string",
  "SourceArn": "string"
},
"EmailConfigurationFailure": "string",
"EmailVerificationMessage": "string",
"EmailVerificationSubject": "string",
"EstimatedNumberOfUsers": number,
"Id": "string",
"LambdaConfig": {
  "CreateAuthChallenge": "string",
  "CustomEmailSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "CustomMessage": "string",
  "CustomSMSSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "DefineAuthChallenge": "string",
  "KMSKeyID": "string",
  "PostAuthentication": "string",
  "PostConfirmation": "string",
  "PreAuthentication": "string",
  "PreSignUp": "string",
  "PreTokenGeneration": "string",
  "UserMigration": "string",
  "VerifyAuthChallengeResponse": "string"
},
"LastModifiedDate": number,
"MfaConfiguration": "string",
"Name": "string",
"Policies": {
  "PasswordPolicy": {
    "MinimumLength": number,
    "RequireLowercase": boolean,
    "RequireNumbers": boolean,
    "RequireSymbols": boolean,
    "RequireUppercase": boolean,
    "TemporaryPasswordValidityDays": number
  }
},
"SchemaAttributes": [
  {
    "AttributeDataType": "string",
    "DeveloperOnlyAttribute": boolean,
    "Mutable": boolean,
    "Name": "string",
    "NumberAttributeConstraints": {
      "MaxValue": "string",
      "MinValue": "string"
    },
    "Required": boolean,
    "StringAttributeConstraints": {
      "MaxLength": "string",
      "MinLength": "string"
    }
  }
]
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserPool (p. 145)

A container for the user pool details.

Type: UserPoolType (p. 490) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.
HTTP Status Code: 400
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
InvalidSmsRoleAccessPolicyException
This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400
InvalidSmsRoleTrustRelationshipException
This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400
LimitExceededException
This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400
NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
UserPoolTaggingException
This exception is thrown when a user pool tag can't be set or updated.

Examples

Example

The following example creates a user pool with all configurable properties set to an example value. The resulting user pool allows sign-in with username or email address, has optional MFA, and has a Lambda function assigned to each possible trigger.

Sample Request

```
POST HTTP/1.1
Host: cognito-idp.us-east-1.amazonaws.com
X-Amz-Date: 20230613T200059Z
Accept-Encoding: identity
X-Amz-Target: AWSCognitoIdentityProviderService.CreateUserPool
```
User-Agent: <UserAgentString>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>, Signature=<Signature>
Content-Length: <PayloadSizeBytes>
{
    "AccountRecoverySetting": {
        "RecoveryMechanisms": [
            {
                "Name": "verified_email",
                "Priority": 1
            }
        ]
    },
    "AdminCreateUserConfig": {
        "AllowAdminCreateUserOnly": false,
        "InviteMessageTemplate": {
            "EmailMessage": "Your username is {username} and temporary password is {####}.",
            "EmailSubject": "Your sign-in information",
            "SMSMessage": "Your username is {username} and temporary password is {####}"}
    },
    "AliasAttributes": ["email"],
    "AutoVerifiedAttributes": ["email"],
    "DeviceConfiguration": {
        "ChallengeRequiredOnNewDevice": true,
        "DeviceOnlyRememberedOnUserPrompt": true
    },
    "DeletionProtection": "ACTIVE",
    "EmailConfiguration": {
        "ConfigurationSet": "my-test-ses-configuration-set",
        "EmailSendingAccount": "DEVELOPER",
        "From": "support@example.com",
        "ReplyToEmailAddress": "support@example.com",
        "SourceArn": "arn:aws:ses:us-east-1:123456789012:identity/support@example.com"
    },
    "EmailVerificationMessage": "Your verification code is {####}.",
    "EmailVerificationSubject": "Verify your email address",
    "LambdaConfig": {
        "KMSKeyID": "arn:aws:kms:us-east-1:123456789012:key/a6c4f8e2-0c65-47db-925f-87854bc9e357",
        "CustomEmailSender": {
            "LambdaVersion": "V1_0"
        },
        "CustomSMSSender": {
            "LambdaVersion": "V1_0"
        },
    },
    "MfaConfiguration": "OPTIONAL",
    "Policies": {
        "Policies": ["PolicyName1", "PolicyName2"]
    }
}
"PasswordPolicy": {
  "MinimumLength": 6,
  "RequireLowercase": true,
  "RequireNumbers": true,
  "RequireSymbols": true,
  "RequireUppercase": true,
  "TemporaryPasswordValidityDays": 7
},

"PoolName": "my-test-user-pool",

"Schema": [
  {
    "AttributeDataType": "Number",
    "DeveloperOnlyAttribute": true,
    "Mutable": true,
    "Name": "mydev",
    "NumberAttributeConstraints": {
      "MaxValue": "99",
      "MinValue": "1"
    },
    "Required": false,
    "StringAttributeConstraints": {
      "MaxLength": "99",
      "MinLength": "1"
    }
  }
],

"SmsAuthenticationMessage": "Your verification code is {####}.",

"SmsConfiguration": {
  "ExternalId": "my-role-external-id",
  "SnsCallerArn": "arn:aws:iam::123456789012:role/service-role/test-cognito-SMS-Role"
},

"SmsVerificationMessage": "Your verification code is {####}.",

"UserAttributeUpdateSettings": {
  "AttributesRequireVerificationBeforeUpdate": [
    "email"
  ]
},

"UsernameConfiguration": {
  "CaseSensitive": true
},

"UserPoolAddOns": {
  "AdvancedSecurityMode": "OFF"
},

"UserPoolTags": {
  "my-test-tag-key": "my-test-tag-key"
},

"VerificationMessageTemplate": {
  "DefaultEmailOption": "CONFIRM_WITH_CODE",
  "EmailMessage": "Your confirmation code is {####}",
  "EmailMessageByLink": "Choose this link to {##verify your email##}",
  "EmailSubject": "Here is your confirmation code",
  "EmailSubjectByLink": "Here is your confirmation link",
  "SmsMessage": "Your confirmation code is {####}"}

Sample Response

HTTP/1.1 200 OK
Date: Tue, 13 Jun 2023 20:00:59 GMT
Content-Type: application/x-amz-json-1.0
Content-Length: <PayloadSizeBytes>
vary: origin
Amazon Cognito User Pools API Reference

Examples

vary: access-control-request-method
vary: access-control-request-headers
x-amzn-requestid: a1b2c3d4-e5f6-a1b2-c3d4-EXAMPLE11111
Connection: keep-alive

{
  "UserPool": {
    "AccountRecoverySetting": {
      "RecoveryMechanisms": [
        {
          "Name": "verified_email",
          "Priority": 1
        }
      ]
    },
    "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": false,
      "InviteMessageTemplate": {
        "EmailMessage": "Your username is {username} and temporary password is {####}.",
        "EmailSubject": "Your sign-in information",
        "SMSMessage": "Your username is {username} and temporary password is {####}"
      },
      "UnusedAccountValidityDays": 7
    },
    "AliasAttributes": [
      "email"
    ],
    "AutoVerifiedAttributes": [
      "email"
    ],
    "CreationDate": 1689721665.239,
    "DeletionProtection": "ACTIVE",
    "DeviceConfiguration": {
      "ChallengeRequiredOnNewDevice": true,
      "DeviceOnlyRememberedOnUserPrompt": true
    },
    "EmailConfiguration": {
      "ConfigurationSet": "my-test-ses-configuration-set",
      "EmailSendingAccount": "DEVELOPER",
      "From": "support@example.com",
      "ReplyToEmailAddress": "support@example.com",
      "SourceArn": "arn:aws:ses:us-east-1:123456789012:identity/support@example.com"
    },
    "EmailVerificationMessage": "Your verification code is {####}.",
    "EmailVerificationSubject": "Verify your email address",
    "EstimatedNumberOfUsers": 0,
    "Id": "us-east-1_EXAMPLE",
    "LambdaConfig": {
      "CustomEmailSender": {
        "LambdaVersion": "V1_0"
      },
      "CustomSMSender": {
        "LambdaVersion": "V1_0"
      },
      "KMSKeyID": "arn:aws:kms:us-east-1:767671399759:key/4d43904c-8edf-4bb4-9fca-fb1a80e41cbe",
      "PreAuthenticationChallenge": "arn:aws:lambda:us-east-1:123456789012:function:MyFunction"
    },
    "PasswordConfiguration": {
      "PasswordPolicy": {
        "MinimumLength": 6,
        "MinimumNumbers": 1,
        "MinimumUpperCases": 0,
        "MinimumLowerCases": 0,
        "MinimumSymbols": 0,
        "AllowSamePasswordAsLastN": 5
      }
    },
    "Origin": "null",
    "PublicUserPolicy": {
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Action": "cognito-idp:.adminGetUser",
        },
        {
          "Effect": "Allow",
          "Action": "cognito-idp:userPool_exists",
        }
      ]
    },
    "SmsConfiguration": {
      "SmsVerificationMessage": "Your verification code is {####}.",
      "SmsVerificationCodeValiditySeconds": 180
    },
    "UserAttributes": [
      {"Name": "email", "Value": "abc@example.com"}]
  }
}
},
"LastModifiedDate": 1689721665.239,
"MfaConfiguration": "OPTIONAL",
"Name": "my-test-user-pool",
"Policies": {
  "PasswordPolicy": {
    "MinimumLength": 6,
    "RequireLowercase": true,
    "RequireNumbers": true,
    "RequireSymbols": true,
    "RequireUppercase": true,
    "TemporaryPasswordValidityDays": 7
  }
},
"SchemaAttributes": [
  {
    "AttributeDataType": "String",
    "DeveloperOnlyAttribute": false,
    "Mutable": false,
    "Name": "sub",
    "Required": true,
    "StringAttributeConstraints": {
      "MaxLength": "2048",
      "MinLength": "1"
    }
  },
  {
    "AttributeDataType": "String",
    "DeveloperOnlyAttribute": false,
    "Mutable": true,
    "Name": "name",
    "Required": false,
    "StringAttributeConstraints": {
      "MaxLength": "2048",
      "MinLength": "0"
    }
  },
  {
    "AttributeDataType": "String",
    "DeveloperOnlyAttribute": false,
    "Mutable": true,
    "Name": "given_name",
    "Required": false,
    "StringAttributeConstraints": {
      "MaxLength": "2048",
      "MinLength": "0"
    }
  },
  {
    "AttributeDataType": "String",
    "DeveloperOnlyAttribute": false,
    "Mutable": true,
    "Name": "family_name",
    "Required": false,
    "StringAttributeConstraints": {
      "MaxLength": "2048",
      "MinLength": "0"
    }
  }
]
"MinLength": "0"
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "middle_name",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "nickname",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "preferred_username",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "profile",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "picture",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
},
{
  "AttributeDataType": "String",
  "DeveloperOnlyAttribute": false,
  "Mutable": true,
  "Name": "website",
  "Required": false,
  "StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
  }
}
"MinLength": "0"
}
],
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "email",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
],
{
"AttributeDataType": "Boolean",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "email_verified",
"Required": false
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "gender",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
],
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "birthdate",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "10",
  "MinLength": "10"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "zoneinfo",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "locale",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "zoneinfo",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "locale",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "zoneinfo",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "locale",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "zoneinfo",
"Required": false,
"StringAttributeConstraints": {
  "MaxLength": "2048",
  "MinLength": "0"
}
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
Examples

```
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "phone_number",
"Required": false,
"StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
}
},
{
"AttributeDataType": "Boolean",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "phone_number_verifie",
"Required": false
},
{
"AttributeDataType": "String",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "address",
"Required": false,
"StringAttributeConstraints": {
    "MaxLength": "2048",
    "MinLength": "0"
}
},
{
"AttributeDataType": "Number",
"DeveloperOnlyAttribute": false,
"Mutable": true,
"Name": "updated_at",
"NumberAttributeConstraints": {
    "MinValue": "0"
},
"Required": false
},
{
"AttributeDataType": "Number",
"DeveloperOnlyAttribute": true,
"Mutable": true,
"Name": "dev:custom:mydev",
"NumberAttributeConstraints": {
    "MaxValue": "99",
    "MinValue": "1"
},
"Required": false
},
"SmsAuthenticationMessage": "Your verification code is {####}.",
"SmsConfiguration": {
    "ExternalId": "my-role-external-id",
    "SnsCallerArn": "arn:aws:iamp:123456789012:role/service-role/test-cognito-SMS-Role",
    "SnsRegion": "us-east-1"
},
"SmsVerificationMessage": "Your verification code is {####}.",
"UserAttributeUpdateSettings": {
    "AttributesRequireVerificationBeforeUpdate": ["email"
},
"UserPoolAddOns": {
    "AdvancedSecurityMode": "OFF"
}
"UserPoolTags": {
    "my-test-tag-key": "my-test-tag-value"
},
"UsernameConfiguration": {
    "CaseSensitive": true
},
"VerificationMessageTemplate": {
    "DefaultEmailOption": "CONFIRM_WITH_CODE",
    "EmailMessage": "Your confirmation code is {####}",
    "EmailMessageByLink": "Choose this link to {##verify your email##}",
    "EmailSubject": "Here is your confirmation code",
    "EmailSubjectByLink": "Here is your confirmation link",
    "SmsMessage": "Your confirmation code is {####}"
}
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateUserPoolClient

Creates the user pool client.

When you create a new user pool client, token revocation is automatically activated. For more information about revoking tokens, see RevokeToken.

**Important**
If you don't provide a value for an attribute, Amazon Cognito sets it to its default value.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```json
{
  "AccessTokenValidity": number,
  "AllowedOAuthFlows": [ "string" ],
  "AllowedOAuthFlowsUserPoolClient": boolean,
  "AllowedOAuthScopes": [ "string" ],
  "AnalyticsConfiguration": {
    "ApplicationArn": "string",
    "ApplicationId": "string",
    "ExternalId": "string",
    "RoleArn": "string",
    "UserDataShared": boolean
  },
  "AuthSessionValidity": number,
  "CallbackURLs": [ "string" ],
  "ClientName": "string",
  "DefaultRedirectURI": "string",
  "EnablePropagateAdditionalUserContextData": boolean,
  "EnableTokenRevocation": boolean,
  "ExplicitAuthFlows": [ "string" ],
  "GenerateSecret": boolean,
  "LogoutURLs": [ "string" ],
  "PreventUserExistenceErrors": "string",
  "ReadAttributes": [ "string" ],
  "RefreshTokenValidity": number,
  "SupportedIdentityProviders": [ "string" ],
  "TokenValidityUnits": {
    "AccessToken": "string",
    "IdToken": "string",
    "RefreshToken": "string"
  },
  "UserPoolId": "string",
  "WriteAttributes": [ "string" ]
}
```
Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**AccessTokenValidity (p. 157)**

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for AccessTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.

For example, when you set AccessTokenValidity to 10 and TokenValidityUnits to hours, your user can authorize access with their access token for 10 hours.

The default time unit for AccessTokenValidity in an API request is hours. **Valid range** is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your access tokens are valid for one hour.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

**AllowedOAuthFlows (p. 157)**

The allowed OAuth flows.

- **code**
  
  Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the `/oauth2/token` endpoint.

- **implicit**
  
  Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

  client_credentials
  
  Issue the access token from the `/oauth2/token` endpoint directly to a non-person user using a combination of the client ID and client secret.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: code | implicit | client_credentials

Required: No

**AllowedOAuthFlowsUserPoolClient (p. 157)**

Set to `true` to use OAuth 2.0 features in your user pool app client.

AllowedOAuthFlowsUserPoolClient must be `true` before you can configure the following features in your app client.

- **CallbackURLs**: Callback URLs.
- **LogoutURLs**: Sign-out redirect URLs.
- **AllowedOAuthScopes**: OAuth 2.0 scopes.
• AllowedOAuthFlows: Support for authorization code, implicit, and client credentials OAuth 2.0 grants.

To use OAuth 2.0 features, configure one of these features in the Amazon Cognito console or set AllowedOAuthFlowsUserPoolClient to true in a CreateUserPoolClient or UpdateUserPoolClient API request. If you don’t set a value for AllowedOAuthFlowsUserPoolClient in a request with the AWS CLI or SDKs, it defaults to false.

Type: Boolean

Required: No

AllowedOAuthScopes (p. 157)

The allowed OAuth scopes. Possible values provided by OAuth are phone, email, openid, and profile. Possible values provided by AWS are aws.cognito.signin.user.admin. Custom scopes created in Resource Servers are also supported.

Type: Array of strings

Array Members: Maximum number of 50 items.

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [\x21\x23-\x5B\x5D-\x7E]+

Required: No

AnalyticsConfiguration (p. 157)

The user pool analytics configuration for collecting metrics and sending them to your Amazon Pinpoint campaign.

Note
In AWS Regions where Amazon Pinpoint isn’t available, user pools only support sending events to Amazon Pinpoint projects in AWS Region us-east-1. In Regions where Amazon Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

Type: AnalyticsConfigurationType (p. 400) object

Required: No

AuthSessionValidity (p. 157)

Amazon Cognito creates a session token for each API request in an authentication flow. AuthSessionValidity is the duration, in minutes, of that session token. Your user pool native user must respond to each authentication challenge before the session expires.

Type: Integer


Required: No

CallbackURLs (p. 157)

A list of allowed redirect (callback) URLs for the IdPs.

A redirect URI must:
- Be an absolute URI.
- Be registered with the authorization server.
• Not include a fragment component.

See [OAuth 2.0 - Redirection Endpoint](#).

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.


Pattern: $[^\p{L}\p{M}\p{S}\p{N}\p{P}]+$

Required: No

**ClientName (p. 157)**

The client name for the user pool client you would like to create.

Type: String


Pattern: \[\w\s+=,.@-]+\]

Required: Yes

**DefaultRedirectURI (p. 157)**

The default redirect URI. Must be in the CallbackURLs list.

A redirect URI must:
• Be an absolute URI.
• Be registered with the authorization server.
• Not include a fragment component.

See [OAuth 2.0 - Redirection Endpoint](#).

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported.

Type: String


Pattern: $[^\p{L}\p{M}\p{S}\p{N}\p{P}]+$

Required: No

**EnablePropagateAdditionalUserContextData (p. 157)**

Activates the propagation of additional user context data. For more information about propagation of user context data, see [Adding advanced security to a user pool](#).

If you don't include this parameter, you can't send device fingerprint information, including source IP address, to Amazon Cognito advanced security. You can only activate EnablePropagateAdditionalUserContextData in an app client that has a client secret.

Type: Boolean
Required: No

EnableTokenRevocation (p. 157)

Activates or deactivates token revocation. For more information about revoking tokens, see RevokeToken.

If you don't include this parameter, token revocation is automatically activated for the new user pool client.

Type: Boolean

Required: No

ExplicitAuthFlows (p. 157)

The authentication flows that you want your user pool client to support. For each app client in your user pool, you can sign in your users with any combination of one or more flows, including with a user name and Secure Remote Password (SRP), a user name and password, or a custom authentication process that you define with Lambda functions.

Note
If you don't specify a value for ExplicitAuthFlows, your user client supports ALLOW_REFRESH_TOKEN_AUTH, ALLOW_USER_SRP_AUTH, and ALLOW_CUSTOM_AUTH.

Valid values include:
- ALLOW_ADMIN_USER_PASSWORD_AUTH: Enable admin based user password authentication flow ADMIN_USER_PASSWORD_AUTH. This setting replaces the ADMIN_NO_SRP_AUTH setting. With this authentication flow, your app passes a user name and password to Amazon Cognito in the request, instead of using the Secure Remote Password (SRP) protocol to securely transmit the password.
- ALLOW_CUSTOM_AUTH: Enable Lambda trigger based authentication.
- ALLOW_USER_PASSWORD_AUTH: Enable user password-based authentication. In this flow, Amazon Cognito receives the password in the request instead of using the SRP protocol to verify passwords.
- ALLOW_USER_SRP_AUTH: Enable SRP-based authentication.
- ALLOW_REFRESH_TOKEN_AUTH: Enable authflow to refresh tokens.

In some environments, you will see the values ADMIN_NO_SRP_AUTH, CUSTOM_AUTH_FLOW_ONLY, or USER_PASSWORD_AUTH. You can't assign these legacy ExplicitAuthFlows values to user pool clients at the same time as values that begin with ALLOW_, like ALLOW_USER_SRP_AUTH.

Type: Array of strings

Valid Values: ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH | ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH | ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH

Required: No

GenerateSecret (p. 157)

Boolean to specify whether you want to generate a secret for the user pool client being created.

Type: Boolean

Required: No

IdTokenValidity (p. 157)

The ID token time limit. After this limit expires, your user can't use their ID token. To specify the time unit for IdTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.
For example, when you set IdTokenValidity as 10 and TokenValidityUnits as hours, your user can authenticate their session with their ID token for 10 hours.

The default time unit for IdTokenValidity in an API request is hours. Valid range is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your ID tokens are valid for one hour.

Type: Integer
Valid Range: Minimum value of 1. Maximum value of 86400.
Required: No

LogoutURLs (p. 157)
A list of allowed logout URLs for the IdPs.
Type: Array of strings
Array Members: Minimum number of 0 items. Maximum number of 100 items.
Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+
Required: No

PreventUserExistenceErrors (p. 157)
Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to ENABLED and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to LEGACY, those APIs return a UserNotFoundException exception if the user doesn't exist in the user pool.

Valid values include:
• ENABLED - This prevents user existence-related errors.
• LEGACY - This represents the early behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:
• AdminInitiateAuth (p. 45)
• AdminRespondToAuthChallenge (p. 74)
• InitiateAuth (p. 250)
• RespondToAuthChallenge (p. 297)
• ForgotPassword (p. 213)
• ConfirmForgotPassword (p. 115)
• ConfirmSignUp (p. 120)
• ResendConfirmationCode (p. 292)

Type: String
Valid Values: LEGACY | ENABLED
Required: No
**ReadAttributes (p. 157)**

The list of user attributes that you want your app client to have read-only access to. After your user authenticates in your app, their access token authorizes them to read their own attribute value for any attribute in this list. An example of this kind of activity is when your user selects a link to view their profile information. Your app makes a [GetUser](https://docs.aws.amazon.com/cognito/user-pools/latest/APIReference/API_GetUser.html) API request to retrieve and display your user's profile data.

When you don't specify the ReadAttributes for your app client, your app can read the values of `email_verified`, `phone_number_verified`, and the Standard attributes of your user pool. When your user pool has read access to these default attributes, ReadAttributes doesn't return any information. Amazon Cognito only populates ReadAttributes in the API response if you have specified your own custom set of read attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

**RefreshTokenValidity (p. 157)**

The refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for RefreshTokenValidity as seconds, minutes, hours, or days, set a `TokenValidityUnits` value in your API request.

For example, when you set `RefreshTokenValidity` as 10 and `TokenValidityUnits` as days, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for `RefreshTokenValidity` in an API request is days. You can't set `RefreshTokenValidity` to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. *Valid range* is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your refresh tokens are valid for 30 days.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

**SupportedIdentityProviders (p. 157)**

A list of provider names for the identity providers (IdPs) that are supported on this client. The following are supported: COGNITO, Facebook, Google, SignInWithApple, and LoginWithAmazon. You can also specify the names that you configured for the SAML and OIDC IdPs in your user pool, for example MySAMLIdP or MyOIDCIdP.

Type: Array of strings


Pattern: `[^\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+`

Required: No

**TokenValidityUnits (p. 157)**

The units in which the validity times are represented. The default unit for RefreshToken is days, and default for ID and access tokens are hours.

Type: [TokenValidityUnitsType](https://docs.aws.amazon.com/cognito/user-pools/latest/APIReference/TokenValidityUnitsType.html) object
Required: No

**UserPoolId (p. 157)**

The user pool ID for the user pool where you want to create a user pool client.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

**WriteAttributes (p. 157)**

The list of user attributes that you want your app client to have write access to. After your user authenticates in your app, their access token authorizes them to set or modify their own attribute value for any attribute in this list. An example of this kind of activity is when you present your user with a form to update their profile information and they change their last name. Your app then makes an **UpdateUserAttributes** API request and sets family_name to the new value.

When you don't specify the WriteAttributes for your app client, your app can write the values of the Standard attributes of your user pool. When your user pool has write access to these default attributes, WriteAttributes doesn't return any information. Amazon Cognito only populates WriteAttributes in the API response if you have specified your own custom set of write attributes.

If your app client allows users to sign in through an IdP, this array must include all attributes that you have mapped to IdP attributes. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. If your app client does not have write access to a mapped attribute, Amazon Cognito throws an error when it tries to update the attribute. For more information, see [Specifying IdP Attribute Mappings for Your user pool](#).

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

---

## Response Syntax

```
{
  "UserPoolClient": {
    "AccessTokenValidity": number,
    "AllowedOAuthFlows": [ "string" ],
    "AllowedOAuthFlowsUserPoolClient": boolean,
    "AllowedOAuthScopes": [ "string" ],
    "AnalyticsConfiguration": {
      "ApplicationArn": "string",
      "ApplicationId": "string",
      "ExternalId": "string",
      "RoleArn": "string",
      "UserDataShared": boolean
    },
    "AuthSessionValidity": number,
    "CallbackURLs": [ "string" ],
    "ClientId": "string",
    "ClientName": "string",
    "ClientSecret": "string",
    "CreationDate": number,
    "DefaultRedirectURI": "string",
  }
}
```
"EnablePropagateAdditionalUserContextData": boolean,
"EnableTokenRevocation": boolean,
"ExplicitAuthFlows": ["string"],
"IdTokenValidity": number,
"LastModifiedDate": number,
"LogoutURLs": ["string"],
"PreventUserExistenceErrors": "string",
"ReadAttributes": ["string"],
"RefreshTokenValidity": number,
"SupportedIdentityProviders": ["string"],
"TokenValidityUnits": {
  "AccessToken": "string",
  "IdToken": "string",
  "RefreshToken": "string"
},
"UserPoolId": "string",
"WriteAttributes": ["string"]
}
HTTP Status Code: 400
ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
ScopeDoesNotExistException
This exception is thrown when the specified scope doesn't exist.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

Examples

Example
The following example creates an app client with all configurable properties set to an example value. The resulting user pool client connects to an analytics client, allows sign-in with username and password, and has two external identity providers associated with it.

Sample Request

```
POST HTTP/1.1
Host: cognito-idp.us-east-1.amazonaws.com
X-Amz-Date: 20230613T200059Z
Accept-Encoding: identity
X-Amz-Target: AWSCognitoIdentityProviderService.CreateUserPoolClient
User-Agent: <UserAgentString>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>, Signature=<Signature>
Content-Length: <PayloadSizeBytes>

{
    "AccessTokenValidity": 6,
    "AllowedOAuthFlows": [
        "code"
    ],
    "AllowedOAuthFlowsUserPoolClient": true,
    "AllowedOAuthScopes": [
        "aws.cognito.signin.user.admin",
        "openid"
    ],
    "AnalyticsConfiguration": {
        "ApplicationId": "d70b2ba36a8c4dc5a04a0451a31a1e12",
        "ExternalId": "my-external-id",
        "RoleArn": "arn:aws:iam::123456789012:role/test-cognitouserpool-role",
        "UserDataShared": true
    },
    "CallbackURLs": [
        "https://example.com",
        "http://localhost",
        "myapp://example"
    ],
    "ClientName": "my-test-app-client",
    "DefaultRedirectURI": "https://example.com",
    "ClientMetadata": {}
}
```
"ExplicitAuthFlows": [
  "ALLOW_ADMIN_USER_PASSWORD_AUTH",
  "ALLOW_USER_PASSWORD_AUTH",
  "ALLOW_REFRESH_TOKEN_AUTH"
],
"GenerateSecret": true,
"IdTokenValidity": 6,
"LogoutURLs": [
  "https://example.com/logout"
],
"PreventUserExistenceErrors": "ENABLED",
"ReadAttributes": [
  "email",
  "address",
  "preferred_username"
],
"RefreshTokenValidity": 6,
"SupportedIdentityProviders": [
  "SignInWithApple",
  "MySSO"
],
"TokenValidityUnits": {
  "AccessToken": "hours",
  "IdToken": "minutes",
  "RefreshToken": "days"
},
"UserPoolId": "us-east-1_EXAMPLE",
"WriteAttributes": [
  "family_name",
  "email"
]
}

Sample Response

HTTP/1.1 200 OK
Date: Tue, 13 Jun 2023 20:00:59 GMT
Content-Type: application/x-amz-json-1.0
Content-Length: <PayloadSizeBytes>
vary: origin
vary: access-control-request-method
vary: access-control-request-headers
x-amzn-requestid: a1b2c3d4-e5f6-a1b2-c3d4-EXAMPLE111111
Connection: keep-alive

{
  "UserPoolClient": {
    "AccessTokenValidity": 6,
    "AllowedOAuthFlows": [
      "code"
    ],
    "AllowedOAuthFlowsUserPoolClient": true,
    "AllowedOAuthScopes": [
      "aws.cognito.signin.user.admin",
      "openid"
    ],
    "AnalyticsConfiguration": {
      "ApplicationId": "d70b2ba36a8c4dc5a04a0451a31a1e12",
      "ExternalId": "my-external-id",
      "RoleArn": "arn:aws:iam::123456789012:role/test-cognitouserpool-role",
      "UserDataShared": true
    },
    "AuthSessionValidity": 3,
    "CallbackURLs": [  
      "https://example.com/callback"
    ]
  }
}
"https://example.com",
"http://localhost",
"myapp://example"
],
"ClientId": "26cb2c60k7nbmas7rbme9b6pp",
"ClientName": "my-test-app-client",
"ClientSecret": "13ka4h7u28d9oo44tqppq9djqsfvhu8rk4d21ghvpu0k8fj1c2z9",
"CreationDate": 1689885426.107,
"DefaultRedirectURI": "https://example.com",
"EnablePropagateAdditionalUserContextData": false,
"EnableTokenRevocation": true,
"ExplicitAuthFlows": [
  "ALLOW_USER_PASSWORD_AUTH",
  "ALLOW_ADMIN_USER_PASSWORD_AUTH",
  "ALLOW_REFRESH_TOKEN_AUTH"
],
"IdTokenValidity": 6,
"LastModifiedDate": 1689885426.107,
"LogoutURLs": [
  "https://example.com/logout"
],
"PreventUserExistenceErrors": "ENABLED",
"ReadAttributes": [
  "address",
  "preferred_username",
  "email"
],
"RefreshTokenValidity": 6,
"SupportedIdentityProviders": [
  "SignInWithApple",
  "MySSO"
],
"TokenValidityUnits": {
  "AccessToken": "hours",
  "IdToken": "minutes",
  "RefreshToken": "days"
},
"UserPoolId": "us-east-1_EXAMPLE",
"WriteAttributes": [
  "family_name",
  "email"
]
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
CreateUserPoolDomain

Creates a new domain for a user pool.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
  "CustomDomainConfig": {
    "CertificateArn": "string",
    "Domain": "string",
    "UserPoolId": "string"
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**CustomDomainConfig (p. 169)**

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Provide this parameter only if you want to use a custom domain for your user pool. Otherwise, you can exclude this parameter and use the Amazon Cognito hosted domain instead.

For more information about the hosted domain and custom domains, see Configuring a User Pool Domain.

Type: CustomDomainConfigType (p. 415) object

Required: No

**Domain (p. 169)**

The domain string. For custom domains, this is the fully-qualified domain name, such as auth.example.com. For Amazon Cognito prefix domains, this is the prefix alone, such as auth.

Type: String


Pattern: ^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$
Required: Yes

**UserPoolId (p. 169)**

The user pool ID.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

### Response Syntax

```json
{
  "CloudFrontDomain": "string"
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CloudFrontDomain (p. 170)**

The Amazon CloudFront endpoint that you use as the target of the alias that you set up with your Domain Name Service (DNS) provider.

Type: String


Pattern: `^\[a-z0-9]([-\[a-z0-9\-]{0,61}\[a-z0-9\])?\$`

### Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**LimitExceededException**

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400
NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteGroup

Deletes a group.

Calling this action requires developer credentials.

Request Syntax

```
{
  "GroupName": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**GroupName (p. 172)**

The name of the group.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

**UserPoolId (p. 172)**

The user pool ID for the user pool.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteIdentityProvider

Deletes an IdP for a user pool.

Request Syntax

```json
{
  "ProviderName": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ProviderName (p. 174)**

The IdP name.

- Type: String
- Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+`
- Required: Yes

**UserPoolId (p. 174)**

The user pool ID.

- Type: String
- Pattern: `[\w-]+_[0-9a-zA-Z]+`
- Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**ConcurrentModificationException**

- This exception is thrown if two or more modifications are happening concurrently.
- HTTP Status Code: 400
InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnsupportedIdentityProviderException

This exception is thrown when the specified identifier isn't supported.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteResourceServer

Deletes a resource server.

Request Syntax

```json
{
   "Identifier": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**Identifier (p. 176)**

The identifier for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[\x21\x23-\x5B\x5D-\x7E]+`

Required: Yes

**UserPoolId (p. 176)**

The user pool ID for the user pool that hosts the resource server.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteUser

Allows a user to delete their own user profile.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can’t use IAM credentials to authorize requests, and you can’t grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```json
{
   "AccessToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AccessToken (p. 178)**

A valid access token that Amazon Cognito issued to the user whose user profile you want to delete.

Type: String

Pattern: [A-Za-z0-9-_.=]+

Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400
**NotAuthorizedException**
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
**PasswordResetRequiredException**
This exception is thrown when a password reset is required.

HTTP Status Code: 400
**ResourceNotFoundException**
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
**TooManyRequestsException**
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
**UserNotConfirmedException**
This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400
**UserNotFoundException**
This exception is thrown when a user isn't found.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
DeleteUserAttributes

Deletes the attributes for a user.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```
{
    "AccessToken": "string",
    "UserAttributeNames": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

AccessToken (p. 180)

A valid access token that Amazon Cognito issued to the user whose attributes you want to delete.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

UserAttributeNames (p. 180)

An array of strings representing the user attribute names you want to delete.

For custom attributes, you must prepend the custom: prefix to the front of the attribute name.

Type: Array of strings


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).
ForbiddenException
   This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.
   HTTP Status Code: 400

InternalErrorException
   This exception is thrown when Amazon Cognito encounters an internal error.
   HTTP Status Code: 500

InvalidParameterException
   This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
   HTTP Status Code: 400

NotAuthorizedException
   This exception is thrown when a user isn't authorized.
   HTTP Status Code: 400

PasswordResetRequiredException
   This exception is thrown when a password reset is required.
   HTTP Status Code: 400

ResourceNotFoundException
   This exception is thrown when the Amazon Cognito service can't find the requested resource.
   HTTP Status Code: 400

TooManyRequestsException
   This exception is thrown when the user has made too many requests for a given operation.
   HTTP Status Code: 400

UserNotConfirmedException
   This exception is thrown when a user isn't confirmed successfully.
   HTTP Status Code: 400

UserNotFoundException
   This exception is thrown when a user isn't found.
   HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
See Also

- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteUserPool

Deletes the specified Amazon Cognito user pool.

Request Syntax

```json
{
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserPoolId (p. 183)**

The user pool ID for the user pool you want to delete.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400
**TooManyRequestsException**
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400
**UserImportInProgressException**
This exception is thrown when you're trying to modify a user pool while a user import job is in progress for that pool.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
DeleteUserPoolClient

Allows the developer to delete the user pool client.

Request Syntax

```
{
   "ClientId": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientId (p. 185)**

The app client ID of the app associated with the user pool.

- Type: String
- Pattern: `[\w+]`
- Required: Yes

**UserPoolId (p. 185)**

The user pool ID for the user pool where you want to delete the client.

- Type: String
- Pattern: `[\w-]+[0-9a-zA-Z]+`
- Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**ConcurrentModificationException**

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400
InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DeleteUserPoolDomain

Deletes a domain for a user pool.

**Request Syntax**

```json
{
    "Domain": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**Domain (p. 187)**

The domain string. For custom domains, this is the fully-qualified domain name, such as auth.example.com. For Amazon Cognito prefix domains, this is the prefix alone, such as auth.

Type: String


Pattern: `^[a-z0-9](?:[a-z0-9-]{0,61}[a-z0-9])?$`

Required: Yes

**UserPoolId (p. 187)**

The user pool ID.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeIdentityProvider

Gets information about a specific IdP.

Request Syntax

```json
{
  "ProviderName": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ProviderName (p. 189)**

The IdP name.

Type: String


Pattern: \p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}+

Required: Yes

**UserPoolId (p. 189)**

The user pool ID.

Type: String


Pattern: [\w-]+_\[0-9a-zA-Z]+

Required: Yes

Response Syntax

```json
{
  "IdentityProvider": {
    "AttributeMapping": {
      "string": "string"
    },
    "CreationDate": number,
    "IdpIdentifiers": [ "string" ],
    "LastModifiedDate": number,
    "ProviderDetails": {
      "string": "string"
    },
    "ProviderName": "string",
    "ProviderType": "string"
  }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**IdentityProvider (p. 189)**

The identity provider details.

Type: `IdentityProviderType (p. 433)` object

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).  

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
DescribeResourceServer

Describes a resource server.

Request Syntax

```json
{
    "Identifier": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**Identifier (p. 192)**

The identifier for the resource server

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [\x21\x23-\x5B\x5D-\x7E]+

Required: Yes

**UserPoolId (p. 192)**

The user pool ID for the user pool that hosts the resource server.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```json
{
    "ResourceServer": {
        "Identifier": "string",
        "Name": "string",
        "Scopes": [
            {
                "ScopeDescription": "string",
                "ScopeName": "string"
            }
        ],
        "UserPoolId": "string"
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**ResourceServer (p. 192)**

The resource server.

Type: [ResourceServerType (p. 454)](#)

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
See Also

- AWS SDK for Ruby V3
DescribeRiskConfiguration

Describes the risk configuration.

Request Syntax

```
{
    "ClientId": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientId (p. 195)**

The app client ID.

Type: String


Pattern: `[\w+]`

Required: No

**UserPoolId (p. 195)**

The user pool ID.

Type: String


Pattern: `[\w-]+[\d-\w]+`

Required: Yes

Response Syntax

```
{
    "RiskConfiguration": {
        "AccountTakeoverRiskConfiguration": {
            "Actions": {
                "HighAction": {
                    "EventAction": "string",
                    "Notify": boolean
                },
                "LowAction": {
                    "EventAction": "string",
                    "Notify": boolean
                }
            }
        }
    }
```

API Version 2016-04-18
"MediumAction": {
   "EventAction": "string",
   "Notify": boolean
}
},
"NotifyConfiguration": {
   "BlockEmail": {
      "HtmlBody": "string",
      "Subject": "string",
      "TextBody": "string"
   },
   "From": "string",
   "MfaEmail": {
      "HtmlBody": "string",
      "Subject": "string",
      "TextBody": "string"
   },
   "NoActionEmail": {
      "HtmlBody": "string",
      "Subject": "string",
      "TextBody": "string"
   },
   "ReplyTo": "string",
   "SourceArn": "string"
}
},
"ClientId": "string",
"CompromisedCredentialsRiskConfiguration": {
   "Actions": {
      "EventAction": "string"
   },
   "EventFilter": [ "string" ]
},
"LastModifiedDate": number,
"RiskExceptionConfiguration": {
   "BlockedIPRangeList": [ "string" ],
   "SkippedIPRangeList": [ "string" ]
},
"UserPoolId": "string"

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**RiskConfiguration (p. 195)**

The risk configuration.

Type: [RiskConfigurationType (p. 456)](#) object

### Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolAddOnNotEnabledException

This exception is thrown when user pool add-ons aren't enabled.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeUserImportJob

Describes the user import job.

Request Syntax

```
{
  "JobId": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**JobId (p. 198)**

The job ID for the user import job.

Type: String


Pattern: import-[0-9a-zA-Z-]+

Required: Yes

**UserPoolId (p. 198)**

The user pool ID for the user pool that the users are being imported into.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```
{
  "UserImportJob": {
    "CloudWatchLogsRoleArn": "string",
    "CompletionDate": number,
    "CompletionMessage": "string",
    "CreationDate": number,
    "FailedUsers": number,
    "ImportedUsers": number,
    "JobId": "string",
    "JobName": "string",
    "PreSignedUrl": "string",
    "SkippedUsers": number,
    "StartDate": number,
  }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**UserImportJob (p. 198)**

The job object that represents the user import job.

Type: `UserImportJobType (p. 473)` object

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](##).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeUserPool

Returns the configuration information and metadata of the specified user pool.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
• Signing AWS API Requests
• Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

UserPoolId (p. 201)

The user pool ID for the user pool you want to describe.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```
{
  "UserPool": {
    "AccountRecoverySetting": {
      "RecoveryMechanisms": [
        {
          "Name": "string",
          "Priority": number
        }
      ]
    },
    "AdminCreateUserConfig": {
      "AllowAdminCreateUserOnly": boolean,
      "InviteMessageTemplate": {
        "EmailMessage": "string",
      }
    }
  }
}
```
"EmailSubject": "string",
"SMSMessage": "string"
},
"UnusedAccountValidityDays": number,
"AliasAttributes": [ "string" ],
"Arn": "string",
"AutoVerifiedAttributes": [ "string" ],
"CreationDate": number,
"CustomDomain": "string",
"DeletionProtection": "string",
"DeviceConfiguration": {
  "ChallengeRequiredOnNewDevice": boolean,
  "DeviceOnlyRememberedOnUserPrompt": boolean
},
"Domain": "string",
"EmailConfiguration": {
  "ConfigurationSet": "string",
  "EmailSendingAccount": "string",
  "From": "string",
  "ReplyToEmailAddress": "string",
  "SourceArn": "string"
},
"EmailConfigurationFailure": "string",
"EmailVerificationMessage": "string",
"EmailVerificationSubject": "string",
"EstimatedNumberOfUsers": number,
"Id": "string",
"LambdaConfig": {
  "CreateAuthChallenge": "string",
  "CustomEmailSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "CustomMessage": "string",
  "CustomSMSSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "DefineAuthChallenge": "string",
  "KMSKeyID": "string",
  "PostAuthentication": "string",
  "PostConfirmation": "string",
  "PreAuthentication": "string",
  "PreSignUp": "string",
  "PreTokenGeneration": "string",
  "UserMigration": "string",
  "VerifyAuthChallengeResponse": "string"
},
"LastModifiedDate": number,
"MfaConfiguration": "string",
"Name": "string",
"Policies": {
  "PasswordPolicy": {
    "MinimumLength": number,
    "RequireLowercase": boolean,
    "RequireNumbers": boolean,
    "RequireSymbols": boolean,
    "RequireUppercase": boolean,
    "TemporaryPasswordValidityDays": number
  }
},
"SchemaAttributes": [
  {
    "AttributeDataType": "string",
    "DeveloperOnlyAttribute": boolean,
    "DeveloperOnlyName": "string",
    "MaxLength": number,
    "MinLength": number,
    "Name": "string",
    "Required": boolean
  }
]
"Mutable": boolean,
"Name": "string",
"NumberAttributeConstraints": {
  "MaxValue": "string",
  "MinValue": "string"
},
"Required": boolean,
"StringAttributeConstraints": {
  "MaxLength": "string",
  "MinLength": "string"
}
],
"SmsAuthenticationMessage": "string",
"SmsConfiguration": {
  "ExternalId": "string",
  "SmsCallerArn": "string",
  "SnsRegion": "string"
},
"SmsConfigurationFailure": "string",
"SmsVerificationMessage": "string",
"Status": "string",
"UserAttributeUpdateSettings": {
  "AttributesRequireVerificationBeforeUpdate": [ "string" ]
},
"UsernameAttributes": [ "string" ],
"UsernameConfiguration": {
  "CaseSensitive": boolean
},
"UserPoolAddOns": {
  "AdvancedSecurityMode": "string"
},
"UserPoolTags": {
  "string": "string"
},
"VerificationMessageTemplate": {
  "DefaultEmailOption": "string",
  "EmailMessage": "string",
  "EmailMessageByLink": "string",
  "EmailSubject": "string",
  "EmailSubjectByLink": "string",
  "SmsMessage": "string"
}
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserPool (p. 201)

The container of metadata returned by the server to describe the pool.

Type: UserPoolType (p. 490) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).
InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserPoolTaggingException

This exception is thrown when a user pool tag can't be set or updated.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeUserPoolClient

Client method for returning the configuration information and metadata of the specified user pool app client.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
  "ClientId": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

ClientId (p. 205)

The app client ID of the app associated with the user pool.

Type: String


Pattern: `[\w]+`

Required: Yes

UserPoolId (p. 205)

The user pool ID for the user pool you want to describe.

Type: String


Pattern: `[\w-]+[\0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
}
```
"UserPoolClient": {  
  "AccessTokenValidity": number,  
  "AllowedOAuthFlows": [ "string" ],  
  "AllowedOAuthFlowsUserPoolClient": boolean,  
  "AllowedOAuthScopes": [ "string" ],  
  "AnalyticsConfiguration": {  
    "ApplicationArn": "string",  
    "ApplicationId": "string",  
    "ExternalId": "string",  
    "RoleArn": "string",  
    "UserDataShared": boolean  
  },  
  "AuthSessionValidity": number,  
  "CallbackURLs": [ "string" ],  
  "ClientId": "string",  
  "ClientName": "string",  
  "ClientSecret": "string",  
  "CreationDate": number,  
  "DefaultRedirectURI": "string",  
  "EnablePropagateAdditionalUserContextData": boolean,  
  "EnableTokenRevocation": boolean,  
  "ExplicitAuthFlows": [ "string" ],  
  "IdTokenValidity": number,  
  "LastModifiedDate": number,  
  "LogoutURLs": [ "string" ],  
  "PreventUserExistenceErrors": "string",  
  "ReadAttributes": [ "string" ],  
  "RefreshTokenValidity": number,  
  "SupportedIdentityProviders": [ "string" ],  
  "TokenValidityUnits": {  
    "AccessToken": "string",  
    "IdToken": "string",  
    "RefreshToken": "string"  
  },  
  "UserPoolId": "string",  
  "WriteAttributes": [ "string" ]  
},

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserPoolClient (p. 205)

The user pool client from a server response to describe the user pool client.

Type: UserPoolClientType (p. 479) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
DescribeUserPoolDomain

Gets information about a domain.

Request Syntax

```json
{
  "Domain": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**Domain (p. 208)**

The domain string. For custom domains, this is the fully-qualified domain name, such as auth.example.com. For Amazon Cognito prefix domains, this is the prefix alone, such as auth.

Type: String


Pattern: ^[a-z0-9]{0,61}[a-z0-9]$?

Required: Yes

Response Syntax

```json
{
  "DomainDescription": {
    "AWSAccountId": "string",
    "CloudFrontDistribution": "string",
    "CustomDomainConfig": {
      "CertificateArn": "string"
    },
    "Domain": "string",
    "S3Bucket": "string",
    "Status": "string",
    "UserPoolId": "string",
    "Version": "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**DomainDescription (p. 208)**

A domain description object containing information about the domain.
Type: DomainDescriptionType (p. 421) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ForgetDevice

Forgets the specified device.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```
{
    "AccessToken": "string",
    "DeviceKey": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 210)**

A valid access token that Amazon Cognito issued to the user whose registered device you want to forget.

Type: String

Pattern: [A-Za-z0-9-_= .]+

Required: No

**DeviceKey (p. 210)**

The device key.

Type: String


Pattern: \w+_[0-9a-f-]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).
ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
See Also

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ForgotPassword

Calling this API causes a message to be sent to the end user with a confirmation code that is required to change the user's password. For the Username parameter, you can use the username or user alias. The method used to send the confirmation code is sent according to the specified AccountRecoverySetting. For more information, see Recovering User Accounts in the Amazon Cognito Developer Guide. To use the confirmation code for resetting the password, call ConfirmForgotPassword.

If neither a verified phone number nor a verified email exists, this API returns InvalidParameterException. If your app client has a client secret and you don't provide a SECRET_HASH parameter, this API returns NotAuthorizedException.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

**Request Syntax**

```json
{
    "AnalyticsMetadata": {
        "AnalyticsEndpointId": "string"
    },
    "ClientId": "string",
    "ClientMetadata": {
        "string": "string"
    },
    "SecretHash": "string",
    "UserContextData": {
        "EncodedData": "string",
        "IpAddress": "string"
    },
    "Username": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.
AnalyticsMetadata (p. 213)

The Amazon Pinpoint analytics metadata that contributes to your metrics for ForgotPassword calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No

ClientId (p. 213)

The ID of the client associated with the user pool.

Type: String


Pattern: [\w+]+

Required: Yes

ClientMetadata (p. 213)

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ForgotPassword API action, Amazon Cognito invokes any functions that are assigned to the following triggers: pre sign-up, custom message, and user migration. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your ForgotPassword request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see Customizing user pool Workflows with Lambda Triggers in the Amazon Cognito Developer Guide.

Note

When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

SecretHash (p. 213)

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String

Pattern: \[w+=/]+

Required: No

**UserContextData (p. 213)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: UserContextDataType (p. 472) object

Required: No

**Username (p. 213)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: \[p\{L\}p\{M\}p\{S\}p\{N\}p\{P\}\]+

Required: Yes

**Response Syntax**

```json
{
    "CodeDeliveryDetails": {
        "AttributeName": "string",
        "DeliveryMedium": "string",
        "Destination": "string"
    }
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CodeDeliveryDetails (p. 215)**

The code delivery details returned by the server in response to the request to reset a password.

Type: CodeDeliveryDetailsType (p. 410) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 503).

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

API Version 2016-04-18

215
HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidEmailRoleAccessPolicyException**

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

**LimitExceededException**

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

**UnexpectedLambdaException**

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

**UserLambdaValidationException**

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
GetCSVHeader

Gets the header information for the comma-separated value (CSV) file to be used as input for the user import job.

Request Syntax

```json
{
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserPoolId (p. 218)**

The user pool ID for the user pool that the users are to be imported into.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```json
{
    "CSVHeader": [ "string" ],
    "UserPoolId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CSVHeader (p. 218)**

The header information of the CSV file for the user import job.

Type: Array of strings

Length Constraints: Minimum length of 0. Maximum length of 131072.

**UserPoolId (p. 218)**

The user pool ID for the user pool that the users are to be imported into.
Type: String
Pattern: \[\w-\]+\[0-9a-zA-Z\]+

Errors
For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetDevice

Gets the device.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```
{
  "AccessToken": "string",
  "DeviceKey": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AccessToken (p. 220)**

A valid access token that Amazon Cognito issued to the user whose device information you want to request.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: No

**DeviceKey (p. 220)**

The device key.

Type: String


Pattern: [\w-]+_[0-9a-f-]+

Required: Yes

**Response Syntax**

```
{
  "Device": {
    "DeviceAttributes": [
      {
        "Name": "string",
        "Value": "string"
      }
    ],
  }
}
```
"DeviceCreateDate": number,
"DeviceKey": "string",
"DeviceLastAuthenticatedDate": number,
"DeviceLastModifiedDate": number
}

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Device (p. 220)**

The device.

Type: [DeviceType](p. 420) object

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidUserPoolConfigurationException**

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**PasswordResetRequiredException**

This exception is thrown when a password reset is required.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserNotConfirmedException**

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](https://aws.amazon.com/cli/)
- [AWS SDK for .NET](https://aws.amazon.com/sdk-net/)
- [AWS SDK for C++](https://aws.amazon.com/sdk-cpp/)
- [AWS SDK for Go](https://aws.amazon.com/sdk-go/)
- [AWS SDK for Java V2](https://aws.amazon.com/sdk-java/)
- [AWS SDK for JavaScript](https://aws.amazon.com/sdk-js/)
- [AWS SDK for PHP V3](https://aws.amazon.com/sdk-php-v3/)
- [AWS SDK for Python](https://aws.amazon.com/sdk-python/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/sdk-ruby-v3/)
GetGroup

Gets a group.

Calling this action requires developer credentials.

Request Syntax

```
{
   "GroupName": "string",
   "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**GroupName (p. 223)**

The name of the group.

Type: String


Pattern: \p{L}\p{M}\p{S}\p{N}\p{P}+

Required: Yes

**UserPoolId (p. 223)**

The user pool ID for the user pool.

Type: String


Pattern: \w-+\[0-9a-zA-Z]+

Required: Yes

Response Syntax

```
{
   "Group": {
      "CreationDate": number,
      "Description": "string",
      "GroupName": "string",
      "LastModifiedDate": number,
      "Precedence": number,
      "RoleArn": "string",
      "UserPoolId": "string"
   }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Group (p. 223)**

The group object for the group.

Type: [GroupType (p. 430)]

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
GetIdentityProviderByIdentifier

Gets the specified IdP.

**Request Syntax**

```json
{
   "IdpIdentifier": "string",
   "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**IdpIdentifier (p. 226)**

The IdP identifier.

- Type: String
- Pattern: \[\w\s+=.@-]+\n
  Required: Yes

**UserPoolId (p. 226)**

The user pool ID.

- Type: String
- Pattern: \[\w-]+\_[0-9a-zA-Z]+\n
  Required: Yes

**Response Syntax**

```json
{
   "IdentityProvider": {
      "AttributeMapping": {
         "string": "string"
      },
      "CreationDate": number,
      "IdpIdentifiers": [ "string" ],
      "LastModifiedDate": number,
      "ProviderDetails": {
         "string": "string"
      },
      "ProviderName": "string",
      "ProviderType": "string"
   }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

IdentityProvider (p. 226)

The identity provider details.

Type: IdentityProviderType (p. 433) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetLogDeliveryConfiguration

Gets the detailed activity logging configuration for a user pool.

Request Syntax

```
{
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserPoolId (p. 229)**

The ID of the user pool where you want to view detailed activity logging configuration.

Type: String


Pattern: `\w-+\[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
  "LogDeliveryConfiguration": {
    "LogConfigurations": [
      {
        "CloudWatchLogsConfiguration": {
          "LogGroupArn": "string",
          "EventSource": "string",
          "LogLevel": "string"
        }
      },
      "UserPoolId": "string"
    }
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LogDeliveryConfiguration (p. 229)**

The detailed activity logging configuration of the requested user pool.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetSigningCertificate

This method takes a user pool ID, and returns the signing certificate. The issued certificate is valid for 10 years from the date of issue.

Amazon Cognito issues and assigns a new signing certificate annually. This process returns a new value in the response to GetSigningCertificate, but doesn't invalidate the original certificate.

Request Syntax

```
{
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserPoolId (p. 231)**

The user pool ID.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```
{
    "Certificate": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Certificate (p. 231)**

The signing certificate.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetUICustomization

Gets the user interface (UI) Customization information for a particular app client's app UI, if any such information exists for the client. If nothing is set for the particular client, but there is an existing pool level customization (the app clientId is ALL), then that information is returned. If nothing is present, then an empty shape is returned.

Request Syntax

```
{
    "ClientId": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientId (p. 233)**

The client ID for the client app.

Type: String


Pattern: \[\w+\]+

Required: No

**UserPoolId (p. 233)**

The user pool ID for the user pool.

Type: String


Pattern: \[\w-\]_\[0-9a-zA-Z\]+

Required: Yes

Response Syntax

```
{
    "UICustomization": {
        "ClientId": "string",
        "CreationDate": number,
        "CSS": "string",
        "CSSVersion": "string",
        "ImageUrl": "string",
        "LastModifiedDate": number,
        "UserPoolId": "string"
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**UICustomization (p. 233)**

The UI customization information.

Type: [UICustomizationType (p. 469)]

*Errors*

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

- **InternalErrorException**
  
  This exception is thrown when Amazon Cognito encounters an internal error.
  
  HTTP Status Code: 500

- **InvalidParameterException**
  
  This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
  
  HTTP Status Code: 400

- **NotAuthorizedException**
  
  This exception is thrown when a user isn't authorized.
  
  HTTP Status Code: 400

- **ResourceNotFoundException**
  
  This exception is thrown when the Amazon Cognito service can't find the requested resource.
  
  HTTP Status Code: 400

- **TooManyRequestsException**
  
  This exception is thrown when the user has made too many requests for a given operation.
  
  HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
See Also

- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GetUser

Gets the user attributes and metadata for a user.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```
{
  "AccessToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AccessToken (p. 236)**

A non-expired access token for the user whose information you want to query.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

**Response Syntax**

```
{
  "MFAOptions": [
    {
      "AttributeName": "string",
      "DeliveryMedium": "string"
    }
  ],
  "PreferredMfaSetting": "string",
  "UserAttributes": [
    {
      "Name": "string",
      "Value": "string"
    }
  ],
  "UserMFASettingList": [ "string" ],
  "Username": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**MFAOptions (p. 236)**

This response parameter is no longer supported. It provides information only about SMS MFA configurations. It doesn't provide information about time-based one-time password (TOTP) software token MFA configurations. To look up information about either type of MFA configuration, use UserMFASettingList instead.

Type: Array of MFAOptionType (p. 442) objects

**PreferredMfaSetting (p. 236)**

The user's preferred MFA setting.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

**UserAttributes (p. 236)**

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the custom: prefix to the attribute name.

Type: Array of AttributeType (p. 403) objects

**UserMFASettingList (p. 236)**

The MFA options that are activated for the user. The possible values in this list are SMS_MFA and SOFTWARE_TOKEN_MFA.

Type: Array of strings

Length Constraints: Minimum length of 0. Maximum length of 131072.

**Username (p. 236)**

The username of the user that you requested.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\+]n

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 503).

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

PasswordResetRequiredException
This exception is thrown when a password reset is required.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

UserNotConfirmedException
This exception is thrown when a user isn't confirmed successfully.
HTTP Status Code: 400

UserNotFoundException
This exception is thrown when a user isn't found.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
**GetUserAttributeVerificationCode**

Generates a user attribute verification code for the specified attribute name. Sends a message to a user with a code that they must return in a VerifyUserAttribute request.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-identity-iam.html).

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](https://docs.aws.amazon.com/cognito/latest/developerguide/sms-auth.html) in the [Amazon Cognito Developer Guide](https://docs.aws.amazon.com/cognito/latest/developerguide/using-authentication.html).

**Request Syntax**

```json
{
    "AccessToken": "string",
    "AttributeName": "string",
    "ClientMetadata": {
        "string": "string"
    }
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](https://docs.aws.amazon.com/cognito/latest/developerguide/common-parameters.html).

The request accepts the following data in JSON format.

**AccessToken (p. 239)**

A non-expired access token for the user whose attribute verification code you want to generate.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

**AttributeName (p. 239)**

The attribute name returned by the server response to get the user attribute verification code.

Type: String

Amazon Cognito User Pools API Reference
Response Syntax

Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\]+

Required: Yes

**ClientMetadata (p. 239)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the GetUserAttributeVerificationCode API action, Amazon Cognito invokes the function that is assigned to the custom message trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your GetUserAttributeVerificationCode request. In your function code in AWS Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see [Customizing user pool Workflows with Lambda Triggers](#) in the *Amazon Cognito Developer Guide*.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

### Response Syntax

```json
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
      "DeliveryMedium": "string",
      "Destination": "string"
   }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CodeDeliveryDetails (p. 240)**

The code delivery details returned by the server in response to the request to get the user attribute verification code.

---

API Version 2016-04-18

240
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidEmailRoleAccessPolicyException**

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

**LimitExceededException**

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400
NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
• AWS SDK for Ruby V3
GetUserPoolMfaConfig

Gets the user pool multi-factor authentication (MFA) configuration.

Request Syntax

```
{
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**UserPoolId (p. 244)**

The user pool ID.

- Type: String
- Pattern: [\w-]+_[0-9a-zA-Z]+
- Required: Yes

Response Syntax

```
{
  "MfaConfiguration": "string",
  "SmsMfaConfiguration": {
    "SmsAuthenticationMessage": "string",
    "SmsConfiguration": {
      "ExternalId": "string",
      "SnsCallerArn": "string",
      "SnsRegion": "string"
    }
  },
  "SoftwareTokenMfaConfiguration": {
    "Enabled": boolean
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**MfaConfiguration (p. 244)**

The multi-factor authentication (MFA) configuration. Valid values include:
• OFF MFA won't be used for any users.
• ON MFA is required for all users to sign in.
• OPTIONAL MFA will be required only for individual users who have an MFA factor activated.

Type: String  
Valid Values: OFF | ON | OPTIONAL

SmsMfaConfiguration (p. 244)
The SMS text message multi-factor authentication (MFA) configuration.

Type: SmsMfaConfigType (p. 463) object

SoftwareTokenMfaConfiguration (p. 244)
The software token multi-factor authentication (MFA) configuration.

Type: SoftwareTokenMfaConfigType (p. 465) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException  
This exception is thrown when Amazon Cognito encounters an internal error.  
HTTP Status Code: 500

InvalidParameterException  
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.  
HTTP Status Code: 400

NotAuthorizedException  
This exception is thrown when a user isn't authorized.  
HTTP Status Code: 400

ResourceNotFoundException  
This exception is thrown when the Amazon Cognito service can't find the requested resource.  
HTTP Status Code: 400

TooManyRequestsException  
This exception is thrown when the user has made too many requests for a given operation.  
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
See Also

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
GlobalSignOut

Invalidate the identity, access, and refresh tokens that Amazon Cognito issued to a user. Call this operation when your user signs out of your app. This results in the following behavior.

- Amazon Cognito no longer accepts **token-authorized** user operations that you authorize with a signed-out user's access tokens. For more information, see Using the Amazon Cognito user pools API and user pool endpoints.
  
  Amazon Cognito returns an **Access Token has been revoked** error when your app attempts to authorize a user pools API request with a revoked access token that contains the scope aws.cognito.signin.user.admin.

- Amazon Cognito no longer accepts a signed-out user's ID token in a **GetId** request to an identity pool with **ServerSideTokenCheck** enabled for its user pool IdP configuration in **CognitoIdentityProvider**.

- Amazon Cognito no longer accepts a signed-out user's refresh tokens in refresh requests.

Other requests might be valid until your user's token expires.

**Note**

Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Request Syntax**

```
{
  "AccessToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 247)**

A valid access token that Amazon Cognito issued to the user who you want to sign out.

Type: String

Pattern: [A-Za-z0-9-_.]+

Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 503).
ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
InitiateAuth

Initiates sign-in for a user in the Amazon Cognito user directory. You can't sign in a user with a federated IdP with InitiateAuth. For more information, see Adding user pool sign-in through a third party.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

**Request Syntax**

```json
{
  "AnalyticsMetadata": {
    "AnalyticsEndpointId": "string"
  },
  "AuthFlow": "string",
  "AuthParameters": {
    "string": "string"
  },
  "ClientId": "string",
  "ClientMetadata": {
    "string": "string"
  },
  "UserContextData": {
    "EncodedData": "string",
    "IpAddress": "string"
  }
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 250)**

The Amazon Pinpoint analytics metadata that contributes to your metrics for InitiateAuth calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No
**AuthFlow (p. 250)**

The authentication flow for this call to run. The API action will depend on this value. For example:

- **REFRESH_TOKEN_AUTH** takes in a valid refresh token and returns new tokens.
- **USER_SRP_AUTH** takes in USERNAME and SRP_A and returns the SRP variables to be used for next challenge execution.
- **USER_PASSWORD_AUTH** takes in USERNAME and PASSWORD and returns the next challenge or tokens.

Valid values include:

- **USER_SRP_AUTH**: Authentication flow for the Secure Remote Password (SRP) protocol.
- **REFRESH_TOKEN_AUTH/REFRESH_TOKEN**: Authentication flow for refreshing the access token and ID token by supplying a valid refresh token.
- **CUSTOM_AUTH**: Custom authentication flow.
- **USER_PASSWORD_AUTH**: Non-SRP authentication flow; user name and password are passed directly. If a user migration Lambda trigger is set, this flow will invoke the user migration Lambda if it doesn't find the user name in the user pool.

**ADMIN_NO_SRP_AUTH** isn't a valid value.

Type: String

Valid Values: USER_SRP_AUTH | REFRESH_TOKEN_AUTH | REFRESH_TOKEN | CUSTOM_AUTH | ADMIN_NO_SRP_AUTH | USER_PASSWORD_AUTH | ADMIN_USER_PASSWORD_AUTH

Required: Yes

**AuthParameters (p. 250)**

The authentication parameters. These are inputs corresponding to the AuthFlow that you’re invoking. The required values depend on the value of AuthFlow:

- For **USER_SRP_AUTH**: USERNAME (required), SRP_A (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For **USER_PASSWORD_AUTH**: USERNAME (required), PASSWORD (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For **REFRESH_TOKEN_AUTH/REFRESH_TOKEN**: REFRESH_TOKEN (required), SECRET_HASH (required if the app client is configured with a client secret), DEVICE_KEY.
- For **CUSTOM_AUTH**: USERNAME (required), SECRET_HASH (if app client is configured with client secret), DEVICE_KEY. To start the authentication flow with password verification, include ChallengeName: SRP_A and SRP_A: (The SRP_A Value).

For more information about SECRET_HASH, see Computing secret hash values. For information about DEVICE_KEY, see Working with user devices in your user pool.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ClientId (p. 250)**

The app client ID.

Type: String

Pattern: \[\w+\]+
Required: Yes

**ClientMetadata (p. 250)**
A map of custom key-value pairs that you can provide as input for certain custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the InitiateAuth API action, Amazon Cognito invokes the Lambda functions that are specified for various triggers. The ClientMetadata value is passed as input to the functions for only the following triggers:
- Pre signup
- Pre authentication
- User migration

When Amazon Cognito invokes the functions for these triggers, it passes a JSON payload, which the function receives as input. This payload contains a `validationData` attribute, which provides the data that you assigned to the ClientMetadata parameter in your InitiateAuth request. In your function code in Lambda, you can process the `validationData` value to enhance your workflow for your specific needs.

When you use the InitiateAuth API action, Amazon Cognito also invokes the functions for the following triggers, but it doesn't provide the ClientMetadata value as input:
- Post authentication
- Custom message
- Pre token generation
- Create auth challenge
- Define auth challenge

For more information, see [Customizing user pool Workflows with Lambda Triggers](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools-working-with-lambda-triggers.html) in the *Amazon Cognito Developer Guide*.

**Note**
When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:
- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**UserContextData (p. 250)**
Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: [UserContextDataType (p. 472)](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools-api-ref.html#cognito-users-pools-api-ref-request-parameters) object
Response Syntax

```
{
    "AuthenticationResult": {
        "AccessToken": "string",
        "ExpiresIn": number,
        "IdToken": "string",
        "NewDeviceMetadata": {
            "DeviceGroupKey": "string",
            "DeviceKey": "string"
        },
        "RefreshToken": "string",
        "TokenType": "string"
    },
    "ChallengeName": "string",
    "ChallengeParameters": {
        "string": "string"
    },
    "Session": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AuthenticationResult (p. 253)**

The result of the authentication response. This result is only returned if the caller doesn't need to pass another challenge. If the caller does need to pass another challenge before it gets tokens, ChallengeName, ChallengeParameters, and Session are returned.

Type: [AuthenticationResultType](p. 404) object

**ChallengeName (p. 253)**

The name of the challenge that you're responding to with this call. This name is returned in the InitiateAuth response if you must pass another challenge.

Valid values include the following:

**Note**

All of the following challenges require USERNAME and SECRET_HASH (if applicable) in the parameters.

- **SMS_MFA**: Next challenge is to supply an SMS_MFA_CODE, delivered via SMS.
- **PASSWORD_VERIFIER**: Next challenge is to supply PASSWORD_CLAIM_SIGNATURE, PASSWORD_CLAIM_SECRET_BLOCK, and TIMESTAMP after the client-side SRP calculations.
- **CUSTOM_CHALLENGE**: This is returned if your custom authentication flow determines that the user should pass another challenge before tokens are issued.
- **DEVICE_SRP_AUTH**: If device tracking was activated on your user pool and the previous challenges were passed, this challenge is returned so that Amazon Cognito can start tracking this device.
- **DEVICE_PASSWORD_VERIFIER**: Similar to PASSWORD_VERIFIER, but for devices only.
- **NEW_PASSWORD_REQUIRED**: For users who are required to change their passwords after successful first login.
Respond to this challenge with NEW_PASSWORD and any required attributes that Amazon Cognito returned in the requiredAttributes parameter. You can also set values for attributes that aren't required by your user pool and that your app client can write. For more information, see RespondToAuthChallenge.

Note
In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In RespondToAuthChallenge, set a value for any keys that Amazon Cognito returned in the requiredAttributes parameter, then use the UpdateUserAttributes API operation to modify the value of any additional attributes.

- MFA_SETUP: For users who are required to setup an MFA factor before they can sign in. The MFA types activated for the user pool will be listed in the challenge parameters MFAS_CAN_SETUP value.

To set up software token MFA, use the session returned here from InitiateAuth as an input to AssociateSoftwareToken. Use the session returned by VerifySoftwareToken as an input to RespondToAuthChallenge with challenge name MFA_SETUP to complete sign-in. To set up SMS MFA, an administrator should help the user to add a phone number to their account, and then the user should call InitiateAuth again to restart sign-in.

Type: String

Invalid Values: SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED

ChallengeParameters (p. 253)

The challenge parameters. These are returned in the InitiateAuth response if you must pass another challenge. The responses in this parameter should be used to compute inputs to the next call (RespondToAuthChallenge).

All challenges require USERNAME and SECRET_HASH (if applicable).

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Session (p. 253)

The session that should pass both ways in challenge-response calls to the service. If the caller must pass another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next RespondToAuthChallenge API call.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.
HTTP Status Code: 400
**InternalErrorException**
This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500
**InvalidLambdaResponseException**
This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.
HTTP Status Code: 400
**InvalidParameterException**
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400
**InvalidSmsRoleAccessPolicyException**
This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.
HTTP Status Code: 400
**InvalidSmsRoleTrustRelationshipException**
This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.
HTTP Status Code: 400
**InvalidUserPoolConfigurationException**
This exception is thrown when the user pool configuration is not valid.
HTTP Status Code: 400
**NotAuthorizedException**
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400
**PasswordResetRequiredException**
This exception is thrown when a password reset is required.
HTTP Status Code: 400
**ResourceNotFoundException**
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400
**TooManyRequestsException**
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400
**UnexpectedLambdaException**
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.
HTTP Status Code: 400
UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

Examples

Example

The following example signs in the user mytestuser with analytics data, client metadata, and user context data for advanced security.

Sample Request

```
POST / HTTP/1.1
Content-Type: application/x-amz-json-1.1
X-Amz-Target: AWSCognitoIdentityProviderService.InitiateAuth
User-Agent: <UserAgentString>
Accept: */*
Host: cognito-idp.us-east-1.amazonaws.com
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: <PayloadSizeBytes>
{
    "AuthFlow": "USER_PASSWORD_AUTH",
    "ClientId": "1example23456789",
    "AuthParameters": {
        "USERNAME": "mytestuser",
        "PASSWORD": "This-is-my-test-99!",
        "SECRET_HASH": "oT5ZkS8ctnrhYeeGsGTvOzPhoc/Jd1c05fueBWFVmp8="
    },
    "AnalyticsMetadata": {
        "AnalyticsEndpointId": "d70b2ba36a8c4dc5a04a8451a31a1e12"
    },
    "UserContextData": {
        "EncodedData": "AmazonCognitoAdvancedSecurityData_object",
        "IpAddress": "192.0.2.1"
    },
    "ClientMetadata": {
        "MyTestKey": "MyTestValue"
    }
}
```

Sample Response

```
HTTP/1.1 200 OK
```

API Version 2016-04-18
Example

The following example exchanges a refresh token for access and ID tokens.

Sample Request

POST / HTTP/1.1
Content-Type: application/x-amz-json-1.1
X-Amz-Target: AWSCognitoIdentityProviderService.InitiateAuth
User-Agent: <UserAgentString>
Accept: */*
Host: cognito-idp.us-east-1.amazonaws.com
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Content-Length: 1964

{
  "AuthFlow": "REFRESH_TOKEN",
  "ClientId": "1example23456789",
  "AuthParameters": {
    "REFRESH_TOKEN": "eyJ123abcEXAMPLE",
    "SECRET_HASH": "7P85/EXAMPLE"
  }
}

Sample Response

HTTP/1.1 200 OK
Date: Tue, 13 Jun 2023 20:00:59 GMT
Content-Type: application/x-amz-json-1.0
Content-Length: <PayloadSizeBytes>
vary: origin
vary: access-control-request-method
vary: access-control-request-headers
x-amzn-requestid: a1b2c3d4-e5f6-a1b2-c3d4-EXAMPLE11111
Connection: keep-alive

{
  "AuthenticationResult": {
    "AccessToken": "eyJra456defEXAMPLE",
    "ExpiresIn": 3600,
    "IdToken": "eyJra789ghiEXAMPLE",
    "TokenType": "Bearer"
  }
}
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
ListDevices

Lists the sign-in devices that Amazon Cognito has registered to the current user.

**Note**

Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).  

**Request Syntax**

```
{
    "AccessToken": "string",
    "Limit": number,
    "PaginationToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**AccessToken (p. 259)**

A valid access token that Amazon Cognito issued to the user whose list of devices you want to view.

- Type: String
- Pattern: [A-Za-z0-9-_=.]+  
- Required: Yes

**Limit (p. 259)**

The limit of the device request.

- Type: Integer
- Valid Range: Minimum value of 0. Maximum value of 60.
- Required: No

**PaginationToken (p. 259)**

This API operation returns a limited number of results. The pagination token is an identifier that you can present in an additional API request with the same parameters. When you include the pagination token, Amazon Cognito returns the next set of items after the current list. Subsequent requests return a new pagination token. By use of this token, you can paginate through the full list of items.

- Type: String
- Length Constraints: Minimum length of 1.
- Pattern: \S+  
- Required: No
Response Syntax

```
{
    "Devices": [
        {
            "DeviceAttributes": [
                {
                    "Name": "string",
                    "Value": "string"
                }
            ],
            "DeviceCreateDate": number,
            "DeviceKey": "string",
            "DeviceLastAuthenticatedDate": number,
            "DeviceLastModifiedDate": number
        }
    ],
    "PaginationToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Devices (p. 260)**

The devices returned in the list devices response.

Type: Array of **DeviceType (p. 420)** objects

**PaginationToken (p. 260)**

The identifier that Amazon Cognito returned with the previous request to this operation. When you include a pagination token in your request, Amazon Cognito returns the next set of items in the list. By use of this token, you can paginate through the full list of items.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

Errors

For information about the errors that are common to all actions, see **Common Errors (p. 503)**.

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListGroups

Lists the groups associated with a user pool.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
  "Limit": number,
  "NextToken": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

Limit (p. 263)

The limit of the request to list groups.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

NextToken (p. 263)

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String


Pattern: [\S]+

Required: No

UserPoolId (p. 263)

The user pool ID for the user pool.

Type: String
Response Syntax

```json
{
    "Groups": [
        {
            "CreationDate": number,
            "Description": "string",
            "GroupName": "string",
            "LastModifiedDate": number,
            "Precedence": number,
            "RoleArn": "string",
            "UserPoolId": "string"
        }
    ],
    "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Groups** *(p. 264)*

The group objects for the groups.

Type: Array of **GroupType** *(p. 430)* objects

**NextToken** *(p. 264)*

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String


Pattern: [\S]+

Errors

For information about the errors that are common to all actions, see **Common Errors** *(p. 503)*.

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListIdentityProviders

Lists information about all IdPs for a user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```json
{
    "MaxResults": number,
    "NextToken": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**MaxResults (p. 266)**

The maximum number of IdPs to return.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

**NextToken (p. 266)**

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: \[\S\]+

Required: No

**UserPoolId (p. 266)**

The user pool ID.

Type: String

Response Syntax

```json
{
  "NextToken": "string",
  "Providers": [
    {
      "CreationDate": number,
      "LastModifiedDate": number,
      "ProviderName": "string",
      "ProviderType": "string"
    }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 267)**

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

**Providers (p. 267)**

A list of IdP objects.

Type: Array of ProviderDescription (p. 450) objects

Array Members: Minimum number of 0 items. Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

API Version 2016-04-18

267
**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
ListResourceServers

Lists the resource servers for a user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "MaxResults": number,
  "NextToken": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**MaxResults (p. 269)**

The maximum number of resource servers to return.

Type: Integer


Required: No

**NextToken (p. 269)**

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

Required: No

**UserPoolId (p. 269)**

The user pool ID for the user pool.

Type: String

Pattern: \[\w\-]+[0-9a-zA-Z]+

Required: Yes

Response Syntax

```
{
    "NextToken": "string",
    "ResourceServers": [
    {
        "Identifier": "string",
        "Name": "string",
        "Scopes": [
        {
            "ScopeDescription": "string",
            "ScopeName": "string"
        }
    ],
    "UserPoolId": "string"
    }
]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 270)**

A pagination token.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

**ResourceServers (p. 270)**

The resource servers.

Type: Array of **ResourceServerType (p. 454)** objects

Errors

For information about the errors that are common to all actions, see **Common Errors (p. 503)**.

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListTagsForResource

Lists the tags that are assigned to an Amazon Cognito user pool.

A tag is a label that you can apply to user pools to categorize and manage them in different ways, such as by purpose, owner, environment, or other criteria.

You can use this action up to 10 times per second, per account.

Request Syntax

```json
{
    "ResourceArn": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters](#). The request accepts the following data in JSON format.

**ResourceArn (p. 272)**

The Amazon Resource Name (ARN) of the user pool that the tags are assigned to.

- Type: String
- Pattern: `arn:[:\w+=/,.@-]+[:\w+=/,.@-]*:[0-9]+[:\w+=/,.@-]+(:[:\w+=/,.@-]+)?(:[:\w+=/,.@-]+)?
- Required: Yes

Response Syntax

```json
{
    "Tags": {
        "string": "string"
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**Tags (p. 272)**

The tags that are assigned to the user pool.

- Type: String to string map
Key Length Constraints: Minimum length of 1. Maximum length of 128.
Value Length Constraints: Minimum length of 0. Maximum length of 256.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListUserImportJobs

Lists user import jobs for a user pool.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
• Signing AWS API Requests
• Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```
{
    "MaxResults": number,
    "PaginationToken": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

MaxResults (p. 274)

The maximum number of import jobs you want the request to return.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: Yes

PaginationToken (p. 274)

This API operation returns a limited number of results. The pagination token is an identifier that you can present in an additional API request with the same parameters. When you include the pagination token, Amazon Cognito returns the next set of items after the current list. Subsequent requests return a new pagination token. By use of this token, you can paginate through the full list of items.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

Required: No

UserPoolId (p. 274)

The user pool ID for the user pool that the users are being imported into.

Type: String

Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```json
{
    "PaginationToken": "string",
    "UserImportJobs": [
        {
            "CloudWatchLogsRoleArn": "string",
            "CompletionDate": number,
            "CompletionMessage": "string",
            "CreationDate": number,
            "FailedUsers": number,
            "ImportedUsers": number,
            "JobId": "string",
            "JobName": "string",
            "PreSignedUrl": "string",
            "SkippedUsers": number,
            "StartDate": number,
            "Status": "string",
            "UserPoolId": "string"
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**PaginationToken (p. 275)**

The identifier that Amazon Cognito returned with the previous request to this operation. When you include a pagination token in your request, Amazon Cognito returns the next set of items in the list. By use of this token, you can paginate through the full list of items.

Type: String

Length Constraints: Minimum length of 1.

Pattern: [\S]+

**UserImportJobs (p. 275)**

The user import jobs.

Type: Array of UserImportJobType (p. 473) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).
InternalErrorException
This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListUserPoolClients

Lists the clients that have been created for the specified user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "MaxResults": "number",
  "NextToken": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**MaxResults (p. 277)**

The maximum number of results you want the request to return when listing the user pool clients.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60.

Required: No

**NextToken (p. 277)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String


Pattern: \[\S\]+

Required: No

**UserPoolId (p. 277)**

The user pool ID for the user pool where you want to list user pool clients.

Type: String

Response Syntax

```json
{
  "NextToken": "string",
  "UserPoolClients": [
    {
      "ClientId": "string",
      "ClientName": "string",
      "UserPoolId": "string"
    }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 278)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String


Pattern: \[\S\]+

**UserPoolClients (p. 278)**

The user pool clients in the response that lists user pool clients.

Type: Array of UserPoolClientDescription (p. 478) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
ListUserPools

Lists the user pools associated with an AWS account.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```
{
    "MaxResults": number,
    "NextToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

- **MaxResults** *(p. 280)*
  
  The maximum number of results you want the request to return when listing the user pools.

  Type: Integer

  Valid Range: Minimum value of 1. Maximum value of 60.

  Required: Yes

- **NextToken** *(p. 280)*
  
  An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

  Type: String

  Length Constraints: Minimum length of 1.

  Pattern: \[\S]+

  Required: No

**Response Syntax**

```
{
    "NextToken": "string",
    "UserPools": []
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 280)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Length Constraints: Minimum length of 1.

Pattern: \[\S\]+

**UserPools (p. 280)**

The user pools from the response to list users.

Type: Array of UserPoolDescriptionType (p. 487) objects

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListUsers

Lists users and their basic details in a user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```json
{
    "AttributesToGet": [ "string" ],
    "Filter": "string",
    "Limit": number,
    "PaginationToken": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AttributesToGet (p. 283)**

A JSON array of user attribute names, for example `given_name`, that you want Amazon Cognito to include in the response for each user. When you don't provide an AttributesToGet parameter, Amazon Cognito returns all attributes for each user.

Use AttributesToGet with required attributes in your user pool, or in conjunction with Filter. Amazon Cognito returns an error if not all users in the results have set a value for the attribute you request. Attributes that you can't filter on, including custom attributes, must have a value set in every user profile before an AttributesToGet parameter returns results.

Type: Array of strings


Pattern: `\[\p{L}\p{M}\p{S}\p{N}\p{P}\]+`

Required: No

**Filter (p. 283)**

A filter string of the form "AttributeName Filter-Type "AttributeValue"". Quotation marks within the filter string must be escaped using the backslash (`\`) character. For example, "family_name = " Reddy" ".

API Version 2016-04-18
283
Request Parameters

- **AttributeName**: The name of the attribute to search for. You can only search for one attribute at a time.
- **Filter-Type**: For an exact match, use =, for example, "given_name = "Jon"". For a prefix ("starts with") match, use ^=, for example, "given_name ^= "Jon""
- **AttributeValue**: The attribute value that must be matched for each user.

If the filter string is empty, ListUsers returns all users in the user pool.

You can only search for the following standard attributes:

- username (case-sensitive)
- email
- phone_number
- name
- given_name
- family_name
- preferred_username
- cognito:user_status (called **Status** in the Console) (case-insensitive)
- status (called **Enabled** in the Console) (case-sensitive)
- sub

Custom attributes aren't searchable.

**Note**

You can also list users with a client-side filter. The server-side filter matches no more than one attribute. For an advanced search, use a client-side filter with the --query parameter of the list-users action in the AWS CLI. When you use a client-side filter, ListUsers returns a paginated list of zero or more users. You can receive multiple pages in a row with zero results. Repeat the query with each pagination token that is returned until you receive a null pagination token value, and then review the combined result.


For more information, see [Searching for Users Using the ListUsers API](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools-list-users.html) and [Examples of Using the ListUsers API](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools-list-users-examples.html) in the Amazon Cognito Developer Guide.

**Type**: String

**Length Constraints**: Maximum length of 256.

**Required**: No

**Limit (p. 283)**

Maximum number of users to be returned.

**Type**: Integer

**Valid Range**: Minimum value of 0. Maximum value of 60.

**Required**: No

**PaginationToken (p. 283)**

This API operation returns a limited number of results. The pagination token is an identifier that you can present in an additional API request with the same parameters. When you include the pagination token
token, Amazon Cognito returns the next set of items after the current list. Subsequent requests return a new pagination token. By use of this token, you can paginate through the full list of items.

Type: String
Length Constraints: Minimum length of 1.
Pattern: \[\S\]+
Required: No

UserPoolId (p. 283)

The user pool ID for the user pool on which the search should be performed.
Type: String
Pattern: \[\w\-\]+_[0-9a-zA-Z]+
Required: Yes

Response Syntax

```
{
  "PaginationToken": "string",
  "Users": [
    {
      "Attributes": [
        {
          "Name": "string",
          "Value": "string"
        }
      ],
      "Enabled": boolean,
      "MFAOptions": [
        {
          "AttributeName": "string",
          "DeliveryMedium": "string"
        }
      ],
      "UserCreateDate": number,
      "UserLastModifiedDate": number,
      "Username": "string",
      "UserStatus": "string"
    }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

PaginationToken (p. 285)

The identifier that Amazon Cognito returned with the previous request to this operation. When you include a pagination token in your request, Amazon Cognito returns the next set of items in the list. By use of this token, you can paginate through the full list of items.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

Examples

Example

This request submits a value for all possible parameters for ListUsers. By iterating the PaginationToken, you can page through and collect all users in a user pool.
Sample Request

POST HTTP/1.1
Host: cognito-idp.us-east-1.amazonaws.com
X-Amz-Date: 20230613T200059Z
Accept-Encoding: identity
X-Amz-Target: AWSCognitoIdentityProviderService.ListUsers
User-Agent: <UserAgentString>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>, Signature=<Signature>
Content-Length: <PayloadSizeBytes>

{
    "AttributesToGet": [
        "email",
        "sub"
    ],
    "Filter": "\"email\"^\"testuser\"",
    "Limit": 3,
    "PaginationToken": "abcd1234EXAMPLE",
    "UserPoolId": "us-east-1_EXAMPLE"
}

Sample Response

HTTP/1.1 200 OK
Date: Tue, 13 Jun 2023 20:00:59 GMT
Content-Type: application/x-amz-json-1.0
Content-Length: <PayloadSizeBytes>

{
    "PaginationToken": "efgh5678EXAMPLE",
    "Users": [
    {
        "Attributes": [
            {
                "Name": "sub",
                "Value": "eaad0219-2117-439f-8d46-4db20e59268f"
            },
            {
                "Name": "email",
                "Value": "testuser@example.com"
            }
        ],
        "Enabled": true,
        "UserCreateDate": 1682955829.578,
        "UserLastModifiedDate": 1689030181.63,
        "UserStatus": "CONFIRMED",
        "Username": "testuser"
    },
    {
        "Attributes": [
            {
                "Name": "sub",
                "Value": "3b994cfd-0b07-4581-be46-3c82f9a70c90"
            },
            {
                "Name": "email",
                "Value": "testuser@example.com"
            }
        ],
        "Enabled": true,
        "UserCreateDate": 1682955829.578,
        "UserLastModifiedDate": 1689030181.63,
        "UserStatus": "CONFIRMED",
        "Username": "testuser"
    }
]
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ListUsersInGroup

Lists the users in the specified group.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "GroupName": "string",
  "Limit": number,
  "NextToken": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**GroupName (p. 289)**

The name of the group.

Type: String


Pattern: \[p{L}\p{M}\p{S}\p{N}\p{P}]*

Required: Yes

**Limit (p. 289)**

The maximum number of users that you want to retrieve before pagination.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 60.

Required: No

**NextToken (p. 289)**

An identifier that was returned from the previous call to this operation, which can be used to return the next set of items in the list.

Type: String

Pattern: \[\S\]+

Required: No

**UserPoolId (p. 289)**

The user pool ID for the user pool.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: Yes

### Response Syntax

```
{
    "NextToken": "string",
    "Users": [ 
        { 
            "Attributes": [ 
                { 
                    "Name": "string",
                    "Value": "string"
                } 
            ],
            "Enabled": boolean,
            "MFAOptions": [ 
                { 
                    "AttributeName": "string",
                    "DeliveryMedium": "string"
                } 
            ],
            "UserCreateDate": number,
            "UserLastModifiedDate": number,
            "UserName": "string",
            "UserStatus": "string"
        } 
    ]
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 290)**

An identifier that you can use in a later request to return the next set of items in the list.

Type: String


Pattern: \[\S\]+
Users (p. 290)

A list of users in the group, and their attributes.

Type: Array of UserType (p. 497) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
ResendConfirmationCode

Resends the confirmation (for confirmation of registration) to a specific user in the user pool.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with [Amazon Pinpoint](#). Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In **sandbox mode**, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](#) in the Amazon Cognito Developer Guide.

**Request Syntax**

```
{
    "AnalyticsMetadata": {
        "AnalyticsEndpointId": "string"
    },
    "ClientId": "string",
    "ClientMetadata": {
        "string": "string"
    },
    "SecretHash": "string",
    "UserContextData": {
        "EncodedData": "string",
        "IpAddress": "string"
    },
    "Username": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 292)**

The Amazon Pinpoint analytics metadata that contributes to your metrics for ResendConfirmationCode calls.

Type: [AnalyticsMetadataType (p. 402)](#) object

Required: No
Request Parameters

**ClientId (p. 292)**

The ID of the client associated with the user pool.

Type: String


Pattern: `[\w+]`+

Required: Yes

**ClientMetadata (p. 292)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the ResendConfirmationCode API action, Amazon Cognito invokes the function that is assigned to the custom message trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a clientMetadata attribute, which provides the data that you assigned to the ClientMetadata parameter in your ResendConfirmationCode request. In your function code in Lambda, you can process the clientMetadata value to enhance your workflow for your specific needs.

For more information, see [Customizing user pool Workflows with Lambda Triggers](#) in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won't do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**SecretHash (p. 292)**

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String


Pattern: `[\w+=/\-]+`+

Required: No

**UserContextData (p. 292)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.
Type: UserContextDataType (p. 472) object

Required: No

**Username (p. 292)**

The username of the user that you want to query or modify. The value of this parameter is typically your user's username, but it can be any of their alias attributes. If username isn't an alias attribute in your user pool, you can also use their sub in this request.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

### Response Syntax

```
{
   "CodeDeliveryDetails": {
      "AttributeName": "string",
      "DeliveryMedium": "string",
      "Destination": "string"
   }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CodeDeliveryDetails (p. 294)**

The code delivery details returned by the server in response to the request to resend the confirmation code.

Type: CodeDeliveryDetailsType (p. 410) object

### Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400
InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

LimitExceededException

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.
HTTP Status Code: 400

**UserLambdaValidationException**

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
RespondToAuthChallenge

Some API operations in a user pool generate a challenge, like a prompt for an MFA code, for device authentication that bypasses MFA, or for a custom authentication challenge. A RespondToAuthChallenge API request provides the answer to that challenge, like a code or a secure remote password (SRP). The parameters of a response to an authentication challenge vary with the type of challenge.

For more information about custom authentication challenges, see Custom authentication challenge Lambda triggers.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Note
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Request Syntax

```json
{
    "AnalyticsMetadata": {
        "AnalyticsEndpointId": "string"
    },
    "ChallengeName": "string",
    "ChallengeResponses": {
        "string": "string"
    },
    "ClientId": "string",
    "ClientMetadata": {
        "string": "string"
    },
    "Session": "string",
    "UserContextData": {
        "EncodedData": "string",
        "IpAddress": "string"
    }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.
AnalyticsMetadata (p. 297)

The Amazon Pinpoint analytics metadata that contributes to your metrics for RespondToAuthChallenge calls.

Type: AnalyticsMetadataType (p. 402) object

Required: No

ChallengeName (p. 297)

The challenge name. For more information, see InitiateAuth.

ADMIN_NO_SRP_AUTH isn't a valid value.

Type: String

Valid Values: SMS_MFA | SOFTWARE_TOKEN_MFA | SELECT_MFA_TYPE | MFA_SETUP | PASSWORD_VERIFIER | CUSTOM_CHALLENGE | DEVICE_SRP_AUTH | DEVICE_PASSWORD_VERIFIER | ADMIN_NO_SRP_AUTH | NEW_PASSWORD_REQUIRED

Required: Yes

ChallengeResponses (p. 297)

The responses to the challenge that you received in the previous request. Each challenge has its own required response parameters. The following examples are partial JSON request bodies that highlight challenge-response parameters.

**Important**
You must provide a SECRET_HASH parameter in all challenge responses to an app client that has a client secret.

**SMS_MFA**

```
"ChallengeName": "SMS_MFA", "ChallengeResponses": {
"SMS_MFA_CODE": "[SMS_code]", "USERNAME": "[username]"
}
```

**PASSWORD_VERIFIER**

```
"ChallengeName": "PASSWORD_VERIFIER", "ChallengeResponses": {
"PASSWORD_CLAIM_SIGNATURE": "[claim_signature]",
"PASSWORD_CLAIM_SECRET_BLOCK": "[secret_block]", "TIMESTAMP": [timestamp], "USERNAME": "[username]"
}
```

Add "DEVICE_KEY" when you sign in with a remembered device.

**CUSTOM_CHALLENGE**

```
"ChallengeName": "CUSTOM_CHALLENGE", "ChallengeResponses": {
"USERNAME": "[username]", "ANSWER": "[challenge_answer]"
}
```

Add "DEVICE_KEY" when you sign in with a remembered device.

**NEW_PASSWORD_REQUIRED**

```
"ChallengeName": "NEW_PASSWORD_REQUIRED", "ChallengeResponses": {
"NEW_PASSWORD": "[new_password]", "USERNAME": "[username]"
}
```

To set any required attributes that InitiateAuth returned in an requiredAttributes parameter, add "userAttributes.[attribute_name]": "[attribute_value]". This parameter can also set values for writable attributes that aren't required by your user pool.

**Note**
In a NEW_PASSWORD_REQUIRED challenge response, you can't modify a required attribute that already has a value. In RespondToAuthChallenge, set a value for any
keys that Amazon Cognito returned in the `requiredAttributes` parameter, then use the `UpdateUserAttributes` API operation to modify the value of any additional attributes.

**SOFTWARE_TOKEN_MFA**

```json
"ChallengeName": "SOFTWARE_TOKEN_MFA", "ChallengeResponses":
{"USERNAME": "[username]", "SOFTWARE_TOKEN_MFA_CODE":
[authenticator_code]}
```

**DEVICE_SRP_AUTH**

```json
"ChallengeName": "DEVICE_SRP_AUTH", "ChallengeResponses": {
"USERNAME": "[username]", "DEVICE_KEY": "[device_key]", "SRP_A": "[srp_a]"
```

**DEVICE_PASSWORD_VERIFIER**

```json
"ChallengeName": "DEVICE_PASSWORD_VERIFIER", "ChallengeResponses":
{"DEVICE_KEY": "[device_key]", "PASSWORD_CLAIM_SIGNATURE":
[claim_signature]", "PASSWORD_CLAIM_SECRET_BLOCK": "[secret_block]",
"TIMESTAMP": [timestamp], "USERNAME": "[username]"
```

**MFA_SETUP**

```json
"ChallengeName": "MFA_SETUP", "ChallengeResponses": {
"USERNAME": "[username]"}, "SESSION": "[Session ID from VerifySoftwareToken]"
```

**SELECT_MFA_TYPE**

```json
"ChallengeName": "SELECT_MFA_TYPE", "ChallengeResponses": {
"USERNAME": "[username]", "ANSWER": "[SMS_MFA or SOFTWARE_TOKEN_MFA]"
```

For more information about `SECRET_HASH`, see [Computing secret hash values](#). For information about `DEVICE_KEY`, see [Working with user devices in your user pool](#).

**ClientID (p. 297)**

The app client ID.

Type: String


Pattern: `[\w+]`+

Required: Yes

**ClientMetadata (p. 297)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the `RespondToAuthChallenge` API action, Amazon Cognito invokes any functions that are assigned to the following triggers: `post authentication`, `pre token generation`, `define auth challenge`, `create auth challenge`, and `verify auth challenge`. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains
a `clientMetadata` attribute, which provides the data that you assigned to the `ClientMetadata` parameter in your `RespondToAuthChallenge` request. In your function code in AWS Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see [Customizing user pool Workflows with Lambda Triggers](https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-pools-trigger-lambda.html) in the *Amazon Cognito Developer Guide*.

**Note**
When you use the `ClientMetadata` parameter, remember that Amazon Cognito won’t do the following:

- Store the `ClientMetadata` value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the `ClientMetadata` parameter serves no purpose.
- Validate the `ClientMetadata` value.
- Encrypt the `ClientMetadata` value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**Session (p. 297)**

The session that should be passed both ways in challenge-response calls to the service. If `InitiateAuth` or `RespondToAuthChallenge` API call determines that the caller must pass another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String


Required: No

**UserContextData (p. 297)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: `UserContextDataType (p. 472)` object

Required: No

### Response Syntax

```json
{
  "AuthenticationResult": {
    "AccessToken": "string",
    "ExpiresIn": number,
    "IdToken": "string",
    "NewDeviceMetadata": {
      "DeviceGroupKey": "string",
      "DeviceKey": "string"
    },
    "RefreshToken": "string",
  }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**AuthenticationResult (p. 300)**

The result returned by the server in response to the request to respond to the authentication challenge.

Type: `AuthenticationResultType` (p. 404) object

**ChallengeName (p. 300)**

The challenge name. For more information, see [InitiateAuth](#).

Type: String

Valid Values: `SMS_MFA` | `SOFTWARE_TOKEN_MFA` | `SELECT_MFA_TYPE` | `MFA_SETUP` | `PASSWORD_VERIFIER` | `CUSTOM_CHALLENGE` | `DEVICE_SRP_AUTH` | `DEVICE_PASSWORD_VERIFIER` | `ADMIN_NO_SRP_AUTH` | `NEW_PASSWORD_REQUIRED`

**ChallengeParameters (p. 300)**

The challenge parameters. For more information, see [InitiateAuth](#).

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

**Session (p. 300)**

The session that should be passed both ways in challenge-response calls to the service. If the caller must pass another challenge, they return a session with other challenge parameters. This session should be passed as it is to the next `RespondToAuthChallenge` API call.

Type: String


Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**AliasExistsException**

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception
indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

**CodeMismatchException**

This exception is thrown if the provided code doesn't match what the server was expecting.

HTTP Status Code: 400

**ExpiredCodeException**

This exception is thrown if a code has expired.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidLambdaResponseException**

This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidPasswordException**

This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

**InvalidUserPoolConfigurationException**

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400
MFAMethodNotFoundException

This exception is thrown when Amazon Cognito can't find a multi-factor authentication (MFA) method.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException

This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException

This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
RevokeToken

RevokeToken

Revokes all of the access tokens generated by, and at the same time as, the specified refresh token. After a token is revoked, you can't use the revoked token to access Amazon Cognito user APIs, or to authorize access to your resource server.

Note

Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```
{
   "ClientId": "string",
   "ClientSecret": "string",
   "Token": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ClientId (p. 305)**

The client ID for the token that you want to revoke.

Type: String


Pattern: `[\w+]*`

Required: Yes

**ClientSecret (p. 305)**

The secret for the client ID. This is required only if the client ID has a secret.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `[\w+]*`

Required: No

**Token (p. 305)**

The refresh token that you want to revoke.

Type: String

Pattern: `[A-Za-z0-9-_.]+`

Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnauthorizedException

Exception that is thrown when the request isn't authorized. This can happen due to an invalid access token in the request.

HTTP Status Code: 400

UnsupportedOperationException

Exception that is thrown when you attempt to perform an operation that isn't enabled for the user pool client.

HTTP Status Code: 400

UnsupportedTokenTypeException

Exception that is thrown when an unsupported token is passed to an operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
SetLogDeliveryConfiguration

Sets up or modifies the detailed activity logging configuration of a user pool.

Request Syntax

```json
{
    "LogConfigurations": [
        {
            "CloudWatchLogsConfiguration": {
                "LogGroupArn": "string",
                "EventSource": "string",
                "LogLevel": "string"
            }
        },
    ],
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**LogConfigurations (p. 308)**

A collection of all of the detailed activity logging configurations for a user pool.

Type: Array of LogConfigurationType (p. 439) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: Yes

**UserPoolId (p. 308)**

The ID of the user pool where you want to configure detailed activity logging.

Type: String


Pattern: `\[\w-]+@[0-9a-zA-Z]+`

Required: Yes

Response Syntax

```json
{
    "LogDeliveryConfiguration": {
        "LogConfigurations": [
            {
                "CloudWatchLogsConfiguration": {
                    "LogGroupArn": "string"
                }
            }
        ]
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**LogDeliveryConfiguration** *(p. 308)*

The detailed activity logging configuration that you applied to the requested user pool.

Type: *LogDeliveryConfigurationType* *(p. 440)* object

Errors

For information about the errors that are common to all actions, see *Common Errors* *(p. 503)*.

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for .NET](https://docs.aws.amazon.com/sdkfornet/latest/reference/com.amazonaws.cognito.idp.html)
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
SetRiskConfiguration

Configures actions on detected risks. To delete the risk configuration for UserPoolId or ClientId, pass null values for all four configuration types.

To activate Amazon Cognito advanced security features, update the user pool to include the UserPoolAddOns key AdvancedSecurityMode.

See UpdateUserPool (p. 366).

Request Syntax

```
[  
  "AccountTakeoverRiskConfiguration": {  
    "Actions": {  
      "HighAction": {  
        "EventAction": "string",
        "Notify": boolean
      },  
      "LowAction": {  
        "EventAction": "string",
        "Notify": boolean
      },  
      "MediumAction": {  
        "EventAction": "string",
        "Notify": boolean
      }
    },  
    "NotifyConfiguration": {  
      "BlockEmail": {  
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },  
      "From": "string",
      "MfaEmail": {  
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },  
      "NoActionEmail": {  
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },  
      "ReplyTo": "string",
      "SourceArn": "string"
    }
  },  
  "ClientId": "string",
  "CompromisedCredentialsRiskConfiguration": {  
    "Actions": {  
      "EventAction": "string"
    },  
    "EventFilter": [ "string" ]
  },  
  "RiskExceptionConfiguration": {  
    "BlockedIPRangeList": [ "string" ],
    "SkippedIPRangeList": [ "string" ]
  },  
  "UserPoolId": "string"
]
```
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccountTakeoverRiskConfiguration (p. 311)**

The account takeover risk configuration.

Type: AccountTakeoverRiskConfigurationType (p. 398) object

Required: No

**ClientId (p. 311)**

The app client ID. If ClientId is null, then the risk configuration is mapped to userPoolId. When the client ID is null, the same risk configuration is applied to all the clients in the userPool.

Otherwise, ClientId is mapped to the client. When the client ID isn't null, the user pool configuration is overridden and the risk configuration for the client is used instead.

Type: String


Pattern: \[\w+\]+

Required: No

**CompromisedCredentialsRiskConfiguration (p. 311)**

The compromised credentials risk configuration.

Type: CompromisedCredentialsRiskConfigurationType (p. 412) object

Required: No

**RiskExceptionConfiguration (p. 311)**

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 458) object

Required: No

**UserPoolId (p. 311)**

The user pool ID.

Type: String


Pattern: \[\w\-]+[0-9a-zA-Z]+

Required: Yes

Response Syntax

{ }
"RiskConfiguration": {
  "AccountTakeoverRiskConfiguration": {
    "Actions": {
      "HighAction": {
        "EventAction": "string",
        "Notify": boolean
      },
      "LowAction": {
        "EventAction": "string",
        "Notify": boolean
      },
      "MediumAction": {
        "EventAction": "string",
        "Notify": boolean
      }
    },
    "NotifyConfiguration": {
      "BlockEmail": {
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },
      "From": "string",
      "MFAEmail": {
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },
      "NoActionEmail": {
        "HtmlBody": "string",
        "Subject": "string",
        "TextBody": "string"
      },
      "ReplyTo": "string",
      "SourceArn": "string"
    },
    "ClientId": "string",
    "CompromisedCredentialsRiskConfiguration": {
      "Actions": {
        "EventAction": "string"
      },
      "EventFilter": [ "string" ]
    },
    "LastModifiedDate": number,
    "RiskExceptionConfiguration": {
      "BlockedIPRangeList": [ "string" ],
      "SkippedIPRangeList": [ "string" ]
    },
    "UserPoolId": "string"
  }
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

RiskConfiguration (p. 312)

The risk configuration.

Type: RiskConfigurationType (p. 456) object
Errors

For information about the errors that are common to all actions, see [Common Errors](#).

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidEmailRoleAccessPolicyException**

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**UserPoolAddOnNotEnabledException**

This exception is thrown when user pool add-ons aren't enabled.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
See Also

- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
SetUICustomization

Sets the user interface (UI) customization information for a user pool's built-in app UI.

You can specify app UI customization settings for a single client (with a specific clientId) or for all clients (by setting the clientId to ALL). If you specify ALL, the default configuration is used for every client that has no previously set UI customization. If you specify UI customization settings for a particular client, it will no longer return to the ALL configuration.

**Note**
To use this API, your user pool must have a domain associated with it. Otherwise, there is no place to host the app's pages, and the service will throw an error.

**Request Syntax**

```json
{
   "ClientId": "string",
   "CSS": "string",
   "ImageFile": blob,
   "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](p. 501).

The request accepts the following data in JSON format.

**ClientId (p. 316)**

The client ID for the client app.

Type: String


Pattern: \w+

Required: No

**CSS (p. 316)**

The CSS values in the UI customization.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**ImageFile (p. 316)**

The uploaded logo image for the UI customization.

Type: Base64-encoded binary data object

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No
**UserPoolId (p. 316)**

The user pool ID for the user pool.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

**Response Syntax**

```
{
    "UICustomization": {
        "ClientId": "string",
        "CreationDate": number,
        "CSS": "string",
        "CSSVersion": "string",
        "ImageUrl": "string",
        "LastModifiedDate": number,
        "UserPoolId": "string"
    }
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**UICustomization (p. 317)**

The UI customization information.

Type: `UICustomizationType (p. 469)` object

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)](Common-Errors).  

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
SetUserMFAPreference

Set the user's multi-factor authentication (MFA) method preference, including which MFA factors are activated and if any are preferred. Only one factor can be set as preferred. The preferred MFA factor will be used to authenticate a user if multiple factors are activated. If multiple options are activated and no preference is set, a challenge to choose an MFA option will be returned during sign-in. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts unless device tracking is turned on and the device has been trusted. If you want MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```
{
    "AccessToken": "string",
    "SMSMfaSettings": {
        "Enabled": boolean,
        "PreferredMfa": boolean
    },
    "SoftwareTokenMfaSettings": {
        "Enabled": boolean,
        "PreferredMfa": boolean
    }
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**AccessToken (p. 319)**

A valid access token that Amazon Cognito issued to the user whose MFA preference you want to set.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

**SMSMfaSettings (p. 319)**

The SMS text message multi-factor authentication (MFA) settings.

Type: [SMSMfaSettingsType](#) object

Required: No

**SoftwareTokenMfaSettings (p. 319)**

The time-based one-time password (TOTP) software token MFA settings.
Type: `SoftwareTokenMfaSettingsType (p. 466)` object

Required: No

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
- **AWS Command Line Interface**
- **AWS SDK for .NET**
- **AWS SDK for C++**
- **AWS SDK for Go**
- **AWS SDK for Java V2**
- **AWS SDK for JavaScript**
- **AWS SDK for PHP V3**
- **AWS SDK for Python**
- **AWS SDK for Ruby V3**
SetUserPoolMfaConfig

Sets the user pool multi-factor authentication (MFA) configuration.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see [SMS message settings for Amazon Cognito user pools](https://docs.aws.amazon.com/cognito/latest/developerguide/sms-message-settings.html) in the Amazon Cognito Developer Guide.

**Request Syntax**

```json
{
  "MfaConfiguration": "string",
  "SmsMfaConfiguration": {
    "SmsAuthenticationMessage": "string",
    "SmsConfiguration": {
      "ExternalId": "string",
      "SnsCallerArn": "string",
      "SnsRegion": "string"
    }
  },
  "SoftwareTokenMfaConfiguration": {
    "Enabled": boolean
  },
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](https://docs.aws.amazon.com/cognito/latest/developerguide/common-parameters.html).

The request accepts the following data in JSON format.

**MfaConfiguration (p. 322)**

The MFA configuration. If you set the MfaConfiguration value to 'ON', only users who have set up an MFA factor can sign in. To learn more, see [Adding Multi-Factor Authentication (MFA) to a user pool](https://docs.aws.amazon.com/cognito/latest/developerguide/mfa-configure.html).

Valid values include:

- **OFF** MFA won't be used for any users.
- **ON** MFA is required for all users to sign in.
- **OPTIONAL** MFA will be required only for individual users who have an MFA factor activated.

Type: String

Valid Values: OFF | ON | OPTIONAL

Required: No
**SmsMfaConfiguration (p. 322)**

The SMS text message MFA configuration.

Type: `SmsMfaConfigType (p. 463)` object

Required: No

**SoftwareTokenMfaConfiguration (p. 322)**

The software token MFA configuration.

Type: `SoftwareTokenMfaConfigType (p. 465)` object

Required: No

**UserPoolId (p. 322)**

The user pool ID.

Type: String


Pattern: `\[\w-]+_[0-9a-zA-Z]+`

Required: Yes

**Response Syntax**

```
{
   "MfaConfiguration": "string",
   "SmsMfaConfiguration": {
      "SmsAuthenticationMessage": "string",
      "SmsConfiguration": {
         "ExternalId": "string",
         "SnsCallerArn": "string",
         "SnsRegion": "string"
      }
   },
   "SoftwareTokenMfaConfiguration": {
      "Enabled": boolean
   }
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**MfaConfiguration (p. 323)**

The MFA configuration. Valid values include:

- **OFF** MFA won't be used for any users.
- **ON** MFA is required for all users to sign in.
- **OPTIONAL** MFA will be required only for individual users who have an MFA factor enabled.

Type: String

Valid Values: OFF | ON | OPTIONAL
**SmsMfaConfiguration (p. 323)**

The SMS text message MFA configuration.

Type: `SmsMfaConfigType (p. 463)` object

**SoftwareTokenMfaConfiguration (p. 323)**

The software token MFA configuration.

Type: `SoftwareTokenMfaConfigType (p. 465)` object

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**InvalidSmsRoleAccessPolicyException**

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

**InvalidSmsRoleTrustRelationshipException**

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust `cognito-idp.amazonaws.com` or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
SetUserSettings

This action is no longer supported. You can use it to configure only SMS MFA. You can't use it to configure time-based one-time password (TOTP) software token MFA. To configure either type of MFA, use SetUserMFAPreference instead.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```
{
    "AccessToken": "string",
    "MFAOptions": [
        {
            "AttributeName": "string",
            "DeliveryMedium": "string"
        }
    ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 326)**
A valid access token that Amazon Cognito issued to the user whose user settings you want to configure.

Type: String
Pattern: [A-Za-z0-9-_.]+
Required: Yes

**MFAOptions (p. 326)**
You can use this parameter only to set an SMS configuration that uses SMS for delivery.

Type: Array of MFAOptionType (p. 442) objects
Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).
**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**PasswordResetRequiredException**

This exception is thrown when a password reset is required.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**UserNotConfirmedException**

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

---

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](https://aws.amazon.com/cli/)
- [AWS SDK for .NET](https://aws.amazon.com/sdk-for-net/)
- [AWS SDK for C++](https://aws.amazon.com/sdk-for-cpp/)
- [AWS SDK for Go](https://aws.amazon.com/sdk-for-go/)
- [AWS SDK for Java V2](https://aws.amazon.com/sdk-for-java/)
- [AWS SDK for JavaScript](https://aws.amazon.com/sdk-for-javascript/)
- [AWS SDK for PHP V3](https://aws.amazon.com/sdk-for-php/)
- [AWS SDK for Python](https://aws.amazon.com/sdk-for-python/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/sdk-for-ruby/)

---

API Version 2016-04-18

327
SignUp

Registers the user in the specified user pool and creates a user name, password, and user attributes.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Note
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Request Syntax

```
{
  "AnalyticsMetadata": {
    "AnalyticsEndpointId": "string"
  },
  "ClientId": "string",
  "ClientMetadata": {
    "string": "string"
  },
  "Password": "string",
  "SecretHash": "string",
  "UserAttributes": [
    {
      "Name": "string",
      "Value": "string"
    }
  ],
  "UserContextData": {
    "EncodedData": "string",
    "IpAddress": "string"
  },
  "Username": "string",
  "ValidationData": [
    {
      "Name": "string",
      "Value": "string"
    }
  ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).
The request accepts the following data in JSON format.

**AnalyticsMetadata (p. 329)**

The Amazon Pinpoint analytics metadata that contributes to your metrics for SignUp calls.

Type: [AnalyticsMetadataType](p. 402) object

Required: No

**ClientId (p. 329)**

The ID of the client associated with the user pool.

Type: String


Pattern: \[\w+]+

Required: Yes

**ClientMetadata (p. 329)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action triggers.

You create custom workflows by assigning AWS Lambda functions to user pool triggers. When you use the SignUp API action, Amazon Cognito invokes any functions that are assigned to the following triggers: pre sign-up, custom message, and post confirmation. When Amazon Cognito invokes any of these functions, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the `ClientMetadata` parameter in your SignUp request. In your function code in Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see [Customizing user pool Workflows with Lambda Triggers](#) in the Amazon Cognito Developer Guide.

**Note**

When you use the ClientMetadata parameter, remember that Amazon Cognito won’t do the following:

- Store the ClientMetadata value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn’t include triggers, the ClientMetadata parameter serves no purpose.
- Validate the ClientMetadata value.
- Encrypt the ClientMetadata value. Don’t use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**Password (p. 329)**

The password of the user you want to register.

Type: String

Length Constraints: Maximum length of 256.
Request Parameters

Pattern: [\S]+  
Required: Yes

**SecretHash (p. 329)**

A keyed-hash message authentication code (HMAC) calculated using the secret key of a user pool client and username plus the client ID in the message.

Type: String


Pattern: [\w+=/]+  
Required: No

**UserAttributes (p. 329)**

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the custom: prefix to the attribute name.

Type: Array of **AttributeType (p. 403)** objects  
Required: No

**UserContextData (p. 329)**

Contextual data about your user session, such as the device fingerprint, IP address, or location. Amazon Cognito advanced security evaluates the risk of an authentication event based on the context that your app generates and passes to Amazon Cognito when it makes API requests.

Type: **UserContextDataType (p. 472)** object  
Required: No

**Username (p. 329)**

The username of the user that you want to sign up. The value of this parameter is typically a username, but can be any alias attribute in your user pool.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+  
Required: Yes

**ValidationData (p. 329)**

Temporary user attributes that contribute to the outcomes of your pre sign-up Lambda trigger. This set of key-value pairs are for custom validation of information that you collect from your users but don't need to retain.

Your Lambda function can analyze this additional data and act on it. Your function might perform external API operations like logging user attributes and validation data to Amazon CloudWatch Logs. Validation data might also affect the response that your function returns to Amazon Cognito, like automatically confirming the user if they sign up from within your network.

For more information about the pre sign-up Lambda trigger, see Pre sign-up Lambda trigger.

Type: Array of **AttributeType (p. 403)** objects  
Required: No
Response Syntax

```
{
  "CodeDeliveryDetails": {
    "AttributeName": "string",
    "DeliveryMedium": "string",
    "Destination": "string"
  },
  "UserConfirmed": boolean,
  "UserSub": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**CodeDeliveryDetails (p. 332)**

The code delivery details returned by the server response to the user registration request.

Type: CodeDeliveryDetailsType (p. 410) object

**UserConfirmed (p. 332)**

A response from the server indicating that a user registration has been confirmed.

Type: Boolean

**UserSub (p. 332)**

The UUID of the authenticated user. This isn't the same as username.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidEmailRoleAccessPolicyException
This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidLambdaResponseException
This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.

HTTP Status Code: 400

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidPasswordException
This exception is thrown when Amazon Cognito encounters an invalid password.

HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException
This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException
This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnexpectedLambdaException
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.

HTTP Status Code: 400

UserLambdaValidationException
This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.
HTTP Status Code: 400

**UsernameExistsException**

This exception is thrown when Amazon Cognito encounters a user name that already exists in the user pool.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
StartUserImportJob

Starts the user import.

Request Syntax

```json
{
  "JobId": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**JobId (p. 335)**

The job ID for the user import job.

Type: String


Pattern: import-[0-9a-zA-Z-]+

Required: Yes

**UserPoolId (p. 335)**

The user pool ID for the user pool that the users are being imported into.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

Response Syntax

```json
{
  "UserImportJob": {
    "CloudWatchLogsRoleArn": "string",
    "CompletionDate": number,
    "CompletionMessage": "string",
    "CreationDate": number,
    "FailedUsers": number,
    "ImportedUsers": number,
    "JobId": "string",
    "JobName": "string",
    "PreSignedUrl": "string",
    "SkippedUsers": number,
    "StartDate": number,
    "JobStatus": "string",
    "StatusMessage": "string"
  }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserImportJob (p. 335)

The job object that represents the user import job.

Type: UserImportJobType (p. 473) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
StopUserImportJob

Stops the user import job.

Request Syntax

```json
{
    "JobId": "string",
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**JobId (p. 338)**

The job ID for the user import job.

- Type: String
- Pattern: import-[0-9a-zA-Z-]+
- Required: Yes

**UserPoolId (p. 338)**

The user pool ID for the user pool that the users are being imported into.

- Type: String
- Pattern: []w-]+_[0-9a-zA-Z]+
- Required: Yes

Response Syntax

```json
{
    "UserImportJob": {
        "CloudWatchLogsRoleArn": "string",
        "CompletionDate": number,
        "CompletionMessage": "string",
        "CreationDate": number,
        "FailedUsers": number,
        "ImportedUsers": number,
        "JobId": "string",
        "JobName": "string",
        "PreSignedUrl": "string",
        "SkippedUsers": number,
        "StartDate": number,
    }
}
```
"Status": "string",
"UserPoolId": "string"
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

UserImportJob (p. 338)

The job object that represents the user import job.

Type: UserImportJobType (p. 473) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PreconditionNotMetException

This exception is thrown when a precondition is not met.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
TagResource

Assigns a set of tags to an Amazon Cognito user pool. A tag is a label that you can use to categorize and manage user pools in different ways, such as by purpose, owner, environment, or other criteria.

Each tag consists of a key and value, both of which you define. A key is a general category for more specific values. For example, if you have two versions of a user pool, one for testing and another for production, you might assign an Environment tag key to both user pools. The value of this key might be Test for one user pool, and Production for the other.

Tags are useful for cost tracking and access control. You can activate your tags so that they appear on the Billing and Cost Management console, where you can track the costs associated with your user pools. In an AWS Identity and Access Management policy, you can constrain permissions for user pools based on specific tags or tag values.

You can use this action up to 5 times per second, per account. A user pool can have as many as 50 tags.

Request Syntax

```json
{
  "ResourceArn": "string",
  "Tags": {
    "string": "string"
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ResourceArn (p. 341)**

The Amazon Resource Name (ARN) of the user pool to assign the tags to.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?

Required: Yes

**Tags (p. 341)**

The tags to assign to the user pool.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UntagResource

Removes the specified tags from an Amazon Cognito user pool. You can use this action up to 5 times per second, per account.

Request Syntax

```json
{
   "ResourceArn": "string",
   "TagKeys": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**ResourceArn (p. 343)**

The Amazon Resource Name (ARN) of the user pool that the tags are assigned to.

Type: String


Pattern: `arn::[\w+=/,.@-]+:[\w+=/,.@-]+:[(\[\w+=/,.@-]*+\[\w+=/,.@-]+(\[\w+=/,.@-]+?):[0-9]+\[\w+=/,.@-]+(\[\w+=/,.@-]+?):\[\w+=/,.@-]+(\[\w+=/,.@-]+)?]`

Required: Yes

**TagKeys (p. 343)**

The keys of the tags to remove from the user pool.

Type: Array of strings


Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateAuthEventFeedback

Provides the feedback for an authentication event, whether it was from a valid user or not. This feedback is used for improving the risk evaluation decision for the user pool as part of Amazon Cognito advanced security.

**Note**

Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```
{
    "EventId": "string",
    "FeedbackToken": "string",
    "FeedbackValue": "string",
    "Username": "string",
    "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**EventId (p. 345)**

The event ID.

Type: String


Pattern: \[\w+-\]+

Required: Yes

**FeedbackToken (p. 345)**

The feedback token.

Type: String

Pattern: [A-Za-z0-9-_\.]+

Required: Yes

**FeedbackValue (p. 345)**

The authentication event feedback value. When you provide a `FeedbackValue` value of `valid`, you tell Amazon Cognito that you trust a user session where Amazon Cognito has evaluated some level of risk. When you provide a `FeedbackValue` value of `invalid`, you tell Amazon Cognito that you don't trust a user session, or you don't believe that Amazon Cognito evaluated a high-enough risk level.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
TooManyRequestsException
   This exception is thrown when the user has made too many requests for a given operation.
   HTTP Status Code: 400

UserNotFoundException
   This exception is thrown when a user isn't found.
   HTTP Status Code: 400

UserPoolAddOnNotEnabledException
   This exception is thrown when user pool add-ons aren't enabled.
   HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateDeviceStatus

Updates the device status.

Note
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```json
{
    "AccessToken": "string",
    "DeviceKey": "string",
    "DeviceRememberedStatus": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 348)**

A valid access token that Amazon Cognito issued to the user whose device status you want to update.

Type: String

Pattern: [A-Za-z0-9-_=.]+

Required: Yes

**DeviceKey (p. 348)**

The device key.

Type: String


Pattern: [\w-]+[0-9a-f-]+

Required: Yes

**DeviceRememberedStatus (p. 348)**

The status of whether a device is remembered.

Type: String

Valid Values: remembered | not_remembered

Required: No
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalServerError

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateGroup

Updates the specified group with the specified attributes.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

**Request Syntax**

```
{
  "Description": "string",
  "GroupName": "string",
  "Precedence": number,
  "RoleArn": "string",
  "UserPoolId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#).

The request accepts the following data in JSON format.

**Description (p. 351)**

A string containing the new description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

**GroupName (p. 351)**

The name of the group.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]*

Required: Yes

**Precedence (p. 351)**

The new precedence value for the group. For more information about this parameter, see [CreateGroup](#).

Type: Integer
Valid Range: Minimum value of 0.

Required: No

**RoleArn (p. 351)**

The new role Amazon Resource Name (ARN) for the group. This is used for setting the `cognito:roles` and `cognito:preferred_role` claims in the token.

Type: String


Pattern: `arn:\[[w+=/,.@-]+:\[[w+=/,.@-]*):[0-9]+:\[[w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?`?

Required: No

**UserPoolId (p. 351)**

The user pool ID for the user pool.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

**Response Syntax**

```json
{
"Group": {
    "CreationDate": "number",
    "Description": "string",
    "GroupName": "string",
    "LastModifiedDate": "number",
    "Precedence": "number",
    "RoleArn": "string",
    "UserPoolId": "string"
}
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Group (p. 352)**

The group object for the group.

Type: **GroupType (p. 430)** object

**Errors**

For information about the errors that are common to all actions, see **Common Errors (p. 503)**.
**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateIdentityProvider

Updates IdP information for a user pool.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

### Request Syntax

```json
{
    "AttributeMapping": {
        "string": "string"
    },
    "IdpIdentifiers": [ "string" ],
    "ProviderDetails": {
        "string": "string"
    },
    "ProviderName": "string",
    "UserPoolId": "string"
}
```

### Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**AttributeMapping (p. 354)**

The IdP attribute mapping to be changed.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 32.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**IdpIdentifiers (p. 354)**

A list of IdP identifiers.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 50 items.


Pattern: `[\w\s+=.@-]+`
Required: No

ProviderDetails (p. 354)

The IdP details to be updated, such as MetadataURL and MetadataFile.
Type: String to string map
Key Length Constraints: Minimum length of 0. Maximum length of 131072.
Value Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

ProviderName (p. 354)

The IdP name.
Type: String
Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+
Required: Yes

UserPoolId (p. 354)

The user pool ID.
Type: String
Pattern: [\w-]+_[0-9a-zA-Z]+
Required: Yes

Response Syntax

```
{
    "IdentityProvider": {
        "AttributeMapping": {
            "string": "string"
        },
        "CreationDate": number,
        "IdpIdentifiers": [ "string" ],
        "LastModifiedDate": number,
        "ProviderDetails": {
            "string": "string"
        },
        "ProviderName": "string",
        "ProviderType": "string",
        "UserPoolId": "string"
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.
IdentityProvider (p. 355)

The identity provider details.

Type: identityProviderType (p. 433) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ConcurrentModificationException

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UnsupportedIdentityProviderException

This exception is thrown when the specified identifier isn't supported.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V3
UpdateResourceServer

Updates the name and scopes of resource server. All other fields are read-only.

**Important**
If you don't provide a value for an attribute, it is set to the default value.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more
- [Signing AWS API Requests](#)
- [Using the Amazon Cognito user pools API and user pool endpoints](#)

Request Syntax

```json
{
    "Identifier": "string",
    "Name": "string",
    "Scopes": [
        {
            "ScopeDescription": "string",
            "ScopeName": "string"
        }
    ],
    "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see [Common Parameters (p. 501)](#).

The request accepts the following data in JSON format.

**Identifier (p. 358)**

The identifier for the resource server.

- **Type:** String
- **Length Constraints:** Minimum length of 1. Maximum length of 256.
- **Pattern:** `[^\x21\x23-\x5B\x5D-\x7E]+`

- **Required:** Yes

**Name (p. 358)**

The name of the resource server.

- **Type:** String
- **Length Constraints:** Minimum length of 1. Maximum length of 256.
- **Pattern:** `[^\s+=,.@-]+`

API Version 2016-04-18

358
Required: Yes

**Scopes (p. 358)**

The scope values to be set for the resource server.

Type: Array of `ResourceServerScopeType (p. 453)` objects

Array Members: Maximum number of 100 items.

Required: No

**UserPoolId (p. 358)**

The user pool ID for the user pool.

Type: String


Pattern: `[\w-]+_[0-9a-zA-Z]+`

Required: Yes

### Response Syntax

```json
{
  "ResourceServer": {
    "Identifies": "string",
    "Name": "string",
    "Scopes": [
      {
        "ScopeDescription": "string",
        "ScopeName": "string"
      }
    ],
    "UserPoolId": "string"
  }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**ResourceServer (p. 359)**

The resource server.

Type: `ResourceServerType (p. 454)` object

### Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)](#).

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.
HTTP Status Code: 500
InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
NotAuthorizedException
This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400
TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateUserAttributes

With this operation, your users can update one or more of their attributes with their own credentials. You authorize this API request with the user’s access token. To delete an attribute from your user, submit the attribute in your API request with a blank value. Custom attribute values in this request must include the custom: prefix.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can’t use IAM credentials to authorize requests, and you can’t grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

**Request Syntax**

```json
{
   "AccessToken": "string",
   "ClientMetadata": {
      "string": "string"
   },
   "UserAttributes": [
      {
         "Name": "string",
         "Value": "string"
      }
   ]
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 361)**

A valid access token that Amazon Cognito issued to the user whose user attributes you want to update.

Type: String

Pattern: [A-Za-z0-9-_.]+
Required: Yes

**ClientMetadata (p. 361)**

A map of custom key-value pairs that you can provide as input for any custom workflows that this action initiates.

You create custom workflows by assigning Lambda functions to user pool triggers. When you use the UpdateUserAttributes API action, Amazon Cognito invokes the function that is assigned to the custom message trigger. When Amazon Cognito invokes this function, it passes a JSON payload, which the function receives as input. This payload contains a `clientMetadata` attribute, which provides the data that you assigned to the `ClientMetadata` parameter in your UpdateUserAttributes request. In your function code in Lambda, you can process the `clientMetadata` value to enhance your workflow for your specific needs.

For more information, see [Customizing user pool Workflows with Lambda Triggers](https://docs.aws.amazon.com/cognito/userguide/user-pools-custom-flow.html) in the *Amazon Cognito Developer Guide*.

**Note**

When you use the `ClientMetadata` parameter, remember that Amazon Cognito won't do the following:

- Store the `ClientMetadata` value. This data is available only to AWS Lambda triggers that are assigned to a user pool to support custom workflows. If your user pool configuration doesn't include triggers, the `ClientMetadata` parameter serves no purpose.
- Validate the `ClientMetadata` value.
- Encrypt the `ClientMetadata` value. Don't use Amazon Cognito to provide sensitive information.

Type: String to string map

Key Length Constraints: Minimum length of 0. Maximum length of 131072.

Value Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**UserAttributes (p. 361)**

An array of name-value pairs representing user attributes.

For custom attributes, you must prepend the `custom:` prefix to the attribute name.

If you have set an attribute to require verification before Amazon Cognito updates its value, this request doesn't immediately update the value of that attribute. After your user receives and responds to a verification message to verify the new value, Amazon Cognito updates the attribute value. Your user can sign in and receive messages with the original attribute value until they verify the new value.

Type: Array of `AttributeType (p. 403)` objects

Required: Yes

**Response Syntax**

```json
{
    "CodeDeliveryDetailsList": [
        {
            "AttributeName": "string",
            "AttributeType": "string",
            "DeliveryAddress": "string",
            "DeliveryPhone": "string",
            "DefaultDeliveryCountry": "string",
            "DefaultDeliveryRegion": "string",
            "DefaultDeliveryType": "string",
            "Email": "string",
            "EmailVerificationAttributes": [
                "string"
            ],
            "Phone": "string",
            "PreferredDeliveryEmail": "string",
            "PreferredDeliveryPhone": "string",
            "PreferredDeliveryType": "string",
            "Reason": "string",
            "ResponseCode": "string",
            "ResponseDescription": "string",
            "ResponseMetadata": "string",
            "UserValidationStatus": "string"
        }
    ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**CodeDeliveryDetailsList (p. 362)**

The code delivery details list from the server for the request to update user attributes.

Type: Array of [CodeDeliveryDetailsType (p. 410)] objects

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**AliasExistsException**

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

**CodeDeliveryFailureException**

This exception is thrown when a verification code fails to deliver successfully.

HTTP Status Code: 400

**CodeMismatchException**

This exception is thrown if the provided code doesn't match what the server was expecting.

HTTP Status Code: 400

**ExpiredCodeException**

This exception is thrown if a code has expired.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500
InvalidEmailRoleAccessPolicyException
This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.
HTTP Status Code: 400

InvalidLambdaResponseException
This exception is thrown when Amazon Cognito encounters an invalid AWS Lambda response.
HTTP Status Code: 400

InvalidParameterException
This exception is thrown when the Amazon Cognito service encounters an invalid parameter.
HTTP Status Code: 400

InvalidSmsRoleAccessPolicyException
This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.
HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException
This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.
HTTP Status Code: 400

NotAuthorizedException
This exception is thrown when a user isn't authorized.
HTTP Status Code: 400

PasswordResetRequiredException
This exception is thrown when a password reset is required.
HTTP Status Code: 400

ResourceNotFoundException
This exception is thrown when the Amazon Cognito service can't find the requested resource.
HTTP Status Code: 400

TooManyRequestsException
This exception is thrown when the user has made too many requests for a given operation.
HTTP Status Code: 400

UnexpectedLambdaException
This exception is thrown when Amazon Cognito encounters an unexpected exception with AWS Lambda.
HTTP Status Code: 400

UserLambdaValidationException
This exception is thrown when the Amazon Cognito service encounters a user validation exception with the AWS Lambda service.
HTTP Status Code: 400

**UserNotConfirmedException**

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

**UserNotFoundException**

This exception is thrown when a user isn't found.

HTTP Status Code: 400

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS Command Line Interface](#)
- [AWS SDK for .NET](#)
- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for JavaScript](#)
- [AWS SDK for PHP V3](#)
- [AWS SDK for Python](#)
- [AWS SDK for Ruby V3](#)
UpdateUserPool

**Note**
This action might generate an SMS text message. Starting June 1, 2021, US telecom carriers require you to register an origination phone number before you can send SMS messages to US phone numbers. If you use SMS text messages in Amazon Cognito, you must register a phone number with Amazon Pinpoint. Amazon Cognito uses the registered number automatically. Otherwise, Amazon Cognito users who must receive SMS messages might not be able to sign up, activate their accounts, or sign in.

If you have never used SMS text messages with Amazon Cognito or any other AWS service, Amazon Simple Notification Service might place your account in the SMS sandbox. In sandbox mode, you can send messages only to verified phone numbers. After you test your app while in the sandbox environment, you can move out of the sandbox and into production. For more information, see SMS message settings for Amazon Cognito user pools in the Amazon Cognito Developer Guide.

Updates the specified user pool with the specified attributes. You can get a list of the current user pool settings using DescribeUserPool.

**Important**
If you don't provide a value for an attribute, Amazon Cognito sets it to its default value.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```
{
    "AccountRecoverySetting": {
        "RecoveryMechanisms": [
            {
                "Name": "string",
                "Priority": number
            }
        ],
    },
    "AdminCreateUserConfig": {
        "AllowAdminCreateUserOnly": boolean,
        "InviteMessageTemplate": {
            "EmailMessage": "string",
            "EmailSubject": "string",
            "SMSMessage": "string"
        },
        "UnusedAccountValidityDays": number
    },
    "AutoVerifiedAttributes": [ "string" ],
    "DeletionProtection": "string",
    "DeviceConfiguration": {
        "ChallengeRequiredOnNewDevice": boolean,
        "DeviceOnlyRememberedOnUserPrompt": boolean
    }
}
```

API Version 2016-04-18
Amazon Cognito User Pools API Reference
Request Syntax

```
"EmailConfiguration": {
  "ConfigurationSet": "string",
  "EmailSendingAccount": "string",
  "From": "string",
  "ReplyToEmailAddress": "string",
  "SourceArn": "string"
},
"EmailVerificationMessage": "string",
"EmailVerificationSubject": "string",
"LambdaConfig": {
  "CreateAuthChallenge": "string",
  "CustomEmailSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "CustomMessage": "string",
  "CustomSMSSender": {
    "LambdaArn": "string",
    "LambdaVersion": "string"
  },
  "DefineAuthChallenge": "string",
  "KMSKeyID": "string",
  "PostAuthentication": "string",
  "PostConfirmation": "string",
  "PreAuthentication": "string",
  "PreSignUp": "string",
  "PreTokenGeneration": "string",
  "UserMigration": "string",
  "VerifyAuthChallengeResponse": "string"
},
"MfaConfiguration": "string",
"Policies": {
  "PasswordPolicy": {
    "MinimumLength": number,
    "RequireLowercase": boolean,
    "RequireNumbers": boolean,
    "RequireSymbols": boolean,
    "RequireUppercase": boolean,
    "TemporaryPasswordValidityDays": number
  }
},
"SmsAuthenticationMessage": "string",
"SmsConfiguration": {
  "ExternalId": "string",
  "SnsCallerArn": "string",
  "SnsRegion": "string"
},
"SmsVerificationMessage": "string",
"UserAttributeUpdateSettings": {
  "AttributesRequireVerificationBeforeUpdate": [ "string" ]
},
"UserPoolAddOns": {
  "AdvancedSecurityMode": "string"
},
"UserPoolId": "string",
"UserPoolTags": {
  "string": "string"
},
"VerificationMessageTemplate": {
  "DefaultEmailOption": "string",
  "EmailMessage": "string",
  "EmailMessageByLink": "string",
  "EmailSubject": "string",
  "EmailSubjectByLink": "string",
  "SmsMessage": "string"
}
```
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccountRecoverySetting (p. 366)**

The available verified method a user can use to recover their password when they call 
ForgotPassword. You can use this setting to define a preferred method when a user has more 
than one method available. With this setting, SMS doesn't qualify for a valid password recovery 
mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of 
this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS 
is preferred through email.

Type: AccountRecoverySettingType (p. 395) object

Required: No

**AdminCreateUserConfig (p. 366)**

The configuration for AdminCreateUser requests.

Type: AdminCreateUserConfigType (p. 399) object

Required: No

**AutoVerifiedAttributes (p. 366)**

The attributes that are automatically verified when Amazon Cognito requests to update user pools.

Type: Array of strings

Valid Values: phone_number | email

Required: No

**DeletionProtection (p. 366)**

When active, DeletionProtection prevents accidental deletion of your user pool. Before you can 
delete a user pool that you have protected against deletion, you must deactivate this feature.

When you try to delete a protected user pool in a DeleteUserPool API request, Amazon Cognito 
returns an InvalidParameterException error. To delete a protected user pool, send a new 
DeleteUserPool request after you deactivate deletion protection in an UpdateUserPool API 
request.

Type: String

Valid Values: ACTIVE | INACTIVE

Required: No

**DeviceConfiguration (p. 366)**

The device-remembering configuration for a user pool. A null value indicates that you have 
deactivated device remembering in your user pool.

**Note**

When you provide a value for any DeviceConfiguration field, you activate the Amazon 
Cognito device-remembering feature.
EmailConfiguration (p. 366)

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for email invitation and verification messages from your user pool.

EmailVerificationMessage (p. 366)

This parameter is no longer used. See VerificationMessageTemplateType.

EmailVerificationSubject (p. 366)

This parameter is no longer used. See VerificationMessageTemplateType.

LambdaConfig (p. 366)

The Lambda configuration information from the request to update the user pool.

MfaConfiguration (p. 366)

Possible values include:

- OFF - MFA tokens aren't required and can't be specified during user registration.
- ON - MFA tokens are required for all user registrations. You can only specify ON when you're initially creating a user pool. You can use the SetUserPoolMfaConfig API operation to turn MFA "ON" for existing user pools.
- OPTIONAL - Users have the option when registering to create an MFA token.

Policies (p. 366)

A container with the policies you want to update in a user pool.
Type: `UserPoolPolicyType (p. 489)` object
Required: No

**SmsAuthenticationMessage (p. 366)**

The contents of the SMS authentication message.
Type: String
Pattern: `.\{####\}.\`
Required: No

**SmsConfiguration (p. 366)**

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.
Type: `SmsConfigurationType (p. 461)` object
Required: No

**SmsVerificationMessage (p. 366)**

This parameter is no longer used. See `VerificationMessageTemplateType`.
Type: String
Pattern: `.\{####\}.\`
Required: No

**UserAttributeUpdateSettings (p. 366)**

The settings for updates to user attributes. These settings include the property `AttributesRequireVerificationBeforeUpdate`, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see `Verifying updates to email addresses and phone numbers`.
Type: `UserAttributeUpdateSettingsType (p. 471)` object
Required: No

**UserPoolAddOns (p. 366)**

User pool add-ons. Contains settings for activation of advanced security features. To log user security information but take no action, set to `AUDIT`. To configure automatic security responses to risky traffic to your user pool, set to `ENFORCED`.

For more information, see `Adding advanced security to a user pool`.
Type: `UserPoolAddOnsType (p. 477)` object
Required: No

**UserPoolId (p. 366)**

The user pool ID for the user pool you want to update.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

ConcurrentModificationException

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidEmailRoleAccessPolicyException

This exception is thrown when Amazon Cognito isn't allowed to use your email identity. HTTP status code: 400.

HTTP Status Code: 400

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
InvalidSmsRoleAccessPolicyException

This exception is returned when the role provided for SMS configuration doesn't have permission to publish using Amazon SNS.

HTTP Status Code: 400

InvalidSmsRoleTrustRelationshipException

This exception is thrown when the trust relationship is not valid for the role provided for SMS configuration. This can happen if you don't trust cognito-idp.amazonaws.com or the external ID provided in the role does not match what is provided in the SMS configuration for the user pool.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserImportInProgressException

This exception is thrown when you're trying to modify a user pool while a user import job is in progress for that pool.

HTTP Status Code: 400

UserPoolTaggingException

This exception is thrown when a user pool tag can't be set or updated.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateUserPoolClient

Updates the specified user pool app client with the specified attributes. You can get a list of the current user pool app client settings using `DescribeUserPoolClient`.

**Important**
If you don't provide a value for an attribute, Amazon Cognito sets it to its default value.

You can also use this operation to enable token revocation for user pool clients. For more information about revoking tokens, see `RevokeToken`.

**Note**
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

**Learn more**
- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

**Request Syntax**

```json
{
  "AccessTokenValidity": number,
  "AllowedOAuthFlows": [ "string" ],
  "AllowedOAuthFlowsUserPoolClient": boolean,
  "AllowedOAuthScopes": [ "string" ],
  "AnalyticsConfiguration": {
    "ApplicationArn": "string",
    "ApplicationId": "string",
    "ExternalId": "string",
    "RoleArn": "string",
    "UserDataShared": boolean
  },
  "AuthSessionValidity": number,
  "CallbackURLs": [ "string" ],
  "ClientId": "string",
  "ClientName": "string",
  "DefaultRedirectURI": "string",
  "EnableTokenRevocation": boolean,
  "ExplicitAuthFlows": [ "string" ],
  "IdTokenValidity": number,
  "LogoutURLs": [ "string" ],
  "PreventUserExistenceErrors": "string",
  "ReadAttributes": [ "string" ],
  "RefreshTokenValidity": number,
  "SupportedIdentityProviders": [ "string" ],
  "TokenValidityUnits": {
    "AccessToken": "string",
    "IdToken": "string",
    "RefreshToken": "string"
  },
  "UserPoolId": "string",
  "WriteAttributes": [ "string" ]
}
```

API Version 2016-04-18
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessTokenValidity (p. 373)**

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for AccessTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.

For example, when you set AccessTokenValidity to 10 and TokenValidityUnits to hours, your user can authorize access with their access token for 10 hours.

The default time unit for AccessTokenValidity in an API request is hours. *Valid range* is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your access tokens are valid for one hour.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

**AllowedOAuthFlows (p. 373)**

The allowed OAuth flows.

- **code**
  
  Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the /oauth2/token endpoint.

- **implicit**
  
  Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

- **client_credentials**
  
  Issue the access token from the /oauth2/token endpoint directly to a non-person user using a combination of the client ID and client secret.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: code | implicit | client_credentials

Required: No

**AllowedOAuthFlowsUserPoolClient (p. 373)**

Set to true to use OAuth 2.0 features in your user pool app client.

AllowedOAuthFlowsUserPoolClient must be true before you can configure the following features in your app client.

- **CallBackURLs**: Callback URLs.
- **LogoutURLs**: Sign-out redirect URLs.
- **AllowedOAuthScopes**: OAuth 2.0 scopes.
Request Parameters

- **AllowedOAuthFlows**: Support for authorization code, implicit, and client credentials OAuth 2.0 grants.

  To use OAuth 2.0 features, configure one of these features in the Amazon Cognito console or set `AllowedOAuthFlowsUserPoolClient` to `true` in a `CreateUserPoolClient` or `UpdateUserPoolClient` API request. If you don't set a value for `AllowedOAuthFlowsUserPoolClient` in a request with the AWS CLI or SDKs, it defaults to `false`.

  Type: Boolean

  Required: No

- **AllowedOAuthScopes** *(p. 373)*

  The allowed OAuth scopes. Possible values provided by OAuth are `phone`, `email`, `openid`, and `profile`. Possible values provided by AWS are `aws.cognito.signin.user.admin`. Custom scopes created in Resource Servers are also supported.

  Type: Array of strings

  Array Members: Maximum number of 50 items.

  Length Constraints: Minimum length of 1. Maximum length of 256.

  Pattern: `[^\x21\x23-\x5B\x5D-\x7E]+`

  Required: No

- **AnalyticsConfiguration** *(p. 373)*

  The Amazon Pinpoint analytics configuration necessary to collect metrics for this user pool.

  **Note**

  In AWS Regions where Amazon Pinpoint isn't available, user pools only support sending events to Amazon Pinpoint projects in us-east-1. In Regions where Amazon Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

  Type: `AnalyticsConfigurationType` *(p. 400)* object

  Required: No

- **AuthSessionValidity** *(p. 373)*

  Amazon Cognito creates a session token for each API request in an authentication flow. `AuthSessionValidity` is the duration, in minutes, of that session token. Your user pool native user must respond to each authentication challenge before the session expires.

  Type: Integer


  Required: No

- **CallbackURLs** *(p. 373)*

  A list of allowed redirect (callback) URLs for the IdPs.

  A redirect URI must:
  - Be an absolute URI.
  - Be registered with the authorization server.
  - Not include a fragment component.
See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported.

_type: Array of strings

_array_members: Minimum number of 0 items. Maximum number of 100 items.

_length_constraints: Minimum length of 1. Maximum length of 1024.

_pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

_required: No

**ClientId (p. 373)**

The ID of the client associated with the user pool.

_type: String

_length_constraints: Minimum length of 1. Maximum length of 128.

_pattern: [\w+]+

_required: Yes

**ClientName (p. 373)**

The client name from the update user pool client request.

_type: String

_length_constraints: Minimum length of 1. Maximum length of 128.

_pattern: [\w\s+=,.@-]+ 

_required: No

**DefaultRedirectURI (p. 373)**

The default redirect URI. Must be in the CallbackURLs list.

A redirect URI must:

- Be an absolute URI.
- Be registered with the authorization server.
- Not include a fragment component.

See OAuth 2.0 - Redirection Endpoint.

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only.

App callback URLs such as myapp://example are also supported.

_type: String

_length_constraints: Minimum length of 1. Maximum length of 1024.

_pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

_required: No
EnablePropagateAdditionalUserContextData (p. 373)

Activates the propagation of additional user context data. For more information about propagation of user context data, see Adding advanced security to a user pool.

If you don’t include this parameter, you can’t send device fingerprint information, including source IP address, to Amazon Cognito advanced security. You can only activate EnablePropagateAdditionalUserContextData in an app client that has a client secret.

Type: Boolean
Required: No

EnableTokenRevocation (p. 373)

Activates or deactivates token revocation. For more information about revoking tokens, see RevokeToken.

Type: Boolean
Required: No

ExplicitAuthFlows (p. 373)

The authentication flows that you want your user pool client to support. For each app client in your user pool, you can sign in your users with any combination of one or more flows, including with a user name and Secure Remote Password (SRP), a user name and password, or a custom authentication process that you define with Lambda functions.

**Note**
If you don’t specify a value for ExplicitAuthFlows, your user client supports ALLOW_REFRESH_TOKEN_AUTH, ALLOW_USER_SRP_AUTH, and ALLOW_CUSTOM_AUTH.

Valid values include:
- ALLOW_ADMIN_USER_PASSWORD_AUTH: Enable admin based user password authentication flow ADMIN_USER_PASSWORD_AUTH. This setting replaces the ADMIN_NO_SRP_AUTH setting. With this authentication flow, your app passes a user name and password to Amazon Cognito in the request, instead of using the Secure Remote Password (SRP) protocol to securely transmit the password.
- ALLOW_CUSTOM_AUTH: Enable Lambda trigger based authentication.
- ALLOW_USER_PASSWORD_AUTH: Enable user password-based authentication. In this flow, Amazon Cognito receives the password in the request instead of using the SRP protocol to verify passwords.
- ALLOW_USER_SRP_AUTH: Enable SRP-based authentication.
- ALLOW_REFRESH_TOKEN_AUTH: Enable authflow to refresh tokens.

In some environments, you will see the values ADMIN_NO_SRP_AUTH, CUSTOM_AUTH_FLOW_ONLY, or USER_PASSWORD_AUTH. You can’t assign these legacy ExplicitAuthFlows values to user pool clients at the same time as values that begin with ALLOW_, like ALLOW_USER_SRP_AUTH.

Type: Array of strings
Valid Values: ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH | ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH | ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH
Required: No

IdTokenValidity (p. 373)

The ID token time limit. After this limit expires, your user can’t use their ID token. To specify the time unit for IdTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.
For example, when you set IdTokenValidity as 10 and TokenValidityUnits as hours, your user can authenticate their session with their ID token for 10 hours.

The default time unit for IdTokenValidity in an API request is hours. Valid range is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your ID tokens are valid for one hour.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

LogoutURLs (p. 373)

A list of allowed logout URLs for the IdPs.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: No

PreventUserExistenceErrors (p. 373)

Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to ENABLED and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to LEGACY, those APIs return a UserNotFoundException exception if the user doesn't exist in the user pool.

Valid values include:
• ENABLED - This prevents user existence-related errors.
• LEGACY - This represents the early behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:
• AdminInitiateAuth (p. 45)
• AdminRespondToAuthChallenge (p. 74)
• InitiateAuth (p. 250)
• RespondToAuthChallenge (p. 297)
• ForgotPassword (p. 213)
• ConfirmForgotPassword (p. 115)
• ConfirmSignUp (p. 120)
• ResendConfirmationCode (p. 292)

Type: String

Valid Values: LEGACY | ENABLED

Required: No
**ReadAttributes (p. 373)**

The list of user attributes that you want your app client to have read-only access to. After your user authenticates in your app, their access token authorizes them to read their own attribute value for any attribute in this list. An example of this kind of activity is when your user selects a link to view their profile information. Your app makes a [GetUser](https://docs.aws.amazon.com/cognito/user-pools/latest/APIReference/API_GetUser.html) API request to retrieve and display your user's profile data.

When you don't specify the ReadAttributes for your app client, your app can read the values of `email_verified`, `phone_number_verified`, and the Standard attributes of your user pool. When your user pool has read access to these default attributes, ReadAttributes doesn't return any information. Amazon Cognito only populates ReadAttributes in the API response if you have specified your own custom set of read attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

**RefreshTokenValidity (p. 373)**

The refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for `RefreshTokenValidity` as seconds, minutes, hours, or days, set a `TokenValidityUnits` value in your API request.

For example, when you set `RefreshTokenValidity` as 10 and `TokenValidityUnits` as days, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for `RefreshTokenValidity` in an API request is days. You can't set `RefreshTokenValidity` to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. `Valid range` is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your refresh tokens are valid for 30 days.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

**SupportedIdentityProviders (p. 373)**

A list of provider names for the IdPs that this client supports. The following are supported: COGNITO, Facebook, Google, SignInWithApple, LoginWithAmazon, and the names of your own SAML and OIDC providers.

Type: Array of strings


Pattern: `[^\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+`

Required: No

**TokenValidityUnits (p. 373)**

The time units you use when you set the duration of ID, access, and refresh tokens. The default unit for `RefreshToken` is days, and the default for ID and access tokens is hours.

Type: [TokenValidityUnitsType (p. 468)](https://docs.aws.amazon.com/cognito/user-pools/latest/APIReference/API_RegisterForOpenIdConnectProvider.html) object
**UserPoolId (p. 373)**

The user pool ID for the user pool where you want to update the user pool client.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

**WriteAttributes (p. 373)**

The list of user attributes that you want your app client to have write access to. After your user authenticates in your app, their access token authorizes them to set or modify their own attribute value for any attribute in this list. An example of this kind of activity is when you present your user with a form to update their profile information and they change their last name. Your app then makes an `UpdateUserAttributes` API request and sets `family_name` to the new value.

When you don't specify the `WriteAttributes` for your app client, your app can write the values of the Standard attributes of your user pool. When your user pool has write access to these default attributes, `WriteAttributes` doesn't return any information. Amazon Cognito only populates `WriteAttributes` in the API response if you have specified your own custom set of write attributes.

If your app client allows users to sign in through an IdP, this array must include all attributes that you have mapped to IdP attributes. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. If your app client does not have write access to a mapped attribute, Amazon Cognito throws an error when it tries to update the attribute. For more information, see [Specifying IdP Attribute Mappings for Your user pool](https://docs.aws.amazon.com/cognito/userguide/idp-attribute-mapping.html).

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

**Response Syntax**

```
{
    "UserPoolClient": {
        "AccessTokenValidity": number,
        "AllowedOAuthFlows": [ "string" ],
        "AllowedOAuthFlowsUserPoolClient": boolean,
        "AllowedOAuthScopes": [ "string" ],
        "AnalyticsConfiguration": {
            "ApplicationArn": "string",
            "ApplicationId": "string",
            "ExternalId": "string",
            "RoleArn": "string",
            "UserDataShared": boolean
        },
        "AuthSessionValidity": number,
        "CallbackURLs": [ "string" ],
        "ClientId": "string",
        "ClientName": "string",
        "ClientSecret": "string",
        "CreationDate": number,
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**UserPoolClient (p. 380)**

The user pool client value from the response from the server when you request to update the user pool client.

Type: `UserPoolClientType (p. 479)` object

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 503)].

**ConcurrentModificationException**

This exception is thrown if two or more modifications are happening concurrently.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidOAuthFlowException**

This exception is thrown when the specified OAuth flow is not valid.

HTTP Status Code: 400

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400
**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

**ResourceNotFoundException**

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

**ScopeDoesNotExistException**

This exception is thrown when the specified scope doesn't exist.

HTTP Status Code: 400

**TooManyRequestsException**

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
UpdateUserPoolDomain

Updates the Secure Sockets Layer (SSL) certificate for the custom domain for your user pool.

You can use this operation to provide the Amazon Resource Name (ARN) of a new certificate to Amazon Cognito. You can't use it to change the domain for a user pool.

A custom domain is used to host the Amazon Cognito hosted UI, which provides sign-up and sign-in pages for your application. When you set up a custom domain, you provide a certificate that you manage with AWS Certificate Manager (ACM). When necessary, you can use this operation to change the certificate that you applied to your custom domain.

Usually, this is unnecessary following routine certificate renewal with ACM. When you renew your existing certificate in ACM, the ARN for your certificate remains the same, and your custom domain uses the new certificate automatically.

However, if you replace your existing certificate with a new one, ACM gives the new certificate a new ARN. To apply the new certificate to your custom domain, you must provide this ARN to Amazon Cognito.

When you add your new certificate in ACM, you must choose US East (N. Virginia) as the AWS Region. After you submit your request, Amazon Cognito requires up to 1 hour to distribute your new certificate to your custom domain.

For more information about adding a custom domain to your user pool, see Using Your Own Domain for the Hosted UI.

Note
Amazon Cognito evaluates AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you must use IAM credentials to authorize requests, and you must grant yourself the corresponding IAM permission in a policy.

Learn more

- Signing AWS API Requests
- Using the Amazon Cognito user pools API and user pool endpoints

Request Syntax

```json
{
  "CustomDomainConfig": {
    "CertificateArn": "string"
  },
  "Domain": "string",
  "UserPoolId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.
**CustomDomainConfig (p. 383)**

The configuration for a custom domain that hosts the sign-up and sign-in pages for your application. Use this object to specify an SSL certificate that is managed by ACM.

Type: **CustomDomainConfigType (p. 415)** object

Required: Yes

**Domain (p. 383)**

The domain name for the custom domain that hosts the sign-up and sign-in pages for your application. One example might be auth.example.com.

This string can include only lowercase letters, numbers, and hyphens. Don't use a hyphen for the first or last character. Use periods to separate subdomain names.

Type: String


Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`

Required: Yes

**UserPoolId (p. 383)**

The ID of the user pool that is associated with the custom domain whose certificate you're updating.

Type: String


Pattern: `\[\w-]+_[0-9a-zA-Z]+`

Required: Yes

---

**Response Syntax**

```
{
  "CloudFrontDomain": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**CloudFrontDomain (p. 384)**

The Amazon CloudFront endpoint that Amazon Cognito set up when you added the custom domain to your user pool.

Type: String


Pattern: `^[a-z0-9](?:[a-z0-9\-]{0,61}[a-z0-9])?$`
Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

InternalServerErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
VerifySoftwareToken

Use this API to register a user's entered time-based one-time password (TOTP) code and mark the user's software token MFA status as "verified" if successful. The request takes an access token or a session string, but not both.

**Note**
Amazon Cognito doesn't evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can't use IAM credentials to authorize requests, and you can't grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see [Using the Amazon Cognito native and OIDC APIs](#).

**Request Syntax**

```
{
    "AccessToken": "string",
    "FriendlyDeviceName": "string",
    "Session": "string",
    "UserCode": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters](#). The request accepts the following data in JSON format.

**AccessToken (p. 386)**

A valid access token that Amazon Cognito issued to the user whose software token you want to verify.

Type: String

Pattern: [A-Za-z0-9-_.]+

Required: No

**FriendlyDeviceName (p. 386)**

The friendly device name.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**Session (p. 386)**

The session that should be passed both ways in challenge-response calls to the service.

Type: String


Required: No
UserCode (p. 386)
The one-time password computed using the secret code returned by AssociateSoftwareToken.
Type: String
Pattern: [0-9]+
Required: Yes

Response Syntax

```
{
  "Session": "string",
  "Status": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

Session (p. 387)
The session that should be passed both ways in challenge-response calls to the service.
Type: String

Status (p. 387)
The status of the verify software token.
Type: String
Valid Values: SUCCESS | ERROR

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

CodeMismatchException
This exception is thrown if the provided code doesn't match what the server was expecting.
HTTP Status Code: 400

EnableSoftwareTokenMFAException
This exception is thrown when there is a code mismatch and the service fails to configure the software token TOTP multi-factor authentication (MFA).
HTTP Status Code: 400
ForbiddenException

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

InternalErrorException

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

InvalidParameterException

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

InvalidUserPoolConfigurationException

This exception is thrown when the user pool configuration is not valid.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

NotAuthorizedException

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400

PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

SoftwareTokenMFANotFoundException

This exception is thrown when the software token time-based one-time password (TOTP) multi-factor authentication (MFA) isn't activated for the user pool.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400
UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
VerifyUserAttribute

Verifies the specified user attributes in the user pool.

If your user pool requires verification before Amazon Cognito updates the attribute value, VerifyUserAttribute updates the affected attribute to its pending value. For more information, see UserAttributeUpdateSettingsType.

Note

Amazon Cognito doesn’t evaluate AWS Identity and Access Management (IAM) policies in requests for this API operation. For this operation, you can’t use IAM credentials to authorize requests, and you can’t grant IAM permissions in policies. For more information about authorization models in Amazon Cognito, see Using the Amazon Cognito native and OIDC APIs.

Request Syntax

```
{
    "AccessToken": "string",
    "AttributeName": "string",
    "Code": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 501).

The request accepts the following data in JSON format.

**AccessToken (p. 390)**

A valid access token that Amazon Cognito issued to the user whose user attributes you want to verify.

Type: String

Pattern: [A-Za-z-\0-9-_=.]+

Required: Yes

**AttributeName (p. 390)**

The attribute name in the request to verify user attributes.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]*

Required: Yes

**Code (p. 390)**

The verification code in the request to verify user attributes.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 503).

**AliasExistsException**

This exception is thrown when a user tries to confirm the account with an email address or phone number that has already been supplied as an alias for a different user profile. This exception indicates that an account with this email address or phone already exists in a user pool that you've configured to use email address or phone number as a sign-in alias.

HTTP Status Code: 400

**CodeMismatchException**

This exception is thrown if the provided code doesn't match what the server was expecting.

HTTP Status Code: 400

**ExpiredCodeException**

This exception is thrown if a code has expired.

HTTP Status Code: 400

**ForbiddenException**

This exception is thrown when AWS WAF doesn't allow your request based on a web ACL that's associated with your user pool.

HTTP Status Code: 400

**InternalErrorException**

This exception is thrown when Amazon Cognito encounters an internal error.

HTTP Status Code: 500

**InvalidParameterException**

This exception is thrown when the Amazon Cognito service encounters an invalid parameter.

HTTP Status Code: 400

**LimitExceededException**

This exception is thrown when a user exceeds the limit for a requested AWS resource.

HTTP Status Code: 400

**NotAuthorizedException**

This exception is thrown when a user isn't authorized.

HTTP Status Code: 400
PasswordResetRequiredException

This exception is thrown when a password reset is required.

HTTP Status Code: 400

ResourceNotFoundException

This exception is thrown when the Amazon Cognito service can't find the requested resource.

HTTP Status Code: 400

TooManyRequestsException

This exception is thrown when the user has made too many requests for a given operation.

HTTP Status Code: 400

UserNotConfirmedException

This exception is thrown when a user isn't confirmed successfully.

HTTP Status Code: 400

UserNotFoundException

This exception is thrown when a user isn't found.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V3
Data Types

The Amazon Cognito Identity Provider API contains several data types that various actions use. This section describes each data type in detail.

**Note**
The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- AccountRecoverySettingType (p. 395)
- AccountTakeoverActionsType (p. 396)
- AccountTakeoverActionType (p. 397)
- AccountTakeoverRiskConfigurationType (p. 398)
- AdminCreateUserConfigType (p. 399)
- AnalyticsConfigurationType (p. 400)
- AnalyticsMetadataType (p. 402)
- AttributeType (p. 403)
- AuthenticationResultType (p. 404)
- AuthEventType (p. 406)
- ChallengeResponseType (p. 408)
- CloudWatchLogsConfigurationType (p. 409)
- CodeDeliveryDetailsType (p. 410)
- CompromisedCredentialsActionsType (p. 411)
- CompromisedCredentialsRiskConfigurationType (p. 412)
- ContextDataType (p. 413)
- CustomDomainConfigType (p. 415)
- CustomEmailLambdaVersionConfigType (p. 416)
- CustomSMSLambdaVersionConfigType (p. 417)
- DeviceConfigurationType (p. 418)
- DeviceSecretVerifierConfigType (p. 419)
- DeviceType (p. 420)
- DomainDescriptionType (p. 421)
- EmailConfigurationType (p. 423)
- EventContextDataType (p. 426)
- EventFeedbackType (p. 428)
- EventRiskType (p. 429)
- GroupType (p. 430)
- HttpHeaders (p. 432)
- IdentityProviderType (p. 433)
- LambdaConfigType (p. 436)
- LogConfigurationType (p. 439)
- LogDeliveryConfigurationType (p. 440)
- MessageTemplateType (p. 441)
- MFAOptionType (p. 442)
• NewDeviceMetadataType (p. 443)
• NotifyConfigurationType (p. 444)
• NotifyEmailType (p. 446)
• NumberAttributeConstraintsType (p. 447)
• PasswordPolicyType (p. 448)
• ProviderDescription (p. 450)
• ProviderUserIdentifierType (p. 451)
• RecoveryOptionType (p. 452)
• ResourceServerScopeType (p. 453)
• ResourceServerType (p. 454)
• RiskConfigurationType (p. 456)
• RiskExceptionConfigurationType (p. 458)
• SchemaAttributeType (p. 459)
• SmsConfigurationType (p. 461)
• SmsMfaConfigType (p. 463)
• SMSMfaSettingsType (p. 464)
• SoftwareTokenMfaConfigType (p. 465)
• SoftwareTokenMfaSettingsType (p. 466)
• StringAttributeConstraintsType (p. 467)
• TokenValidityUnitsType (p. 468)
• UICustomizationType (p. 469)
• UserAttributeUpdateSettingsType (p. 471)
• UserContextDataType (p. 472)
• UserImportJobType (p. 473)
• UsernameConfigurationType (p. 476)
• UserPoolAddOnsType (p. 477)
• UserPoolClientDescription (p. 478)
• UserPoolClientType (p. 479)
• UserPoolDescriptionType (p. 487)
• UserPoolPolicyType (p. 489)
• UserPoolType (p. 490)
• UserType (p. 497)
• VerificationMessageTemplateType (p. 499)
AccountRecoverySettingType

The data type for AccountRecoverySetting.

Contents

RecoveryMechanisms

The list of RecoveryOptionTypes.

Type: Array of RecoveryOptionType (p. 452) objects

Array Members: Minimum number of 1 item. Maximum number of 2 items.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AccountTakeoverActionsType

Account takeover actions type.

Contents

HighAction

Action to take for a high risk.

Type: AccountTakeoverActionType (p. 397) object

Required: No

LowAction

Action to take for a low risk.

Type: AccountTakeoverActionType (p. 397) object

Required: No

MediumAction

Action to take for a medium risk.

Type: AccountTakeoverActionType (p. 397) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AccountTakeoverActionType

Account takeover action type.

Contents

**EventAction**

The action to take in response to the account takeover action. Valid values are as follows:

- **BLOCK** Choosing this action will block the request.
- **MFA_IF_CONFIGURED** Present an MFA challenge if user has configured it, else allow the request.
- **MFA_REQUIRED** Present an MFA challenge if user has configured it, else block the request.
- **NO_ACTION** Allow the user to sign in.

Type: String

Valid Values: BLOCK | MFA_IF_CONFIGURED | MFA_REQUIRED | NO_ACTION

Required: Yes

**Notify**

Flag specifying whether to send a notification.

Type: Boolean

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/coderun/)
- [AWS SDK for Go](https://golang.org/)
- [AWS SDK for Java V2](https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/ruby-sdk/)

API Version 2016-04-18

397
AccountTakeoverRiskConfigurationType

Configuration for mitigation actions and notification for different levels of risk detected for a potential account takeover.

Contents

Actions

Account takeover risk configuration actions.

Type: AccountTakeoverActionsType (p. 396) object

Required: Yes

NotifyConfiguration

The notify configuration used to construct email notifications.

Type: NotifyConfigurationType (p. 444) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AdminCreateUserConfigType

The configuration for creating a new user profile.

Contents

AllowAdminCreateUserOnly

Set to True if only the administrator is allowed to create user profiles. Set to False if users can sign themselves up via an app.

Type: Boolean
Required: No

InviteMessageTemplate

The message template to be used for the welcome message to new users.

See also Customizing User Invitation Messages.

Type: MessageTemplateType (p. 441) object
Required: No

UnusedAccountValidityDays

The user account expiration limit, in days, after which a new account that hasn't signed in is no longer usable. To reset the account after that time limit, you must call AdminCreateUser again, specifying "RESEND" for the MessageAction parameter. The default value for this parameter is 7.

Note
If you set a value for TemporaryPasswordValidityDays in PasswordPolicy, that value will be used, and UnusedAccountValidityDays will be no longer be an available parameter for that user pool.

Type: Integer
Valid Range: Minimum value of 0. Maximum value of 365.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3

API Version 2016-04-18
399
AnalyticsConfigurationType

The Amazon Pinpoint analytics configuration necessary to collect metrics for a user pool.

**Note**

In Regions where Amazon Pinpoint isn’t available, user pools only support sending events to Amazon Pinpoint projects in us-east-1. In Regions where Amazon Pinpoint is available, user pools support sending events to Amazon Pinpoint projects within that same Region.

**Contents**

**ApplicationArn**

The Amazon Resource Name (ARN) of an Amazon Pinpoint project. You can use the Amazon Pinpoint project to integrate with the chosen user pool Client. Amazon Cognito publishes events to the Amazon Pinpoint project that the app ARN declares.

*Type: String*


*Pattern: \\
\[w+=/,.@-]+:\\\n\[w+=/,.@-]+:\\\n\[w+=/,.@-]*:\\n\[0-9]+:\\\n\[w+=/,.@-]+:\\\n\[w+=/,.@-]+(?:\\\n\[w+=/,.@-]+)?*

*Required: No*

**ApplicationId**

The application ID for an Amazon Pinpoint application.

*Type: String*

*Pattern: ^[0-9a-fA-F]+$*

*Required: No*

**ExternalId**

The external ID.

*Type: String*

*Length Constraints: Minimum length of 0. Maximum length of 131072.*

*Required: No*

**RoleArn**

The ARN of an AWS Identity and Access Management role that authorizes Amazon Cognito to publish events to Amazon Pinpoint analytics.

*Type: String*


*Pattern: \\
\[w+=/,.@-]+:\\\n\[w+=/,.@-]+:\\\n\[w+=/,.@-]*:\\n\[0-9]+:\\\n\[w+=/,.@-]+:\\\n\[w+=/,.@-]+(?:\\\n\[w+=/,.@-]+)?*

*Required: No*

**UserDataShared**

If `UserDataShared` is `true`, Amazon Cognito includes user data in the events that it publishes to Amazon Pinpoint analytics.
Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AnalyticsMetadataType

An Amazon Pinpoint analytics endpoint.

An endpoint uniquely identifies a mobile device, email address, or phone number that can receive messages from Amazon Pinpoint analytics. For more information about AWS Regions that can contain Amazon Pinpoint resources for use with Amazon Cognito user pools, see Using Amazon Pinpoint analytics with Amazon Cognito user pools.

Contents

AnalyticsEndpointId

The endpoint ID.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AttributeType

Specifies whether the attribute is standard or custom.

Contents

Name

The name of the attribute.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: Yes

Value

The value of the attribute.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AuthenticationResultType

The authentication result.

Contents

AccessToken

A valid access token that Amazon Cognito issued to the user who you want to authenticate.

Type: String

Pattern: [A-Za-z0-9-_=\./]+

Required: No

ExpiresIn

The expiration period of the authentication result in seconds.

Type: Integer

Required: No

IdToken

The ID token.

Type: String

Pattern: [A-Za-z0-9-_=\./]+

Required: No

NewDeviceMetadata

The new device metadata from an authentication result.

Type: NewDeviceMetadataType (p. 443) object

Required: No

RefreshToken

The refresh token.

Type: String

Pattern: [A-Za-z0-9-_=\./]+

Required: No

TokenType

The token type.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
AuthEventType

The authentication event type.

Contents

ChallengeResponses

The challenge responses.

Type: Array of ChallengeResponseType (p. 408) objects

Required: No

CreationDate

The date and time, in ISO 8601 format, when the item was created.

Type: Timestamp

Required: No

EventContextData

The user context data captured at the time of an event request. This value provides additional information about the client from which event the request is received.

Type: EventContextDataType (p. 426) object

Required: No

EventFeedback

A flag specifying the user feedback captured at the time of an event request is good or bad.

Type: EventFeedbackType (p. 428) object

Required: No

EventId

The event ID.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

EventResponse

The event response.

Type: String

Valid Values: Pass | Fail | InProgress

Required: No

EventRisk

The event risk.

Type: EventRiskType (p. 429) object
### Required: No

**EventType**

The event type.

Type: String

Valid Values: SignIn | SignUp | ForgotPassword | PasswordChange | ResendCode

Required: No

---

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
ChallengeResponseType

The challenge response type.

Contents

**ChallengeName**

The challenge name.

Type: String

Valid Values: Password | Mfa

Required: No

**ChallengeResponse**

The challenge response.

Type: String

Valid Values: Success | Failure

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/sdk-for-cpp/)
- [AWS SDK for Go](https://github.com/aws/aws-sdk-go)
- [AWS SDK for Java V2](https://aws-sdk-java.github.io/aws-sdk-java/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/sdk-for-ruby/)

CloudWatchLogsConfigurationType

The CloudWatch logging destination of a user pool detailed activity logging configuration.

Contents

LogGroupArn

The Amazon Resource Name (arn) of a CloudWatch Logs log group where your user pool sends logs. The log group must not be encrypted with AWS Key Management Service and must be in the same AWS account as your user pool.

To send logs to log groups with a resource policy of a size greater than 5120 characters, configure a log group with a path that starts with /aws/vendedlogs. For more information, see Enabling logging from certain AWS services.

Type: String


Pattern: arn:[:w+=/,.@-]+[:w+=/,.@-]+(:[:w+=/,.@-]*)?:[0-9]+[:w+=/,.@-]+(:[:w+=/,.@-]+)?(:[:w+=/,.@-]+)?

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
CodeDeliveryDetailsType

The delivery details for an email or SMS message that Amazon Cognito sent for authentication or verification.

Contents

AttributeName

The name of the attribute that Amazon Cognito verifies with the code.

Type: String
Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\]+
Required: No

DeliveryMedium

The method that Amazon Cognito used to send the code.

Type: String
Valid Values: SMS | EMAIL
Required: No

Destination

The email address or phone number destination where Amazon Cognito sent the code.

Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
CompromisedCredentialsActionsType

The compromised credentials actions type.

Contents

EventAction

The event action.

Type: String

Valid Values: BLOCK | NO_ACTION

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
CompromisedCredentialsRiskConfigurationType

The compromised credentials risk configuration type.

Contents

Actions

The compromised credentials risk configuration actions.

Type: CompromisedCredentialsActionsType (p. 411) object

Required: Yes

EventFilter

Perform the action for these events. The default is to perform all events if no event filter is specified.

Type: Array of strings

Valid Values: SIGN_IN | PASSWORD_CHANGE | SIGN_UP

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/csharp/
- [AWS SDK for Go](https://aws.amazon.com/go/
- [AWS SDK for Java V2](https://aws.amazon.com/java/
- [AWS SDK for Ruby V3](https://aws.amazon.com/ruby/
ContextDataType

Contextual user data type used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Contents

HttpHeaders

HttpHeaders received on your server in same order.

Type: Array of HttpHeaders (p. 432) objects

Required: Yes

IpAddress

The source IP address of your user's device.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes

ServerName

Your server endpoint where this API is invoked.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes

ServerPath

Your server path where this API is invoked.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes

EncodedData

Encoded device-fingerprint details that your app collected with the Amazon Cognito context data collection library. For more information, see Adding user device and session data to API requests.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
See Also

- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
CustomDomainConfigType

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Contents

CertificateArn

The Amazon Resource Name (ARN) of an AWS Certificate Manager SSL certificate. You use this certificate for the subdomain of your custom domain.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(?:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
CustomEmailLambdaVersionConfigType

A custom email sender AWS Lambda configuration type.

Contents

LambdaArn

The Amazon Resource Name (ARN) of the Lambda function that Amazon Cognito activates to send email notifications to users.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:[\w+=/,.@-]+)?(:[\w+=/,.@-]+)?

Required: Yes

LambdaVersion

Signature of the "request" attribute in the "event" information Amazon Cognito passes to your custom email Lambda function. The only supported value is V1_0.

Type: String

Valid Values: V1_0

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/sdk-for-cpp)
- [AWS SDK for Go](https://aws.amazon.com/sdk-for-go)
- [AWS SDK for Java V2](https://aws.amazon.com/sdk-for-java)
- [AWS SDK for Ruby V3](https://aws.amazon.com/sdk-for-ruby)
CustomSMSLambdaVersionConfigType

A custom SMS sender AWS Lambda configuration type.

Contents

LambdaArn

The Amazon Resource Name (ARN) of the Lambda function that Amazon Cognito activates to send SMS notifications to users.

Type: String


Pattern: \n+\[\w+=/,.@-]+:[\w+=/,.@-]+((\[\w+=/,.@-]*):\[0-9]+):\[\w+=/,.@-]+(:(\[\w+=/,.@-]+))?

Required: Yes

LambdaVersion

Signature of the "request" attribute in the "event" information that Amazon Cognito passes to your custom SMS Lambda function. The only supported value is V1_0.

Type: String

Valid Values: V1_0

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
DeviceConfigurationType

The device-remembering configuration for a user pool. A DescribeUserPool request returns a null value for this object when the user pool isn't configured to remember devices. When device remembering is active, you can remember a user's device with a ConfirmDevice API request. Additionally, when the property DeviceOnlyRememberedOnUserPrompt is true, you must follow ConfirmDevice with a UpdateDeviceStatus API request that sets the user's device to remembered or not_remembered.

To sign in with a remembered device, include DEVICE_KEY in the authentication parameters in your user's InitiateAuth request. If your app doesn't include a DEVICE_KEY parameter, the response from Amazon Cognito includes newly-generated DEVICE_KEY and DEVICE_GROUP_KEY values under NewDeviceMetadata. Store these values to use in future device-authentication requests.

**Note**
When you provide a value for any property of DeviceConfiguration, you activate the device remembering for the user pool.

### Contents

**ChallengeRequiredOnNewDevice**

When true, a remembered device can sign in with device authentication instead of SMS and time-based one-time password (TOTP) factors for multi-factor authentication (MFA).

**Note**
Whether or not ChallengeRequiredOnNewDevice is true, users who sign in with devices that have not been confirmed or remembered must still provide a second factor in a user pool that requires MFA.

Type: Boolean

Required: No

**DeviceOnlyRememberedOnUserPrompt**

When true, Amazon Cognito doesn't automatically remember a user's device when your app sends a ConfirmDevice API request. In your app, create a prompt for your user to choose whether they want to remember their device. Return the user's choice in an UpdateDeviceStatus API request.

When DeviceOnlyRememberedOnUserPrompt is false, Amazon Cognito immediately remembers devices that you register in a ConfirmDevice API request.

Type: Boolean

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
DeviceSecretVerifierConfigType

The device verifier against which it is authenticated.

Contents

PasswordVerifier

The password verifier.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

Salt

The salt

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
DeviceType

The device type.

Contents

DeviceAttributes

The device attributes.

Type: Array of [AttributeType](p. 403) objects

Required: No

DeviceCreateDate

The creation date of the device.

Type: Timestamp

Required: No

DeviceKey

The device key.

Type: String


Pattern: [\w-]+_[0-9a-f-]+

Required: No

DeviceLastAuthenticatedDate

The date when the device was last authenticated.

Type: Timestamp

Required: No

DeviceLastModifiedDate

The date and time, in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format, when the item was modified.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/cybersecurity/resources/aws-sdk-cpp/)
- [AWS SDK for Go](https://github.com/aws/aws-sdk-go)
- [AWS SDK for Java V2](https://aws.amazon.com/cybersecurity/resources/aws-sdk-java/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/cybersecurity/resources/aws-sdk-ruby/)

API Version 2016-04-18
DomainDescriptionType

A container for information about a domain.

Contents

AWSAccountId

The AWS ID for the user pool owner.

Type: String

Length Constraints: Maximum length of 12.

Pattern: [0-9]+

Required: No

CloudFrontDistribution

The Amazon CloudFront endpoint that you use as the target of the alias that you set up with your Domain Name Service (DNS) provider.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

CustomDomainConfig

The configuration for a custom domain that hosts the sign-up and sign-in webpages for your application.

Type: CustomDomainConfigType (p. 415) object

Required: No

Domain

The domain string. For custom domains, this is the fully-qualified domain name, such as auth.example.com. For Amazon Cognito prefix domains, this is the prefix alone, such as auth.

Type: String


Pattern: ^[a-z0-9](?:-[a-z0-9\-]{0,61}[a-z0-9])?$

Required: No

S3Bucket

The Amazon S3 bucket where the static files for this domain are stored.

Type: String


Pattern: ^[0-9A-Za-z\-\._/]*(?<!\.)$

Required: No
**Status**

The domain status.

Type: String

Valid Values: CREATING | DELETING | UPDATING | ACTIVE | FAILED

Required: No

**UserPoolId**

The user pool ID.

Type: String


Pattern: [\w-]+[\0-9a-zA-Z]+

Required: No

**Version**

The app version.

Type: String


Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
EmailConfigurationType

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages from your user pool.

**Note**
Amazon Cognito can send email messages with Amazon Simple Email Service resources in the AWS Region where you created your user pool, and in alternate Regions in some cases. For more information on the supported Regions, see [Email settings for Amazon Cognito user pools](#).

## Contents

### ConfigurationSet

The set of configuration rules that can be applied to emails sent using Amazon Simple Email Service. A configuration set is applied to an email by including a reference to the configuration set in the headers of the email. Once applied, all of the rules in that configuration set are applied to the email. Configuration sets can be used to apply the following types of rules to emails:

- **Event publishing**
  
  Amazon Simple Email Service can track the number of send, delivery, open, click, bounce, and complaint events for each email sent. Use event publishing to send information about these events to other AWS services such as Amazon CloudWatch.

- **IP pool management**
  
  When leasing dedicated IP addresses with Amazon Simple Email Service, you can create groups of IP addresses, called dedicated IP pools. You can then associate the dedicated IP pools with configuration sets.

  Type: String

  Length Constraints: Minimum length of 1. Maximum length of 64.

  Pattern: `^[a-zA-Z0-9_-]+$`

  Required: No

### EmailSendingAccount

Specifies whether Amazon Cognito uses its built-in functionality to send your users email messages, or uses your Amazon Simple Email Service email configuration. Specify one of the following values:

- **COGITO_DEFAULT**
  
  When Amazon Cognito emails your users, it uses its built-in email functionality. When you use the default option, Amazon Cognito allows only a limited number of emails each day for your user pool. For typical production environments, the default email limit is less than the required delivery volume. To achieve a higher delivery volume, specify DEVELOPER to use your Amazon SES email configuration.

  To look up the email delivery limit for the default option, see [Limits](#) in the [Amazon Cognito Developer Guide](#).

  The default FROM address is `no-reply@verificationemail.com`. To customize the FROM address, provide the Amazon Resource Name (ARN) of an Amazon SES verified email address for the `SourceArn` parameter.

- **DEVELOPER**
  
  When Amazon Cognito emails your users, it uses your Amazon SES configuration. Amazon Cognito calls Amazon SES on your behalf to send email from your verified email address. When
you use this option, the email delivery limits are the same limits that apply to your Amazon SES verified email address in your AWS account. If you use this option, provide the ARN of an Amazon SES verified email address for the SourceArn parameter.

Before Amazon Cognito can email your users, it requires additional permissions to call Amazon SES on your behalf. When you update your user pool with this option, Amazon Cognito creates a service-linked role, which is a type of role in your AWS account. This role contains the permissions that allow you to access Amazon SES and send email messages from your email address. For more information about the service-linked role that Amazon Cognito creates, see Using Service-Linked Roles for Amazon Cognito in the Amazon Cognito Developer Guide.

Type: String

Valid Values: COGNITO_DEFAULT | DEVELOPER

Required: No

From

Either the sender’s email address or the sender’s name with their email address. For example, testuser@example.com or Test User <testuser@example.com>. This address appears before the body of the email.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

ReplyToEmailAddress

The destination to which the receiver of the email should reply.

Type: String

Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+@[\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: No

SourceArn

The ARN of a verified email address or an address from a verified domain in Amazon SES. You can set a SourceArn email from a verified domain only with an API request. You can set a verified email address, but not an address in a verified domain, in the Amazon Cognito console. Amazon Cognito uses the email address that you provide in one of the following ways, depending on the value that you specify for the EmailSendingAccount parameter:

- If you specify COGNITO_DEFAULT, Amazon Cognito uses this address as the custom FROM address when it emails your users using its built-in email account.
- If you specify DEVELOPER, Amazon Cognito emails your users with this address by calling Amazon SES on your behalf.

The Region value of the SourceArn parameter must indicate a supported AWS Region of your user pool. Typically, the Region in the SourceArn and the user pool Region are the same. For more information, see Amazon SES email configuration regions in the Amazon Cognito Developer Guide.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:(\[\w+=/,.@-\]+)?:[0-9]+:[\w+=/,.@-]+:(\[\w+=/,.@-\]+)?(\[\w+=/,.@-\]+)?
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
EventContextDataType

Specifies the user context data captured at the time of an event request.

Contents

City

The user's city.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

Country

The user's country.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

DeviceName

The user's device name.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

IpAddress

The source IP address of your user's device.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

Timezone

The user's time zone.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for Ruby V3
EventFeedbackType

Specifies the event feedback type.

Contents

FeedbackValue

The authentication event feedback value. When you provide a FeedbackValue value of valid, you tell Amazon Cognito that you trust a user session where Amazon Cognito has evaluated some level of risk. When you provide a FeedbackValue value of invalid, you tell Amazon Cognito that you don't trust a user session, or you don't believe that Amazon Cognito evaluated a high-enough risk level.

Type: String

Valid Values: Valid | Invalid

Required: Yes

Provider

The provider.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: Yes

FeedbackDate

The event feedback date.

Type: Timestamp

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
EventRiskType

The event risk type.

Contents

CompromisedCredentialsDetected

Indicates whether compromised credentials were detected during an authentication event.

Type: Boolean
Required: No

RiskDecision

The risk decision.

Type: String

Valid Values: NoRisk | AccountTakeover | Block
Required: No

RiskLevel

The risk level.

Type: String

Valid Values: Low | Medium | High
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
**GroupType**

The group type.

## Contents

**CreationDate**

The date and time, in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format, when the item was created.

Type: Timestamp

Required: No

**Description**

A string containing the description of the group.

Type: String

Length Constraints: Maximum length of 2048.

Required: No

**GroupName**

The name of the group.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}] *

Required: No

**LastModifiedDate**

The date and time, in [ISO 8601](https://en.wikipedia.org/wiki/ISO_8601) format, when the item was modified.

Type: Timestamp

Required: No

**Precedence**

A non-negative integer value that specifies the precedence of this group relative to the other groups that a user can belong to in the user pool. Zero is the highest precedence value. Groups with lower Precedence values take precedence over groups with higher or null Precedence values. If a user belongs to two or more groups, it is the group with the lowest precedence value whose role ARN is given in the user's tokens for the cognito:roles and cognito:preferred_role claims.

Two groups can have the same Precedence value. If this happens, neither group takes precedence over the other. If two groups with the same Precedence have the same role ARN, that role is used in the cognito:preferred_role claim in tokens for users in each group. If the two groups have different role ARNs, the cognito:preferred_role claim isn't set in users' tokens.

The default Precedence value is null.

Type: Integer

Valid Range: Minimum value of 0.
Required: No

**RoleArn**

The role Amazon Resource Name (ARN) for the group.

Type: String


Pattern: arn:[:\w+=/,.@-]+[:\w+=/,.@-+][:\w+=/,.@-]+([:\w+=/,.@-]+)?([:\w+=/,.@-]+)?

Required: No

**UserPoolId**

The user pool ID for the user pool.

Type: String


Pattern: [:\w-]+_[0-9a-zA-Z]+

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)

API Version 2016-04-18

431
HeaderView

The HTTP header.

Contents

headerName

The header name.
Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

headerValue

The header value.
Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
IdentityProviderType

A container for information about an IdP.

Contents

**AttributeMapping**

A mapping of IdP attributes to standard and custom user pool attributes.

- Type: String to string map
- Key Length Constraints: Minimum length of 1. Maximum length of 32.
- Value Length Constraints: Minimum length of 0. Maximum length of 131072.
- Required: No

**CreationDate**

The date and time, in ISO 8601 format, when the item was created.

- Type: Timestamp
- Required: No

**IdpIdentifiers**

A list of IdP identifiers.

- Type: Array of strings
- Array Members: Minimum number of 0 items. Maximum number of 50 items.
- Pattern: `\w\s+=.@-]+`
- Required: No

**LastModifiedDate**

The date and time, in ISO 8601 format, when the item was modified.

- Type: Timestamp
- Required: No

**ProviderDetails**

The IdP details. The following list describes the provider detail keys for each IdP type.

- For Google and Login with Amazon:
  - client_id
  - client_secret
  - authorize_scopes

- For Facebook:
  - client_id
  - client_secret
  - authorize_scopes
  - api_version
- For Sign in with Apple:
  - client_id
  - team_id
  - key_id
  - private_key

  You can submit a private_key when you add or update an IdP. Describe operations don't return the private key.

- authorize_scopes

- For OIDC providers:
  - client_id
  - client_secret
  - attributes_request_method
  - oidc_issuer
  - authorize_scopes

  The following keys are only present if Amazon Cognito didn't discover them at the oidc_issuer URL.
  - authorize_url
  - token_url
  - attributes_url
  - jwks_uri

  Amazon Cognito sets the value of the following keys automatically. They are read-only.
  - attributes_url_add_attributes

- For SAML providers:
  - MetadataFile or MetadataURL
  - IDPSignout optional

  Type: String to string map

  Key Length Constraints: Minimum length of 0. Maximum length of 131072.
  Value Length Constraints: Minimum length of 0. Maximum length of 131072.

  Required: No

**ProviderName**

The IdP name.

Type: String


Pattern: `[\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]++`

Required: No

**ProviderType**

The IdP type.

Type: String

Valid Values: SAML | Facebook | Google | LoginWithAmazon | SignInWithApple | OIDC
Required: No

**UserPoolId**

The user pool ID.

Type: String


Pattern: `[\w-]+[0-9a-zA-Z]+`

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
LambdaConfigType

 Specifies the configuration for AWS Lambda triggers.

Contents

CreateAuthChallenge

Creates an authentication challenge.

Type: String


Pattern: arn:[\w=:/,.@-]+:[\w=:/,.@-]+:(\[\w=:/,.@-\]?:[0-9]+:)?(\[\w=:/,.@-\]+)?

Required: No

CustomEmailSender

A custom email sender Lambda trigger.

Type: CustomEmailLambdaVersionConfigType (p. 416) object

Required: No

CustomMessage

A custom Message AWS Lambda trigger.

Type: String


Pattern: arn:[\w=:/,.@-]+:[\w=:/,.@-]+:(\[\w=:/,.@-\]?:[0-9]+:)?(\[\w=:/,.@-\]+)?

Required: No

CustomSMSSender

A custom SMS sender Lambda trigger.

Type: CustomSMSSLambdaVersionConfigType (p. 417) object

Required: No

DefineAuthChallenge

Defines the authentication challenge.

Type: String


Pattern: arn:[\w=:/,.@-]+:[\w=:/,.@-]+:(\[\w=:/,.@-\]?:[0-9]+:)?(\[\w=:/,.@-\]+)?

Required: No
KMSKeyID

The Amazon Resource Name (ARN) of an AWS KMS key. Amazon Cognito uses the key to encrypt codes and temporary passwords sent to CustomEmailSender and CustomSMSSender.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:\[\w+=/,.@-]+)?(:\[\w+=/,.@-]+)?

Required: No

PostAuthentication

A post-authentication AWS Lambda trigger.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:\[\w+=/,.@-]+)?(:\[\w+=/,.@-]+)?

Required: No

PostConfirmation

A post-confirmation AWS Lambda trigger.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:\[\w+=/,.@-]+)?(:\[\w+=/,.@-]+)?

Required: No

PreAuthentication

A pre-authentication AWS Lambda trigger.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:\[\w+=/,.@-]+)?(:\[\w+=/,.@-]+)?

Required: No

PreSignUp

A pre-registration AWS Lambda trigger.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[\w+=/,.@-]*:[0-9]+:[\w+=/,.@-]+(:\[\w+=/,.@-]+)?(:\[\w+=/,.@-]+)?

Required: No
**PreTokenGeneration**

A Lambda trigger that is invoked before token generation.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[(\[\w+=/,.@-]+)?:[0-9]+:[\w+=/,.@-]+:(\[\w+=/,.@-]+)?(\[\w+=/,.@-]+)?

Required: No

**UserMigration**

The user migration Lambda config type.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[(\[\w+=/,.@-]+)?:[0-9]+:[\w+=/,.@-]+:(\[\w+=/,.@-]+)?(\[\w+=/,.@-]+)?

Required: No

**VerifyAuthChallengeResponse**

Verifies the authentication challenge response.

Type: String


Pattern: arn:[\w+=/,.@-]+:[\w+=/,.@-]+:[(\[\w+=/,.@-]+)?:[0-9]+:[\w+=/,.@-]+:(\[\w+=/,.@-]+)?(\[\w+=/,.@-]+)?

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
LogConfigurationType

The logging parameters of a user pool.

Contents

EventSource

The source of events that your user pool sends for detailed activity logging.

Type: String

Valid Values: userNotification

Required: Yes

LogLevel

The error level selection of logs that a user pool sends for detailed activity logging.

Type: String

Valid Values: ERROR

Required: Yes

CloudWatchLogsConfiguration

The CloudWatch logging destination of a user pool.

Type: CloudWatchLogsConfigurationType (p. 409) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
LogDeliveryConfigurationType

The logging parameters of a user pool.

Contents

LogConfigurations

The detailed activity logging destination of a user pool.

Type: Array of LogConfigurationType (p. 439) objects

Array Members: Minimum number of 0 items. Maximum number of 1 item.

Required: Yes

UserPoolId

The ID of the user pool where you configured detailed activity logging.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
MessageTemplateType

The message template structure.

Contents

EmailMessage

The message template for email messages. EmailMessage is allowed only if EmailSendingAccount is DEVELOPER.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*\{####\}
\[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*

Required: No

EmailSubject

The subject line for email messages. EmailSubject is allowed only if EmailSendingAccount is DEVELOPER.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*

Required: No

SNSMessage

The message template for SMS messages.

Type: String


Pattern: .*\{####\}.*

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
MFAOptionType

This data type is no longer supported. Applies only to SMS multi-factor authentication (MFA) configurations. Does not apply to time-based one-time password (TOTP) software token MFA configurations.

To set either type of MFA configuration, use the AdminSetUserMFAPreference (p. 82) or setUserMFAPreference (p. 319) actions.

To look up information about either type of MFA configuration, use the AdminGetUser:UserMFASettingList (p. 43) or GetUser:UserMFASettingList (p. 237) responses.

Contents

AttributeName

The attribute name of the MFA option type. The only valid value is phone_number.

Type: String
Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}]+
Required: No

DeliveryMedium

The delivery medium to send the MFA code. You can use this parameter to set only the SMS delivery medium value.

Type: String
Valid Values: SMS | EMAIL
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
NewDeviceMetadataType

The new device metadata type.

Contents

DeviceGroupKey

The device group key.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

DeviceKey

The device key.

Type: String


Pattern: \w-+\[0-9a-f-]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
NotifyConfigurationType

The notify configuration type.

Contents

SourceArn

The Amazon Resource Name (ARN) of the identity that is associated with the sending authorization policy. This identity permits Amazon Cognito to send for the email address specified in the From parameter.

Type: String


Pattern: `arn:[:\w+=/,.@-]+::[:\w+=/,.@-]+(:[0-9]+::[:\w+=/,.@-]+)?(:[:\w+=/,.@-]+)?`?

Required: Yes

BlockEmail

Email template used when a detected risk event is blocked.

Type: NotifyEmailType (p. 446) object

Required: No

From

The email address that is sending the email. The address must be either individually verified with Amazon Simple Email Service, or from a domain that has been verified with Amazon SES.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

MfaEmail

The multi-factor authentication (MFA) email template used when MFA is challenged as part of a detected risk.

Type: NotifyEmailType (p. 446) object

Required: No

NoActionEmail

The email template used when a detected risk event is allowed.

Type: NotifyEmailType (p. 446) object

Required: No

ReplyTo

The destination to which the receiver of an email should reply to.

Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
NotifyEmailType

The notify email type.

Contents

Subject

The email subject.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\s]+

Required: Yes

HtmlBody

The email HTML body.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\s*]+

Required: No

TextBody

The email text body.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\s*]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
NumberAttributeConstraintsType

The minimum and maximum values of an attribute that is of the number data type.

Contents

MaxValue

The maximum value of an attribute that is of the number data type.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

MinValue

The minimum value of an attribute that is of the number data type.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
PasswordPolicyType

The password policy type.

Contents

MinimumLength

The minimum length of the password in the policy that you have set. This value can't be less than 6.

Type: Integer


Required: No

RequireLowercase

In the password policy that you have set, refers to whether you have required users to use at least one lowercase letter in their password.

Type: Boolean

Required: No

RequireNumbers

In the password policy that you have set, refers to whether you have required users to use at least one number in their password.

Type: Boolean

Required: No

RequireSymbols

In the password policy that you have set, refers to whether you have required users to use at least one symbol in their password.

Type: Boolean

Required: No

RequireUppercase

In the password policy that you have set, refers to whether you have required users to use at least one uppercase letter in their password.

Type: Boolean

Required: No

TemporaryPasswordValidityDays

The number of days a temporary password is valid in the password policy. If the user doesn't sign in during this time, an administrator must reset their password. Defaults to 7. If you submit a value of 0, Amazon Cognito treats it as a null value and sets TemporaryPasswordValidityDays to its default value.

Note
When you set TemporaryPasswordValidityDays for a user pool, you can no longer set a value for the legacy UnusedAccountValidityDays parameter in that user pool.
Type: Integer

Valid Range: Minimum value of 0. Maximum value of 365.

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
ProviderDescription

A container for IdP details.

## Contents

### CreationDate

The date and time, in ISO 8601 format, when the item was created.

Type: Timestamp

Required: No

### LastModifiedDate

The date the provider was last modified.

Type: Timestamp

Required: No

### ProviderName

The IdP name.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+

Required: No

### ProviderType

The IdP type.

Type: String

Valid Values: SAML | Facebook | Google | LoginWithAmazon | SignInWithApple | OIDC

Required: No

## See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
ProviderUserIdentifierType

A container for information about an IdP for a user pool.

Contents

ProviderAttributeName

The name of the provider attribute to link to, such as NameID.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

ProviderAttributeValue

The value of the provider attribute to link to, such as xxxxx_account.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

ProviderName

The name of the provider, such as Facebook, Google, or Login with Amazon.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
RecoveryOptionType

A map containing a priority as a key, and recovery method name as a value.

Contents

Name

The recovery method for a user.

Type: String

Valid Values: verified_email | verified_phone_number | admin_only

Required: Yes

Priority

A positive integer specifying priority of a method with 1 being the highest priority.

Type: Integer


Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
ResourceServerScopeType

A resource server scope.

Contents

ScopeDescription

A description of the scope.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Required: Yes

ScopeName

The name of the scope.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [\x21\x23-\x2E\x30-\x5B\x5D-\x7E]+

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
ResourceServerType

A container for information about a resource server for a user pool.

**Contents**

**Identifier**

The identifier for the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [\x21\x23-\x5B\x5D-\x7E]+

Required: No

**Name**

The name of the resource server.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [\w\s+=,.@-]+

Required: No

**Scopes**

A list of scopes that are defined for the resource server.

Type: Array of ResourceServerScopeType (p. 453) objects

Array Members: Maximum number of 100 items.

Required: No

**UserPoolId**

The user pool ID for the user pool that hosts the resource server.

Type: String


Pattern: [\w-]+_[0-9a-zA-Z]+

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
• AWS SDK for Ruby V3
RiskConfigurationType

The risk configuration type.

Contents

AccountTakeoverRiskConfiguration

The account takeover risk configuration object, including the NotifyConfiguration object and Actions to take if there is an account takeover.

Type: AccountTakeoverRiskConfigurationType (p. 398) object

Required: No

ClientId

The app client ID.

Type: String


Pattern: \[\w+]+

Required: No

CompromisedCredentialsRiskConfiguration

The compromised credentials risk configuration object, including the EventFilter and the EventAction.

Type: CompromisedCredentialsRiskConfigurationType (p. 412) object

Required: No

LastModifiedDate

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp

Required: No

RiskExceptionConfiguration

The configuration to override the risk decision.

Type: RiskExceptionConfigurationType (p. 458) object

Required: No

UserPoolId

The user pool ID.

Type: String


Pattern: \[\w-]+[\0-9a-zA-Z]+

Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
RiskExceptionConfigurationType

The type of the configuration to override the risk decision.

Contents

BlockedIPRangeList

Overrides the risk decision to always block the pre-authentication requests. The IP range is in CIDR notation, a compact representation of an IP address and its routing prefix.

Type: Array of strings

Array Members: Maximum number of 200 items.

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

SkippedIPRangeList

Risk detection isn't performed on the IP addresses in this range list. The IP range is in CIDR notation.

Type: Array of strings

Array Members: Maximum number of 200 items.

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
SchemaAttributeType

A list of the user attributes and their properties in your user pool. The attribute schema contains standard attributes, custom attributes with a custom: prefix, and developer attributes with a dev: prefix. For more information, see User pool attributes.

Developer-only attributes are a legacy feature of user pools, are read-only to all app clients. You can create and update developer-only attributes only with IAM-authenticated API operations. Use app client read/write permissions instead.

Contents

AttributeDataType

The data format of the values for your attribute. When you choose an AttributeDataType, Amazon Cognito validates the input against the data type. A custom attribute value in your user's ID token is always a string, for example "custom:isMember" : "true" or "custom:YearsAsMember" : "12".

Type: String

Valid Values: String | Number | DateTime | Boolean

Required: No

DeveloperOnlyAttribute

Note

You should use WriteAttributes in the user pool client to control how attributes can be mutated for new use cases instead of using DeveloperOnlyAttribute.

Specifies whether the attribute type is developer only. This attribute can only be modified by an administrator. Users won't be able to modify this attribute using their access token. For example, DeveloperOnlyAttribute can be modified using AdminUpdateUserAttributes but can't be updated using UpdateUserAttributes.

Type: Boolean

Required: No

Mutable

Specifies whether the value of the attribute can be changed.

Any user pool attribute whose value you map from an IdP attribute must be mutable, with a parameter value of true. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. If an attribute is immutable, Amazon Cognito throws an error when it attempts to update the attribute. For more information, see Specifying Identity Provider Attribute Mappings for Your User Pool.

Type: Boolean

Required: No

Name

The name of your user pool attribute. When you create or update a user pool, adding a schema attribute creates a custom or developer-only attribute. When you add an attribute with a Name value of MyAttribute, Amazon Cognito creates the custom attribute custom:MyAttribute. When DeveloperOnlyAttribute is true, Amazon Cognito creates your attribute as dev:MyAttribute.
In an operation that describes a user pool, Amazon Cognito returns this value as `value` for standard attributes, `custom:value` for custom attributes, and `dev:value` for developer-only attributes.

Type: String


Pattern: `\[\p{L}\p{M}\p{S}\p{N}\p{P}]\+`

Required: No

**Required**

Specifies whether a user pool attribute is required. If the attribute is required and the user doesn't provide a value, registration or sign-in will fail.

Type: Boolean

Required: No

**StringAttributeConstraints**

Specifies the constraints for an attribute of the string type.

Type: `StringAttributeConstraintsType` (p. 467) object

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
The SMS configuration type is the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

**Contents**

**SnsCallerArn**

The Amazon Resource Name (ARN) of the Amazon SNS caller. This is the ARN of the IAM role in your AWS account that Amazon Cognito will use to send SMS messages. SMS messages are subject to a spending limit.

Type: String


Pattern: `arn:\[\w+=/,.@-\]+:\[\w+=/,.@-\]+:(\[\w+=/,.@-\]+):\[0-9\]+:\[\w+=/,.@-\]+(:\[\w+=/,.@-\]+)?(:\[\w+=/,.@-\]+)?`?

Required: Yes

**ExternalId**

The external ID provides additional security for your IAM role. You can use an ExternalId with the IAM role that you use with Amazon SNS to send SMS messages for your user pool. If you provide an ExternalId, your Amazon Cognito user pool includes it in the request to assume your IAM role. You can configure the role trust policy to require that Amazon Cognito, and any principal, provide the ExternalId. If you use the Amazon Cognito Management Console to create a role for SMS multi-factor authentication (MFA), Amazon Cognito creates a role with the required permissions and a trust policy that demonstrates use of the ExternalId.

For more information about the ExternalId of a role, see [How to use an external ID when granting access to your AWS resources to a third party](#).

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**SnsRegion**

The AWS Region to use with Amazon SNS integration. You can choose the same Region as your user pool, or a supported Legacy Amazon SNS alternate Region.

Amazon Cognito resources in the Asia Pacific (Seoul) AWS Region must use your Amazon SNS configuration in the Asia Pacific (Tokyo) Region. For more information, see [SMS message settings for Amazon Cognito user pools](#).

Type: String


Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Amazon Cognito User Pools API Reference
SmsMfaConﬁgType

SmsMfaConﬁgType
The SMS text message multi-factor authentication (MFA) conﬁguration type.

Contents
SmsAuthenticationMessage
The SMS authentication message that will be sent to users with the code they must sign in. The
message must contain the ‘{####}’ placeholder, which is replaced with the code. If the message isn't
included, and default message will be used.
Type: String
Pattern: .*\{####\}.*
Required: No
SmsConﬁguration
The SMS conﬁguration with the settings that your Amazon Cognito user pool must use to send
an SMS message from your AWS account through Amazon Simple Notiﬁcation Service. To request
Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity
and Access Management (IAM) role that you provide for your AWS account.
Type: SmsConﬁgurationType (p. 461) object
Required: No

See Also
For more information about using this API in one of the language-speciﬁc AWS SDKs, see the following:
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for Ruby V3

API Version 2016-04-18
463


SMSMfaSettingsType

The type used for enabling SMS multi-factor authentication (MFA) at the user level. Phone numbers don't need to be verified to be used for SMS MFA. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted. If you would like MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

Contents

Enabled

Specifies whether SMS text message MFA is activated. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted.

Type: Boolean
Required: No

PreferredMfa

Specifies whether SMS is the preferred MFA method.

Type: Boolean
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
SoftwareTokenMfaConfigType

The type used for enabling software token MFA at the user pool level.

Contents

Enabled

Specifies whether software token MFA is activated.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
SoftwareTokenMfaSettingsType

The type used for enabling software token MFA at the user level. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted. If you want MFA to be applied selectively based on the assessed risk level of sign-in attempts, deactivate MFA for users and turn on Adaptive Authentication for the user pool.

Contents

**Enabled**

Specifies whether software token MFA is activated. If an MFA type is activated for a user, the user will be prompted for MFA during all sign-in attempts, unless device tracking is turned on and the device has been trusted.

Type: Boolean

Required: No

**PreferredMfa**

Specifies whether software token MFA is the preferred MFA method.

Type: Boolean

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
StringAttributeConstraintsType

The constraints associated with a string attribute.

Contents

MaxLength

The maximum length.
Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

MinLength

The minimum length.
Type: String
Length Constraints: Minimum length of 0. Maximum length of 131072.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](https://aws.amazon.com/sdk-for-c/)
- [AWS SDK for Go](https://aws.amazon.com/sdk-for-go/)
- [AWS SDK for Java V2](https://aws.amazon.com/sdk-for-java/)
- [AWS SDK for Ruby V3](https://aws.amazon.com/sdk-for-ruby/)

API Version 2016-04-18
TokenValidityUnitsType

The data type TokenValidityUnits specifies the time units you use when you set the duration of ID, access, and refresh tokens.

Contents

AccessToken

A time unit of seconds, minutes, hours, or days for the value that you set in the AccessTokenValidity parameter. The default AccessTokenValidity time unit is hours. AccessTokenValidity duration can range from five minutes to one day.

Type: String

Valid Values: seconds | minutes | hours | days

Required: No

IdToken

A time unit of seconds, minutes, hours, or days for the value that you set in the IdTokenValidity parameter. The default IdTokenValidity time unit is hours. IdTokenValidity duration can range from five minutes to one day.

Type: String

Valid Values: seconds | minutes | hours | days

Required: No

RefreshToken

A time unit of seconds, minutes, hours, or days for the value that you set in the RefreshTokenValidity parameter. The default RefreshTokenValidity time unit is days. RefreshTokenValidity duration can range from 60 minutes to 10 years.

Type: String

Valid Values: seconds | minutes | hours | days

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UICustomizationType

A container for the UI customization information for a user pool's built-in app UI.

Contents

ClientId

The client ID for the client app.

Type: String


Pattern: \[\w+\]+

Required: No

CreationDate

The date and time, in ISO 8601 format, when the item was created.

Type: Timestamp

Required: No

CSS

The CSS values in the UI customization.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

CSSVersion

The CSS version number.

Type: String

Required: No

ImageUrl

The logo image for the UI customization.

Type: String

Required: No

LastModifiedDate

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp

Required: No

UserPoolId

The user pool ID for the user pool.
Type: String


Pattern: \[\w-]+_\[0-9a-zA-Z\]+ 

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserAttributeUpdateSettingsType

The settings for updates to user attributes. These settings include the property AttributesRequireVerificationBeforeUpdate, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to email addresses and phone numbers.

Contents

AttributesRequireVerificationBeforeUpdate

Requires that your user verifies their email address, phone number, or both before Amazon Cognito updates the value of that attribute. When you update a user attribute that has this option activated, Amazon Cognito sends a verification message to the new phone number or email address. Amazon Cognito doesn't change the value of the attribute until your user responds to the verification message and confirms the new value.

You can verify an updated email address or phone number with a VerifyUserAttribute API request. You can also call the AdminUpdateUserAttributes API and set email_verified or phone_number_verified to true.

When AttributesRequireVerificationBeforeUpdate is false, your user pool doesn't require that your users verify attribute changes before Amazon Cognito updates them. In a user pool where AttributesRequireVerificationBeforeUpdate is false, API operations that change attribute values can immediately update a user's email or phone_number attribute.

Type: Array of strings

Valid Values: phone_number | email

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserContextDataType

Contextual data, such as the user's device fingerprint, IP address, or location, used for evaluating the risk of an unexpected event by Amazon Cognito advanced security.

Contents

EncodedData

Encoded device-fingerprint details that your app collected with the Amazon Cognito context data collection library. For more information, see Adding user device and session data to API requests.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

IpAddress

The source IP address of your user's device.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserImportJobType

The user import job type.

Contents

CloudWatchLogsRoleArn

The role Amazon Resource Name (ARN) for the Amazon CloudWatch Logging role for the user import job. For more information, see "Creating the CloudWatch Logs IAM Role" in the Amazon Cognito Developer Guide.

Type: String


Pattern: arn:\[w+=/,.@-]+:\[w+=/,.@-]+(?:[0-9]+:)?\[w+=/,.@-]+(:\[w+=/,.@-]+)?(?:[\w+=/,.@-]+)\

Required: No

CompletionDate

The date when the user import job was completed.

Type: Timestamp

Required: No

CompletionMessage

The message returned when the user import job is completed.

Type: String


Pattern: [\w]+

Required: No

CreationDate

The date and time, in ISO 8601 format, when the item was created.

Type: Timestamp

Required: No

FailedUsers

The number of users that couldn't be imported.

Type: Long

Required: No

ImportedUsers

The number of users that were successfully imported.

Type: Long
Required: No

**JobId**

The job ID for the user import job.

Type: String


Pattern: import-[0-9a-zA-Z-]+

Required: No

**JobName**

The job name for the user import job.

Type: String


Pattern: \[\w\s+=,.@-]+

Required: No

**PreSignedUrl**

The pre-signed URL to be used to upload the .csv file.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 2048.

Required: No

**SkippedUsers**

The number of users that were skipped.

Type: Long

Required: No

**StartDate**

The date when the user import job was started.

Type: Timestamp

Required: No

**Status**

The status of the user import job. One of the following:

- **Created** - The job was created but not started.
- **Pending** - A transition state. You have started the job, but it has not begun importing users yet.
- **InProgress** - The job has started, and users are being imported.
- **Stopping** - You have stopped the job, but the job has not stopped importing users yet.
- **Stopped** - You have stopped the job, and the job has stopped importing users.
- **Succeeded** - The job has completed successfully.
- **Failed** - The job has stopped due to an error.
- **Expired** - You created a job, but did not start the job within 24-48 hours. All data associated with the job was deleted, and the job can't be started.
Type: String

Valid Values: Created | Pending | InProgress | Stopping | Expired | Stopped | Failed | Succeeded

Required: No

**UserPoolId**

The user pool ID for the user pool that the users are being imported into.

Type: String


Pattern: \[\w-]+_[0-9a-zA-Z]+

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UsernameConfigurationType

The username configuration type.

Contents

CaseSensitive

Specifies whether user name case sensitivity will be applied for all users in the user pool through Amazon Cognito APIs. For most use cases, set case sensitivity to True (case insensitive) as a best practice. When usernames and email addresses are case insensitive, users can sign in as the same user when they enter a different capitalization of their user name.

Valid values include:

- **True**
  - Enables case sensitivity for all username input. When this option is set to **True**, users must sign in using the exact capitalization of their given username, such as "UserName". This is the default value.

- **False**
  - Enables case insensitivity for all username input. For example, when this option is set to **False**, users can sign in using username, USERNAME, or UserName. This option also enables both preferred_username and email alias to be case insensitive, in addition to the username attribute.

Type: Boolean

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++]
- [AWS SDK for Go]
- [AWS SDK for Java V2]
- [AWS SDK for Ruby V3]
UserPoolAddOnsType

User pool add-ons. Contains settings for activation of advanced security features. To log user security information but take no action, set to AUDIT. To configure automatic security responses to risky traffic to your user pool, set to ENFORCED.

For more information, see Adding advanced security to a user pool.

Contents

AdvancedSecurityMode

The operating mode of advanced security features in your user pool.

Type: String

Valid Values: OFF | AUDIT | ENFORCED

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserPoolClientDescription

The description of the user pool client.

Contents

ClientId

The ID of the client associated with the user pool.

Type: String


Pattern: \[\w+]+

Required: No

ClientName

The client name from the user pool client description.

Type: String


Pattern: \[\w\s+=,.@-]+

Required: No

UserPoolId

The user pool ID for the user pool where you want to describe the user pool client.

Type: String


Pattern: \[\w-]+[\-0-9a-zA-Z]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserPoolClientType

Contains information about a user pool client.

Contents

AccessTokenValidity

The access token time limit. After this limit expires, your user can't use their access token. To specify the time unit for AccessTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.

For example, when you set AccessTokenValidity to 10 and TokenValidityUnits to hours, your user can authorize access with their access token for 10 hours.

The default time unit for AccessTokenValidity in an API request is hours. Valid range is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your access tokens are valid for one hour.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

AllowedOAuthFlows

The allowed OAuth flows.

code

Use a code grant flow, which provides an authorization code as the response. This code can be exchanged for access tokens with the /oauth2/token endpoint.

implicit

Issue the access token (and, optionally, ID token, based on scopes) directly to your user.

client_credentials

Issue the access token from the /oauth2/token endpoint directly to a non-person user using a combination of the client ID and client secret.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 3 items.

Valid Values: code | implicit | client_credentials

Required: No

AllowedOAuthFlowsUserPoolClient

Set to true to use OAuth 2.0 features in your user pool app client.

AllowedOAuthFlowsUserPoolClient must be true before you can configure the following features in your app client.

• CallBackURLs: Callback URLs.
• LogoutURLs: Sign-out redirect URLs.
Amazon Cognito User Pools API Reference

Contents

• AllowedOAuthScopes: OAuth 2.0 scopes.
• AllowedOAuthFlows: Support for authorization code, implicit, and client credentials OAuth 2.0 grants.

To use OAuth 2.0 features, configure one of these features in the Amazon Cognito console or set AllowedOAuthFlowsUserPoolClient to true in a CreateUserPoolClient or UpdateUserPoolClient API request. If you don't set a value for AllowedOAuthFlowsUserPoolClient in a request with the AWS CLI or SDKs, it defaults to false.

Type: Boolean
Required: No

**AllowedOAuthScopes**

The OAuth scopes that your app client supports. Possible values that OAuth provides are phone, email, openid, and profile. Possible values that AWS provides are aws.cognito.signin.user.admin. Amazon Cognito also supports custom scopes that you create in Resource Servers.

Type: Array of strings
Array Members: Maximum number of 50 items.
Length Constraints: Minimum length of 1. Maximum length of 256.
Pattern: [\x21\x23-\x2B\x2D-\x7E]+
Required: No

**AnalyticsConfiguration**

The Amazon Pinpoint analytics configuration for the user pool client.

Note
Amazon Cognito user pools only support sending events to Amazon Pinpoint projects in the US East (N. Virginia) us-east-1 Region, regardless of the Region where the user pool resides.

Type: AnalyticsConfigurationType (p. 400) object
Required: No

**AuthSessionValidity**

Amazon Cognito creates a session token for each API request in an authentication flow. AuthSessionValidity is the duration, in minutes, of that session token. Your user pool native user must respond to each authentication challenge before the session expires.

Type: Integer
Required: No

**CallbackURLs**

A list of allowed redirect (callback) URLs for the IdPs.

A redirect URI must:
• Be an absolute URI.
• Be registered with the authorization server.
• Not include a fragment component.
See [OAuth 2.0 - Redirection Endpoint](#).

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.


Pattern: `\[p{L}\p{M}\p{S}\p{N}\p{P}]`+

Required: No

**ClientId**

The ID of the client associated with the user pool.

Type: String


Pattern: `\[\w\+]`+

Required: No

**ClientName**

The client name from the user pool request of the client type.

Type: String


Pattern: `\[\w\s+=,.@-\]`+

Required: No

**ClientSecret**

The client secret from the user pool request of the client type.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 64.

Pattern: `\[\w\+]`+

Required: No

**CreationDate**

The date and time, in [ISO 8601](#) format, when the item was created.

Type: Timestamp

Required: No

**DefaultRedirectURI**

The default redirect URI. Must be in the CallbackURLs list.

A redirect URI must:
• Be an absolute URI.
• Be registered with the authorization server.
• Not include a fragment component.

See [OAuth 2.0 - Redirection Endpoint](#).

Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: No

**EnablePropagateAdditionalUserContextData**

When EnablePropagateAdditionalUserContextData is true, Amazon Cognito accepts an IpAddress value that you send in the UserContextData parameter. The UserContextData parameter sends information to Amazon Cognito advanced security for risk analysis. You can send UserContextData when you sign in Amazon Cognito native users with the InitiateAuth andRespondToAuthChallenge API operations.

When EnablePropagateAdditionalUserContextData is false, you can't send your user's source IP address to Amazon Cognito advanced security with unauthenticated API operations. EnablePropagateAdditionalUserContextData doesn't affect whether you can send a source IP address in a ContextData parameter with the authenticated API operations AdminInitiateAuth and AdminRespondToAuthChallenge.

You can only activate EnablePropagateAdditionalUserContextData in an app client that has a client secret. For more information about propagation of user context data, see [Adding user device and session data to API requests](#).

Type: Boolean

Required: No

**EnableTokenRevocation**

Indicates whether token revocation is activated for the user pool client. When you create a new user pool client, token revocation is activated by default. For more information about revoking tokens, see [RevokeToken](#).

Type: Boolean

Required: No

**ExplicitAuthFlows**

The authentication flows that you want your user pool client to support. For each app client in your user pool, you can sign in your users with any combination of one or more flows, including with a user name and Secure Remote Password (SRP), a user name and password, or a custom authentication process that you define with Lambda functions.

**Note**

If you don't specify a value for ExplicitAuthFlows, your user client supports ALLOW_REFRESH_TOKEN_AUTH, ALLOW_USER_SRP_AUTH, and ALLOW_CUSTOM_AUTH.

Valid values include:
• ALLOW_ADMIN_USER_PASSWORD_AUTH: Enable admin based user password authentication flow.
  ADMIN_USER_PASSWORD_AUTH. This setting replaces the ADMIN_NO_SRP_AUTH setting. With this
  authentication flow, your app passes a user name and password to Amazon Cognito in the request,
  instead of using the Secure Remote Password (SRP) protocol to securely transmit the password.
• ALLOW_CUSTOM_AUTH: Enable Lambda trigger based authentication.
• ALLOW_USER_PASSWORD_AUTH: Enable user password-based authentication. In this flow,
  Amazon Cognito receives the password in the request instead of using the SRP protocol to verify
  passwords.
• ALLOW_USER_SRP_AUTH: Enable SRP-based authentication.
• ALLOW_REFRESH_TOKEN_AUTH: Enable authflow to refresh tokens.

In some environments, you will see the values ADMIN_NO_SRP_AUTH, CUSTOM_AUTH_FLOW_ONLY,
or USER_PASSWORD_AUTH. You can't assign these legacy ExplicitAuthFlows values to user pool
clients at the same time as values that begin with ALLOW_, like ALLOW_USER_SRP_AUTH.

Type: Array of strings

Valid Values: ADMIN_NO_SRP_AUTH | CUSTOM_AUTH_FLOW_ONLY | USER_PASSWORD_AUTH
| ALLOW_ADMIN_USER_PASSWORD_AUTH | ALLOW_CUSTOM_AUTH |
ALLOW_USER_PASSWORD_AUTH | ALLOW_USER_SRP_AUTH | ALLOW_REFRESH_TOKEN_AUTH

Required: No

IdTokenValidity

The ID token time limit. After this limit expires, your user can't use their ID token. To specify the time
unit for IdTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.

For example, when you set IdTokenValidity as 10 and TokenValidityUnits as hours, your
user can authenticate their session with their ID token for 10 hours.

The default time unit for IdTokenValidity in an API request is hours. Valid range is displayed
below in seconds.

If you don't specify otherwise in the configuration of your app client, your ID tokens are valid for one
hour.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 86400.

Required: No

LastModifiedDate

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp

Required: No

LogoutURLs

A list of allowed logout URLs for the IdPs.

Type: Array of strings

Array Members: Minimum number of 0 items. Maximum number of 100 items.

PreventUserExistenceErrors

Errors and responses that you want Amazon Cognito APIs to return during authentication, account confirmation, and password recovery when the user doesn't exist in the user pool. When set to ENABLED and the user doesn't exist, authentication returns an error indicating either the username or password was incorrect. Account confirmation and password recovery return a response indicating a code was sent to a simulated destination. When set to LEGACY, those APIs return a UserNotFoundException exception if the user doesn't exist in the user pool.

Valid values include:
- ENABLED - This prevents user existence-related errors.
- LEGACY - This represents the old behavior of Amazon Cognito where user existence related errors aren't prevented.

This setting affects the behavior of following APIs:
- AdminInitiateAuth (p. 45)
- AdminRespondToAuthChallenge (p. 74)
- InitiateAuth (p. 250)
- RespondToAuthChallenge (p. 297)
- ForgotPassword (p. 213)
- ConfirmForgotPassword (p. 115)
- ConfirmSignUp (p. 120)
- ResendConfirmationCode (p. 292)

ReadAttributes

The list of user attributes that you want your app client to have read-only access to. After your user authenticates in your app, their access token authorizes them to read their own attribute value for any attribute in this list. An example of this kind of activity is when your user selects a link to view their profile information. Your app makes a GetUser API request to retrieve and display your user's profile data.

When you don't specify the ReadAttributes for your app client, your app can read the values of email_verified, phone_number_verified, and the Standard attributes of your user pool. When your user pool has read access to these default attributes, ReadAttributes doesn't return any information. Amazon Cognito only populates ReadAttributes in the API response if you have specified your own custom set of read attributes.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

RefreshTokenValidity

The refresh token time limit. After this limit expires, your user can't use their refresh token. To specify the time unit for RefreshTokenValidity as seconds, minutes, hours, or days, set a TokenValidityUnits value in your API request.
For example, when you set RefreshTokenValidity as 10 and TokenValidityUnits as days, your user can refresh their session and retrieve new access and ID tokens for 10 days.

The default time unit for RefreshTokenValidity in an API request is days. You can't set RefreshTokenValidity to 0. If you do, Amazon Cognito overrides the value with the default value of 30 days. Valid range is displayed below in seconds.

If you don't specify otherwise in the configuration of your app client, your refresh tokens are valid for 30 days.

Type: Integer

Valid Range: Minimum value of 0. Maximum value of 315360000.

Required: No

**SupportedIdentityProviders**

A list of provider names for the IdPs that this client supports. The following are supported: COGNITO, Facebook, Google, SignInWithApple, LoginWithAmazon, and the names of your own SAML and OIDC providers.

Type: Array of strings


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\p{Z}]+

Required: No

**TokenValidityUnits**

The time units used to specify the token validity times of each token type: ID, access, and refresh.

Type: [TokenValidityUnitsType (p. 468)](p.468) object

Required: No

**UserPoolId**

The user pool ID for the user pool client.

Type: String


Pattern: \[\w-]+_\[0-9a-zA-Z]+\]

Required: No

**WriteAttributes**

The list of user attributes that you want your app client to have write access to. After your user authenticates in your app, their access token authorizes them to set or modify their own attribute value for any attribute in this list. An example of this kind of activity is when you present your user with a form to update their profile information and they change their last name. Your app then makes an [UpdateUserAttributes](p.468) API request and sets family_name to the new value.

When you don't specify the WriteAttributes for your app client, your app can write the values of the Standard attributes of your user pool. When your user pool has write access to these default attributes, WriteAttributes doesn't return any information. Amazon Cognito only populates WriteAttributes in the API response if you have specified your own custom set of write attributes.
If your app client allows users to sign in through an IdP, this array must include all attributes that you have mapped to IdP attributes. Amazon Cognito updates mapped attributes when users sign in to your application through an IdP. If your app client does not have write access to a mapped attribute, Amazon Cognito throws an error when it tries to update the attribute. For more information, see Specifying IdP Attribute Mappings for Your user pool.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserPoolDescriptionType

A user pool description.

Contents

**CreationDate**

The date and time, in ISO 8601 format, when the item was created.

Type: Timestamp
Required: No

**Id**

The ID in a user pool description.

Type: String
Pattern: \[\w-]+_[0-9a-zA-Z]+
Required: No

**LambdaConfig**

The AWS Lambda configuration information in a user pool description.

Type: LambdaConfigType (p. 436) object
Required: No

**LastModifiedDate**

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp
Required: No

**Name**

The name in a user pool description.

Type: String
Pattern: \[\w\s+=,.@-]+
Required: No

**Status**

This member has been deprecated.

The user pool status in a user pool description.

Type: String
Valid Values: Enabled | Disabled
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- [AWS SDK for C++](#)
- [AWS SDK for Go](#)
- [AWS SDK for Java V2](#)
- [AWS SDK for Ruby V3](#)
UserPoolPolicyType

The policy associated with a user pool.

Contents

PasswordPolicy

The password policy.

Type: PasswordPolicyType (p. 448) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserPoolType

A container for information about the user pool.

Contents

AccountRecoverySetting

The available verified method a user can use to recover their password when they call ForgotPassword. You can use this setting to define a preferred method when a user has more than one method available. With this setting, SMS doesn't qualify for a valid password recovery mechanism if the user also has SMS multi-factor authentication (MFA) activated. In the absence of this setting, Amazon Cognito uses the legacy behavior to determine the recovery method where SMS is preferred through email.

Type: AccountRecoverySettingType (p. 395) object

Required: No

AdminCreateUserConfig

The configuration for AdminCreateUser requests.

Type: AdminCreateUserConfigType (p. 399) object

Required: No

AliasAttributes

The attributes that are aliased in a user pool.

Type: Array of strings

Valid Values: phone_number | email | preferred_username

Required: No

Arn

The Amazon Resource Name (ARN) for the user pool.

Type: String


Pattern: arn:\[w+=/,.@-]+:\[w+=/,.@-]*:([\w+=/,.@-]*):[0-9]+:([\w+=/,.@-]+(:[^\w+=/,.@-]+)?(:[^\w+=/,.@-]+)?

Required: No

AutoVerifiedAttributes

The attributes that are auto-verified in a user pool.

Type: Array of strings

Valid Values: phone_number | email

Required: No

CreationDate

The date and time, in ISO 8601 format, when the item was created.
Type: Timestamp
Required: No

**CustomDomain**

A custom domain name that you provide to Amazon Cognito. This parameter applies only if you use a custom domain to host the sign-up and sign-in pages for your application. An example of a custom domain name might be auth.example.com.

For more information about adding a custom domain to your user pool, see [Using Your Own Domain for the Hosted UI](#).

Type: String


Pattern: `^[a-z0-9](?:[a-z0-9-]{0,61}[a-z0-9])?$`

Required: No

**DeletionProtection**

When active, DeletionProtection prevents accidental deletion of your user pool. Before you can delete a user pool that you have protected against deletion, you must deactivate this feature.

When you try to delete a protected user pool in a DeleteUserPool API request, Amazon Cognito returns an InvalidParameterException error. To delete a protected user pool, send a new DeleteUserPool request after you deactivate deletion protection in an UpdateUserPool API request.

Type: String

Valid Values: ACTIVE | INACTIVE

Required: No

**DeviceConfiguration**

The device-remembering configuration for a user pool. A null value indicates that you have deactivated device remembering in your user pool.

**Note**

When you provide a value for any DeviceConfiguration field, you activate the Amazon Cognito device-remembering feature.

Type: [DeviceConfigurationType](#) object

Required: No

**Domain**

The domain prefix, if the user pool has a domain associated with it.

Type: String


Pattern: `^[a-z0-9](?:[a-z0-9-]{0,61}[a-z0-9])?$`

Required: No

**EmailConfiguration**

The email configuration of your user pool. The email configuration type sets your preferred sending method, AWS Region, and sender for messages from your user pool.
Type: `EmailConfigurationType (p. 423)` object

Required: No

**EmailConfigurationFailure**

Deprecated. Review error codes from API requests with EventSource: cognito-idp.amazonaws.com in AWS CloudTrail for information about problems with user pool email configuration.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**EmailVerificationMessage**

This parameter is no longer used. See `VerificationMessageTemplateType`.

Type: String


Pattern: `\([\p{L}\p{M}\p{S}\p{N}\p{P}\s]*\{####\}\[\p{L}\p{M}\p{S}\p{N}\p{P}\s\]+`.

Required: No

**EmailVerificationSubject**

This parameter is no longer used. See `VerificationMessageTemplateType`.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

Pattern: `\([\p{L}\p{M}\p{S}\p{N}\p{P}\s]+`.

Required: No

**EstimatedNumberOfUsers**

A number estimating the size of the user pool.

Type: Integer

Required: No

**Id**

The ID of the user pool.

Type: String


Pattern: `\([w-]+(\[0-9a-zA-Z]\]+`.

Required: No

**LambdaConfig**

The AWS Lambda triggers associated with the user pool.

Type: `LambdaConfigType (p. 436)` object
LastModifiedDate

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp

MfaConfiguration

Can be one of the following values:
- OFF - MFA tokens aren't required and can't be specified during user registration.
- ON - MFA tokens are required for all user registrations. You can only specify required when you're initially creating a user pool.
- OPTIONAL - Users have the option when registering to create an MFA token.

Type: String

Valid Values: OFF | ON | OPTIONAL

Name

The name of the user pool.

Type: String


Pattern: [\w\s+=,.@-]+

Required: No

Policies

The policies associated with the user pool.

Type: UserPoolPolicyType (p. 489) object

Required: No

SchemaAttributes

A list of the user attributes and their properties in your user pool. The attribute schema contains standard attributes, custom attributes with a custom: prefix, and developer attributes with a dev: prefix. For more information, see User pool attributes.

Developer-only attributes are a legacy feature of user pools, are read-only to all app clients. You can create and update developer-only attributes only with IAM-authenticated API operations. Use app client read/write permissions instead.

Type: Array of SchemaAttributeType (p. 459) objects

Array Members: Minimum number of 1 item. Maximum number of 50 items.

Required: No

SmsAuthenticationMessage

The contents of the SMS authentication message.

Type: String

Pattern: .*\{#####\}.*

Required: No

**SmsConfiguration**

The SMS configuration with the settings that your Amazon Cognito user pool must use to send an SMS message from your AWS account through Amazon Simple Notification Service. To send SMS messages with Amazon SNS in the AWS Region that you want, the Amazon Cognito user pool uses an AWS Identity and Access Management (IAM) role in your AWS account.

Type: [SmsConfigurationType (p. 461)](#) object

Required: No

**SmsConfigurationFailure**

The reason why the SMS configuration can't send the messages to your users.

This message might include comma-separated values to describe why your SMS configuration can't send messages to user pool end users.

InvalidSmsRoleAccessPolicyException

The AWS Identity and Access Management role that Amazon Cognito uses to send SMS messages isn't properly configured. For more information, see [SmsConfigurationType](#).

**SNSSandbox**

The AWS account is in the SNS SMS Sandbox and messages will only reach verified end users. This parameter won't get populated with SNSSandbox if the user creating the user pool doesn't have SNS permissions. To learn how to move your AWS account out of the sandbox, see [Moving out of the SMS sandbox](#).

Type: String

Length Constraints: Minimum length of 0. Maximum length of 131072.

Required: No

**SmsVerificationMessage**

This parameter is no longer used. See [VerificationMessageTemplateType](#).

Type: String


Pattern: .*\{#####\}.*

Required: No

**Status**

*This member has been deprecated.*

This parameter is no longer used.

Type: String

Valid Values: Enabled | Disabled

Required: No
UserAttributeUpdateSettings

The settings for updates to user attributes. These settings include the property AttributesRequireVerificationBeforeUpdate, a user-pool setting that tells Amazon Cognito how to handle changes to the value of your users' email address and phone number attributes. For more information, see Verifying updates to email addresses and phone numbers.

Type: UserAttributeUpdateSettingsType (p. 471) object

Required: No

UsernameAttributes

Specifies whether a user can use an email address or phone number as a username when they sign up.

Type: Array of strings

Valid Values: phone_number | email

Required: No

UsernameConfiguration

Case sensitivity of the username input for the selected sign-in option. For example, when case sensitivity is set to False, users can sign in using either "username" or "Username". This configuration is immutable once it has been set. For more information, see UsernameConfigurationType.

Type: UsernameConfigurationType (p. 476) object

Required: No

UserPoolAddOns

User pool add-ons. Contains settings for activation of advanced security features. To log user security information but take no action, set to AUDIT. To configure automatic security responses to risky traffic to your user pool, set to ENFORCED.

For more information, see Adding advanced security to a user pool.

Type: UserPoolAddOnsType (p. 477) object

Required: No

UserPoolTags

The tags that are assigned to the user pool. A tag is a label that you can apply to user pools to categorize and manage them in different ways, such as by purpose, owner, environment, or other criteria.

Type: String to string map

Key Length Constraints: Minimum length of 1. Maximum length of 128.

Value Length Constraints: Minimum length of 0. Maximum length of 256.

Required: No

VerificationMessageTemplate

The template for verification messages.

Type: VerificationMessageTemplateType (p. 499) object
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
UserType

A user profile in a Amazon Cognito user pool.

Contents

Attributes

A container with information about the user type attributes.

Type: Array ofAttributeType (p. 403) objects

Enabled

Specifies whether the user is enabled.

Type: Boolean

Required: No

MFAOptions

The MFA options for the user.

Type: Array ofMFAOptionType (p. 442) objects

Required: No

UserCreateDate

The creation date of the user.

Type: Timestamp

Required: No

UserLastModifiedDate

The date and time, in ISO 8601 format, when the item was modified.

Type: Timestamp

Required: No

Username

The user name of the user you want to describe.

Type: String


Pattern: [\p{L}\p{M}\p{S}\p{N}\p{P}]+

Required: No

UserStatus

The user status. This can be one of the following:

- UNCONFIRMED - User has been created but not confirmed.
- CONFIRMED - User has been confirmed.
• EXTERNAL_PROVIDER - User signed in with a third-party IdP.
• UNKNOWN - User status isn't known.
• RESET_REQUIRED - User is confirmed, but the user must request a code and reset their password before they can sign in.
• FORCE_CHANGE_PASSWORD - The user is confirmed and the user can sign in using a temporary password, but on first sign-in, the user must change their password to a new value before doing anything else.

Type: String

Valid Values: UNCONFIRMED | CONFIRMED | ARCHIVED | COMPROMISED | UNKNOWN | RESET_REQUIRED | FORCE_CHANGE_PASSWORD

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java V2
• AWS SDK for Ruby V3
VerificationMessageTemplateType

The template for verification messages.

Contents

DefaultEmailOption

The default email option.

Type: String

Valid Values: CONFIRM_WITH_LINK | CONFIRM_WITH_CODE

Required: No

EmailMessage

The template for email messages that Amazon Cognito sends to your users. You can set an EmailMessage template only if the value of EmailSendingAccount is DEVELOPER. When your EmailSendingAccount is DEVELOPER, your user pool sends email messages with your own Amazon SES configuration.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]***\[####\] \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*

Required: No

EmailMessageByLink

The email message template for sending a confirmation link to the user. You can set an EmailMessageByLink template only if the value of EmailSendingAccount is DEVELOPER. When your EmailSendingAccount is DEVELOPER, your user pool sends email messages with your own Amazon SES configuration.

Type: String


Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]***\[##\[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]##\] \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]*

Required: No

EmailSubject

The subject line for the email message template. You can set an EmailSubject template only if the value of EmailSendingAccount is DEVELOPER. When your EmailSendingAccount is DEVELOPER, your user pool sends email messages with your own Amazon SES configuration.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

Pattern: \[\p{L}\p{M}\p{S}\p{N}\p{P}\s*]

Required: No
EmailSubjectByLink

The subject line for the email message template for sending a confirmation link to the user. You can set an EmailSubjectByLink template only if the value of EmailSendingAccount is DEVELOPER. When your EmailSendingAccount is DEVELOPER, your user pool sends email messages with your own Amazon SES configuration.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 140.

Pattern: \[\P{L}\P{M}\P{S}\P{N}\P{P}\s]+

Required: No

SmsMessage

The template for SMS messages that Amazon Cognito sends to your users.

Type: String


Pattern: .*\{####\}.*

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java V2
- AWS SDK for Ruby V3
Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signing AWS API requests in the IAM User Guide.

Action
The action to be performed.
- Type: string
- Required: Yes

Version
The API version that the request is written for, expressed in the format YYYY-MM-DD.
- Type: string
- Required: Yes

X-Amz-Algorithm
The hash algorithm that you used to create the request signature.
- Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.
- Type: string
- Valid Values: AWS4-HMAC-SHA256
- Required: Conditional

X-Amz-Credential
The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string ("aws4_request"). The value is expressed in the following format: access_key/YYYYMMDD/region/service/aws4_request.
- For more information, see Create a signed AWS API request in the IAM User Guide.
- Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.
- Type: string
- Required: Conditional

X-Amz-Date
The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.
- Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Elements of an AWS API request signature in the IAM User Guide.

API Version 2016-04-18
**X-Amz-Security-Token**

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS STS, see [AWS services that work with IAM](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_faqs.html) in the *IAM User Guide*.

Condition: If you're using temporary security credentials from AWS STS, you must include the security token.

**X-Amz-Signature**

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

**X-Amz-SignedHeaders**

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see [Create a signed AWS API request](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_useTokens_for_S3_apiKeys.html) in the *IAM User Guide*.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.
Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

**AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

**IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

**InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

**InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

**InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

**NotAuthorized**

You do not have permission to perform this action.

HTTP Status Code: 400

**OptInRequired**

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

**RequestExpired**

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

**ServiceUnavailable**

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

**ThrottlingException**

The request was denied due to request throttling.
HTTP Status Code: 400
**ValidationError**

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400