Amazon WorkMail

Administrator Guide

Version 1.0

Amazon WorkMail: Administrator Guide

Table of Contents

What is Amazon WorkMail?

Amazon WorkMail is a secure, managed business email and calendaring service with support for existing desktop and mobile email clients. Amazon WorkMail users can access their email, contacts, and calendars using Microsoft Outlook, their browser, or their native iOS and Android email applications. You can integrate Amazon WorkMail with your existing corporate directory and control both the keys that encrypt your data and the location in which your data is stored.

For a list of supported AWS Regions and endpoints, see AWS Regions and Endpoints.

Topics

Amazon WorkMail system requirements

Amazon WorkMail works with all major mobile devices and operating systems that support the Exchange ActiveSync protocol. These devices include the iPad, iPhone, Android, and Windows Phone. Users of macOS can add their Amazon WorkMail account to their Mail, Calendar, and Contacts apps.

If you have a valid Microsoft Outlook license, you can access Amazon WorkMail using the following versions of Microsoft Outlook:

- Outlook 2007, Outlook 2010, Outlook 2013, Outlook 2016, and Outlook 2019
- Outlook 2010 and Outlook 2013 Click-to-Run
- Outlook for Mac 2011, Outlook 2016 for Mac, and Outlook 2019 for Mac

The Amazon WorkMail web application is accessed at `https://alias.awsapps.com/mail`. Amazon WorkMail can also be used with IMAP clients. For more information, see Setting up email clients for Amazon WorkMail in the *Amazon WorkMail User Guide*.

Amazon WorkMail concepts

The terminology and concepts that are central to your understanding and use of Amazon WorkMail are described below.

Organization

A tenant setup for Amazon WorkMail.

Alias

A globally unique name to identify your organization. The alias is used to access the Amazon WorkMail web application (https://alias.awsapps.com/mail).

Domain

The web address that comes after the @ symbol in an email address. You can add a domain that receives mail and delivers it to mailboxes in your organization.

Test mail domain

A domain is automatically configured during setup that can be used for testing Amazon WorkMail. The test mail domain is *alias*.awsapps.com and is used as the default domain if you do not configure your own domain. The test mail domain is subject to different limits. For more information, see Amazon WorkMail quotas (p. 50).

Directory

An AWS Simple AD, AWS Managed AD, or AD Connector created in AWS Directory Service. If you create an organization using the Amazon WorkMail Quick setup, we create a WorkMail directory for you. You cannot view a WorkMail directory in AWS Directory Service.

User

A user created in the AWS Directory Service. When a user is enabled for Amazon WorkMail, they receive their own mailbox to access. When a user is disabled, they cannot access Amazon WorkMail.

Group

A group used in AWS Directory Service. A group can be used as a distribution list or a security group in Amazon WorkMail. Groups do not have their own mailboxes.

Resource

A resource represents a meeting room or equipment resource that can be booked by Amazon WorkMail users.

Mobile device policy

Various IT policy rules that control the security features and behavior of a mobile device.

Related AWS services

The following services are used along with Amazon WorkMail:

- **AWS Directory Service**—You can integrate Amazon WorkMail with an existing AWS Simple AD, AWS Managed AD, or AD Connector. Create a directory in the AWS Directory Service and then enable Amazon WorkMail for this directory. After you've configured this integration, you can choose which users you would like to enable for Amazon WorkMail from a list of users in your existing directory, and users can log in using their existing Active Directory credentials. For more information, see AWS Directory Service Administration Guide.
- **Amazon Simple Email Service**—Amazon WorkMail uses Amazon SES to send all outgoing email. The test mail domain and your domains are available for management in the Amazon SES console. There is no cost for outgoing email sent from Amazon WorkMail. For more information, see Amazon Simple Email Service Developer Guide.
- **AWS Identity and Access Management**—The AWS Management Console requires your user name and password so that any service you use can determine whether you have permission to access its resources. We recommend that you avoid using AWS account credentials to access AWS because AWS account credentials cannot be revoked or limited in any way. Instead, we recommend that you create an IAM user and add the user to an IAM group with administrative permissions. You can then access the console using the IAM user credentials.

  If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console. For more information, see Create individual IAM users in the *IAM User Guide*.

- **AWS Key Management Service**—Amazon WorkMail is integrated with AWS KMS for encryption of customer data. Key management can be performed from the AWS KMS console. For more information, see What is the AWS Key Management Service in the *AWS Key Management Service Developer Guide*.

Amazon WorkMail pricing

With Amazon WorkMail, there are no upfront fees or commitments. You pay only for active user accounts. For more specific information about pricing, see Pricing.

Amazon WorkMail resources

The following related resources can help you as you work with this service.

- **Classes & Workshops** – Links to role-based and specialty courses as well as self-paced labs to help sharpen your AWS skills and gain practical experience.
- **AWS Developer Tools** – Links to developer tools, SDKs, IDE toolkits, and command line tools for developing and managing AWS applications.
- **AWS Whitepapers** – Links to a comprehensive list of technical AWS whitepapers, covering topics such as architecture, security, and economics and authored by AWS Solutions Architects or other technical experts.
- **AWS Support Center** – The hub for creating and managing your AWS Support cases. Also includes links to other helpful resources, such as forums, technical FAQs, service health status, and AWS Trusted Advisor.
- **AWS Support** – The primary webpage for information about AWS Support, a one-on-one, fast-response support channel to help you build and run applications in the cloud.
- **Contact Us** – A central contact point for inquiries concerning AWS billing, account, events, abuse, and other issues.
- **AWS Site Terms** – Detailed information about our copyright and trademark; your account, license, and site access; and other topics.

Prerequisites

To use Amazon WorkMail you'll need an AWS account. If you haven't signed up for AWS yet, complete the following tasks to get set up.

Topics

Get an AWS account and your root user credentials

To access AWS, you must sign up for an AWS account.

To sign up for an AWS account

1. Open https://portal.aws.amazon.com/billing/signup.
2. Follow the online instructions.

   Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

AWS sends you a confirmation email after the sign-up process is complete. At any time, you can view your current account activity and manage your account by going to https://aws.amazon.com/ and choosing **My Account**.

Access keys consist of an access key ID and secret access key, which are used to sign programmatic requests that you make to AWS. If you don't have access keys, you can create them from the AWS Management Console. As a best practice, do not use the AWS account root user access keys for any task where it's not required. Instead, create a new administrator IAM user with access keys for yourself.

The only time that you can view or download the secret access key is when you create the keys. You cannot recover them later. However, you can create new access keys at any time. You must also have permissions to perform the required IAM actions. For more information, see Permissions Required to Access IAM Resources in the *IAM User Guide*.

To create access keys for an IAM user

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane, choose **Users**.
3. Choose the name of the user whose access keys you want to create, and then choose the **Security credentials** tab.
4. In the **Access keys** section, choose **Create access key**.
5. To view the new access key pair, choose **Show**. You will not have access to the secret access key again after this dialog box closes. Your credentials will look something like this:

   - Access key ID: AKIAIOSFODNN7EXAMPLE
   - Secret access key: wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
6. To download the key pair, choose **Download .csv file**. Store the keys in a secure location. You will not have access to the secret access key again after this dialog box closes.

Keep the keys confidential in order to protect your AWS account and never email them. Do not share them outside your organization, even if an inquiry appears to come from AWS or Amazon.com. No one who legitimately represents Amazon will ever ask you for your secret key.

7. After you download the `.csv` file, choose **Close**. When you create an access key, the key pair is active by default, and you can use the pair right away.

Related topics

- What Is IAM? in the *IAM User Guide*
- AWS Security Credentials in *AWS General Reference*

Create AWS Identity and Access Management users and groups

The AWS Management Console requires your username and password so that the service can determine whether you have permission to access its resources. We recommend that you avoid using root account credentials to access AWS because root account credentials cannot be revoked or limited in any way. Instead, use AWS Identity and Access Management (IAM) to create an IAM user and add the user to an IAM group with administrative permissions. You can then access the console using the credentials for the IAM user.

If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM console. For more information, see Create Individual IAM Users in *IAM User Guide*.

Grant IAM users permissions for Amazon WorkMail

By default, IAM users don't have permissions to manage Amazon WorkMail resources; you must attach an AWS managed policy (**AmazonWorkMailFullAccess** or **AmazonWorkMailReadOnlyAccess**) or create a customer managed policy that explicitly grants IAM users those permissions, and attach the policy to the specific IAM users or groups that require those permissions. For more information, see Identity and access management for Amazon WorkMail (p. 13).

Security in Amazon WorkMail

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security *of* the cloud and security *in* the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS compliance programs. To learn about the compliance programs that apply to Amazon WorkMail, see AWS Services in Scope by Compliance Program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using Amazon WorkMail. The following topics show you how to configure Amazon WorkMail to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Amazon WorkMail resources.

Topics

Data protection in Amazon WorkMail

The AWS shared responsibility model applies to data protection in Amazon WorkMail. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. This content includes the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see the Data Privacy FAQ. For information about data protection in Europe, see the AWS Shared Responsibility Model and GDPR blog post on the *AWS Security Blog*.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM). That way each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.

- Use SSL/TLS to communicate with AWS resources. We recommend TLS 1.2 or later.
- Set up API and user activity logging with AWS CloudTrail.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.
- If you require FIPS 140-2 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-2.

We strongly recommend that you never put sensitive identifying information, such as your customers' account numbers, into free-form fields such as a **Name** field. This includes when you work with Amazon WorkMail or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into Amazon WorkMail or other services might get picked up for inclusion in diagnostic logs. When you provide a URL to an external server, don't include credentials information in the URL to validate your request to that server.

How Amazon WorkMail uses AWS KMS

Amazon WorkMail transparently encrypts all messages in the mailboxes of all Amazon WorkMail organizations before the messages are written to disk and transparently decrypts the messages when users access them. There is no option to disable encryption. To protect the encryption keys that protect the messages, Amazon WorkMail is integrated with AWS Key Management Service (AWS KMS).

Amazon WorkMail also provides an option for enabling users to send signed or encrypted email. This encryption feature does not use AWS KMS. For more information, see Enabling signed or encrypted email (p. 99).

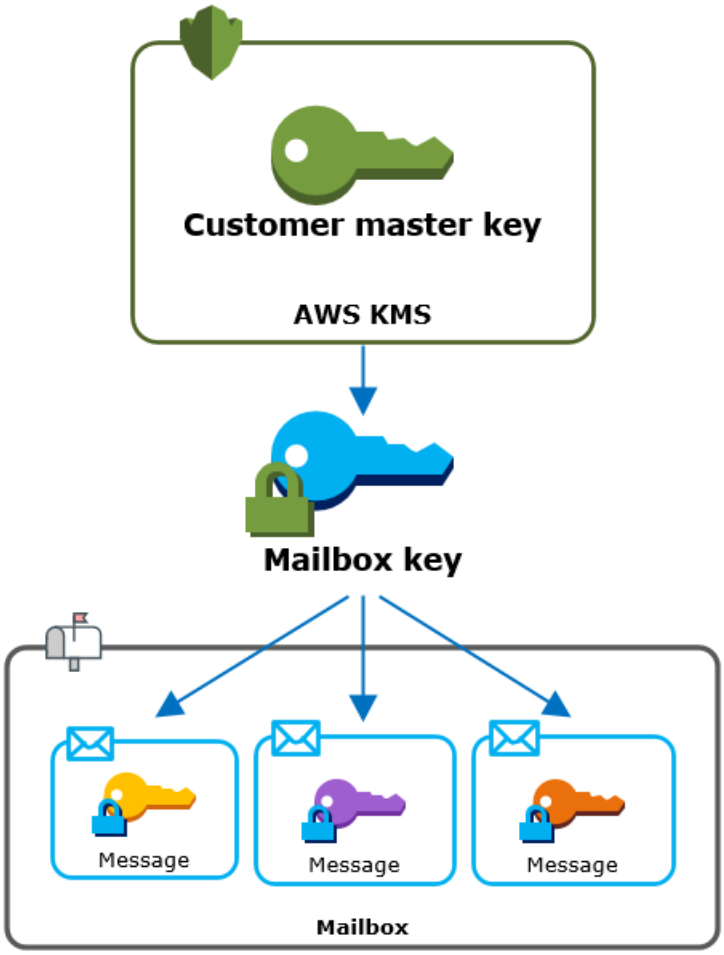
Topics

Amazon WorkMail encryption

In Amazon WorkMail, each organization can contain multiple mailboxes, one for each user in the organization. All messages, including email and calendar items, are stored in the user's mailbox.

To protect the contents of the mailboxes in your Amazon WorkMail organizations, Amazon WorkMail encrypts all mailbox messages before they are written to disk. No customer-provided information is stored in plaintext.

Each message is encrypted under a unique data encryption key. The message key is protected by a mailbox key, which is a unique encryption key that is used only for that mailbox. The mailbox key is encrypted under an AWS KMS customer master key (CMK) for the organization that never leaves AWS KMS unencrypted. The following diagram shows the relationship of the encrypted messages, encrypted message keys, encrypted mailbox key, and the CMK for the organization in AWS KMS.



A CMK for the organization

When you create an Amazon WorkMail organization, you have the option to select an AWS KMS customer master key (CMK) for the organization. This CMK protects all mailbox keys in that organization.

You can either select the default AWS managed CMK for Amazon WorkMail, or select an existing customer managed CMK that you own and manage. For more information, see customer master keys (CMKs) in the *AWS Key Management Service Developer Guide*. You can select the same CMK or a different CMK for each of your organizations, but you cannot change the CMK once you have selected it.

> **Important**
> Amazon WorkMail supports only symmetric CMKs. You cannot use an asymmetric CMK to encrypt data in Amazon WorkMail. For help determining whether a CMK is symmetric or asymmetric, see Identifying symmetric and asymmetric CMKs in the *AWS Key Management Service Developer Guide*.

To find the CMK for your organization, use the AWS CloudTrail log entry that records calls to AWS KMS.

A unique encryption key for each mailbox

When you create a new mailbox, Amazon WorkMail generates a unique 256-bit Advanced Encryption Standard (AES) symmetric encryption key for the mailbox, known as its *mailbox key*, outside of AWS KMS. Amazon WorkMail uses the mailbox key to protect the encryption keys for each message in the mailbox.

To protect the mailbox key, Amazon WorkMail calls AWS KMS to encrypt the mailbox key under the CMK for the organization. Then it stores the encrypted mailbox key in the mailbox metadata.

Note
Amazon WorkMail uses a symmetric mailbox encryption key to protect message keys. Previously, Amazon WorkMail protected each mailbox with an asymmetric key pair. It used the public key to encrypt each message key and the private key to decrypt it. The private mailbox key was protected by the CMK for the organization. Existing mailboxes might still use an asymmetric mailbox key pair. This change does not affect the security of the mailbox or its messages.

A unique encryption key for each message

When a message is added to the mailbox, Amazon WorkMail generates a unique 256-bit AES symmetric encryption key for the message outside of AWS KMS. It uses this *message key* to encrypt the message. Amazon WorkMail encrypts the message key under the mailbox key and stores the encrypted message key with the message. Then, it encrypts the mailbox key under the CMK for the organization.

Creating a new mailbox

When Amazon WorkMail creates a new mailbox, it uses the following process to prepare the mailbox to hold encrypted messages.

- Amazon WorkMail generates a unique 256-bit AES symmetric encryption key for the mailbox outside of AWS KMS.
- Amazon WorkMail calls the AWS KMS Encrypt operation. It passes in the mailbox key and the identifier of the customer master key (CMK) for the organization. AWS KMS returns a ciphertext of the mailbox key encrypted under the CMK.
- Amazon WorkMail stores the encrypted mailbox key with the mailbox metadata.

Encrypting a mailbox message

To encrypt a message, Amazon WorkMail uses the following process.

1. Amazon WorkMail generates a unique 256-bit AES symmetric key for the message. It uses the plaintext message key and the Advanced Encryption Standard (AES) algorithm to encrypt the message outside of AWS KMS.
2. To protect the message key under the mailbox key, Amazon WorkMail needs to decrypt the mailbox key, which is always stored in its encrypted form.

   Amazon WorkMail calls the AWS KMS Decrypt operation and passes in the encrypted mailbox key. AWS KMS uses the CMK for the organization to decrypt the mailbox key and it returns the plaintext mailbox key to Amazon WorkMail.
3. Amazon WorkMail uses the plaintext mailbox key and the Advanced Encryption Standard (AES) algorithm to encrypt the message key outside of AWS KMS.
4. Amazon WorkMail stores the encrypted message key in the metadata of the encrypted message so it is available to decrypt it.

Decrypting a mailbox message

To decrypt a message, Amazon WorkMail uses the following process.

1. Amazon WorkMail calls the AWS KMS Decrypt operation and passes in the encrypted mailbox key. AWS KMS uses the CMK for the organization to decrypt the mailbox key and it returns the plaintext mailbox key to Amazon WorkMail.
2. Amazon WorkMail uses the plaintext mailbox key and the Advanced Encryption Standard (AES) algorithm to decrypt the encrypted message key outside of AWS KMS.

3. Amazon WorkMail uses the plaintext message key to decrypt the encrypted message.

Caching mailbox keys

To improve performance and minimize calls to AWS KMS, Amazon WorkMail caches each plaintext mailbox key for each client locally for up to one minute. At the end of the caching period, the mailbox key is removed. If the mailbox key for that client is required during the caching period, Amazon WorkMail can get it from the cache instead of calling AWS KMS. The mailbox key is protected in the cache and is never written to disk in plaintext.

Authorizing use of the CMK

When Amazon WorkMail uses a customer master key (CMK) in cryptographic operations, it acts on behalf of the mailbox administrator.

To use the AWS KMS customer master key (CMK) for a secret on your behalf, the administrator must have the following permissions. You can specify these required permissions in an IAM policy or key policy.

- `kms:Encrypt`
- `kms:Decrypt`
- `kms:CreateGrant`

To allow the CMK to be used only for requests that originate in Amazon WorkMail, you can use the kms:ViaService condition key with the `workmail.<region>.amazonaws.com` value.

You can also use the keys or values in the encryption context (p. 11) as a condition for using the CMK for cryptographic operations. For example, you can use a string condition operator in an IAM or key policy document or use a grant constraint in a grant.

Key policy for the AWS managed CMK

The key policy for the AWS managed CMK for Amazon WorkMail gives users permission to use the CMK for specified operations only when Amazon WorkMail makes the request on the user's behalf. The key policy does not allow any user to use the CMK directly.

This key policy, like the policies of all AWS managed keys, is established by the service. You cannot change the key policy, but you can view it at any time. For details, see Viewing a key policy in the *AWS Key Management Service Developer Guide*.

The policy statements in the key policy have the following effect:

- Allow users in the account and Region to use the CMK for cryptographic operations and to create grants, but only when the request comes from Amazon WorkMail on their behalf. The `kms:ViaService` condition key enforces this restriction.
- Allows the AWS account to create IAM policies that allow users to view CMK properties and revoke grants.

The following is a key policy for an example AWS managed CMK for Amazon WorkMail.

```
{
  "Version" : "2012-10-17",
  "Id" : "auto-workmail-1",
  "Statement" : [ {
    "Sid" : "Allow access through WorkMail for all principals in the account that are
 authorized to use WorkMail",
```

```
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
    },
    "Action" : [ "kms:Decrypt", "kms:CreateGrant", "kms:ReEncrypt*", "kms:DescribeKey",
 "kms:Encrypt" ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "workmail.us-east-1.amazonaws.com",
        "kms:CallerAccount" : "111122223333"
      }
    }
  }, {
    "Sid" : "Allow direct access to key metadata to the account",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : [ "kms:Describe*", "kms:List*", "kms:Get*", "kms:RevokeGrant" ],
    "Resource" : "*"
  } ]
}
```

Using grants to authorize Amazon WorkMail

In addition to key policies, Amazon WorkMail uses grants to add permissions to the CMK for each organization. To view the grants on the CMK in your account, use the ListGrants operation.

Amazon WorkMail uses grants to add the following permissions to the CMK for the organization.

- Add the `kms:Encrypt` permission to allow Amazon WorkMail to encrypt the mailbox key.
- Add the `kms:Decrypt` permission to allow Amazon WorkMail to use the CMK to decrypt the mailbox key. Amazon WorkMail requires this permission in a grant because the request to read mailbox messages uses the security context of the user who is reading the message. The request does not use the credentials of the AWS account. Amazon WorkMail creates this grant when you select a CMK for the organization.

To create the grants, Amazon WorkMail calls CreateGrant on behalf of the user who created the organization. Permission to create the grant comes from the key policy. This policy allows account users to call `CreateGrant` on the CMK for the organization when Amazon WorkMail makes the request on an authorized user's behalf.

The key policy also allows the account root to revoke the grant on the AWS managed key. However, if you revoke the grant, Amazon WorkMail cannot decrypt the encrypted data in your mailboxes.

Amazon WorkMail encryption context

An encryption context is a set of key-value pairs that contain arbitrary nonsecret data. When you include an encryption context in a request to encrypt data, AWS KMS cryptographically binds the encryption context to the encrypted data. To decrypt the data, you must pass in the same encryption context. For more information, see Encryption context in the *AWS Key Management Service Developer Guide*.

Amazon WorkMail uses the same encryption context format in all AWS KMS cryptographic operations. You can use the encryption context to identify a cryptographic operation in audit records and logs, such as AWS CloudTrail, and as a condition for authorization in policies and grants.

In its Encrypt and Decrypt requests to AWS KMS, Amazon WorkMail uses an encryption context where the key is `aws:workmail:arn` and the value is the Amazon Resource Name (ARN) of the organization.

```
"aws:workmail:arn":"arn:aws:workmail:region:account ID:organization/organization-ID"
```

For example, the following encryption context includes an example organization ARN in the Europe (Ireland) (`eu-west-1`) Region.

```
"aws:workmail:arn":"arn:aws:workmail:eu-west-1:111122223333:organization/m-
a123b4c5de678fg9h0ij1k2lm234no56"
```

Monitoring Amazon WorkMail interaction with AWS KMS

You can use AWS CloudTrail and Amazon CloudWatch Logs to track the requests that Amazon WorkMail sends to AWS KMS on your behalf.

Encrypt

When you create a new mailbox, Amazon WorkMail generates a mailbox key and calls AWS KMS to encrypt the mailbox key. Amazon WorkMail sends an Encrypt request to AWS KMS with the plaintext mailbox key and an identifier for the CMK of the Amazon WorkMail organization.

The event that records the `Encrypt` operation is similar to the following example event. The user is the Amazon WorkMail service. The parameters include the CMK ID (`keyId`) and the encryption context for the Amazon WorkMail organization. Amazon WorkMail also passes in the mailbox key, but that is not recorded in the CloudTrail log.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
    },
    "eventTime": "2019-02-19T10:01:09Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Encrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/m-
a123b4c5de678fg9h0ij1k2lm234no56"
        },
        "keyId": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    },
    "responseElements": null,
    "requestID": "76e96b96-7e24-4faf-a2d6-08ded2eaf63c",
    "eventID": "d5a59c18-128a-4082-aa5b-729f7734626a",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "d08e60f1-097e-4a00-b7e9-10bc3872d50c"
}
```

Decrypt

When you add, view, or delete a mailbox message, Amazon WorkMail asks AWS KMS to decrypt the mailbox key. Amazon WorkMail sends a Decrypt request to AWS KMS with the encrypted mailbox key and an identifier for the CMK of the Amazon WorkMail organization.

The event that records the `Decrypt` operation is similar to the following example event. The user is the Amazon WorkMail service. The parameters include the encrypted mailbox key (as a ciphertext blob), which is not recorded in the log, and the encryption context for the Amazon WorkMail organization. AWS KMS derives the ID of the CMK from the ciphertext.


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
    },
    "eventTime": "2019-02-20T11:51:10Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/m-
a123b4c5de678fg9h0ij1k2lm234no56"
        }
    },
    "responseElements": null,
    "requestID": "4a32dda1-34d9-4100-9718-674b8e0782c9",
    "eventID": "ea9fd966-98e9-4b7b-b377-6e5a397a71de",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "241e1e5b-ff64-427a-a5b3-7949164d0214"
}
```


Identity and access management for Amazon WorkMail

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be *authenticated* (signed in) and *authorized* (have permissions) to use Amazon WorkMail resources. IAM is an AWS service that you can use with no additional charge.

Topics

Audience

How you use AWS Identity and Access Management (IAM) differs, depending on the work that you do in Amazon WorkMail.

Service user – If you use the Amazon WorkMail service to do your job, then your administrator provides you with the credentials and permissions that you need. As you use more Amazon WorkMail features to do your work, you might need additional permissions. Understanding how access is managed can help you request the right permissions from your administrator. If you cannot access a feature in Amazon WorkMail, see Troubleshooting Amazon WorkMail identity and access (p. 25).

Service administrator – If you're in charge of Amazon WorkMail resources at your company, you probably have full access to Amazon WorkMail. It's your job to determine which Amazon WorkMail features and resources your employees should access. You must then submit requests to your IAM administrator to change the permissions of your service users. Review the information on this page to understand the basic concepts of IAM. To learn more about how your company can use IAM with Amazon WorkMail, see How Amazon WorkMail works with IAM (p. 17).

IAM administrator – If you're an IAM administrator, you might want to learn details about how you can write policies to manage access to Amazon WorkMail. To view example Amazon WorkMail identity-based policies that you can use in IAM, see Amazon WorkMail identity-based policy examples (p. 21).

Authenticating With identities

Authentication is how you sign in to AWS using your identity credentials. For more information about signing in using the AWS Management Console, see Signing in to the AWS Management Console as an IAM user or root user in the *IAM User Guide*.

You must be *authenticated* (signed in to AWS) as the AWS account root user, an IAM user, or by assuming an IAM role. You can also use your company's single sign-on authentication or even sign in using Google or Facebook. In these cases, your administrator previously set up identity federation using IAM roles. When you access AWS using credentials from another company, you are assuming a role indirectly.

To sign in directly to the AWS Management Console, use your password with your root user email address or your IAM user name. You can access AWS programmatically using your root user or IAM users access keys. AWS provides SDK and command line tools to cryptographically sign your request using your credentials. If you don't use AWS tools, you must sign the request yourself. Do this using *Signature Version 4*, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 signing process in the *AWS General Reference*.

Regardless of the authentication method that you use, you might also be required to provide additional security information. For example, AWS recommends that you use multi-factor authentication (MFA) to increase the security of your account. To learn more, see Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.

AWS account root user

When you first create an AWS account, you begin with a single sign-in identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account *root user* and is accessed by signing in with the email address and password that you used to create the account. We strongly recommend that you do not use the root user for your everyday tasks, even the administrative

ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks.

IAM users and groups

An *IAM user* is an identity within your AWS account that has specific permissions for a single person or application. An IAM user can have long-term credentials such as a user name and password or a set of access keys. To learn how to generate access keys, see Managing access keys for IAM users in the *IAM User Guide*. When you generate access keys for an IAM user, make sure you view and securely save the key pair. You cannot recover the secret access key in the future. Instead, you must generate a new access key pair.

An *IAM group* is an identity that specifies a collection of IAM users. You can't sign in as a group. You can use groups to specify permissions for multiple users at a time. Groups make permissions easier to manage for large sets of users. For example, you could have a group named *IAMAdmins* and give that group permissions to administer IAM resources.

Users are different from roles. A user is uniquely associated with one person or application, but a role is intended to be assumable by anyone who needs it. Users have permanent long-term credentials, but roles provide temporary credentials. To learn more, see When to create an IAM user (instead of a role) in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity within your AWS account that has specific permissions. It is similar to an IAM user, but is not associated with a specific person. You can temporarily assume an IAM role in the AWS Management Console by switching roles. You can assume a role by calling an AWS CLI or AWS API operation or by using a custom URL. For more information about methods for using roles, see Using IAM roles in the *IAM User Guide*.

IAM roles with temporary credentials are useful in the following situations:

- **Temporary IAM user permissions** – An IAM user can assume an IAM role to temporarily take on different permissions for a specific task.
- **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as *federated users*. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated users and roles in the *IAM User Guide*.
- **Cross-account access** – You can use an IAM role to allow someone (a trusted principal) in a different account to access resources in your account. Roles are the primary way to grant cross-account access. However, with some AWS services, you can attach a policy directly to a resource (instead of using a role as a proxy). To learn the difference between roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.
- **Cross-service access** – Some AWS services use features in other AWS services. For example, when you make a call in a service, it's common for that service to run applications in Amazon EC2 or store objects in Amazon S3. A service might do this using the calling principal's permissions, using a service role, or using a service-linked role.
  - **Principal permissions** – When you use an IAM user or role to perform actions in AWS, you are considered a principal. Policies grant permissions to a principal. When you use some services, you might perform an action that then triggers another action in a different service. In this case, you must have permissions to perform both actions. To see whether an action requires additional dependent actions in a policy, see in the *Service Authorization Reference*.
  - **Service role** – A service role is an IAM role that a service assumes to perform actions on your behalf. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. An IAM administrator can create, modify, and delete a service role from within

IAM. For more information, see Creating a role to delegate permissions to an AWS service in the *IAM User Guide*.

- **Service-linked role** – A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

- **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the EC2 instance. To assign an AWS role to an EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the EC2 instance to get temporary credentials. For more information, see Using an IAM role to grant permissions to applications running on Amazon EC2 instances in the *IAM User Guide*.

To learn whether to use IAM roles or IAM users, see When to create an IAM role (instead of a user) in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to IAM identities or AWS resources. A policy is an object in AWS that, when associated with an identity or resource, defines their permissions. You can sign in as the root user or an IAM user, or you can assume an IAM role. When you then make a request, AWS evaluates the related identity-based or resource-based policies. Permissions in the policies determine whether the request is allowed or denied. Most policies are stored in AWS as JSON documents. For more information about the structure and contents of JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

Every IAM entity (user or role) starts with no permissions. In other words, by default, users can do nothing, not even change their own password. To give a user permission to do something, an administrator must attach a permissions policy to a user. Or the administrator can add the user to a group that has the intended permissions. When an administrator gives permissions to a group, all users in that group are granted those permissions.

IAM policies define permissions for an action regardless of the method that you use to perform the operation. For example, suppose that you have a policy that allows the `iam:GetRole` action. A user with that policy can get role information from the AWS Management Console, the AWS CLI, or the AWS API.

Identity-based policies

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Creating IAM policies in the *IAM User Guide*.

Identity-based policies can be further categorized as *inline policies* or *managed policies*. Inline policies are embedded directly into a single user, group, or role. Managed policies are standalone policies that you can attach to multiple users, groups, and roles in your AWS account. Managed policies include AWS managed policies and customer managed policies. To learn how to choose between a managed policy or an inline policy, see Choosing between managed policies and inline policies in the *IAM User Guide*.

Resource-based policies

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM *role trust policies* and Amazon S3 *bucket policies*. In services that support resource-

based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.

Resource-based policies are inline policies that are located in that service. You can't use AWS managed policies from IAM in a resource-based policy.

Access Control Lists (ACLs)

Access control lists (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they do not use the JSON policy document format.

Amazon S3, AWS WAF, and Amazon VPC are examples of services that support ACLs. To learn more about ACLs, see Access control list (ACL) overview in the *Amazon Simple Storage Service Developer Guide*.

Other policy types

AWS supports additional, less-common policy types. These policy types can set the maximum permissions granted to you by the more common policy types.

- **Permissions boundaries** – A permissions boundary is an advanced feature in which you set the maximum permissions that an identity-based policy can grant to an IAM entity (IAM user or role). You can set a permissions boundary for an entity. The resulting permissions are the intersection of entity's identity-based policies and its permissions boundaries. Resource-based policies that specify the user or role in the `Principal` field are not limited by the permissions boundary. An explicit deny in any of these policies overrides the allow. For more information about permissions boundaries, see Permissions boundaries for IAM entities in the *IAM User Guide*.
- **Service control policies (SCPs)** – SCPs are JSON policies that specify the maximum permissions for an organization or organizational unit (OU) in AWS Organizations. AWS Organizations is a service for grouping and centrally managing multiple AWS accounts that your business owns. If you enable all features in an organization, then you can apply service control policies (SCPs) to any or all of your accounts. The SCP limits permissions for entities in member accounts, including each AWS account root user. For more information about Organizations and SCPs, see How SCPs work in the *AWS Organizations User Guide*.
- **Session policies** – Session policies are advanced policies that you pass as a parameter when you programmatically create a temporary session for a role or federated user. The resulting session's permissions are the intersection of the user or role's identity-based policies and the session policies. Permissions can also come from a resource-based policy. An explicit deny in any of these policies overrides the allow. For more information, see Session policies in the *IAM User Guide*.

Multiple policy types

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see Policy evaluation logic in the *IAM User Guide*.

How Amazon WorkMail works with IAM

Before you use IAM to manage access to Amazon WorkMail, you should understand what IAM features are available to use with Amazon WorkMail. To get a high-level view of how Amazon WorkMail and other AWS services work with IAM, see AWS services that work with IAM in the *IAM User Guide*.

Topics

Amazon WorkMail identity-based policies

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. Amazon WorkMail supports specific actions, resources, and condition keys. To learn about all of the elements that you use in a JSON policy, see IAM JSON Policy Elements Reference in the *IAM User Guide*.

Actions

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Action` element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Policy actions usually have the same name as the associated AWS API operation. There are some exceptions, such as *permission-only actions* that don't have a matching API operation. There are also some operations that require multiple actions in a policy. These additional actions are called *dependent actions*.

Include actions in a policy to grant permissions to perform the associated operation.

Policy actions in Amazon WorkMail use the following prefix before the action: `workmail:`. For example, to grant someone permission to retrieve a list of users with the Amazon WorkMail `ListUsers` API operation, you include the `workmail:ListUsers` action in their policy. Policy statements must include either an `Action` or `NotAction` element. Amazon WorkMail defines its own set of actions that describe tasks that you can perform with this service.

To specify multiple actions in a single statement, separate them with commas as follows:

```
"Action": [
      "workmail:ListUsers",
      "workmail:DeleteUser"
```

You can specify multiple actions using wildcards (*). For example, to specify all actions that begin with the word `List`, include the following action:

```
"Action": "workmail:List*"
```

To see a list of Amazon WorkMail actions, see Actions defined by Amazon WorkMail in the *IAM User Guide*.

Resources

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Resource` JSON policy element specifies the object or objects to which the action applies. Statements must include either a `Resource` or a `NotResource` element. As a best practice, specify a resource using its Amazon Resource Name (ARN). You can do this for actions that support a specific resource type, known as *resource-level permissions*.

For actions that don't support resource-level permissions, such as listing operations, use a wildcard (*) to indicate that the statement applies to all resources.

```
"Resource": "*"
```

Amazon WorkMail supports resource-level permissions for Amazon WorkMail organizations.

The Amazon WorkMail organization resource has the following ARN:

```
arn:aws:workmail:${Region}:${Account}:organization/${OrganizationId}
```

For more information about the format of ARNs, see Amazon Resource Names (ARNs) and AWS service namespaces.

For example, to specify the `m-n1pq2345678r901st2u3vx45x6789yza` organization in your statement, use the following ARN.

```
"Resource": "arn:aws:workmail:us-east-1:111122223333:organization/m-
n1pq2345678r901st2u3vx45x6789yza"
```

To specify all organizations that belong to a specific account, use the wildcard (*):

```
"Resource": "arn:aws:workmail:us-east-1:111122223333:organization/*"
```

Some Amazon WorkMail actions, such as those for creating resources, cannot be performed on a specific resource. In those cases, you must use the wildcard (*).

```
"Resource": "*"
```

To see a list of Amazon WorkMail resource types and their ARNs, see Resources defined by Amazon WorkMail in the *IAM User Guide*. To learn with which actions you can specify for the ARN of each resource, see Actions, resources, and condition keys for Amazon WorkMail.

Condition keys

Amazon WorkMail does not provide any service-specific condition keys, but it does support using the following global condition keys.

- `aws:CurrentTime`
- `aws:EpochTime`
- `aws:MultiFactorAuthAge`
- `aws:MultiFactorAuthPresent`
- `aws:PrincipalOrgID`
- `aws:PrincipalArn`
- `aws:RequestedRegion`
- `aws:SecureTransport`
- `aws:UserAgent`

The following example policy grants access to the Amazon WorkMail console only from MFA authenticated IAM principals in the `eu-west-1` AWS Region.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ses:Describe*",
                "ses:Get*",
                "workmail:Describe*",
                "workmail:Get*",
                "workmail:List*",
                "workmail:Search*",
                "lambda:ListFunctions",
                "iam:ListRoles",
                "logs:DescribeLogGroups",
                "cloudwatch:GetMetricData"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestedRegion": [
                        "eu-west-1"
                    ]
                },
                "Bool": {
                    "aws:MultiFactorAuthPresent": true
                }
            }
        }
    ]
}
```

To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

Examples

To view examples of Amazon WorkMail identity-based policies, see Amazon WorkMail identity-based policy examples (p. 21).

Amazon WorkMail resource-based policies

Amazon WorkMail does not support resource-based policies.

Authorization based on Amazon WorkMail tags

You can attach tags to Amazon WorkMail resources or pass tags in a request to Amazon WorkMail. To control access based on tags, you provide tag information in the condition element of a policy using the `workmail:ResourceTag/key-name`, `aws:RequestTag/key-name`, or `aws:TagKeys` condition keys. For more information about tagging Amazon WorkMail resources, see Tagging an organization (p. 57).

Amazon WorkMail IAM roles

An IAM role is an entity within your AWS account that has specific permissions.

Using temporary credentials with Amazon WorkMail

You can use temporary credentials to sign in with federation, assume an IAM role, or to assume a cross-account role. You obtain temporary security credentials by calling AWS STS API operations such as AssumeRole or GetFederationToken.

Amazon WorkMail supports using temporary credentials.

Service-linked roles

Service-linked roles allow AWS services to access resources in other services to complete an action on your behalf. Service-linked roles appear in your IAM account and are owned by the service. An IAM administrator can view but not edit the permissions for service-linked roles.

Amazon WorkMail supports service-linked roles. For details about creating or managing Amazon WorkMail service-linked roles, see Using service-linked roles for Amazon WorkMail (p. 26).

Service roles

This feature allows a service to assume a service role on your behalf. This role allows the service to access resources in other services to complete an action on your behalf. Service roles appear in your IAM account and are owned by the account. This means that an IAM administrator can change the permissions for this role. However, doing so might break the functionality of the service.

Amazon WorkMail supports service roles.

Amazon WorkMail identity-based policy examples

By default, IAM users and roles don't have permission to create or modify Amazon WorkMail resources. They also can't perform tasks using the AWS Management Console, AWS CLI, or AWS API. An IAM administrator must create IAM policies that grant users and roles permission to perform specific API operations on the specified resources they need. The administrator must then attach those policies to the IAM users or groups that require those permissions.

To learn how to create an IAM identity-based policy using these example JSON policy documents, see Creating policies on the JSON tab in the *IAM User Guide*.

Topics

Policy best practices

Identity-based policies are very powerful. They determine whether someone can create, access, or delete Amazon WorkMail resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get started using AWS managed policies** – To start using Amazon WorkMail quickly, use AWS managed policies to give your employees the permissions they need. These policies are already available in your account and are maintained and updated by AWS. For more information, see Get started using permissions with AWS managed policies in the *IAM User Guide*.
- **Grant least privilege** – When you create custom policies, grant only the permissions required to perform a task. Start with a minimum set of permissions and grant additional permissions as necessary. Doing so is more secure than starting with permissions that are too lenient and then trying to tighten them later. For more information, see Grant least privilege in the *IAM User Guide*.
- **Enable MFA for sensitive operations** – For extra security, require IAM users to use multi-factor authentication (MFA) to access sensitive resources or API operations. For more information, see Using multi-factor authentication (MFA) in AWS in the *IAM User Guide*.
- **Use policy conditions for extra security** – To the extent that it's practical, define the conditions under which your identity-based policies allow access to a resource. For example, you can write conditions to specify a range of allowable IP addresses that a request must come from. You can also write conditions

to allow requests only within a specified date or time range, or to require the use of SSL or MFA. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.

Using the Amazon WorkMail console

To access the Amazon WorkMail console, you must have a minimum set of permissions. These permissions must allow you to list and view details about the Amazon WorkMail resources in your AWS account. If you create an identity-based policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities (IAM users or roles) with that policy.

To ensure that those entities can still use the Amazon WorkMail console, also attach the following AWS managed policy, **AmazonWorkMailFullAccess**, to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*.

The **AmazonWorkMailFullAccess** policy grants an IAM user full access to Amazon WorkMail resources. This policy gives the user access to all Amazon WorkMail, AWS Key Management Service, Amazon Simple Email Service, and AWS Directory Service operations. This also includes several Amazon EC2 operations that Amazon WorkMail needs to perform on your behalf. The `logs` and `cloudwatch` permissions are required for email event logging and viewing metrics in the Amazon WorkMail console. For more information, see Logging and monitoring in Amazon WorkMail (p. 28).

```
{
        "Version": "2012-10-17",
        "Statement": [
                        {
                          "Effect": "Allow",
                          "Action": [
                                     "ds:AuthorizeApplication",
                                     "ds:CheckAlias",
                                     "ds:CreateAlias",
                                     "ds:CreateDirectory",
                                     "ds:CreateIdentityPoolDirectory",
                                     "ds:DeleteAlias",
                                     "ds:DeleteDirectory",
                                     "ds:DescribeDirectories",
                                     "ds:GetDirectoryLimits",
                                     "ds:ListAuthorizedApplications",
                                     "ds:UnauthorizeApplication",
                                     "ec2:AuthorizeSecurityGroupEgress",
                                     "ec2:AuthorizeSecurityGroupIngress",
                                     "ec2:CreateNetworkInterface",
                                     "ec2:CreateSecurityGroup",
                                     "ec2:CreateSubnet",
                                     "ec2:CreateTags",
                                     "ec2:CreateVpc",
                                     "ec2:DeleteSecurityGroup",
                                     "ec2:DeleteSubnet",
                                     "ec2:DeleteVpc",
                                     "ec2:DescribeAvailabilityZones",
                                     "ec2:DescribeRouteTables",
                                     "ec2:DescribeSubnets",
                                     "ec2:DescribeVpcs",
                                     "ec2:RevokeSecurityGroupEgress",
                                     "ec2:RevokeSecurityGroupIngress",
                                     "kms:DescribeKey",
                                     "kms:ListAliases"
                                     "lambda:ListFunctions",
                                     "route53:ChangeResourceRecordSets",
                                     "route53:ListHostedZones",
                                     "route53:ListResourceRecordSets",
                                     "route53domains:CheckDomainAvailability",
                                     "route53domains:ListDomains",
```

```
                                    "ses:*",
                                    "workmail:*",
                                    "iam:ListRoles",
                                    "logs:DescribeLogGroups",
                                    "logs:CreateLogGroup",
                                    "logs:PutRetentionPolicy",
                                    "cloudwatch:GetMetricData"
                                    ],
                            "Resource": "*"
                        },
                        {
                            "Effect": "Allow",
                            "Action": "iam:CreateServiceLinkedRole",
                            "Resource": "*",
                            "Condition": {
                                "StringEquals": {
                                    "iam:AWSServiceName": "events.workmail.amazonaws.com"
                                }
                            }
                        },
                        {
                            "Effect": "Allow",
                            "Action": [
                                "iam:DeleteServiceLinkedRole",
                                "iam:GetServiceLinkedRoleDeletionStatus"
                             ],
                            "Resource": "arn:aws:iam::*:role/aws-service-role/
events.workmail.amazonaws.com/AWSServiceRoleForAmazonWorkMailEvents*"
                        },
                        {
                            "Effect": "Allow",
                            "Action": "iam:PassRole",
                            "Resource": "arn:aws:iam::*:role/*workmail*",
                            "Condition": {
                                "StringLike": {
                                    "iam:PassedToService": "events.workmail.amazonaws.com"
                                }
                            }
                         }
    ]
}
```

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, allow access to only the actions that match the API operation that you're trying to perform.

Allow users to view their own permissions

This example shows how you might create a policy that allows IAM users to view the inline and managed policies that are attached to their user identity. This policy includes permissions to complete this action on the console or programmatically using the AWS CLI or AWS API.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
```

```
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Allow users read-only access to Amazon WorkMail resources

The following policy statement grants an IAM user read-only access to Amazon WorkMail resources. This policy gives the same level of access as the AWS managed policy **AmazonWorkMailReadOnlyAccess**. Either policy gives the user access to all of the Amazon WorkMail `Describe` operations. Access to the AWS Directory Service `DescribeDirectories` operation is needed to obtain information about your AWS Directory Service directories. Access to the Amazon SES service is needed to obtain information about the configured domains. Access to AWS Key Management Service is needed to obtain information about the used encryption keys. The `logs` and `cloudwatch` permissions are required for email event logging and viewing metrics in the Amazon WorkMail console. For more information, see Logging and monitoring in Amazon WorkMail (p. 28).

```
{
        "Version": "2012-10-17",
        "Statement": [
                        {
                         "Effect": "Allow",
                         "Action": [
                                    "ses:Describe*",
                                    "ses:Get*",
                                    "workmail:Describe*",
                                    "workmail:Get*",
                                    "workmail:List*",
                                    "workmail:Search*",
                                    "lambda:ListFunctions",
                                    "iam:ListRoles",
                                    "logs:DescribeLogGroups",
                                    "cloudwatch:GetMetricData"
                                   ],
                         "Resource": "*"
                        }
                ]
  }
}
```

Troubleshooting Amazon WorkMail identity and access

Use the following information to help you diagnose and fix common issues that you might encounter when working with Amazon WorkMail and IAM.

Topics

I am not authorized to perform an action in Amazon WorkMail

If the AWS Management Console tells you that you're not authorized to perform an action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password.

The following example error occurs when the `mateojackson` IAM user tries to use the console to view details about a group but does not have `workmail:DescribeGroup` permissions.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 workmail:DescribeGroup on resource: group
```

In this case, Mateo asks his administrator to update his policies to allow him to access the `group` resource using the `workmail:DescribeGroup` action.

I am not authorized to perform iam:PassRole

If you receive an error that you're not authorized to perform the `iam:PassRole` action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password. Ask that person to update your policies to allow you to pass a role to Amazon WorkMail.

Some AWS services allow you to pass an existing role to that service, instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in Amazon WorkMail. However, the action requires the service to have permissions granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform: iam:PassRole
```

In this case, Mary asks her administrator to update her policies to allow her to perform the `iam:PassRole` action.

I want to view my access keys

After you create your IAM user access keys, you can view your access key ID at any time. However, you can't view your secret access key again. If you lose your secret key, you must create a new access key pair.

Access keys consist of two parts: an access key ID (for example, `AKIAIOSFODNN7EXAMPLE`) and a secret access key (for example, `wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY`). Like a user name and password, you must use both the access key ID and secret access key together to authenticate your requests. Manage your access keys as securely as you do your user name and password.

Important
Do not provide your access keys to a third party, even to help find your canonical user ID. By doing this, you might give someone permanent access to your account.

When you create an access key pair, you are prompted to save the access key ID and secret access key in a secure location. The secret access key is available only at the time you create it. If you lose your secret access key, you must add new access keys to your IAM user. You can have a maximum of two access keys. If you already have two, you must delete one key pair before creating a new one. To view instructions, see Managing access keys in the *IAM User Guide*.

I'm an administrator and want to allow others to access Amazon WorkMail

To allow others to access Amazon WorkMail, you must create an IAM entity (user or role) for the person or application that needs access. They will use the credentials for that entity to access AWS. You must then attach a policy to the entity that grants them the correct permissions in Amazon WorkMail.

To get started right away, see Creating your first IAM delegated user and group in the *IAM User Guide*.

I want to allow people outside of my AWS sccount to access my Amazon WorkMail resources

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether Amazon WorkMail supports these features, see How Amazon WorkMail works with IAM (p. 17).
- To learn how to provide access to your resources across AWS accounts that you own, see Providing access to an IAM user in another AWS account that you own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing access to AWS accounts owned by third parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing access to externally authenticated users (identity federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see How IAM roles differ from resource-based policies in the *IAM User Guide*.

Using service-linked roles for Amazon WorkMail

Amazon WorkMail uses AWS Identity and Access Management (IAM) service-linked roles. A service-linked role is a unique type of IAM role that is linked directly to Amazon WorkMail. Service-linked roles are predefined by Amazon WorkMail and include all the permissions that the service requires to call other AWS services on your behalf.

A service-linked role makes setting up Amazon WorkMail easier because you don't have to manually add the necessary permissions. Amazon WorkMail defines the permissions of its service-linked roles, and

unless defined otherwise, only Amazon WorkMail can assume its roles. The defined permissions include the trust policy and the permissions policy, and that permissions policy cannot be attached to any other IAM entity.

You can delete a service-linked role only after first deleting the related resources. This protects your Amazon WorkMail resources because you can't inadvertently remove permission to access the resources.

For information about other services that support service-linked roles, see AWS services that work with IAM and look for the services that have **Yes** in the **Service-linked role** column. Choose a **Yes** with a link to view the service-linked role documentation for that service.

Service-linked role permissions for Amazon WorkMail

Amazon WorkMail uses the service-linked role named **AmazonWorkMailEvents** – Amazon WorkMail uses this service-linked role to enable access to AWS services and resources used or managed by Amazon WorkMail events, such as monitoring email events logged by CloudWatch. For more information about enabling email event logging for Amazon WorkMail, see Tracking messages (p. 75).

The AmazonWorkMailEvents service-linked role trusts the following services to assume the role:

- `events.workmail.amazonaws.com`

The role permissions policy allows Amazon WorkMail to complete the following actions on the specified resources:

- Action: `logs:CreateLogGroup` on `all AWS resources`
- Action: `logs:CreateLogStream` on `all AWS resources`
- Action: `logs:PutLogEvents` on `all AWS resources`

You must configure permissions to allow an IAM entity (such as a user, group, or role) to create, edit, or delete a service-linked role. For more information, see Service-linked role permissions in the *IAM User Guide*.

Creating a service-linked role for Amazon WorkMail

You don't need to manually create a service-linked role. When you turn on Amazon WorkMail event logging and use the default settings in the Amazon WorkMail console, Amazon WorkMail creates the service-linked role for you.

If you delete this service-linked role, and then need to create it again, you can use the same process to recreate the role in your account. When you turn on Amazon WorkMail event logging and use the default settings, Amazon WorkMail creates the service-linked role for you again.

Editing a service-linked role for Amazon WorkMail

Amazon WorkMail does not allow you to edit the AmazonWorkMailEvents service-linked role. After you create a service-linked role, you cannot change the name of the role because various entities might reference the role. However, you can edit the description of the role using IAM. For more information, see Editing a Service-Linked Role in the *IAM User Guide*.

Deleting a service-linked role for Amazon WorkMail

If you no longer need to use a feature or service that requires a service-linked role, we recommend that you delete that role. That way you don't have an unused entity that is not actively monitored

or maintained. However, you must clean up the resources for your service-linked role before you can manually delete it.

Note
If the Amazon WorkMail service is using the role when you try to delete the resources, then the deletion might fail. If that happens, wait for a few minutes and try the operation again.

To delete Amazon WorkMail resources used by AmazonWorkMailEvents

1. Turn off Amazon WorkMail event logging.
   a. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
   b. Choose the organization alias for which to delete the AmazonWorkMailEvents role.
   c. Choose **Organization settings**, **Monitoring**.
   d. For **Log settings**, choose **Edit**.
   e. Clear the check box for **Enable mail events**.
   f. Choose **Save**.
2. Delete the Amazon CloudWatch log group.
   a. Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
   b. Choose **Logs**.
   c. For **Log Groups**, select the log group to delete.
   d. For **Actions**, choose **Delete log group**.
   e. Choose **Yes, Delete**.

To manually delete the service-linked role using IAM

Use the IAM console, the AWS CLI, or the AWS API to delete the AmazonWorkMailEvents service-linked role. For more information, see Deleting a service-linked role in the *IAM User Guide*.

Supported Regions for Amazon WorkMail service-linked roles

Amazon WorkMail supports using service-linked roles in all of the Regions where the service is available. For more information, see Amazon WorkMail Regions and Endpoints.

Logging and monitoring in Amazon WorkMail

Monitoring your email flow is important to maintaining the health of your Amazon WorkMail organization. Monitoring the email sending activity for your organization helps protect your domain reputation. Monitoring can also help you track emails that are sent and received. For more information about how to enable email event logging, see Tracking messages (p. 75).

AWS provides the following monitoring tools to watch Amazon WorkMail, report when something is wrong, and take automatic actions when appropriate:

- *Amazon CloudWatch* monitors your AWS resources and the applications you run on AWS in real time. For example, when you enable email event logging for Amazon WorkMail, CloudWatch can track emails sent and received for your organization. For more information about monitoring Amazon WorkMail with CloudWatch, see Monitoring Amazon WorkMail with Amazon CloudWatch (p. 29). For more information about CloudWatch, see the Amazon CloudWatch User Guide.
- *Amazon CloudWatch Logs* enables you to monitor, store, and access your email event logs for Amazon WorkMail when email event logging is enabled in the Amazon WorkMail console. CloudWatch Logs can

monitor information in the log files, and you can archive your log data in highly durable storage. For more information about tracking Amazon WorkMail messages using CloudWatch Logs, see Tracking messages (p. 75). For more information about CloudWatch Logs, see the Amazon CloudWatch Logs User Guide.

- *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account, and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see Logging Amazon WorkMail API calls with AWS CloudTrail (p. 36).

Topics

Monitoring Amazon WorkMail with Amazon CloudWatch

You can monitor Amazon WorkMail using CloudWatch, which collects raw data and processes it into readable, near real-time metrics. These statistics are kept for 15 months, so that you can access historical information and gain a better perspective on how your web application or service is performing. You can also set alarms that watch for certain thresholds, and send notifications or take actions when those thresholds are met. For more information, see the Amazon CloudWatch User Guide.

CloudWatch metrics for Amazon WorkMail

Amazon WorkMail sends the following metrics and dimension information to CloudWatch.

The `AWS/WorkMail` namespace includes the following metrics.

Metric	Description
`OrganizationEmailReceived`	The number of emails received by your Amazon WorkMail organization. If 1 email is addressed to 10 recipients in your organization, the `OrganizationEmailReceived` count is 1. Units: Count
`MailboxEmailDelivered`	The number of emails delivered to individual mailboxes in your Amazon WorkMail organization. If 1 email is successfully delivered to 10 recipients in your organization, the `MailboxEmailDelivered` count is 10. Units: Count
`IncomingEmailBounced`	The number of incoming emails that bounced due to full mailboxes or non-existent mailboxes. This metric is counted for each intended recipient. For example, if 1 email is sent to 10 recipients in your organization, and 2 of the recipients have full mailboxes resulting in a bounce response, the `IncomingEmailBounced` count is 2.

Metric	Description
	Units: Count
`OutgoingEmailBounced`	The number of outgoing emails that could not be delivered, counted for each intended recipient. For example, if 1 email is sent to 10 recipients, and 2 emails could not be delivered, the `OutgoingEmailBounced` count is 2. Units: Count
`OutgoingEmailSent`	The number of emails successfully sent from your Amazon WorkMail organization. This metric is counted for each recipient of a successfully sent email. For example, if 1 email is sent to 10 recipients, and the email was successfully delivered to 8 of the recipients, the `OutgoingEmailSent` count is 8 . Units: Count

CloudWatch event logs for Amazon WorkMail

When you turn on email event logging for your Amazon WorkMail organization, Amazon WorkMail logs email events with CloudWatch. For more information about turning on email event logging, see Tracking messages (p. 75).

The following tables describe the events that Amazon WorkMail logs with CloudWatch, when the events are transmitted, and what the event fields contain.

`ORGANIZATION_EMAIL_RECEIVED`

This event is logged when your Amazon WorkMail organization receives an email message.

Field	Description
recipients	The intended recipients of the message.
sender	The email address of the user who sent the email message on behalf of another user. This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
subject	The email message subject.
messageId	The SMTP message ID.
spamVerdict	Indicates whether the message is marked as spam by Amazon SES. For more information, see

Field	Description
	Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
dkimVerdict	Indicates whether the DomainKeys Identified Mail (DKIM) check passed. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
dmarcVerdict	Indicates whether the Domain-based Message Authentication, Reporting & Conformance (DMARC) check passed. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
dmarcPolicy	Appears only when the dmarcVerdict field contains "FAIL". Indicates the action to take on the email when the DMARC check fails (NONE, QUARANTINE, or REJECT). This is set by the owner of the sending email domain.
spfVerdict	Indicates whether the Sender Policy Framework (SPF) check passed. For more information, see Contents of Notifications for Amazon SES Email Receiving in the *Amazon Simple Email Service Developer Guide*.
messageTimestamp	Indicates when the message is received.

`MAILBOX_EMAIL_DELIVERED`

This event is logged when a message is delivered to a mailbox in your organization. This is logged once for each mailbox to which a message is delivered, so a single `ORGANIZATION_EMAIL_RECEIVED` event can result in multiple `MAILBOX_EMAIL_DELIVERED` events.

Field	Description
recipient	The mailbox to which the message is delivered.
folder	The mailbox folder where the message is placed.

`RULE_APPLIED`

This event is logged when an incoming or outgoing message triggers an email flow rule.

Field	Description
ruleName	The name of the rule.
ruleType	The type of rule applied (INBOUND_RULE, OUTBOUND_RULE, MAILBOX_RULE). Inbound and outbound rules apply to your Amazon WorkMail

Field	Description
	organization. Mailbox rules apply only to specified mailboxes. For more information, see Managing email flows (p. 61).
ruleActions	Actions taken based on the rule. Different recipients of the message might have different actions, such as a bounced email or a successfully delivered email.
targetFolder	Intended destination folder for a `Move` or `Copy` MAILBOX_RULE.
targetRecipient	Intended recipient of a `Forward` or `Redirect` MAILBOX_RULE.

`JOURNALING_INITIATED`

This event is logged when Amazon WorkMail sends an email to the journaling address specified by your organization administrator. This is only transmitted if journaling is configured for your organization. For more information, see Using email journaling with Amazon WorkMail (p. 111).

Field	Description
journalingAddress	The email address to which the journaling message is sent.

`INCOMING_EMAIL_BOUNCED`

This event is logged when an incoming message cannot be delivered to a target recipient. Bounced emails can be caused by reasons such as the target mailbox not existing, or the mailbox being full. This is logged once for each recipient that resulted in a bounced email. For example, if an incoming message is addressed to three recipients and two of them have full mailboxes, two INCOMING_EMAIL_BOUNCED events are logged.

Field	Description
bouncedRecipient	The intended recipient for which Amazon WorkMail bounced the message.

`OUTGOING_EMAIL_SUBMITTED`

This event is logged when a user in your organization submits an email message for sending. This is logged before the message leaves Amazon WorkMail, so this event does not indicate whether the email is successfully delivered.

Field	Description
recipients	The recipients of the message as specified by the sender. Includes all recipients on the To, CC, and BCC lines.

Field	Description
sender	The email address of the user who sent the email message on behalf of another user. This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
subject	The email message subject.

`OUTGOING_EMAIL_SENT`

This event is logged when an outgoing email is successfully delivered to a target recipient. This is logged once for each successful recipient, so a single `OUTGOING_EMAIL_SUBMITTED` can result in multiple `OUTGOING_EMAIL_SENT` entries.

Field	Description
recipient	The recipient of the successfully delivered email.
sender	The email address of the user who sent the email message on behalf of another user. This field is set only when an email is sent on behalf of another user.
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
messageId	The SMTP message ID.

`OUTGOING_EMAIL_BOUNCED`

This event is logged when an outgoing message cannot be delivered to a target recipient. Bounced emails can be caused by reasons such as the target mailbox not existing, or the mailbox being full. This is logged once for each recipient that resulted in a bounced email. For example, if an outgoing message is addressed to three recipients and two of them have full mailboxes, two `OUTGOING_EMAIL_BOUNCED` events are logged.

Field	Description
bouncedRecipient	The intended recipient for which the destination mail server bounced the message.

`DMARC_POLICY_APPLIED`

This event is logged when a DMARC policy is applied to an email sent to your organization.

Field	Description
from	The **From** address, which is usually the email address of the user who sent the message. If the user sent the message as another user or on behalf of another user, this field returns the email address of the user on whose behalf the email was sent, not the email address of the actual sender.
recipients	The intended recipients of the message.
policy	The applied DMARC policy, indicating the action to take on the email when the DMARC check fails (NONE, QUARANTINE, or REJECT). This is the same as the dmarcPolicy field in the ORGANIZATION_EMAIL_RECEIVED event.

Using CloudWatch Insights with Amazon WorkMail

If you have turned on email event logging in the Amazon WorkMail console, you can use Amazon CloudWatch Logs Insights to query your event logs. For more information about turning on email event logging, see Tracking messages (p. 75). For more information about CloudWatch Logs Insights, see Analyze log data with CloudWatch Logs Insights in the *Amazon CloudWatch Logs User Guide*.

The following examples demonstrate how to query CloudWatch Logs for common email events. You run these queries in the CloudWatch console. For instructions about how to run these queries, see Tutorial: Run and modify a sample query in the *Amazon CloudWatch Logs User Guide*.

Example Example: See why User B did not receive an email sent by User A.

The following code example demonstrates how to query for an outgoing email sent by User A to User B, sorted by timestamp.

```
fields @timestamp, traceId
| sort @timestamp asc
| filter (event.from like /(?i)userA@example.com/
and event.eventName = "OUTGOING_EMAIL_SUBMITTED"
and event.recipients.0 like /(?i)userB@example.com/)
```

This returns the sent message and trace ID. Use the trace ID in the following code example to query the event logs for the sent message.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter traceId = "$TRACEID"
```

This returns the email message ID and the email events. `OUTGOING_EMAIL_SENT` indicates that the email was sent. `OUTGOING_EMAIL_BOUNCED` indicates that the email bounced. To see whether the email was received, query using the message ID in the following code example.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter event.messageId like "$MESSAGEID"
```

This should also return the received message, because it has the same message ID. Use the trace ID in the following code example to query for delivery.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter traceId = "$TRACEID"
```

This returns the delivery action and any applicable rule actions.

Example Example: See all mail received from a user or domain

The following code example demonstrates how to query for all mail received from a specified user.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter (event.from like /(?i)user@example.com/ and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED")
```

The following code example demonstrates how to query for all mail received from a specified domain.

```
fields @timestamp, event.eventName
| sort @timestamp asc
| filter (event.from like "example.com" and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED")
```

Example Example: See who sent bounced emails

The following code example demonstrates how to query for outgoing emails that bounced, and also returns the reasons for bouncing.

```
fields @timestamp, event.destination, event.reason
| sort @timestamp desc
| filter event.eventName = "OUTGOING_EMAIL_BOUNCED"
```

The following code example demonstrates how to query for incoming emails that bounced, and also returns the bounced recipients' email addresses and the reasons for bouncing.

```
fields @timestamp, event.bouncedRecipient.emailAddress, event.bouncedRecipient.reason,
 event.bouncedRecipient.status
| sort @timestamp desc
| filter event.eventName = "INCOMING_EMAIL_BOUNCED"
```

Example Example: See which domains are sending spam

The following code example demonstrates how to query for recipients in your organization that are receiving spam.

```
stats count(*) as c by event.recipients.0
| filter (event.eventName = "ORGANIZATION_EMAIL_RECEIVED" and event.spamVerdict = "FAIL")
| sort c desc
```

The following code example demonstrates how to query for the sender of the spam emails.

```
fields @timestamp, event.recipients.0, event.sender, event.from
| sort @timestamp asc
| filter (event.spamVerdict = "FAIL")
```

Example Example: See why an email was sent to a recipient's spam folder

The following code example demonstrates how to query for emails identified as spam, filtered by subject.

```
fields @timestamp, event.recipients.0, event.spamVerdict, event.spfVerdict,
 event.dkimVerdict, event.dmarcVerdict
| sort @timestamp asc
| filter event.subject like /(?i)$SUBJECT/ and event.eventName =
 "ORGANIZATION_EMAIL_RECEIVED"
```

You can also query by the email trace ID to see all events for the email.

Example Example: See emails that match email flow rules

The following code example demonstrates how to query for emails that matched outbound email flow rules.

```
fields @timestamp, event.ruleName, event.ruleActions.0.action
| sort @timestamp desc
| filter event.ruleType = "OUTBOUND_RULE"
```

The following code example demonstrates how to query for emails that matched inbound email flow rules.

```
fields @timestamp, event.ruleName, event.ruleActions.0.action,
 event.ruleActions.0.recipients.0
| sort @timestamp desc
| filter event.ruleType = "INBOUND_RULE"
```

Example Example: See how many emails are received or sent by your organization

The following code example demonstrates how to query for the number of emails received by each recipient in your organization.

```
stats count(*) as c by event.recipient
| filter event.eventName = "MAILBOX_EMAIL_DELIVERED"
| sort c desc
```

The following code example demonstrates how to query for the number of emails sent by each sender in your organization.

```
stats count(*) as c by event.from
| filter event.eventName = "OUTGOING_EMAIL_SUBMITTED"
| sort c desc
```

Logging Amazon WorkMail API calls with AWS CloudTrail

Amazon WorkMail is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in Amazon WorkMail. CloudTrail captures all API calls for Amazon WorkMail as events, including calls from the Amazon WorkMail console and from code calls to the Amazon WorkMail APIs. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Amazon WorkMail. If you don't configure a trail, you can still

view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that was made to Amazon WorkMail, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Amazon WorkMail information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When activity occurs in Amazon WorkMail, that activity is recorded in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing events with CloudTrail event history.

For an ongoing record of events in your AWS account, including events for Amazon WorkMail, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple Regions and Receiving CloudTrail log files from multiple accounts

All Amazon WorkMail actions are logged by CloudTrail and are documented in the Amazon WorkMail API Reference. For example, calls to the `CreateUser`, `CreateAlias`, and `GetRawMessageContent` API operations generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root or IAM user credentials.
- Whether the request was made with temporary security credentials for a role or federated user.
- Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity element.

Understanding Amazon WorkMail log file entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail log files are not an ordered stack trace of the public API calls, so they do not appear in any specific order.

The following example shows a CloudTrail log entry that demonstrates the `CreateUser` action from the Amazon WorkMail API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
```

```
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE"
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T17:49:59Z",
  "eventSource": "workmail.amazonaws.com",
  "eventName": "CreateUser",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
  "requestParameters": {
    "name": "janedoe",
    "displayName": "Jane Doe",
    "organizationId": "m-5b1c980000EXAMPLE"
  },
  "responseElements": {
    "userId": "a3a9176d-EXAMPLE"
  },
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```

The following example shows a CloudTrail log entry that demonstrates the `CreateAlias` action from the Amazon WorkMail API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T18:13:44Z",
  "eventSource": "workmail.amazonaws.com",
  "eventName": "CreateAlias",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
  "requestParameters": {
    "alias": "aliasjamesdoe@testofconsole.awsapps.com",
    "organizationId": "m-5b1c980000EXAMPLE"
    "entityId": "a3a9176d-EXAMPLE"
  },
  "responseElements": null,
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```

The following example shows a CloudTrail log entry that demonstrates the `GetRawMessageContent` action from the Amazon WorkMail Message Flow API.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
```

```
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111111111111:user/WMSDK",
    "accountId": "111111111111",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "WMSDK"
  },
  "eventTime": "2017-12-12T18:13:44Z",
  "eventSource": "workmailMessageFlow.amazonaws.com",
  "eventName": "GetRawMessageContent",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "203.0.113.12",
  "userAgent": "aws-sdk-java/1.11.205 Mac_OS_X/10.11.6 Java_HotSpot(TM)_64-
Bit_Server_VM/25.151-b12 java/1.8.0_151",
  "requestParameters": {
    "messageId": "123A4A5A-67B8-90C1-D23E-45FG67H890J1"
  },
  "responseElements": null,
  "requestID": "dec81e4a-EXAMPLE",
  "eventID": "9f2f09c5-EXAMPLE",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "recipientAccountId": "111111111111"
}
```


Compliance validation for Amazon WorkMail

Third-party auditors assess the security and compliance of Amazon WorkMail as part of multiple AWS compliance programs. These include SOC, ISO, and C5.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading reports in AWS Artifact.

Your compliance responsibility when using Amazon WorkMail is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in Amazon WorkMail

The AWS global infrastructure is built around AWS Regions and Availability Zones. AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected with low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between Availability Zones without interruption.

Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

In addition to the AWS global infrastructure, Amazon WorkMail offers several features to help support your data resiliency and backup needs.

Infrastructure security in Amazon WorkMail

As a managed service, Amazon WorkMail is protected by the AWS global network security procedures that are described in the Amazon Web Services: Overview of Security Processes whitepaper.

You use AWS published API calls to access Amazon WorkMail through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Getting started with Amazon WorkMail

After you have completed the Prerequisites (p. 4), you're ready to get started with Amazon WorkMail. For more information, see Getting started with Amazon WorkMail (p. 41).

You can also learn more about migrating existing mailboxes to Amazon WorkMail, interoperability with Microsoft Exchange, and Amazon WorkMail quotas in the following sections.

Topics

Getting started with Amazon WorkMail

Whether you are a new Amazon WorkMail user or an existing user of Amazon WorkDocs or Amazon WorkSpaces, get started with Amazon WorkMail by completing the following steps.

> **Note**
> Complete the Prerequisites (p. 4) before getting started.

Topics

Step 1: Sign in to the Amazon WorkMail console

You must sign in to the Amazon WorkMail console before you can add users and manage accounts and mailboxes.

To sign in to the Amazon WorkMail console

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, go to the navigation bar and select the AWS Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.

Step 2: Set up your Amazon WorkMail site

1. After you sign in to the Amazon WorkMail console, set up your organization and add a domain. We recommend using a dedicated domain for your Amazon WorkMail organization. For more information, see Creating an organization (p. 54) and Adding a domain (p. 79).
2. (Optional) You can choose to use a free testing domain provided by Amazon WorkMail. If you choose to do this, skip to step 4.

Note
The test domain format is *example*.awsapps.com. You can use the test mail domain with Amazon WorkMail and other supported AWS services as long as you maintain enabled users in your Amazon WorkMail organization. However, you cannot use the test domain for other purposes. Also, the test domain might become available for registration and use by other customers if your Amazon WorkMail organization does not maintain at least one enabled user.

3. If you are using an external domain, verify your domain by adding the appropriate TXT and MX records to your DNS service. Make sure to set your domain as the default for your organization. For more information, see Verifying domains (p. 82) and Choosing the default domain (p. 82).
4. Create new users or enable your existing directory users for Amazon WorkMail. For more information, see Managing user accounts (p. 90).
5. (Optional) If you have existing Microsoft Exchange mailboxes, migrate them to Amazon WorkMail. For more information, see Migrating to Amazon WorkMail (p. 42).

After you've finished setting up your Amazon WorkMail site, you can access Amazon WorkMail using the web application URL.

To locate your Amazon WorkMail web application URL

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation panel, choose **Organization settings**.

The web application URL is on the **General settings** tab and looks like this: https://*alias*.awsapps.com/mail.

Step 3: Set up Amazon WorkMail user access

Choose from the following options to set up Amazon WorkMail user access:

- Set up user access from an existing desktop client using the Microsoft Outlook client. For more information, see Connect Microsoft Outlook to your Amazon WorkMail account.
- Set up user access from a mobile device, such as a Kindle, Android, iPad, or iPhone. For more information, see Getting started with a mobile device.
- Set up user access with any IMAP-compatible client software. For more information, see Connect IMAP clients to Your Amazon WorkMail account.

More resources

- Migrating to Amazon WorkMail (p. 42)
- Interoperability between Amazon WorkMail and Microsoft Exchange (p. 44)
- Amazon WorkMail quotas (p. 50)

Migrating to Amazon WorkMail

You can migrate to Amazon WorkMail from Microsoft Exchange, Microsoft Office 365, G Suite Basic (formerly Google Apps for Work), and many other platforms by working with one of our partners. For more information about our partners, see Amazon WorkMail Features.

Topics

Step 1: Create or enable users in Amazon WorkMail

Before you can migrate your users, you must add the users in Amazon WorkMail to provision the mailbox. For more information, see Creating users (p. 90).

Step 2: Migrate to Amazon WorkMail

You can work with any AWS migration partners to migrate to Amazon WorkMail. For information about these providers, see Amazon WorkMail features.

To migrate your mailboxes, create a dedicated Amazon WorkMail user to act as migration administrator. The following procedure grants permission to that user to access all the mailboxes in your organization.

To create a migration administrator

1. Do one of the following:
   - In the Amazon WorkMail console, create a new user to act as migration administrator. For more information, see Creating users (p. 90).
   - In your Active Directory, create a new user to act as migration administrator, and enable the user for Amazon WorkMail. For more information, see Enabling existing users (p. 91).
2. In the Amazon WorkMail console, for **Organizations**, choose the name of your organization.
3. Choose **Organization settings**, **Migration**, **Edit**.
4. For **Mailbox permissions**, select **On**.
5. Choose **Select user**.
6. Search for and select the user to act as migration administrator, and choose **Select user**.
7. Choose **Save**.

Step 3: Complete the migration to Amazon WorkMail

After you have migrated your email accounts to Amazon WorkMail, verify your DNS records and configure your desktop and mobile clients.

To complete migration to Amazon WorkMail

1. Verify that all DNS records are updated and that they point to Amazon WorkMail. For more information about the required DNS records, see Adding a domain (p. 79).

   Note
   The DNS record update process may take several hours. If any new items appear in a source mailbox while the MX records are being changed, run the migration tool again to migrate new items after the DNS records are updated.
2. For more information about configuring your desktop or mobile clients to use Amazon WorkMail, see Connect Microsoft Outlook to your Amazon WorkMail account in the *Amazon WorkMail User Guide*.

Interoperability between Amazon WorkMail and Microsoft Exchange

Interoperability between Amazon WorkMail and Microsoft Exchange Server allows you to minimize disruption to your users as you migrate mailboxes to Amazon WorkMail, or use Amazon WorkMail for a subset of your corporate mailboxes.

This interoperability allows you to use the same corporate domain for mailboxes across both environments. This way, your users can schedule meetings with bidirectional sharing of calendar free/busy information.

Prerequisites

Before you enable interoperability with Microsoft Exchange, do the following:

- Make sure you have at least one user enabled for Amazon WorkMail, so you can configure the availability settings for Microsoft Exchange. To enable a user, follow the steps in Enable email routing for a user (p. 46).
- Set up an Active Directory (AD) Connector – Setting up an AD Connector with your on-premises directory allows users to continue using their existing corporate credentials. For more information, see Set up AD Connector and Integrate Amazon WorkMail with your on-premises directory.
- Set up your Amazon WorkMail organization – Create an Amazon WorkMail organization that uses the AD Connector that you set up.
- Add your corporate domains to your Amazon WorkMail organization and verify them in the Amazon WorkMail console. Otherwise, emails sent to this alias will bounce. For more information, see Working with domains.
- Migrate mailboxes – Enable users to provision and migrate mailboxes from your on-premises environment to Amazon WorkMail. For more information, see Enable existing users and see Migrating to Amazon WorkMail.

  Note
  Do not update DNS records to point to Amazon WorkMail. This ensures that Microsoft Exchange remains the primary server for incoming email for as long as you want interoperability between the two environments.
- Make sure that the User Principal Names (UPNs) in Active Directory match the users' primary SMTP addresses.

Amazon WorkMail makes HTTPS requests to the Exchange Web Services (EWS) URL on Microsoft Exchange to obtain calendar free/busy information.

- Ensure that the relevant firewall settings are set up to allow access from the internet. The default port for HTTPS requests is port 443.
- Amazon WorkMail can only make successful HTTPS requests to the EWS URL on Microsoft Exchange when a certificate signed by a valid certificate authority (CA) is available in your Microsoft Exchange environment. For more information, see Create an Exchange Server certificate request for a certification authority on the Microsoft Exchange Documentation website.
- You must enable **Basic Authentication** for EWS in Microsoft Exchange. For more information, see Virtual directories: Exchange 2013 on the Microsoft MVP Award Program Blog.

Add domains and enable mailboxes

Add your corporate domains to Amazon WorkMail so that they can be used in email addresses. Ensure that the domains added to Amazon WorkMail are verified, then enable users and groups to provision mailboxes on Amazon WorkMail. Resources cannot be enabled in Amazon WorkMail while in interoperability mode, and should be re-created in Amazon WorkMail after you disable interoperability mode. However, you can still use them to schedule meetings while in interoperability mode. Resources from Microsoft Exchange are always shown in the **Users** tab in Amazon WorkMail.

- For more information, see Add domains, Enable existing users, and Enable an existing group.

> **Note**
> To ensure interoperability with Microsoft Exchange, do not update the DNS records to point to Amazon WorkMail records. Microsoft Exchange remains the primary server for incoming email as long as you want interoperability between the two environments.

Enable interoperability

If you have not created an Amazon WorkMail organization, follow the steps in Integrate Amazon WorkMail with your on-premises directory (Custom Setup) and choose **Enable interoperability** when creating your Amazon WorkMail organization.

If you already have an Amazon WorkMail organization with an AD Connector linked to Active Directory and you also have Microsoft Exchange, contact AWS Support for assistance with enabling Microsoft Exchange interoperability for an existing Amazon WorkMail organization.

Create service accounts in Microsoft Exchange and Amazon WorkMail

To access calendar free/busy information, create a service account on both Microsoft Exchange and Amazon WorkMail. The Microsoft Exchange service account is any user on Microsoft Exchange that has access to the calendar free/busy information of other Exchange users. Access is granted by default; so no special permissions are required.

Similarly, the Amazon WorkMail service account is any user on Amazon WorkMail that has access to calendar free/busy information of other users on Amazon WorkMail. This is also granted by default.

Using an Amazon WorkMail organization that takes advantage of an AD Connector integrated with your on-premises directory means that the Amazon WorkMail service account user must be created in your on-premises directory and then enabled for Amazon WorkMail.

Limitations in interoperability mode

When your organization is in interoperability mode, all user, group, and resource management must be done using the Exchange admin center. Users and groups can be enabled for Amazon WorkMail through the AWS Management Console. For more information, see Enable existing users and Enable an existing group.

When enabling a user or group for Amazon WorkMail, you cannot edit the email addresses or aliases of those users and groups. Those must also be configured via the Exchange admincenter. Amazon WorkMail synchronizes changes in your directory every four hours.

Resources cannot be created or enabled in Amazon WorkMail while in interoperability mode. However, all your Exchange resources are available in the Amazon WorkMail address book and can be used for scheduling meetings as usual.

Enable email routing between Microsoft Exchange and Amazon WorkMail users

When you enable email routing between Microsoft Exchange Server and Amazon WorkMail, users that are configured for Amazon WorkMail can continue using their existing email addresses to send and receive email on Amazon WorkMail. When email routing is enabled, your Microsoft Exchange Server remains the primary SMTP server for incoming email.

Prerequisites for email routing:

- Interoperability mode is enabled for your organization. For more information, see Enable interoperability (p. 45).
- Your domain is added and verified in the Amazon WorkMail console.
- Your Microsoft Exchange Server can send email to the internet. You might need to configure a Send connector.

Enable email routing for a user

We recommend that you carry out the following steps first for test users, before applying the change to your organization.

1. Enable the user you are migrating to Amazon WorkMail. For more information, see Enable existing users.
2. In the Amazon WorkMail console, ensure that there are at least two email addresses associated with the enabled user.
   - *workmailuser*@*orgname*.awsapps.com (this is added automatically, and can be used for tests without your Microsoft Exchange.)
   - *workmailuser*@*yourdomain*.com (this is added automatically, and is the primary Microsoft Exchange address.)

     For more information, see Edit user email addresses.
3. Ensure that you migrate all data from the mailbox in Microsoft Exchange to the mailbox in Amazon WorkMail. For more information, see Migrating to Amazon WorkMail.
4. When all the data is migrated, disable the mailbox for the user on Microsoft Exchange and create a mail user (or mail-enabled user) that has the external SMTP address pointed to Amazon WorkMail. This can be achieved using the following commands in Exchange Management Shell.

   Important
   The steps below erase the contents of the mailbox. Ensure that your data has been migrated to Amazon WorkMail before you attempt to enable email routing. Some mail clients do not seamlessly switch to Amazon WorkMail when this command is run. For more information, see Mail client configuration (p. 47).

```
$old_mailbox = Get-Mailbox exchangeuser
```

```
Disable-Mailbox $old_mailbox
```

```
$new_mailuser = Enable-MailUser $old_mailbox.Identity -
ExternalEmailAddress workmailuser@orgname.awsapps.com -PrimarySmtpAddress
 $old_mailbox.PrimarySmtpAddress
```

```
Set-MailUser $new_mailuser -EmailAddresses $old_mailbox.EmailAddresses -
HiddenFromAddressListsEnabled $old_mailbox.HiddenFromAddressListsEnabled
```

In the above commands, **`orgname`** represents the name of the Amazon WorkMail organization. For more information, see Disabling mailbox and Enabling mail users on Microsoft TechNet.

5. Send a test email to the user (as per the example above, **`workmailuser@yourdomain.com`**). If email routing has been enabled correctly, the user should be able to log in to the Amazon WorkMail mailbox and receive the email.

Note
Microsoft Exchange remains the primary server for incoming email as long as you would like to have interoperability between the two environments. To ensure interoperability with Microsoft Exchange, the DNS records should not be updated to point to Amazon WorkMail until later.

Post setup configuration

The above steps move a user mailbox from Microsoft Exchange Server to Amazon WorkMail, while keeping the user in Microsoft Exchange as a contact. Because the migrated user is now an external mail user, Microsoft Exchange Server imposes additional constraints and there may be additional configuration requirements to complete the migration.

- The user might not be able to send emails to groups by default. To enable this functionality, the user must be added to a safe sender list for all groups. For more information, see Delivery management on Microsoft TechNet.
- The user also might not be able to book resources. To enable this functionality, you must set the `ProcessExternalMeetingMessages` of all resources that the user needs to access. For more information, see Set-CalendarProcessing on Microsoft TechNet.

Mail client configuration

Some mail clients do not switch seamlessly to Amazon WorkMail and require the user to perform additional setup. Different mail clients require different actions to be taken.

- Microsoft Outlook on Windows – Requires MS Outlook to be restarted. At startup, you are required to choose whether to keep using the old mailbox or use a temporary mailbox. Choose the temporary mailbox option, and reconfigure the Microsoft Exchange mailbox from scratch.
- Microsoft Outlook on MacOS – When Outlook is restarted, you see the following message **Outlook was redirected to server orgname.awsapps.com. Do you want this server to configure your settings?**. Accept the suggestion.
- Mail on iOS – The mail app stops receiving emails and generates a **Cannot get mail** error. Reconfigure the Microsoft Exchange mailbox from scratch.

Configure availability settings on Amazon WorkMail

Configure availability settings on Amazon WorkMail and Microsoft Exchange to enable bidirectional sharing of calendar free/busy information.

To configure availability settings in the console

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. In the navigation panel, choose **Organization settings**, **Interoperability Settings**.
3. Choose **Configure availability settings** and provide the following information:

- **Domain** – The domain for which to set interoperability between Amazon WorkMail and Microsoft Exchange.
- **Exchange Web Services (EWS) URL** – The URL to which Amazon WorkMail sends HTTPS requests to access calendar free/busy information of users on Microsoft Exchange. The EWS URL usually looks like the following: **`https://servername.com/EWS/Exchange.asmx`**. You can obtain the EWS URL in one of the following ways:
  - **Using Microsoft Outlook**
    1. Log in to Microsoft Outlook on Windows for any user on your Exchange environment.
    2. Hold the **Ctrl** key and open the context (right-click) menu on the Microsoft Outlook icon in the task bar.
    3. Choose **Test E-mail AutoConfiguration**.
    4. Enter the Microsoft Exchange user's email address and password, and choose **Test**.
    5. From the Results window, copy the value for the **Availability Service URL**.
  - **Using PowerShell**
    - ```
      Get-WebServicesVirtualDirectory |Select name, *url* | fl
      ```

    The external URL returned by the above command is the EWS URL.
- **User email address and password** – These are the credentials of the Microsoft Exchange service account and are encrypted and securely stored by Amazon WorkMail. The email address of the Microsoft Exchange service account should use the Fully Qualified Domain Name (FQDN). For more information, see Create service accounts in Microsoft Exchange and Amazon WorkMail (p. 45).

  If your Active Directory domain is not the same as your Microsoft Exchange domain, use the User Principal Name (UPN) of the Microsoft Exchange Service account. This can be obtained with the following PowerShell command:

  ```
  Get-ADUser exchange_service_account_username | select UserPrincipalName
  ```

  In the above example, **`exchange_service_account_username`** is the username of the Microsoft Exchange Service account.

Configure availability settings in Microsoft Exchange

To redirect all calendar free/busy information requests for enabled users to Amazon WorkMail, set up an availability address space on Microsoft Exchange.

Use the following PowerShell command.

```
$credentials = Get-Credential
```

At the prompt, enter the credentials of the Amazon WorkMail service account. The username should be entered as **`domain\username`** (that is, **`orgname.awsapps.com\workmail_service_account_username`**. Here, **`orgname`** represents the name of the Amazon WorkMail organization. For more information, see Create service accounts in Microsoft Exchange and Amazon WorkMail (p. 45).

```
Add-AvailabilityAddressSpace -ForestName orgname.awsapps.com -AccessMethod OrgWideFB -
Credentials $credentials
```
```

For more information, see Add-AvailabilityAddressSpace on Microsoft Docs.

Disabling interoperability and decommissioning your mail server

After all your Microsoft Exchange mailboxes are configured for Amazon WorkMail, you can disable interoperability. If you have not migrated any users or records, disabling interoperability does not affect any of your configurations.

> **Warning**
> Before disabling interoperability, ensure that you have completed all the required steps. Failure to do so could result in bounced emails or unintended behavior. If you have not completed migration, disabling interoperability may cause disruptions to your organization. You cannot undo this operation.

To disable interoperability support

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. On the **Organizations** page, choose the organization that has interoperability mode enabled and choose **Disable Interoperability**.
4. In the **Disable interoperability with Microsoft Exchange** dialog box, enter the name of the organization and choose **Disable**.

After disabling interoperability support, users and groups that are not enabled for Amazon WorkMail are removed from the address book. You can still enable any missing user or group using the Amazon WorkMail console and they are added to the address book. Resources from Microsoft Exchange cannot be enabled and do not appear in the address book until you complete the step below.

- **Create resources in Amazon WorkMail** – You can create resources in Amazon WorkMail and configure delegates and booking options for these resources. For more information, see Working with resources.
- **Create an AutoDiscover DNS record** – Configure an AutoDiscover DNS record for all mail domains in the organization to enable users to easily connect to their Amazon WorkMail mailboxes from their Microsoft Outlook and mobile clients. For more information, see Use AutoDiscover to configure endpoints.
- **Switch your MX DNS record to Amazon WorkMail** – To deliver all incoming emails to Amazon WorkMail, you have to switch your MX DNS record to Amazon WorkMail. It can take up to 72 hours before the DNS change is propagated to all DNS servers.
- **Decommission your mail server** – After you've verified that all email is being routed directly to Amazon WorkMail, you can decommission your mail server if you do not intend to use it going forward.

Troubleshooting

Solutions to the most commonly encountered Amazon WorkMail interoperability and migration errors are listed below.

Exchange Web Services (EWS) URL is invalid or unreachable – Check that you have the correct EWS URL. For more information, see Configure availability settings on Amazon WorkMail (p. 47).

Connection failure during EWS validation – This is a general error and can be caused by:

- No internet connection in Microsoft Exchange.
- Your firewall is not configured to allow access from the internet. Ensure that port 443 (the default port for HTTPS requests) is open.

If you've confirmed the internet connection and firewall settings, but the error persists, contact AWS Support.

Invalid username and password when configuring Microsoft Exchange interoperability – This is a general error and can be caused by:

- The username is not in the expected form. Use the following pattern.

```
DOMAIN\username
```

- Your Microsoft Exchange server is not configured for Basic Authentication for EWS. For more information, see Virtual directories: Exchange 2013 on the Microsoft MVP Award Program Blog.

User receives emails with winmail.dat attachment – This might happen when encrypted S/MIME email is sent from Exchange to Amazon WorkMail and received in Outlook 2016 for Mac or IMAP client. The solution is to run the following command in the Exchange Management Shell.

Set-RemoteDomain -Identity "Default" -TNEFEnabled $false – If you've confirmed the points above but the error persists, contact AWS Support.

Amazon WorkMail quotas

Amazon WorkMail can be used by enterprise customers as well as small business owners. Although we support most use cases without the need to configure any changes in quotas, we also protect our users and the internet against abuse of the product. Therefore, some customers may run into quotas that we have set. This section describes these quotas and how to change them.

Some quota values can be changed, and some are hard quotas that cannot be changed. For more information about requesting a quota increase, see AWS Service quotas in the *Amazon Web Services General Reference*.

Amazon WorkMail organization and user quotas

You can add up to 25 users to your Amazon WorkMail organization for a 30-day free trial. After this period ends, you are charged for all active users unless you remove them or close your Amazon WorkMail account.

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

> **Note**
> When requesting a quota increase for only a specific organization, include the organization name in your request.

Resource	Default quota	Upper bound for change requests
Amazon WorkMail organizations per AWS account	100	Can be increased depending on the directory type that is used for the organization. You

Resource	Default quota	Upper bound for change requests
		can view AWS Directory Service quotas and request increases from the AWS Directory Service console. For more information, see Service quotas in the *AWS General Reference*.
Users per Amazon WorkMail organization	1,000	Can be increased depending on the directory type that is used for the organization: • Amazon WorkMail directory: up to 10 million users • Simple AD or AD Connector, large: up to 5,000 users* • Simple AD or AD Connector, small: up to 500 users* • Microsoft AD, hosted by AWS Directory Service: depending on your set-up and configuration, up to 10 million users *If you are using Simple AD or AD Connector, see AWS Directory Service for additional information.
Free trial users	Up to 25 users in the first 30 days	The free trial period is only applicable for the first 25 users in any organization. Any additional users are not included in the free trial offer.
Recipients addressed per AWS account per day	100,000 recipients external to the organization, with no hard quota on recipients internal to the organization	There is no upper bound. However, Amazon WorkMail is a business email service and not intended to be used for bulk email services. For bulk email services, see Amazon SES or Amazon Pinpoint.
Recipients addressed per AWS account per day using any of the test domains	200 recipients, regardless of destination	The test mail domain is not intended for long-term usage. We recommend that you add your own domain and use it as the default domain.

Quotas for groups are set by the underlying directory.

WorkMail organization setting quotas

Resource	Default quota
Number of domains per Amazon WorkMail organization	1,000 This is a hard quota and cannot be changed.
Number of sender patterns in email flow rules per rule	250 This is a hard quota and cannot be changed.
Number of sender patterns in email flow rules per organization	1,000 This is a hard quota and cannot be changed.

Per-user quotas

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

Resource	Default quota	Upper quota for change requests
Maximum size of mailbox	50 GB This is a hard quota and cannot be changed.	Not applicable
Maximum number of aliases per user	100 This is a hard quota and cannot be changed.	Not applicable
Recipients addressed per user per day using the domain that you own	10,000 recipients external to the organization, with no hard quota on recipients internal to the organization.	There is no upper bound. However, Amazon WorkMail is a business email service and not intended to be used for bulk email services. For bulk email services, see Amazon SES or Amazon Pinpoint.

Message quotas

All messages that are sent to another user are considered when evaluating these quotas. These include emails, meeting requests, meeting responses, task requests, and messages that are forwarded or redirected automatically as the result of a rule.

Resource	Default quota
Maximum size of incoming message	25 MB

Resource	Default quota
	This is a hard quota and cannot be changed.
Maximum size of outgoing message	25 MB This is a hard quota and cannot be changed.
Number of recipients per message	500 This is a hard quota and cannot be changed.

Working with organizations

In Amazon WorkMail, your organization represents the users in your company. In the Amazon WorkMail console, you see a list of your available organizations. If you don't have any available, you must create an organization in order to use Amazon WorkMail.

Topics

Creating an organization

To use Amazon WorkMail, you must first create an organization. One AWS account can have multiple Amazon WorkMail organizations. When you create an organization, you also select a domain for the organization and set up user directory and encryption settings.

You can either create a new user directory, or integrate Amazon WorkMail with an existing directory such as an on-premises Microsoft Active Directory, AWS Managed Active Directory, or Simple AD. By integrating with your on-premises directory, you can use your existing users and groups in Amazon WorkMail, and users can sign in with their existing credentials. If you're using an existing directory that is on-premises, you must first set up an AD Connector in AWS Directory Service. The AD Connector synchronizes your users and groups to the Amazon WorkMail address book and performs user authentication requests. For more information, see Active Directory Connector in the *AWS Directory Service Administration Guide*.

You also have the option to select a customer managed master key that Amazon WorkMail uses to encrypt the mailbox content. You can either select the default AWS managed master key for Amazon WorkMail, or select an existing customer managed master key in AWS Key Management Service (AWS KMS). For information about creating a new customer managed master key, see Creating keys in the *AWS Key Management Service Developer Guide*. If you are signed in as an AWS Identity and Access Management (IAM) user, make yourself a key administrator on the master key. For more information, see Enabling and disabling keys in the *AWS Key Management Service Developer Guide*.

Considerations

The following are considerations for creating an Amazon WorkMail organization:

- Amazon WorkMail does not currently support managed Microsoft Active Directory services that are shared with multiple accounts.
- If you have an on-premises Active Directory with Microsoft Exchange and an AD Connector, we recommend configuring interoperability settings for your organization. This allows you to minimize

disruption to your users as you migrate mailboxes to Amazon WorkMail, or use Amazon WorkMail for a subset of your corporate mailboxes. For more information, see Interoperability between Amazon WorkMail and Microsoft Exchange (p. 44).

- If you select the **Free test domain** option, you can start using your Amazon WorkMail organization with the provided test domain. The test domain format is *example*.awsapps.com. You can use the test mail domain with Amazon WorkMail and other supported AWS services as long as you maintain enabled users in your Amazon WorkMail organization. However, you cannot use the test domain for other purposes. Also, the test domain might become available for registration and use by other customers if your Amazon WorkMail organization does not maintain at least one enabled user.

Topics

Creating a new organization

Create a new organization in the Amazon WorkMail console.

To create an organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. In the navigation bar, select the AWS Region that meets your needs. For more information, see Regions and Endpoints in the *Amazon Web Services General Reference*.
3. For **Get started**, choose **Create organization**.
4. For **Email domain**, select the domain to use for the email addresses in your organization. Choose from the following options:
   - **Existing Route 53 domain** – Select an existing domain that you manage with an Amazon Route 53 (Route 53) hosted zone.
   - **New Route 53 domain** – Register a new Route 53 domain name to use with Amazon WorkMail.
   - **External domain** – Enter an existing domain that you manage with an external DNS provider.
   - **Free test domain** – Use a free test domain provided by Amazon WorkMail, and add a domain later.
5. (Optional) For **Route 53 hosted zone**, select your Route 53 domain if you are using one.
6. For **Alias**, enter a unique alias for your organization.
7. Under **Advanced settings**, for **User directory**, select one of the following options:
   - **Create new Amazon WorkMail directory** – Creates a new directory for you to add your users.
   - **Use existing directory** – Uses an existing directory to manage your users, such as an on-premises Microsoft Active Directory, AWS Managed Active Directory, or Simple AD.
8. For **Encryption**, select one of the following options:
   - **Use an Amazon WorkMail managed key** – Creates a new encryption key in your account.
   - **Use existing customer managed key (CMK)** – Uses an existing CMK that you have already created in AWS KMS.
9. Choose **Create organization**.

If you are using an external domain, you'll want to verify it by adding the appropriate TXT and MX records to your DNS service. Make sure to set your domain as the default for your organization. For more information, see Verifying domains (p. 82) and Choosing the default domain (p. 82).

When your organization is **Active**, you can add users to it and set up their email clients. For more information, see Managing user accounts (p. 90) and Setting up email clients for Amazon WorkMail.

Integrating an Amazon WorkDocs or Amazon WorkSpaces directory

To use Amazon WorkMail with Amazon WorkDocs or Amazon WorkSpaces, create a compatible directory by using the following steps.

To add a compatible Amazon WorkDocs or Amazon WorkSpaces directory

1. Create a compatible directory using Amazon WorkDocs or Amazon WorkSpaces.
   a. For Amazon WorkDocs instructions, see Getting started with Quick Start in the *Amazon WorkDocs Administration Guide*.
   b. For Amazon WorkSpaces instructions, see Get started with Amazon WorkSpaces Quick Setup in the *Amazon WorkSpaces Administration Guide*.
2. In the Amazon WorkMail console, create your Amazon WorkMail organization and choose to use your existing directory for it. For more information, see Creating a new organization (p. 55).

Organization states and descriptions

After you create an organization, it can have one of the following states.

State	Description
Active	Your organization is healthy and ready for use.
Creating	A workflow is running to create your organization.
Failed	Your organization could not be created.
Impaired	Your organization is malfunctioning or an issue has been detected.
Inactive	Your organization is inactive.
Requested	Your organization creation request is in the queue and waiting to be created.
Validating	All settings for the organization are being health-checked.

Deleting an organization

If you no longer want to use Amazon WorkMail for your organization's email, you can delete your organization from Amazon WorkMail.

Note
This operation cannot be undone, and you will not be able to recover your mailbox data.

To delete an organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

2. If necessary, change the AWS Region. From the navigation bar, select the appropriate Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. On the **Organizations** screen, in the list of organizations, select the organization to delete and choose **Delete**.
4. For **Delete organization**, choose whether to delete or keep the existing user directory, and then enter the name of the organization.
5. Choose **Delete organization**.

   Note
   If you didn't provide your own directory for Amazon WorkMail, we created one for you. If you keep this existing directory when you delete the organization, you will be charged for it unless it is being used by Amazon WorkMail, Amazon WorkDocs, or Amazon WorkSpaces. For pricing information, see Other directory types pricing.
   In order to delete the directory, it cannot have any other AWS applications enabled. For more information, see Deleting a Simple AD directory or Deleting an AD Connector directory in the *AWS Directory Service Administration Guide*.

If an invalid Amazon Simple Email Service (Amazon SES) rule set error message appears when you're attempting to delete an organization, edit the Amazon SES rule in the Amazon SES console and remove the invalid rule set. The rule that you edit should have your Amazon WorkMail organization ID in the rule name. For more information about editing Amazon SES rules, see Editing a receipt rule in the *Amazon Simple Email Service Developer Guide*.

If you need to figure out which rule set is invalid, save the rule first. An error message appears for the invalid rule set.

Tagging an organization

Tagging an Amazon WorkMail organization resource lets you do the following:

- Differentiate between organizations in the AWS Billing and Cost Management console.
- Control access to Amazon WorkMail organization resources by adding them to the `Resource` element of AWS Identity and Access Management (IAM) permission policy statements.

For more information about Amazon WorkMail resource-level permissions, see Resources (p. 18). For more information about controlling access based on tags, see Authorization based on Amazon WorkMail tags (p. 20).

Amazon WorkMail administrators can tag organizations using the Amazon WorkMail console.

To add tags to an Amazon WorkMail organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Tags**.
4. For **Organization tags**, choose **Add new tag**.
5. For **Key**, enter a string to identify the tag.
6. (Optional) For **Value**, enter a value for the tag.
7. (Optional) Repeat steps 4-6 to add more tags to your organization. You can add up to 50 tags.
8. Choose **Submit** to save your changes.

You can view your organization tags in the Amazon WorkMail console.

Developers can also tag organizations using the AWS SDK or AWS Command Line Interface (AWS CLI). For more information, see the `TagResource`, `ListTagsForResource`, and `UntagResource` commands in the Amazon WorkMail API Reference or the AWS CLI Command Reference.

You can remove tags from an organization at any time, using the Amazon WorkMail console.

To remove tags from an Amazon WorkMail organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Tags**.
4. For **Organization tags**, choose **Remove** next to the tag to remove.
5. Choose **Submit** to save your changes.

Working with access control rules

Access control rules for Amazon WorkMail allow administrators to control how their organization's users are granted access to Amazon WorkMail. Each Amazon WorkMail organization has a default access control rule that grants mailbox access to all users added to the organization, no matter which access protocol or IP address they use. Administrators can edit or replace the default rule with one of their own, add a new rule, or delete a rule.

> **Warning**
> If an administrator deletes all access control rules for an organization, Amazon WorkMail blocks all access to the organization's mailboxes.

Administrators can apply access control rules that allow or deny access based on the following criteria:

- **Protocols** – The protocol used to access the mailbox, such as **Autodiscover**, **EWS**, **IMAP**, **SMTP**, **ActiveSync**, **Outlook for Windows**, and **Webmail**.
- **IP addresses** – The IPv4 CIDR ranges used to access the mailbox.
- **Amazon WorkMail users** – The user IDs in your organization that are used to access the mailbox.

Administrators apply access control rules in addition to the user's mailbox and folder permissions. For more information, see Working with mailbox permissions (p. 103) and Sharing folders and folder permissions in the *Amazon WorkMail User Guide*.

> **Note**
> Access control rules do not apply to Amazon WorkMail console or SDK access. Use AWS Identity and Access Management (IAM) roles or policies instead. For more information, see Identity and access management for Amazon WorkMail (p. 13).

Creating access control rules

Create new access control rules from the Amazon WorkMail console.

To create a new access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Access control rules**.

4. Choose **Create rule**.
5. For **Description**, enter a description for the rule.
6. For **Effect**, choose **Allow** or **Deny**. This allows or denies access based on the conditions that you select in the following step.
7. For **This rule applies to requests that ...**, select the conditions to apply to the rule, such as whether to include or exclude specific protocols, IP addresses, or users.
8. (Optional) If you enter IP address ranges or user IDs, choose **Add** to add them to the rule.
9. Choose **Create rule**.

Editing access control rules

Edit new and default access control rules from the Amazon WorkMail console.

To edit an access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Access control rules**.
4. Select the rule to edit.
5. Choose **Edit rule**.
6. Edit the description, effect, and conditions, as needed.
7. Choose **Save changes**.

Testing access control rules

To see how your organization's access control rules are applied, test the rules from the Amazon WorkMail console.

To test access control rules for your organization

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Access control rules**.
4. Choose **Test rules**.
5. For **Request context**, select the protocol to test for.
6. For **Source IP address**, enter the IP address to test for.
7. For **User**, enter the user to test for.
8. Choose **Test**.

The test results appear under **Effect**.

Deleting access control rules

Delete access control rules that you no longer require from the Amazon WorkMail console.

> **Warning**
> If an administrator deletes all access control rules for an organization, Amazon WorkMail blocks all access to the organization's mailboxes.

To delete an access control rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Access control rules**.
4. Select the rule to delete.
5. Choose **Delete rule**.
6. Choose **Delete**.

Setting mailbox retention policies

Set mailbox retention policies for your Amazon WorkMail organization that automatically delete email messages from user mailboxes after a time period that you choose. You can choose which mailbox folders to apply retention policies to, and choose whether to set different retention policies for different folders. Mailbox retention policies apply to the selected folders in all of the user mailboxes in your organization. Users cannot override the retention policies.

To set a mailbox retention policy

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Retention policy**.
4. For **Folder actions**, next to each mailbox folder that you want to include in the policy, select **Delete** or **Permanently delete**.
5. Enter the number of days to keep the email messages in each mailbox folder before deleting them.
6. Choose **Save**.

It can take up to 48 hours to apply the retention policies for your organization. If you choose the **Delete** folder action, users can recover deleted email messages from the Amazon WorkMail web application and supported clients. If you choose the **Permanently delete** folder action, email messages cannot be recovered after they are deleted.

The number of days in a retention policy are counted starting with the day that an email message is placed in the selected folder. For example, if a user moves an email message to their **Deleted Items** folder, the retention policy deletes the email message after the number of days you set for the **Deleted Items** folder. When a user sends an email message and it appears in their **Sent Items** folder, the retention policy deletes the email message after the number of days you set for the **Sent Items** folder. Anytime a user moves an email message from one folder to another, the retention policy deletes the email message after the number of days you set for the destination folder.

Editing your organization's mobile device policy

You can edit your organization's mobile device policy to change the way that mobile devices interact with Amazon WorkMail.

To edit your organization's mobile device policy

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.

2. If necessary, change the AWS Region. From the navigation bar, select the appropriate Region. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Mobile Policies**, and then on the **Mobile policy** screen, choose **Edit**.
5. Update any of the following as necessary:

    a. **Require encryption on device**: Encrypt email data on the mobile device.
    b. **Require encryption on storage card**: Encrypt email data on the mobile device's removable storage.
    c. **Password required**: Require a password to lock a mobile device.
    d. **Allow simple password**: Use the PIN on the device as the password.
    e. **Minimal password length**: Set the number of characters required in a valid password.
    f. **Require alphanumeric password:** Require that passwords are made up of letters and numbers.
    g. **Minimum number of character sets**: Specify the number of character sets required in a password, such as lowercase and uppercase letters, symbols, and numbers.
    h. **Number of failed attempts allowed**: Specify the number of failed login attempts that are allowed before the user is locked out of their account.
    i. **Password expiration**: Specify the number of days before a password expires and must be changed.
    j. **Enable screen lock**: Specify the number of seconds that must elapse without user input to lock the user's screen.
    k. **Enforce password history**: Specify the number of passwords that can be entered before repeating the same password.
6. Choose **Save**.

Managing email flows

Set up *email flow rules* to handle email flows based on email addresses or domains. Email flow rules are based on both the sender's and recipient's email addresses or domains.

To create an email flow rule, specify a *rule action* (p. 61) to apply to an email when a specified *pattern* (p. 63) is matched.

Topics

Inbound email rule actions

Inbound email flow rules help prevent undesirable email from reaching your users' mailboxes. You can use these rules with an AWS Lambda function to process incoming email before it is delivered to your

users' mailboxes. For more information about using Lambda with Amazon WorkMail, see Configuring AWS Lambda for Amazon WorkMail (p. 65). For more information about Lambda, see the *AWS Lambda Developer Guide*.

Inbound email flow rules, also called rule actions, automatically apply to all email messages sent to anyone inside of the Amazon WorkMail organization. This differs from email rules for individual mailboxes.

The following rule actions define how inbound email is handled. For each rule, you specify sender and recipient patterns (p. 63) together with one of the following actions.

Action	Description
Drop email	The email message is ignored. It is not delivered, and the sender is not notified of the non-delivery.
Send bounce response	The email message is not delivered, and the sender is notified of the non-delivery in a bounce message.
Deliver to junk folder	The email message is delivered to users' spam or junk folders, even if it is not originally identified as spam by the Amazon WorkMail spam detection system.
Default	The email message is delivered after being checked by the Amazon WorkMail spam detection system. Spam email is delivered to the junk folder. All other email messages are delivered to the inbox. Other email flow rules with a less specific sender pattern are ignored. To add exceptions to domain-based email flow rules, configure the Default action with a more specific sender pattern. For more information, see Sender and recipient patterns (p. 63).
Never deliver to junk folder	The email message is always delivered to users' inboxes, even if it is identified as spam by the Amazon WorkMail spam detection system. **Important** By not using the default spam detection system, you could expose your users to high-risk content from the addresses that you specify.
Run Lambda	Passes the email message to a Lambda function for processing before or while it is delivered to users' inboxes.

Note
Inbound email is first delivered to Amazon Simple Email Service (Amazon SES) and then to Amazon WorkMail. If Amazon SES blocks an incoming email message, then rule actions won't apply. For example, Amazon SES blocks an email message when a known virus is detected or because of explicit IP filtering rules. Specifying a rule action, such as **Default**, **Deliver to junk folder**, or **Never deliver to junk folder** has no effect.

Outbound email rule actions

Outbound email flow rules can be used to direct email messages via SMTP gateways, or to block senders from sending email messages to specified recipients. For more information about SMTP gateways, see Configuring SMTP gateways (p. 64).

Outbound email flow rules can also be used to pass the email message to an AWS Lambda function for processing after the email is sent. For more information about using Lambda with Amazon WorkMail, see Configuring Lambda for Amazon WorkMail (p. 65). For more information about Lambda, see the *AWS Lambda Developer Guide*.

The following rule actions define how outbound email is handled. For each rule, you specify sender and recipient patterns (p. 63) together with one of the following actions.

Action	Description
Default	The email message is sent via the normal flow.
Drop email	The email message is dropped. It is not sent, and the sender is not notified.
Send bounce response	The email message is not sent, and the sender is notified with a message that the administrator blocked the email message.
Route to SMTP gateway	The email message is sent via a configured SMTP gateway.
Run Lambda	Passes the email message to a Lambda function for processing before or while the email message is sent.

Sender and recipient patterns

An email flow rule can apply to a specific email address, or all email addresses under a specific domain or set of domains. You define a pattern to determine the email addresses that a rule applies to.

Both sender and recipient patterns take one of the following forms:

- **An email address** matches a single email address; for example:

```
mailbox@example.com
```

- **A domain name** matches all email addresses under that domain; for example:

```
example.com
```

- **A wildcard domain** matches all email addresses under that domain and all of its subdomains. A wildcard appears only at the front of a domain; for example:

```
*.example.com
```

- **Star** matches any email addresses under any domain.

```
*
```

Note
The + symbol is not valid inside of sender or recipient patterns.

Multiple patterns can be specified for one rule. For more information, see Inbound email rule actions (p. 61) and Outbound email rule actions (p. 63).

Inbound email flow rules are applied if either the `Sender` or `From` header in an inbound email message matches any patterns. If present, the `Sender` address is matched first. The `From` address is matched if there is no `Sender` header or if the `Sender` header doesn't match any rule. If there are multiple recipients for the email message that match different rules, each rule applies for the matched recipients.

Outbound email flow rules are applied if the recipient and either the `Sender` or `From` header in an outbound email message matches any patterns. If there are multiple recipients for the email message that match different rules, each rule applies for the matched recipients.

If multiple rules match, the action of the most specific rule is applied. An example is when a rule for a specific email address takes precedence over a rule for an entire domain. If multiple rules have the same specificity, the most restrictive action is applied. An example is when a **Drop** action takes precedence over a **Bounce** action. The order of precedence for actions is the same as the order in which they are listed in Inbound email rule actions (p. 61) and Outbound email rule actions (p. 63).

Note
Take care when creating rules with overlapping sender patterns with **Drop** or **Bounce** actions. Unexpected precedence ordering could result in many inbound email messages not being delivered.

Creating an email flow rule

To create an email flow rule, you specify a rule action (p. 61) to apply to an email message when a specified pattern (p. 63) is matched. When you create a new email flow rule, it's applied immediately.

To create a new email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Choose **Create Rule**.
5. Enter a name for your rule.
6. Enter one or more sender patterns and one or more recipient patterns.
7. Select the action to apply to the email.
8. Choose **Create Rule**.

You can test the new email flow rule that you created. For more information, see Testing an email flow rule (p. 74).

Configuring SMTP gateways

You can configure SMTP gateways to use with outbound email flow rules. Outbound email flow rules let you route email messages sent from your Amazon WorkMail organization through an SMTP gateway. For more information, see Outbound email rule actions (p. 63).

Note
SMTP gateways configured for outbound email flow rules must support TLS v1.2 using certificates from major certificate authorities. Only basic authentication is supported.

To configure an SMTP gateway

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. Choose **Organization settings**, **SMTP gateways**.
4. Choose **Create gateway**.
5. Enter a name for the gateway, and the server address and port.
6. For **Basic authentication**, enter the user name and password for authentication with the gateway.
7. Choose **Create gateway**.
8. The SMTP gateway is available for use with outbound email flow rules.

When you configure an SMTP gateway to use in an outbound email flow rule, all outbound email messages that match the rule with the SMTP gateway action are routed to the corresponding SMTP gateway. The SMTP gateway handles the rest of the email delivery.

If Amazon WorkMail is unable to reach the SMTP gateway, the email message is bounced back to the sender. If this occurs, correct the SMTP gateway settings in the Amazon WorkMail console by choosing **Organizations**, the name of your organization, **Organization settings**, **SMTP gateways**.

Configuring AWS Lambda for Amazon WorkMail

Use the **Run Lambda** action in inbound and outbound email flow rules to pass email messages that match the rules to an AWS Lambda function for processing.

Choose from the following configurations for a **Run Lambda** action in Amazon WorkMail.

Synchronous Run Lambda configuration

Email messages that match the flow rule are passed to a Lambda function for processing before they are sent or delivered. Use this configuration to modify email content, and to control inbound or outbound email flow for use cases such as blocking delivery of sensitive email messages, removing attachment, adding disclaimers, and so on.

Asynchronous Run Lambda configuration

Email messages that match the flow rule are passed to a Lambda function for processing while they are sent or delivered. This configuration does not affect email delivery and is used for tasks such as collecting metrics for inbound or outbound email messages.

Whether you choose synchronous or asynchronous configuration, the event object passed to your Lambda function contains metadata for the inbound or outbound email event. You can also use the message ID in the metadata to access the full content of the email message. For more information, see Retrieving message content with AWS Lambda (p. 70). For more information about email events, see Lambda event data (p. 66).

For more information about inbound and outbound email flow rules, see Managing email flows (p. 61). For more information about Lambda, see the *AWS Lambda Developer Guide*.

Note
Currently, Lambda email flow rules reference only Lambda functions in the same AWS Region and AWS account as the Amazon WorkMail organization being configured.

Getting started with AWS Lambda for Amazon WorkMail

To start using AWS Lambda with Amazon WorkMail, we recommend deploying the WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account. The function

has all the necessary resources, and the permissions configured for you. For more examples, see the amazon-workmail-lambda-templates repository on GitHub.

If you choose to create your own Lambda function, you must configure permissions using the AWS Command Line Interface (AWS CLI). In the following example command, replace `MY_FUNCTION_NAME` with the name of your Lambda function, and replace `REGION` with your Amazon WorkMail AWS Region. Available Amazon WorkMail Regions include `us-east-1`, `us-west-2`, and `eu-west-1`.

```
aws --region REGION lambda add-permission --function-name MY_FUNCTION_NAME --statement-id
 AllowWorkMail --action "lambda:InvokeFunction" --principal workmail.REGION.amazonaws.com
```

For more information about using the AWS CLI, see the *AWS Command Line Interface User Guide*.

Configuring synchronous **Run Lambda** rules

To configure a synchronous **Run Lambda** rule, create an email flow rule with the **Run Lambda** action and select the **Run synchronously** check box. For more information about creating mail flow rules, see Creating an email flow rule (p. 64).

To finish creating the synchronous rule, add the Lambda Amazon Resource Name (ARN) and configure the following options.

Fallback action

The action Amazon WorkMail applies if the Lambda function fails to run. This action also applies to any recipients that are omitted from the Lambda response if the **allRecipients** flag is not set. The **Fallback action** cannot be another Lambda action.

Rule timeout (in minutes)

The time period during which the Lambda function is retried if Amazon WorkMail fails to invoke it. The **Fallback action** is applied at the end of this time period.

Note
Synchronous **Run Lambda** rules support the * destination condition only.

Lambda event data

The Lambda function is triggered using the following event data. The presentation of the data varies depending on which programming language is used for the Lambda function.

```
{
    "summaryVersion": "2018-10-10",
    "envelope": {
        "mailFrom" : {
            "address" : "from@example.com"
        },
        "recipients" : [
           { "address" : "recipient1@example.com" },
           { "address" : "recipient2@example.com" }
        ]
    },
    "sender" : {
        "address" :  "sender@example.com"
    },
    "subject" : "Hello From Amazon WorkMail!",
    "messageId": "00000000-0000-0000-0000-000000000000",
    "invocationId": "00000000000000000000000000000000",
```

```
    "flowDirection": "INBOUND",
    "truncated": false
}
```

The event JSON includes the following data.

summaryVersion

The version number for **LambdaEventData**. Only updates when a backwards incompatible change is made in **LambdaEventData**.

envelope

The envelope of the email message, which includes the following fields.

mailFrom

The **From** address, which is usually the email address of the user who sent the email message. If the user sent the email message as another user or on behalf of another user, the **mailFrom** field returns the email address of the user on whose behalf the email message was sent, not the email address of the actual sender.

recipients

A list of recipient email addresses. There is no distinction between **To**, **CC**, or **BCC**.

Note
For inbound email flow rules, this list includes recipients for each domain that is part of the Amazon WorkMail organization in which the rule is created. The Lambda function is invoked separately for each SMTP conversation from the sender, and the recipients field lists the recipients from that SMTP conversation. Recipients with external domains are not included.

sender

The email address of the user who sent the email message on behalf of another user. This field is set only when an email message is sent on behalf of another user.

subject

The email subject line. Truncated when it exceeds the 256 character limit.

messageId

A unique ID used to access the full content of the email message when using the Amazon WorkMail Message Flow SDK.

invocationID

The ID for a unique Lambda invocation. This ID remains the same when a Lambda function is called more than once for the same **LambdaEventData**. Use to detect retries and avoid duplication.

flowDirection

Indicates the direction of the email flow, either **INBOUND** or **OUTBOUND**.

truncated

Applies to the payload size, not the subject line length. When `true`, the payload size exceeds the 128 KB limit, so the list of recipients is truncated in order to meet the limit.

Synchronous **Run Lambda** response schema

When an email flow rule with a synchronous **Run Lambda** action matches an inbound or outbound email message, Amazon WorkMail calls the configured Lambda function and waits for the response before

taking action on the email message. The Lambda function returns a response according to a pre-defined schema that lists the actions, action types, applicable parameters, and recipients that the action applies to.

The following schema is an example of a synchronous **Run Lambda** response. Responses vary based on the programming language used for the Lambda function.

```
{
      'actions': [
      {
        'action' : {
          'type': 'string',
          'parameters': { various }
        },
        'recipients': list of strings,
        'allRecipients': boolean
      }
    ]
}
```

The response JSON includes the following data.

action

The action to take for the recipients.

type

The action type. Action types are not returned for asynchronous **Run Lambda** actions.

Inbound rule action types include **BOUNCE**, **DROP**, **DEFAULT**, **BYPASS_SPAM_CHECK**, and **MOVE_TO_JUNK**. For more information, see Inbound email rule actions (p. 61).

Outbound rule action types include **BOUNCE**, **DROP**, and **DEFAULT**. For more information, see Outbound email rule actions (p. 63).

parameters

Additional action parameters. Supported for the **BOUNCE** action type as a JSON object with the key **bounceMessage** and value **string**. This bounce message is used to create the bounce email message.

recipients

List of email addresses on which the action should be taken. You can add new recipients to the response even if they were not included in the original recipients list. This field is not required if **allRecipients** is true for an action.

> **Note**
> When a Lambda action is called for inbound email, you can only add new recipients that are from your organization. The new recipients are added to the response as **BCC**.

allRecipients

When true, applies the action to all the recipients that are not subject to another specific action in the Lambda response.

Synchronous **Run Lambda** action limits

The following limits apply when Amazon WorkMail invokes Lambda functions for synchronous **Run Lambda** actions:

- Lambda functions must respond within 15 seconds, or be treated as failed invocations.

- Lambda function responses up to 256 KB are allowed.
- Up to 10 unique actions are allowed in the response. Actions greater than 10 are subject to the configured **Fallback action**.
- Up to 500 recipients are allowed for outbound Lambda functions.
- The maximum value for **Rule timeout** is 240 minutes. If the minimum value of 0 is configured, there are no retries before Amazon WorkMail applies the fallback action.

Synchronous **Run Lambda** action failures

If Amazon WorkMail is unable to invoke your Lambda function due to an error, invalid response, or Lambda timeout, Amazon WorkMail retries the invocation with exponential backoff until the **Rule timeout** period completes. Then, the **Fallback action** is applied to all recipients of the email message. For more information, see Configuring synchronous **Run Lambda** rules (p. 66).

Example synchronous **Run Lambda** responses

The following examples demonstrate the structure of common synchronous **Run Lambda** responses.

Example : Remove specified recipients from an email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for removing recipients from an email message.

```
{
    'actions': [{
      'action' : {'type': 'DEFAULT'},
      'allRecipients': True
    },
    {
      'action' : {'type': 'DROP'},
      'recipients' : ['drop-recipient@example.com']
    }]
}
```

Example : Bounce with a custom email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for bouncing with a custom email message.

```
{
      'actions' : [{
        'action' : {
          'type' : 'BOUNCE',
          'parameters' : {
            'bounceMessage' : 'Email in breach of company policy.'
          }
        },
        'allRecipients': True
      }]
}
```

Example : Add recipients to an email message

The following example demonstrates the structure of a synchronous **Run Lambda** response for adding recipients to the email message. This does not update the **To** or **CC** fields of the email message.

```
{
```

```
      'actions': [{
        'action' : { 'type' : 'DEFAULT' },
        'recipients' : [
          'new-recipient@example.com'
         ]
      },
      {
        'action' : { 'type' : 'DEFAULT' },
        'allRecipients' : True
      }]
}
```

For more code examples to use when creating Lambda functions for **Run Lambda** actions, see Amazon WorkMail Lambda templates.

More information about using Lambda with Amazon WorkMail

You can also access the full content of the email message that triggers the Lambda function. For more information, see Retrieving message content with AWS Lambda (p. 70).

Retrieving message content with AWS Lambda

After you configure an AWS Lambda function to manage email flows for Amazon WorkMail, you can access the full content of the email messages that are processed using Lambda. For more information about getting started with Lambda for Amazon WorkMail, see Configuring AWS Lambda for Amazon WorkMail (p. 65).

To access the full content of email messages, use the `GetRawMessageContent` action in the Amazon WorkMail Message Flow API. The email message ID that is passed to your Lambda function upon invocation sends a request to the API. Then, the API responds with the full MIME content of the email message. For more information, see Amazon WorkMail Message Flow in the *Amazon WorkMail API Reference*.

The following example shows how a Lambda function using the Python runtime environment can retrieve the full message content.

Tip
If you start by deploying the Amazon WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account, the system creates a lambda function in your account with all the necessary resources and permission. You can then add your business logic to the lambda function based on your use-case.

```
import boto3
import email
import os

def email_handler(event, context):
    workmail = boto3.client('workmailmessageflow', region_name=os.environ["AWS_REGION"])
    msg_id = event['messageId']
    raw_msg = workmail.get_raw_message_content(messageId=msg_id)

    parsed_msg = email.message_from_bytes(raw_msg['messageContent'].read())
    print(parsed_msg)
```

For more detailed examples of ways to analyze the content of messages that are in transit, see the amazon-workmail-lambda-templates repository on GitHub.

Note
You only use the Amazon WorkMail Message Flow API to access email messages in transit. You can only access the messages within 24 hours of being sent or received. To programmatically

access messages in a user's mailbox, use one of the other protocols supported by Amazon WorkMail, such as IMAP or Exchange Web Services (EWS).

Updating message content with AWS Lambda

After you configure a synchronous AWS Lambda function to manage email flows, you can use the `PutRawMessageContent` action in the Amazon WorkMail Message flow API to update the content of in-transit email messages. For more information about getting started with Lambda functions for Amazon WorkMail, see Configuring synchronous **Run Lambda** rules (p. 66). For more information about the API, see PutRawMessageContent.

Tip
If you start by deploying the Amazon WorkMail Hello World Lambda function from the AWS Serverless Application Repository to your account, the system creates a Lambda function in your account with the necessary resources and permissions. You can then add business logic to the lambda function, based on your use cases.

As you go, remember the following:

1. Use the GetRawMessageContent API to retrieve the original message content. For more information see Retrieving message content with AWS Lambda (p. 70).
2. Once you have the original message, change the MIME content. When you finish, upload the message to an S3 bucket in your account. Ensure that the S3 bucket uses the same AWS account as your Amazon WorkMail operations, and that it uses the same AWS Region as your API calls.
3. For Amazon WorkMail to process requests, your S3 bucket must have the correct policy in order to access the S3 object. For more information, see Example S3 policy.
4. Use the PutRawMessageContent API to send the updated the message content back to Amazon WorkMail.

Note
The `PutRawMessageContent` API ensures that the MIME content of the updated message meets RFC standards, as wells as the criteria mentioned in the RawMessageContent data type. Emails inbound to your Amazon WorkMail organization do not always meet those standards, so the `PutRawMessageContent` API may reject them. In such cases, you can consult the error message returned for more information on how to fix any issues.

Example S3 policy

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "workmail.region.amazonaws.com"
            },
            "Action": [
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": "arn:aws:s3:::My-Test-S3-Bucket/*",
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "true"
                }
            }
        }
    ]
```

```
}
```

The following example shows how a Lambda function uses the Python runtime to update the subject of an in-transit email message.

```
    import boto3
    import os
    import uuid
    import email

    def email_handler(event, context):
        workmail = boto3.client('workmailmessageflow',
 region_name=os.environ["AWS_REGION"])
        s3 = boto3.client('s3', region_name=os.environ["AWS_REGION"])

        msg_id = event['messageId']
        raw_msg = workmail.get_raw_message_content(messageId=msg_id)
        parsed_msg = email.message_from_bytes(raw_msg['messageContent'].read())

        # Updating subject. For more examples, see https://github.com/aws-samples/amazon-
workmail-lambda-templates.
        parsed_msg.replace_header('Subject', "New Subject Updated From Lambda")

        # Store updated email in S3
        key = str(uuid.uuid4());
        s3.put_object(Body=parsed_msg.as_bytes(), Bucket="Your-S3-Bucket", Key=key)

        # Update the email in WorkMail
        s3_reference = {
            'bucket': "Your-S3-Bucket",
            'key': key
        }
        content = {
            's3Reference': s3_reference
        }
        workmail.put_raw_message_content(messageId=msg_id, content=content)
```

For more examples of ways to analyze the content of in-transit messages, see the amazon-workmail-lambda-templates repository on GitHub.

Managing access to the Amazon WorkMail Message Flow API

Use AWS Identity and Access Management (IAM) policies to manage access to the Amazon WorkMail Message Flow API.

The Amazon WorkMail Message Flow API works with a single resource type, an email message in transit. Each email message in transit has a unique Amazon Resource Name (ARN) associated with it.

The following example shows the syntax of an ARN associated with an email message in transit.

```
arn:aws:workmailmessageflow:region:account:message/organization/context/messageID
```

Changeable fields in the preceding example include the following:

- **Region** – The AWS Region for your Amazon WorkMail organization.
- **Account** – The AWS account ID for your Amazon WorkMail organization.
- **Organization** – Your Amazon WorkMail organization ID.
- **Context** – Indicates whether the message is `incoming` to your organization, or `outgoing` from it.

- **Message ID** – The unique email message ID that is passed as input to your Lambda function.

The following example includes example IDs for an ARN associated with an incoming email message in transit.

```
arn:aws:workmailmessageflow:us-east-1:111122223333:message/m-
n1pq2345678r901st2u3vx45x6789yza/incoming/d1234567-8e90-1f23-456g-hjk7lmnop8q9
```

You can use these ARNs as resources in the `Resource` section of your IAM user policies in order to manage access to Amazon WorkMail messages in transit. For more information about granting IAM users permissions for Amazon WorkMail, see Create AWS Identity and Access Management users and groups (p. 5).

Example IAM policies for Amazon WorkMail message flow access

The following example policy grants an IAM entity full read access to all inbound and outbound messages for every Amazon WorkMail organization in your AWS account.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/*",
            "Effect": "Allow"
        }
    ]
}
```

If you have multiple organizations in your AWS account, you can also limit access to one or more organizations. This is useful if certain Lambda functions should only be used for certain organizations.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/organization/
*",
            "Effect": "Allow"
        }
    ]
}
```

You can also choose to grant access to messages depending on whether they are `incoming` to your organization, or `outgoing` from it. To do this, use the qualifier `incoming` or `outgoing` in the ARN.

The following example policy grants access only to messages that are incoming to your organization.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Action": [
                "workmailmessageflow:GetRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/organization/
incoming/*",
            "Effect": "Allow"
        }
    ]
}
```

The following example policy grants an IAM entity full read and update access to all inbound and outbound messages for every Amazon WorkMail organization in your AWS account.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "workmailmessageflow:GetRawMessageContent",
                "workmailmessageflow:PutRawMessageContent"
            ],
            "Resource": "arn:aws:workmailmessageflow:region:account:message/*",
            "Effect": "Allow"
        }
    ]
}
```

Testing an email flow rule

To check your current rule configuration, you can test how the configuration behaves against specific email addresses.

To test an email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Next to **Test configuration**, enter the full email addresses of both the sender and recipient that you want to test.
5. Choose **Test**. The action to be taken for the provided email address is displayed.

Modifying an email flow rule

You can modify the rule action (p. 61) or pattern (p. 63) for an email flow rule. When you modify an email flow rule, the changes are applied immediately.

To modify email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Select the rule and choose **Edit**.
5. Change the patterns or action as required.
6. Choose **Save**.

You can test the new email flow rule that you created. For more information, see Testing an email flow rule (p. 74).

Removing an email flow rule

When you remove an email flow rule, the changes are applied immediately.

To remove an email flow rule

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Inbound/Outbound rules**.
4. Select the rule and choose **Remove**.
5. At the confirmation prompt, choose **Remove**.

Tracking messages

Turn on email event logging in the Amazon WorkMail console to track email messages for your organization. Email event logging uses an AWS Identity and Access Management service-linked role to grant permissions to publish the email event logs to Amazon CloudWatch. For more information about IAM service-linked roles, see Using service-linked roles for Amazon WorkMail (p. 26).

In the CloudWatch event logs, you can use CloudWatch search tools and metrics to track messages and troubleshoot email issues. For more information about the event logs that Amazon WorkMail sends to CloudWatch, see CloudWatch event logs for Amazon WorkMail (p. 30). For more information about CloudWatch Logs, see the Amazon CloudWatch Logs User Guide.

Turning on email event logging

When you turn on email event logging using the default settings, Amazon WorkMail:

- Creates an AWS Identity and Access Management service-linked role – `AmazonWorkMailEvents`.
- Creates a CloudWatch log group – `/aws/workmail/emailevents/`*`organization-alias`*.
- Sets CloudWatch log retention to 30 days.

To turn on email event logging

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Monitoring**.
4. For **Log settings**, choose **Edit**.
5. For **Log Events**, select **Enable mail events**.
6. Do one of the following:

    a. (Recommended) Select **Use default settings**.

    b. (Optional) Clear the check box for **Use default settings**, and select a **Destination Log Group** and **IAM Role**.

    Note
    Choose this option only if you have already created a log group and custom IAM role using the AWS CLI. For more information, see Creating a custom log group and IAM role for email event logging (p. 76).

7. Select **I authorize Amazon WorkMail to publish logs in my account using this configuration**.
8. Choose **Save**.

Creating a custom log group and IAM role for email event logging

We recommend using the default settings when enabling email event logging for Amazon WorkMail. If you require a custom monitoring configuration, you can use the AWS CLI to create a dedicated log group and custom IAM role for email event logging.

To create a custom log group and IAM role for email event logging

1. Using the AWS CLI, create a log group in the same AWS Region as your Amazon WorkMail organization. For more information, see create-log-group in the *AWS CLI Command Reference*.

```
aws --region us-east-1 logs create-log-group --log-group-name workmail-monitoring
```

2. Create a file containing the following policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "events.workmail.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

3. Using the AWS CLI, create an IAM role and attach this file as the role policy document. For more information, see create-role in the *AWS CLI Command Reference*.

```
aws iam create-role --role-name workmail-monitoring-role --assume-role-policy-document
 file://trustpolicyforworkmail.json
```

> **Note**
> If you are a `WorkMailFullAccess` managed policy user, you must include the term `workmail` in the role name. This managed policy only allows you to configure email event logging using roles with `workmail` in the name. For more information, see Granting a user permissions to pass a role to an AWS service in the *IAM User Guide*.

4. Create a file containing the policy for the IAM role you created in the previous step. At minimum, the policy must grant permissions to the role to create log streams and put log events into the log group that you created in step 1.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
```

```
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:us-east-1:111122223333:log-group:workmail-
monitoring*"
        }
    ]
}
```

5. Using the AWS CLI, attach the policy file to the IAM role. For more information, see put-role-policy in the *AWS CLI Command Reference*.

```
aws iam put-role-policy --role-name workmail-monitoring-role --policy-name workmail-
permissions --policy-document file://rolepolicy.json
```

Follow the steps in the previous topic to turn on email event logging using the newly created log group and role. For more information, see Turning on email event logging (p. 75).

Turning off email event logging

Turn off email event logging from the Amazon WorkMail console. If you no longer need to use email event logging, we recommend that you delete the related CloudWatch log group and service-linked role as well. For more information, see Deleting a service-linked role for Amazon WorkMail (p. 27).

To turn off email event logging

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**, **Monitoring**.
4. For **Log settings**, choose **Edit**.
5. Clear the check box for **Enable mail events**.
6. Choose **Save**.

Enforcing DMARC policies on incoming email

Email domains use DNS records for security. They protect your users from common attacks such as spoofing or phishing. DNS records for domains often include DMARC TXT records, which are set by the domain owner that is sending the email. DMARC TXT records include policies that specify actions to take when an email fails a DMARC check. You can choose whether to enforce the DMARC policy on emails being sent to your organization.

New Amazon WorkMail organizations have DMARC enforcement turned on by default.

To turn on DMARC enforcement

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. In the navigation pane, choose **Organization settings**.
4. Choose **Advanced**.
5. For **Inbound DMARC Settings**, choose **Edit**.
6. For **DMARC enforcement**, select **On**.

7. Select the acknowledgment check box.
8. Choose **Save**.

To turn off DMARC enforcement

- Follow steps 1-8 in the previous section, but for step 6, choose **Off** instead of **On**.

Using email event logging to track DMARC enforcement

Turning on DMARC enforcement might result in inbound emails being dropped or marked as spam, depending on how the sender configured their domain. If a sender misconfigures their email domain, your users might stop receiving legitimate emails. To check for emails that aren't being delivered to your users, you can enable email event logging for your Amazon WorkMail organization. Then, you can query your email event logs for inbound emails that are filtered out based on the sender's DMARC policies.

Before you use email event logging to track DMARC enforcement, enable email event logging in the Amazon WorkMail console. To get the most out of your log data, allow some time to pass while email events are logged. For more information and instructions, see the section called “Turning on email event logging” (p. 75).

To use email event logging to track DMARC enforcement

1. In the CloudWatch Insights console, under **Logs**, choose **Insights**.
2. For **Select log group(s)**, select your Amazon WorkMail organization's log group. For example, /aws/workmail/events/organization-alias.
3. Select a time period to query.
4. Run the following query: **stats count() by event.dmarcPolicy | filter event.dmarcVerdict == "FAIL"**
5. Choose **Run query**.

You can also set up custom metrics for these events. For more information, see Creating metric filters.

Working with domains

You can add or remove email domains or make them the default.

Topics

Adding a domain

You can add up to 100 domains to your Amazon WorkMail organization for sending email. When you add a new domain, an Amazon Simple Email Service (Amazon SES) sending authorization policy is automatically added to the domain identity policy. This provides Amazon WorkMail with access to all Amazon SES sending actions for your domain and allows you to redirect email to your domain as well as external domains.

> **Note**
> As a best practice, you should add aliases for postmaster@ and abuse@. You can create distribution groups for these aliases if you want certain users in your organization to receive mail sent to these aliases.

To add a domain

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of the organization to which to add a domain.
4. In the navigation pane, choose **Domains**, **Add domain**.
5. On the **Add domain** screen, enter the domain name to add. Domain names can contain Basic Latin (ASCII) characters.
   - (Optional) If you have a domain that is managed in an Amazon Route 53 public hosted zone, you can choose it from the dropdown menu that appears.
6. Choose **Add domain**.
   - (Optional) If you add a domain for which you are using Route 53 as the DNS service, and the hosted zone does not contain any records for Amazon WorkMail (such as MX records), you are redirected to the **Automatic Configuration** page. Choose **Configure automatically** and follow the

prompts to have Amazon WorkMail automatically insert the DNS records for you, and skip the rest of this procedure.

7. In the console section **Step 1: Verify domain ownership**, the TXT record verifies your ownership of the domain.

   After all your users and distribution groups are created, and mailboxes are successfully migrated, you can update the MX record to start forwarding email to Amazon WorkMail. Updates to DNS records can take up to 48 hours to process. For information about creating DNS records, see step 8.

8. In the console sections **Step 2: Finalize domain setup** and **Step 3: Increase security (recommended)**, the following records are listed:

   - The MX record to deliver incoming email to Amazon WorkMail.
   - The CNAME autodiscover record that allows users to easily configure their Microsoft Outlook or mobile device knowing only their email address and password.
   - The CNAME records for DKIM signing. For more information about DKIM signing, see Authenticating email with DKIM in the *Amazon Simple Email Service Developer Guide*.
   - The TXT record for SPF verification. For more information about SPF verification, see Authenticating email with SPF (p. 89).
   - The TXT record for DMARC. For more information about DMARC records in Amazon WorkMail, see Complying with DMARC using Amazon SES in the *Amazon Simple Email Service Developer Guide*.

   Important
   Some DNS providers automatically append the domain name to the end of DNS records. Adding a record that already contains the domain name (such as _amazonses.example.com) might result in the duplication of the domain name (such as _amazonses.example.com.example.com). To avoid duplicating the domain name in the record name, add a period to the end of the domain name in the DNS record. This indicates to your DNS provider that the record name is fully qualified, meaning that it is no longer relative to the domain name. It also prevents the DNS provider from appending an additional domain name.

   You can copy these records for use with your DNS service. The record names that are copied include the domain name. Depending on which DNS service you use, the domain name might already be added to the domain's DNS record.

   The records on the domain page also include the verification status. After you create a record, choose the refresh icon to see the verification status and record value. For more information about verifying domains, see Verifying domains (p. 82).

   The following table shows the available verification statuses for each record type.

	Verified	Pending	Missing	Failed	Inconsistent
TXT ownership record	Record resolved and verified.	Record not verified yet.	Not applicable	Unable to verify ownership. Record mismatched or unreachable.	Not applicable
MX	Record resolved and verified.	Not applicable	Unable to resolve record.	Not applicable	Value does not match expected record.

	Verified	Pending	Missing	Failed	Inconsistent
AutoDiscover	Record resolved and verified.	Not applicable	Unable to resolve record.	Not applicable	Value does not match expected record.
DKIM CNAME records	Record resolved and verified.	Record not verified yet.	Not applicable	Unable to verify ownership. Record mismatched or unreachable.	Not applicable
SPF TXT record	Record resolved and verified.	Not applicable	Unable to resolve record.	Not applicable	Value does not match expected record.
DMARC TXT record	Record resolved and verified.	Not applicable	Unable to resolve record.	Not applicable	Value does not match expected record.

Note
The AutoDiscover domain verification also checks for correct AutoDiscover setup. After phase 2 and phase 3 verification are complete, a check mark appears next to the **Verified** status.

We recommend that you set the Time to Live (TTL) to 3600 of the MX and autodiscover CNAME record. Reducing the TTL ensures that your mail servers don't use outdated or invalid MX records after updating your MX records or migrating your mailboxes.

9. We recommend configuring your domain as the `MAIL FROM` domain. You can see the status of your `MAIL FROM` domain in the console section **Step 4: Enhance deliverability (recommended)**. For more information, see Configuring a custom MAIL FROM domain in Amazon WorkMail (p. 89).

Removing a domain

When you no longer need a domain, you can delete it.

Note
You can't delete a domain when there are users or groups using the domain as their email address.

To remove a domain

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of the organization from which to remove the domain.
4. In the list of domains, select the check box next to the domain name and choose **Remove**.

5. In the **Remove domain** dialog box, type the name of the domain to remove and choose **Remove**.

Choosing the default domain

To use a domain as default in the email address of your users and groups, you can choose a default domain. Making a domain the default does not change existing email addresses.

To make a domain the default

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of the organization to which to add a default domain.
4. In the list of domains, select the check box next to the domain name and choose **Set as default**.

Verifying domains

After adding a domain for Amazon WorkMail in the console, the next step is to verify the domain. Verifying the domain confirms that you own the domain and that you are using Amazon WorkMail as the email service for the domain.

To verify a domain with Amazon WorkMail, initiate the process using the Amazon WorkMail console. Then add a TXT record to your DNS service as described in Verifying domains in Amazon SES in the *Amazon Simple Email Service Developer Guide*. Use the Amazon WorkMail console to verify that your DNS service is successfully updated with the TXT record for Amazon WorkMail. For more information, see Adding a domain (p. 79).

You can also use `nslookup` or `dig` to confirm that your Amazon WorkMail TXT records and MX records are updated with your DNS service.

Topics

Verifying TXT records and MX records with your DNS service

Confirm that the TXT record that verifies that you own the domain is added correctly to your DNS service. This procedure uses the nslookup tool, which is available for Windows and Linux. On Linux, you can also use dig.

In this procedure for the `nslookup` tool, you first find the DNS servers that serve your domain. Then you query those servers to view the TXT records. You query the DNS servers for your domain because those servers contain the most up-to-date information for your domain. This information can take time to propagate to other DNS servers.

To use nslookup to verify that your TXT record is added to your DNS service

1. Find the name servers for your domain:

a. Open a command prompt.

b. Run the following command to list all of the name servers that serve your domain.

```
nslookup -type=NS example.com
```

You query one of these name servers in the next step.

2. Verify that the TXT record is correctly added.

a. Run the following command using your domain and one of the name servers that you found in step 1.

```
nslookup -type=TXT _amazonses.example.com ns1.name-server.net
```

b. In the output of the command, verify that the string that follows `text =` matches the TXT value you see when you select the domain in the Verified Senders list of the Amazon WorkMail console.

In the example, you are looking for a TXT record under *_amazonses.example.com* with a value of `fmxqxT/icOYx4aA/bEUrDPMeax9/s3frblS+niixmqk=`. If the record is correctly updated, the command should have the following output:

```
_amazonses.example.com text = "fmxqxT/icOYx4aA/bEUrDPMeax9/s3frblS+niixmqk="
```

To use dig to verify that your TXT record is added to your DNS service

1. Open a terminal window.
2. Run the following command to list the TXT records for your domain.

```
dig +short example.com txt
```

3. Verify that the string that follows `TXT` matches the TXT value you see when you select the domain in the Verified Senders list of the Amazon WorkMail console.

To use nslookup to verify that your MX record is added to your DNS service

1. Find the name servers for your domain:

a. Open a command prompt.

b. Run the following command to list all of the name servers that serve your domain.

```
nslookup -type=NS example.com
```

You query one of these name servers in the next step.

2. Verify that the MX record is correctly added:

a. Run the following command using your domain and one of the name servers that you found in step 1.

```
nslookup -type=MX example.com ns1.name-server.net
```

b. In the output of the command, verify that the string that follows `mail exchange =` matches one of the following values:

For the US East (N. Virginia) Region, the record must be: `10 inbound-smtp.us-east-1.amazonaws.com`

For the Europe (Ireland) Region, the record must be: `10 inbound-smtp.eu-west-1.amazonaws.com`

For the US West (Oregon) Region, the record must be: `10 inbound-smtp.us-west-2.amazonaws.com`

Note
`10` represents the MX preference number or priority.

To use dig to verify that your MX record is added to your DNS service

1. Open a terminal window.
2. Run the following command to list the MX records for your domain.

```
dig +short example.com mx
```

3. Verify that the string that follows `MX` matches one of the following values:

   For the US East (N. Virginia) Region, the record must be: `10 inbound-smtp.us-east-1.amazonaws.com`

   For the Europe (Ireland) Region, the record must be: `10 inbound-smtp.eu-west-1.amazonaws.com`

   For the US West (Oregon) Region, the record must be: `10 inbound-smtp.us-west-2.amazonaws.com`

   Note
   `10` represents the MX preference number or priority.

Troubleshooting domain verification

For help troubleshooting domain verification, see the following suggestions:

- **Your DNS service does not allow underscores in TXT record names** – You can omit `_amazonses` from the TXT record name.
- **You want to verify the same domain multiple times and you can't have multiple TXT records with the same name** – You might need to verify your domain name more than once because you're sending from multiple AWS accounts using the same domain in the same Region. If your DNS service does not allow you to have multiple TXT records with the same name, there are two workarounds. The first workaround, if your DNS service allows it, is to assign multiple values to the TXT record. For example, if your DNS is managed by Amazon Route 53, you can set up multiple values for the same TXT record as follows:

  1. In the Route 53 console, choose the `_amazonses` TXT record that you added when you verified your domain in the first Region.
  2. For **Value**, press **Enter** after the first value.
  3. Add the value for the additional Region, and save the record set.

  If you only need to verify your domain twice, another workaround you can try is to verify it one time with `_amazonses` in the TXT record name and then omit `_amazonses` from the record name entirely. We recommend the multiple value solution as a best practice.

- **Amazon WorkMail reports that domain verification failed** – The domain displays a status of "failed" on the **Domains** tab of the Amazon WorkMail console. This means that Amazon WorkMail cannot find the necessary TXT record for your DNS service. Verify that the required TXT record is correctly added to your DNS service by using the procedure in Verifying TXT records and MX records with your DNS service (p. 82), and look for the following possible error.
  - **Your DNS provider appended the domain name to the end of the TXT record** – Adding a TXT record that already contains the domain name (such as _amazonses.example.com) might result in the duplication of the domain name (such as _amazonses.example.com.example.com). To avoid duplicating the domain name in the record name, add a period to the end of the domain name in the TXT record. This indicates to your DNS provider that the record name is fully qualified (that is, no longer relative to the domain name), and prevents the DNS provider from appending an additional domain name.
- **Amazon WorkMail reports that the MX record is Inconsistent** – When migrating from existing mail servers, the MX record might read **Inconsistent**. To resolve this, update your MX record to point to Amazon WorkMail instead of pointing to your previous mail server. The MX record is also returned as **Inconsistent** when a third-party email proxy is used along with Amazon WorkMail. If this is the case, it is safe to ignore the **Inconsistent** warning.

Enabling AutoDiscover to configure endpoints

AutoDiscover enables you to easily configure Microsoft Outlook and mobile clients with only your email address and password. The service also maintains a connection to Amazon WorkMail and updates local settings whenever endpoint or settings changes are made. In addition, AutoDiscover enables your client to use additional Amazon WorkMail features, such as the Offline Address Book, Out-of-Office Assistant, and the ability to view free/busy time in Calendar.

The client performs the following AutoDiscover phases to detect the server endpoint URLs:

- Phase 1: The client performs an SCP lookup against the local Active Directory. If your client isn't domain-joined, AutoDiscover skips this step.
- Phase 2: The client sends a request to the following URLs and validates the results. These endpoints are only available using HTTPS.
  - https://company.tld/autodiscover/autodiscover.xml
  - https://autodiscover.company.tld/autodiscover/autodiscover.xml
- Phase 3: The client performs a DNS lookup to autodiscover.company.tld and sends an unauthenticated GET request to the derived endpoint from the user's email address. If the server returns a 302 redirect, the client resends the AutoDiscover request against the returned HTTPS endpoint.

If all of these phases fail, the client can't be configured automatically, and you must set up the client manually. For information about manually configuring mobile devices, see Manually connect your device.

When you set up your domain in Amazon WorkMail, you are prompted to add the AutoDiscover DNS record. This enables the client to perform phase 3 of the AutoDiscover process. However, these steps don't work for all mobile devices, such as the stock Android email app, and you may need to set up AutoDiscover phase 2 manually.

There are two ways you can set up AutoDiscover phase 2 for your domain:

- By using Route 53 and Amazon CloudFront (recommended)
- By setting up an Apache web server with a reverse proxy

To enable AutoDiscover phase 2 with Route 53 and CloudFront

Note
The following steps show how to proxy https://autodiscover.company.tld/autodiscover/autodiscover.xml. To proxy https://company.tld/autodiscover/autodiscover.xml, remove the "autodiscover." prefix from the domains in the following steps.
For more information about applicable pricing, see Amazon CloudFront pricing and Amazon Route 53 pricing.

1. Get an SSL certificate for autodiscover.company.tld and upload it to AWS Identity and Access Management (IAM) or AWS Certificate Manager. For more information, see Working with server certificates in the *IAM User Guide*, or Getting started in the *AWS Certificate Manager User Guide*.
2. Create a new CloudFront distribution.
   1. Open the CloudFront console at https://console.aws.amazon.com/cloudfront/.
   2. Choose **Create Distribution**.
   3. For **Web**, choose **Get Started**.
   4. Enter the following values for **Origin Settings**:
      - For **Origin Domain Name**, enter the appropriate domain name for your Region: **`autodiscover-service.mail.us-east-1.awsapps.com`**, **`autodiscover-service.mail.eu-west-1.awsapps.com`**, or **`autodiscover-service.mail.us-west-2.awsapps.com`**.
      - **Origin Protocol Policy**: **`Match Viewer`**

        Note
        Leave **Origin path** blank, and do not change the auto-populated value for **Origin ID**.
   5. Select the following values for **Default Cache Behavior Settings**:
      - **Viewer Protocol Policy**: HTTPS Only
      - **Allowed HTTP Methods**: GET, HEAD, OPTIONS, PUT, POST, PATCH, DELETE
      - **Cache Based on Selected Request Headers**: All
      - **Forward Cookies**: All
      - **Query String Forwarding and Caching**: None (Improves Caching)
      - **Smooth Streaming**: No
      - **Restrict Viewer Access**: No
   6. Select the following values for **Distribution Settings**:
      - **Price Class**: Use only US, Canada, and Europe
      - For **Alternate Domain Names (CNAMEs)**, enter **`autodiscover.company.tld`** or **`company.tld`**
      - **SSL Certificate**: Custom SSL Certificate (stored in IAM)
      - **Custom SSL Client Support**: Choose **All Clients** or **Only Clients that Support Server Name Indication (SNI)**. Older versions of Android might not work with the latter option.

        Note
        If you choose **All Clients**, leave **Default Root Object** blank.
      - **Logging**: Choose **On** or **Off**.
      - For **Comment**, enter **`AutoDiscover type2 for autodiscover.company.tld`**
      - For **Distribution State**, choose **Enabled**.
   7. Choose **Create Distribution**.
3. In Route 53, create a record that routes internet traffic for your domain name to your CloudFront distribution:

   Note
   These steps assume that the DNS record for example.com is hosted in Route 53.

1. In the Route 53 console, choose **Hosted Zones** and **example.com**.
2. Choose **Create Record Set**, and then fill in the following fields:
   - **Name**: autodiscover.example.com
   - **Type**: A - IPv4 address
   - **Alias**: Yes
   - **Alias Target**: The CloudFront distribution created above

     Note
     If the CloudFront distribution created above is not present, wait a while and try again later. Change propagation for a new CloudFront distribution can take up to 1 hour.
   - **Evaluate Target Health**: No
3. Choose **Create**.

To enable AutoDiscover phase 2 with an Apache web server

1. Configure the following two directives on an SSL-enabled Apache server.

```
SSLProxyEngine on ProxyPass /autodiscover/autodiscover.xml
https://autodiscover-service.mail.REGION.awsapps.com/autodiscover/autodiscover.xml
```

2. If they are not already enabled, enable the following Apache modules:
   - proxy
   - proxy_http
   - socache_shmcb
   - ssl
3. Confirm that the endpoint is SSL-enabled and configured correctly.

AutoDiscover phase 2 troubleshooting

Post the following requests to your AutoDiscover endpoint to test it for correct configuration. If your AutoDiscover endpoint is configured correctly, it responds with an unauthorized request message.

To make a basic unauthorized request

- Create an unauthenticated POST request to the AutoDiscover endpoint.

If your endpoint is configured correctly, it should return a `401 unauthorized` message.

```
$ curl -X POST -v https://autodiscover.''company.tld''/autodiscover/autodiscover.xml
...
HTTP/1.1 401 Unauthorized
```

Next, run a real request that a mobile device would issue.

To run a real request

1. Create a request.xml file with the following XML content.

```
<?xml version="1.0" encoding="utf-8"?>
<Autodiscover xmlns="http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
requestschema/2006">
```

```
     <Request>
            <EMailAddress>testuser@company.tld</EMailAddress>
            <AcceptableResponseSchema>
             http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
responseschema/2006
            </AcceptableResponseSchema>
     </Request>
</Autodiscover>
```

2. Make the request.

```
$ curl -d @request.xml -u testuser@company.tld -v https://autodiscover.company.tld/
autodiscover/autodiscover.xml
Enter host password for user 'testuser@company.tld':
<?xml version="1.0" encoding="UTF-8"?>
<Autodiscover xmlns="http://schemas.microsoft.com/exchange/autodiscover/
responseschema/2006" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://
www.w3.org/2001/XMLSchema-instance">
<Response xmlns="http://schemas.microsoft.com/exchange/autodiscover/mobilesync/
responseschema/2006">
    <Culture>en:us</Culture>
    <User>
        <DisplayName>User1</DisplayName>
        <EMailAddress>testuser@company.tld</EMailAddress>
    </User>
    <Action>
        <Settings>
            <Server>
                <Type>MobileSync</Type>
                <Url>https://mobile.mail.us-east-1.awsapps.com/Microsoft-Server-
ActiveSync</Url>
                <Name>https://mobile.mail.us-east-1.awsapps.com/Microsoft-Server-
ActiveSync</Name>
            </Server>
        </Settings>
    </Action>
</Response>
```

If the response output is similar, your AutoDiscover endpoint is configured correctly.

Editing domain identity policies

Domain identity policies specify permissions for email actions (such as redirecting email messages). You can redirect email to any email address of your choosing. However, if your domain was added prior to October 13, 2016, you need to update the sending authorization policy manually to support that.

The update is the addition of a new action: `ses:*`. Domains added after October 13, 2016 have this action added by default.

> **Note**
> Exercise caution when editing other sections of the `ses` policy, as incorrect settings can have an adverse effect on Amazon WorkMail functionality.

To update the domain identity policy

1. Sign in to the AWS Management Console and open the Amazon SES console at https://console.aws.amazon.com/ses/home.
2. In the **Navigation** pane of the Amazon SES console, under **Identity Management**, choose **Domains**.
3. In the list of domains, select the domain to edit.

4. In the **Details** pane, expand **Identity Policies**, find the policy to edit, and then choose **Edit Policy**.
5. In the **Edit Policy** pane, under `"Action"`, add `ses:*,`.
6. Choose **Apply Policy**.

The updated actions of the policy should look like the following.

```
"Action": [
    "ses:*",
    "ses:SendBounce",
    "ses:SendRawEmail"
],
```

Authenticating email with SPF

The Sender Policy Framework (SPF) is an email validation standard designed to combat email spoofing. For information about configuring SPF for your Amazon WorkMail-enabled domain, see Authenticating Email with SPF in Amazon SES.

Configuring a custom MAIL FROM domain in Amazon WorkMail

By default, Amazon WorkMail uses a subdomain of `amazonses.com` as the `MAIL FROM` domain for your outgoing email. This can cause delivery failure if the DMARC policy on your domain is only set up for SPF. To resolve this, configure your own domain as the `MAIL FROM` domain. To learn how to set up your email domain as the `MAIL FROM` domain, see Setting up a custom MAIL FROM domain in the *Amazon Simple Email Service Developer Guide*.

For more information about custom `MAIL FROM` domains, see Amazon SES now supports custom MAIL FROM domains.

Working with users

You can create and remove users from Amazon WorkMail. In addition, you can reset their email passwords and wipe the data from their mobile devices.

Topics

Managing user accounts

Create users or enable existing users and edit user email addresses, create user email aliases, edit user details, and reset user passwords from Amazon WorkMail.

Topics

Creating users

When you create users, Amazon WorkMail creates mailboxes for them. Users can log in and access their mail from the Amazon WorkMail web application, mobile device, or Microsoft Outlook on macOS or PC.

To create a user

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users** to see a list of all users in the directory. This includes enabled, disabled, and system users as defined by the underlying directory.
5. To create a user, choose **Create User**.
6. On the **Add the details for your new user** screen, enter the user's first and last name, username, and display name and then choose **Next**.
7. On the **Set up email address and password** screen, enter the user's email address and password, and choose **Add user**.

Enabling existing users

When Amazon WorkMail is integrated with your corporate Active Directory or you already have users available in your Simple AD directory, you can enable these users in Amazon WorkMail.

To enable an existing directory user

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users** to see a list of all the users in the directory, including enabled, disabled, and system users.
5. From the list of disabled users, select the users to enable and choose **Enable user**.
6. In the **Enable user(s)** dialog box, review the primary email address and choose **Enable**.

Editing user email addresses

You can assign multiple email addresses to a single user and the default email address is used as the default sending address for outgoing email.

You can also add one or more email aliases, which can be used to send or receive email from a different address or domain. For more information, see Send as an alias.

To edit a user's email address

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users**, and then in the list of users, select the name of the user to edit.
5. On the **General** tab, choose **Edit**, **Add email address**, and then type the email address to add to this user.
6. To set the new email address as the default, choose **Set as default**.

Editing user details

You can edit a user's first and last name, email address, display name, address, phone number, and company details. You can also set a user's mailbox quota to as little as 1 MB or as much as 51,200 MB (50 GB). Users are notified when they reach 90 percent of their set mailbox quota.

Changing a user's mailbox quota does not affect pricing. For more information about pricing, see Amazon WorkMail Pricing.

> **Note**
> If you are integrating Amazon WorkMail with an AD Connector directory, you can't edit these details from the AWS Management Console. Instead, you must edit them using your Active Directory management tools. Limitations apply when your organization is in interoperability mode. For more information, see Limitations in interoperability mode (p. 45).

To edit a user's details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users** and select the name of the user to edit.
5. On the **General** tab, choose **Edit**, and then update any of the fields as appropriate. To update a user's mailbox quota, choose the **Quota** tab.
6. Choose **Save**.

Resetting user passwords

If a user forgets a password or is having trouble signing in to Amazon WorkMail, you can reset the password. If you are integrating Amazon WorkMail with an AD Connector directory, you have to reset the user password in Active Directory.

To reset a user password

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users**.
5. In the list of users, select the name of the user to edit and choose **Reset password**.
6. In the **Reset Password** dialog box, type the new password and choose **Reset**.

Troubleshooting Amazon WorkMail password policies

If resetting the password is unsuccessful, verify that the new password meets the password policy requirements.

The password policy requirements depend on which directory type is used by your Amazon WorkMail organization.

Amazon WorkMail directory and Simple AD directory password policy

By default, passwords for an Amazon WorkMail directory or Simple AD directory must be:

- Non-empty.
- At least eight characters.
- Less than 64 characters.
- Composed of Basic Latin or Latin-1 supplement characters.

Passwords must also contain characters from three out of five of the following groups:

- Uppercase characters
- Lowercase characters

- Numerical digits
- Special characters (for example, <, ~, or !)
- Latin-1 supplement characters (for example, é, ü, or ñ)

Amazon WorkMail directory password policies cannot be changed.

To change a Simple AD password policy, use the AD administration tools on an Amazon Elastic Compute Cloud (Amazon EC2) Windows instance of your Simple AD directory. For more information, see Installing the Active Directory administration tools in the *AWS Directory Service Administration Guide*.

AWS Managed Microsoft AD Directory password policy

For information about the default password policy for an AWS Managed Microsoft AD directory, see Manage Password Policies for AWS Managed Microsoft AD in the *AWS Directory Service Administration Guide*.

AD Connector password policy

AD Connector uses the password policy of the Active Directory domain that it is connected to.

Managing user mailboxes

You can disable and restore user mailboxes and enable push notifications. For information about managing mailbox permissions, see Working with mailbox permissions (p. 103).

Topics

Disabling user mailboxes

You can disable user mailboxes when they are no longer needed. Amazon WorkMail keeps mailboxes for 30 days before they're permanently removed.

To disable a user's mailbox

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, select **Users**, select the name of the user to disable, and choose **Disable User**.
5. In the **Disable user(s)** dialog box, choose **Disable**.

> **Note**
> To permanently delete a user and their data, use the `DeleteUser` API action for Amazon WorkMail. For more information, see DeleteUser in the *Amazon WorkMail API Reference*.

Restoring disabled mailboxes

Amazon WorkMail retains disabled mailboxes for 30 days before permanently removing them. To restore a mailbox, use the same steps as enabling an existing user.

> **Important**
> Mailboxes cannot be restored if the organization containing them has been deleted. To restore a user's disabled mailbox, the user must be still in the directory. If the user isn't in the directory or if you've re-created them, the mailbox cannot be restored because each mailbox is linked to a unique user ID.

To restore a disabled mailbox

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users** to see a list of enabled, disabled, and system users.
5. From the list of disabled users, select the users to enable and choose **Enable user**.
6. In the **Enable user(s)** dialog box, review the primary email address of the user and choose **Enable**.

Viewing email headers

The information in email headers can help you troubleshoot user email issues.

To view email headers in Amazon WorkMail

1. In the Amazon WorkMail web application, open the email message.
2. Choose **Message options** (the gear and envelope icon).

The email headers appear under **Internet Headers**.

Working with notifications

With the Amazon WorkMail Push Notifications API, you can receive push notifications about changes in your mailbox, including new email and calendar updates. You can register the URLs (or push notification responders) to receive notifications. With this feature, developers can create responsive applications for Amazon WorkMail users, as applications are quickly notified about changes from a user's mailbox.

For more information, see Notification subscriptions, mailbox events, and EWS in Exchange.

You can subscribe specific folders, such as Inbox or Calendar, or all folders for mailbox change events (including NewMail, Created, and Modified).

You can use client libraries such as the EWS Java API or the Managed EWS C# API to access this feature. A complete sample application of a push responder, developed using AWS Lambda and API Gateway (using the AWS Serverless framework), is available here. It uses the EWS Java API.

The following is a sample push subscription request.

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:t="http://
schemas.microsoft.com/exchange/services/2006/types">
   <soap:Body>
      <m:Subscribe xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
         <m:PushSubscriptionRequest>
            <t:FolderIds>
               <t:DistinguishedFolderId Id="inbox" />
            </t:FolderIds>
            <t:EventTypes>
               <t:EventType>NewMailEvent</t:EventType>
               <t:EventType>CopiedEvent</t:EventType>
               <t:EventType>CreatedEvent</t:EventType>
               <t:EventType>DeletedEvent</t:EventType>
               <t:EventType>ModifiedEvent</t:EventType>
               <t:EventType>MovedEvent</t:EventType>
            </t:EventTypes>
            <t:StatusFrequency>1</t:StatusFrequency>
            <t:URL>https://YOUR_PUSH_RESPONDER_URL</t:URL>
         </m:PushSubscriptionRequest>
      </m:Subscribe>
   </soap:Body>
</soap:Envelope>
```

The following is a successful subscription request result.

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/" xmlns:xsd="http://
www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <Header xmlns="http://schemas.xmlsoap.org/soap/envelope/">
      <ServerVersionInfo xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
 MajorVersion="14" MinorVersion="2" MajorBuildNumber="390" Version="Exchange2010_SP2"
 MinorBuildNumber="3" />
   </Header>
   <soap:Body>
      <m:SubscribeResponse xmlns:m="http://schemas.microsoft.com/exchange/services/2006/
messages" xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types">
         <m:ResponseMessages>
            <m:SubscribeResponseMessage ResponseClass="Success">
               <m:ResponseCode>NoError</m:ResponseCode>
               <m:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</m:SubscriptionId>
               <m:Watermark>AAAAAAA=</m:Watermark>
            </m:SubscribeResponseMessage>
         </m:ResponseMessages>
      </m:SubscribeResponse>
   </soap:Body>
</soap:Envelope>
```

Afterwards, notifications are sent to the URL specified in the subscription request. The following is a sample notification.

```
<soap:Envelope
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Header>
        <t:RequestServerVersion
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
 Version="Exchange2010_SP2">
        </t:RequestServerVersion>
    </soap:Header>
```

```
    <soap:Body>
        <m:SendNotification
            xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
            xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
            <m:ResponseMessages>
                <m:SendNotificationResponseMessage ResponseClass="Success">
                    <m:ResponseCode>NoError</m:ResponseCode>
                    <m:Notification>
                        <t:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</t:SubscriptionId>
                        <t:PreviousWatermark>ygwAAAAAAAA=</t:PreviousWatermark>
                        <t:MoreEvents>false</t:MoreEvents>
                        <t:ModifiedEvent>
                            <t:Watermark>ywwAAAAAAAA=</t:Watermark>
                            <t:TimeStamp>2018-02-02T15:15:14Z</t:TimeStamp>
                            <t:FolderId Id="AAB2L089bS1kNDgxOGYwOGE5OTQ0="></t:FolderId>
                            <t:ParentFolderId Id="AAB2L089bS1kNDgxOGYwOGE="></
t:ParentFolderId>
                        </t:ModifiedEvent>
                    </m:Notification>
                </m:SendNotificationResponseMessage>
            </m:ResponseMessages>
        </m:SendNotification>
    </soap:Body>
</soap:Envelope>
```

To acknowledge that the push notification responder has received the notification, it must reply with the following.

```
 <?xml version="1.0"?>
  <s:Envelope xmlns:s= "http://schemas.xmlsoap.org/soap/envelope/">
    <s:Body>
      <SendNotificationResult xmlns="http://schemas.microsoft.com/exchange/services/2006/
messages">
        <SubscriptionStatus>OK</SubscriptionStatus>
      </SendNotificationResult>
    </s:Body>
  </s:Envelope>
```

To unsubscribe from receiving push notifications, clients must send an unsubscribe response in the `SubscriptionStatus` field, similar to the following.

```
 <?xml version="1.0"?>
  <s:Envelope xmlns:s= "http://schemas.xmlsoap.org/soap/envelope/">
    <s:Body>
      <SendNotificationResult xmlns="http://schemas.microsoft.com/exchange/services/2006/
messages">
        <SubscriptionStatus>Unsubscribe</SubscriptionStatus>
      </SendNotificationResult>
    </s:Body>
  </s:Envelope>
```

To verify the health of your push notification responder, Amazon WorkMail sends a “heartbeat” (also called a `StatusEvent`). The frequency with which they are sent is determined by the `StatusFrequency` parameter provided in the initial subscription request. For example, if `StatusFrequency` equals 1, a `StatusEvent` is sent every 1 minute. This value can range between 1 and 1440 minutes. This `StatusEvent` looks like the following.

```
<?xml version="1.0 (http://www.w3.org/TR/REC-xml/)" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Header>
    <t:RequestServerVersion xmlns:t="http://schemas.microsoft.com/exchange/services/2006/
types" xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
 Version="Exchange2010_SP2"/>
</soap:Header>
<soap:Body>
    <m:SendNotification xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
 xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages">
    <m:ResponseMessages>
        <m:SendNotificationResponseMessage ResponseClass="Success">
            <m:ResponseCode>NoError</m:ResponseCode>
            <m:Notification>
                <t:SubscriptionId>hKJETtoAdi9PPW0tZDQ4MThmMDoVYB</t:SubscriptionId>
                <t:PreviousWatermark>AAAAAAAAAAA=</t:PreviousWatermark>
                <t:MoreEvents>false</t:MoreEvents>
                <t:StatusEvent>
                    <t:Watermark>AAAAAAAAAAA=</t:Watermark>
                </t:StatusEvent>
            </m:Notification>
        </m:SendNotificationResponseMessage>
    </m:ResponseMessages>
</m:SendNotification>
</soap:Body>
</soap:Envelope>
```

If a client push notification responder fails to respond (with the same OK status as before), the notification is retried for a maximum of `StatusFrequency` minutes. For example, if `StatusFrequency` equals 5, and the first notification fails, it is retried for a maximum of 5 minutes with an exponential backoff between each retry. If the notification is not delivered after the retry time has expired, the subscription becomes invalidated and no new notifications are delivered. You must create a new subscription to continue to receive notifications about mailbox events. Currently, you can subscribe for a maximum of three subscriptions per mailbox.

Managing mobile devices

Remotely wipe user mobile devices, remove devices from your organization, and view details for devices in your organization from Amazon WorkMail. For information about editing your organization's mobile device policy, see Editing your organization's mobile device policy (p. 60).

Topics

Remotely wiping mobile devices

You can remotely wipe user devices, but they must be online and connected to Amazon WorkMail. Allow 5 minutes for a wipe to start.

> **Warning**
> For most mobile devices, a remote wipe resets the device to factory defaults. All data, including personal files, can be removed when you perform this procedure.

To remotely wipe a user's mobile device

1. Sign in to the AWS Management Console and open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the region. From the navigation bar, select the region that meets your needs. For more information, see Regions and Endpoints in the *Amazon Web Services General Reference*.
3. On the **Organizations** page, in the list of organizations, select your organization's alias.
4. In the navigation pane, choose **Users**.
5. In the list of users, select the user with the device to view.
6. Choose the **Mobile** tab.
7. In the list of devices, select the device to wipe, and then choose **Wipe device**.
8. In the overview, check the status to see if the wipe is requested.
9. After the wipe operation finishes, you can remove the device from the list.

   Important
   To reinstate a device, make sure the device is removed from the list. Otherwise, the device will be wiped again.

Removing user devices from the devices list

If a user is no longer using a specific mobile device or the device is remote wiped, you can remove it from the list. When the user configures the device again, it shows up in the list.

To remove a user's mobile devices from the devices list

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users**, select the user with the device to view, and choose **Mobile**.
5. In the list of devices, select the device to remove and choose **Remove device**.

Viewing mobile device details

You can view the details of a user's mobile device.

Note
Some devices don't send all of their details to the server, so you may not see all available device details.

To view device details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, select the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Users**, select the user with device to view, and choose **Mobile**.
5. In the list of devices, select the device whose details you want to view. Device status codes are listed in the following table.

Status	Description
Provisioning required	A user or administrator has requested that the device be provisioned for use with Amazon WorkMail. Devices are also set to this status if the current policy for that device is modified in the Amazon WorkMail console.
Provisioning succeeded	The device has been successfully provisioned or wiped. In the case of provisioning, the device has enforced the given policy.
Wipe required	An administrator requested a wipe in the Amazon WorkMail console.
Wipe succeeded	The device has been successfully wiped.

Enabling signed or encrypted email

You can use S/MIME to enable users to send signed or encrypted email both inside and outside the organization.

> **Note**
> User certificates in the Global Address List (GAL) are supported only in a connected Active Directory setup.

To enable users to send signed or encrypted emails

1. Set up an Active Directory (AD) Connector. Setting up an AD Connector with your on-premises directory allows users to continue to use their existing corporate credentials.
2. Configure Certificate Autoenrollment to issue and store user certificates automatically in the Active Directory. Amazon WorkMail receives user certificates from the Active Directory and publishes them to the GAL. For more information, see Configure Certificate Autoenrollment.
3. Distribute the generated certificates to users by exporting the certificates from the server running Microsoft Exchange and mailing them.
4. Each user installs the certificate to their email program (such as Windows Outlook) and mobile devices.

Working with groups

Groups can be used as distribution lists in Amazon WorkMail for receiving emails for generic email addresses like sales@example.com or support@example.com. You can create multiple email aliases for a group.

You can also use groups as security groups to share a mailbox or calendar with a certain team. It can take up to 2 hours before newly added groups appear in your Microsoft Outlook offline address book.

Groups do not have their own mailboxes. For information about setting up group permissions, see Managing group permissions (p. 104).

Topics

Create a group

Create groups in the Amazon WorkMail console. You can also create aliases for your groups.

To create a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Groups** to see a list of enabled, disabled, and system groups.
5. To create a new group, choose **Create group**.
6. On the **Add group details** screen, enter the group name and email address, and then choose **Add group members**.
7. On the **Add members to group** screen, for **Search**, enter the user's first name, last name, user name, or group name and press **Enter**.
8. In the list of directory users and groups, select the user or groups to add as a member.
9. Choose the right arrow button to add them to the list of selected users/groups, and then choose **Finish**.

To create a group alias

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Groups** to see a list of enabled, disabled, and system groups.

5. For **Group name**, select the name of the group for which to create the alias.
6. On the **General** tab, choose **Edit**.
7. For **Email aliases**, choose **Add email address**.
8. Enter the alias to create for the group.
9. Choose **Save**.

Enable an existing group

When Amazon WorkMail is integrated with your corporate Active Directory or you already have groups available in your simple Active Directory, you can use these groups as security groups or distribution lists in Amazon WorkMail.

To enable an existing directory group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Groups** to see a list of enabled, disabled, and system groups.
5. From the list of disabled groups, select the groups to enable and choose **Enable Group**.
6. In the **Enable group(s)** dialog box, review the primary email address and choose **Enable**.

Add users to a group

After you create and enable an Amazon WorkMail group, use the Amazon WorkMail console to add users to that group.

> **Note**
> If Amazon WorkMail is integrated with a connected Active Directory service or Microsoft Active Directory, you can manage your group members using Active Directory. This can take a longer time to propagate to Amazon WorkMail.

You can add up to 100 users at a time by using the following procedure.

To add users to a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. Choose **Groups**.
5. Select the name of the group.
6. On the **Group details** page, choose **Members**.
7. Choose **Edit**.
8. Under **Users and groups**, search for the users to add to the group, and select them.
9. Choose **>>**. The users appear under **Group members**.
10. Choose **Save**.

Your changes might take a few minutes to propagate.

Remove users from a group

Use the Amazon WorkMail console to remove users from a group.

> **Note**
> If Amazon WorkMail is integrated with a connected Active Directory or Microsoft Active Directory, you can manage your group members using the Active Directory. This can take a longer time to propagate to Amazon WorkMail.

To remove users from a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. Choose **Groups**.
5. Select the name of the group.
6. On the **Group details** page, choose **Members**.
7. Choose **Edit**.
8. Under **Group members**, search for the users to add to the group, and select them.
9. Choose **<<**. The users no longer appear under **Group members**.
10. Choose **Save**.

Your changes might take a few minutes to propagate.

Disable a group

When you no longer need a group, you can disable it.

To disable a group

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If you require a different AWS Region, change it from the navigation bar. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Groups**.
5. In the list of groups, select the group to disable, and then choose **Disable group**.
6. In the **Disable group(s)** dialog box, choose **Disable**.

> **Note**
> To permanently delete a group, use the `DeleteGroup` API action for Amazon WorkMail. For more information, see DeleteGroup in the *Amazon WorkMail API Reference*.

Working with mailbox permissions

You can use mailbox permissions in Amazon WorkMail to grant users or groups the right to work in other users' mailboxes. Mailbox permissions apply to an entire mailbox, enabling multiple users to access the same mailbox without sharing the credentials for that mailbox. Users with mailbox permissions can read and modify mailbox data and send email from the shared mailbox.

The following list shows the permissions that you can grant:

- **Send On Behalf**: Enables a user or group to send email on behalf of another user. The mailbox owner appears in the **From:** header, and the sender appears in the **Sender:** header.
- **Send As**: Enables a user or group to send email as the mailbox owner, without showing the actual sender of the message. The mailbox owner appears in both the **From:** and **Sender:** headers.
- **Full Access**: Enables full read and write access to the mailbox, including permissions to modify folder-level permissions.

> **Note**
> Granting mailbox permissions to a group extends those permissions to all the members of that group, including members of nested groups.

When you grant mailbox permissions, the Amazon WorkMail AutoDiscover service automatically updates access to those mailboxes for the users or groups you added.

For the Microsoft Outlook client in Windows, users with full access permissions can automatically access the shared mailboxes. Allow up to 60 minutes for the changes to propagate, or restart Microsoft Outlook.

For the Amazon WorkMail web application and other email clients, users with full access permissions can manually open the shared mailboxes. Opened mailboxes stay open, even between sessions, unless the user closes them.

Topics

Mailbox and folder permissions

Mailbox permissions apply to all folders in a mailbox. These permissions can only be enabled by the AWS account holder or the IAM user authorized to call the Amazon WorkMail management API. To apply the permissions to mailboxes or groups as a whole, log in to the AWS Management Console or use the Amazon WorkMail API. You can manage up to 100 mailbox and group permissions from the console. To manage more permissions, use the Amazon WorkMail API.

Folder permissions apply only to a single folder. These permissions can be set by end users, either by using an email client or by using the Amazon WorkMail web application.

Enabling mailbox permissions

You can enable other users to access a mailbox using the Amazon WorkMail web application.

To enable mailbox permissions

1. In the Amazon WorkMail application, on the **User details** page under **Permissions**, choose **Add or remove**.
2. Under **Users and groups**, select the user or group to share your inbox and choose **>>** to add them to the **Permissions** list. Choose **Save**.
3. On the **Permissions** tab, select the level of permissions to grant and choose **Save**.

Updated permissions can take up to five minutes to propagate.

Editing mailbox permissions

You can edit existing mailbox permissions for Amazon WorkMail.

To edit mailbox permissions

1. In the Amazon WorkMail application, on the **User details** page under **Permissions**, choose **Edit**.
2. Select the permissions to change and choose **Save**.

Updated permissions can take up to five minutes to propagate.

Removing mailbox permissions

You can remove existing mailbox permissions for Amazon WorkMail.

To remove mailbox permissions

1. In the Amazon WorkMail application, on the **User details** page under **Permissions**, choose **Add or remove**.
2. Under **Users and groups**, select the user or group and remove them from the **Permissions** list.
3. Choose **Save**.

Updated permissions can take up to five minutes to propagate.

Managing group permissions

You can add or remove group permissions for Amazon WorkMail.

> **Note**
> **Full Access** permissions are not available for groups, because groups do not have a mailbox to access.

To manage group permissions

1. In the Amazon WorkMail application, on the **Groups** page under **WorkMail groups**, select the group to manage.

2. Under **Permissions**, choose **Add or remove**.
3. Under **Users and groups**, select the group to add or remove. Add or remove them from the **Permissions** list and choose **Save**.

   Note
   If you added a group to the **Permissions** list, select the level of permissions to grant under the **Permissions** tab and choose **Save**.

Updated permissions can take up to five minutes to propagate.

Exporting mailbox content

Use the StartMailboxExportJob API action in the *Amazon WorkMail API Reference* to export Amazon WorkMail mailbox content to an Amazon Simple Storage Service (Amazon S3) bucket. This action exports all email messages and calendar items from the specified mailbox to a `.zip` file in the Amazon S3 bucket, in MIME format. Other items, such as contacts and tasks, are not exported.

The time it takes for the mailbox export job to finish is dependent on the size and number of items in the mailbox. Because the mailbox export job takes place over a period of time, it does not represent a snapshot of the mailbox content at a single point in time. To see the status of an export job, use the DescribeMailboxExportJob or ListMailboxExportJobs API actions in the *Amazon WorkMail API Reference*.

When a mailbox export job is completed, the `.zip` file in the Amazon S3 bucket is encrypted using the symmetric AWS Key Management Service (AWS KMS) customer master key (CMK) that you provide. Because AWS KMS encryption is integrated with Amazon S3, the decrypted data is visible to the user who downloads it, as long as the user has access to the AWS KMS CMK.

Prerequisites

The following are prerequisites for exporting mailbox content:

- The ability to program.
- An Amazon WorkMail administrator account.
- An Amazon S3 bucket that does not allow public access. For more information, see Using Amazon S3 block public access in the *Amazon Simple Storage Service Developer Guide* and the *Amazon Simple Storage Service Getting Started Guide*.
- A symmetric AWS KMS CMK. For more information, see Getting started in the *AWS Key Management Service Developer Guide*.
- An AWS Identity and Access Management (IAM) role with a policy that grants permission to write to the Amazon S3 bucket and encrypt the sent files with the AWS KMS CMK. For more information, see How Amazon WorkMail works with IAM (p. 17).

IAM policy examples and role creation

The following example shows an IAM policy that grants permission to write to the Amazon S3 bucket and encrypt the sent files with the AWS KMS CMK. To use this example policy in the following Example: Exporting mailbox content (p. 108) procedure, save the policy as a JSON file with file name `mailbox-export-policy.json`.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:PutObject",
```

```
                "s3:GetBucketPolicyStatus"
            ],
            "Resource": [
                "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET",
                "arn:aws:s3:::AWSDOC-EXAMPLE-BUCKET/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
            ],
            "Resource": [
                "arn:aws:kms:us-east-1:111122223333:key/KEY-ID"
            ],
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "s3.us-east-1.amazonaws.com"
                },
                "StringLike": {
                    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::AWSDOC-EXAMPLE-
BUCKET/S3-PREFIX*"
                }
            }
        }
    ]
}
```

The following example shows an IAM trust policy that is attached to the IAM role you create. To use this example policy in the following Example: Exporting mailbox content (p. 108) procedure, save the policy as a JSON file with file name `mailbox-export-trust-policy.json`.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "export.workmail.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "111122223333"
        }
      }
    }
  ]
}
```

You can use the AWS CLI to create the IAM role in your account by running the following commands.

```
aws iam create-role --role-name WorkmailMailboxExportRole --assume-role-policy-
document file://mailbox-export-trust-policy.json --region us-east-1
```

```
aws iam put-role-policy --role-name WorkmailMailboxExportRole --policy-name MailboxExport
 --policy-document file://mailbox-export-policy.json
```

For more information about the AWS CLI, see the AWS Command Line Interface User Guide.

Example: Exporting mailbox content

After you create the IAM role and policies in the preceding section, complete the following steps to export your mailbox content. You must have your Amazon WorkMail organization ID and user ID (entity ID), which you can access in the Amazon WorkMail console or by using the Amazon WorkMail API.

Example: To export mailbox content

1. Use the AWS CLI to start the mailbox export job.

```
aws workmail start-mailbox-export-job --organization-
id m-a123b4c5de678fg9h0ij1k2lm234no56 --entity-
id S-1-1-11-1111111111-2222222222-3333333333-3333 --kms-key-
arn arn:aws:kms:us-east-1:111122223333:key/KEY-ID --role-arn
 arn:aws:iam::111122223333:role/WorkmailMailboxExportRole --s3-bucket-name AWSDOC-
EXAMPLE-BUCKET --s3-prefix S3-PREFIX
```

2. Use the AWS CLI to monitor the state of the mailbox export jobs for your Amazon WorkMail organization.

```
aws workmail list-mailbox-export-jobs --organization-id m-
a123b4c5de678fg9h0ij1k2lm234no56
```

   Alternatively, use the job ID generated by the **`start-mailbox-export-job`** command to monitor the state of that mailbox export job only.

```
aws workmail describe-mailbox-export-job --organization-id m-
a123b4c5de678fg9h0ij1k2lm234no56 --job-id JOB-ID
```

When the mailbox export job state is **COMPLETED**, the exported mailbox items are available in a `.zip` file in the specified Amazon S3 bucket.

Considerations

The following considerations apply when exporting mailbox jobs for Amazon WorkMail:

- You can run up to 10 concurrent mailbox export jobs for a given Amazon WorkMail organization.
- You can run a mailbox export job for a given mailbox as often as once every 24 hours.
- The following resources must all be in the same AWS Region:
  - Amazon WorkMail organization
  - AWS KMS CMK
  - Amazon S3 bucket

Working with resources

Amazon WorkMail can help your users reserve resources, such as meeting rooms or equipment (projectors, phones, cars, and so on). To book a resource, the user adds the resource to the meeting invite.

Topics

Creating a resource

You can add a new resource to your organization, and allow it to be reserved.

To add a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Resources** and **Add resource**.
5. On the **Add resource details** page, enter values for the **Resource name**, **Description**, **Resource type**, and **Email address** fields.
6. Choose **Create**.

Editing a resource

You can edit a resource's general details (name, description, type, and email address), booking options, and delegates.

To edit general resource details

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Resources**, and select the resource to edit.
5. On the **General** tab, update the details to change: **Resource name**, **Description**, **Resource Type**, or **Email address**.
6. Choose **Save**.

You can configure a resource to accept or decline booking requests automatically.

To enable or disable automatic processing of booking requests

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Resources**, and then select the resource to edit.
5. On the **Booking Options** tab, choose **Edit**.
6. To accept all resource requests automatically, select **Automatically accept all resource requests**.
7. To decline recurring resource requests automatically, select **Automatically decline recurring resource requests**.
8. To decline conflicting resource requests automatically, select **Automatically decline conflicting resource requests**.
9. Choose **Save**.

You can add a delegate to control booking requests for a resource. Resource delegates automatically receive copies of all booking requests and have full access to the resource calendar. In addition, they must accept all booking requests for a resource.

To add a resource delegate

> **Note**
> Before you proceed, follow the process above to clear the **Automatically accept all resource requests** option.

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Resources**, and select the name of the resource to edit.
5. On the **Delegates** tab, choose **Edit**.
6. Select the users or groups to add as delegates, and then use the right arrow to add them to the delegate list.
7. Choose **Save**.

Removing a resource

When you no longer need a resource, you can remove it.

To remove a resource

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. If necessary, change the AWS Region. From the navigation bar, choose the Region that meets your needs. For more information, see Regions and endpoints in the *Amazon Web Services General Reference*.
3. For **Organizations**, choose the name of your organization.
4. In the navigation pane, choose **Resources**.
5. In the list of resources, select the resource to remove, and choose **Remove**.
6. In the **Remove resource(s)** dialog box, choose **Remove**.

Using email journaling with Amazon WorkMail

You can set up journaling to record your email communication, using integrated third-party archiving and eDiscovery tools. This ensures that email storage compliance regulations for privacy protection, data storage, and information protection are met.

Using journaling

Amazon WorkMail journals all email messages that are sent to any user in the specified organization, as well as all email messages sent by users in that organization. A copy of all email messages is sent to an address specified by the system administrator, in a format called `journal record`. This format is compatible with Microsoft email programs. There is no additional charge for email journaling.

Two email addresses are used for email journaling—a journaling email address and a report email address. The journaling email address is the address of a dedicated mailbox or third-party device that is integrated with your account, where journal reports are sent. The report email address is the address of your system administrator, where notifications of failed journal reports are sent.

All journal records are sent from an email address that is automatically added to your domain and looks like the following.

```
amazonjournaling@yourorganization.awsapps.com
```

There is no mailbox associated with this address, and you will not be able to create one using this name or address.

Note
Do not delete the following domain record from the Amazon Simple Email Service (Amazon SES) console, or email journaling stops functioning.

```
yourorganization.awsapps.com
```

Every incoming or outgoing email message generates one journal record, regardless of the number of recipients or user groups. Email that fails to generate a journal record generates an error notification, which is sent to the report email address.

To enable email journaling

1. Open the Amazon WorkMail console at https://console.aws.amazon.com/workmail/.
2. For **Organizations**, choose the name of your organization.
3. On the **Organization settings** screen, choose **Journaling Settings**, **Edit**, **On**.
4. For **Journaling email address**, enter the email address provided by your email journaling provider.

   Note
   We recommend using a dedicated journaling provider.
5. For **Report email address**, enter the email administrator's address.
6. Choose **Save**. The changes are applied immediately.

Document history

The following table describes important changes in each release of the *Amazon WorkMail Administrator Guide*. For notification about updates to this documentation, you can subscribe to an RSS feed.

update-history-change	update-history-description	update-history-date
Console changes for creating an organization (p. 112)	The Amazon WorkMail console experience for creating an organization is updated. For more information, see Creating an organization in the *Amazon WorkMail Administrator Guide*.	October 23, 2020
Exporting mailbox content (p. 112)	Use the `StartMailboxExportJob` API action to export Amazon WorkMail mailbox content to an Amazon Simple Storage Service (Amazon S3) bucket. For more information, see Exporting mailbox content in the *Amazon WorkMail Administrator Guide*.	September 22, 2020
Mailbox retention policies (p. 112)	Set mailbox retention policies for your Amazon WorkMail organization that automatically delete email messages after a time period that you choose. For more information, see Setting mailbox retention policies in the *Amazon WorkMail Administrator Guide*.	May 28, 2020
Synchronous and asynchronous Run Lambda actions (p. 112)	Choose synchronous or asynchronous configurations for **Run Lambda** actions in Amazon WorkMail email flow rules. For more information, see Configuring AWS Lambda for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	May 11, 2020
Working with access control rules (p. 112)	Access control rules allow Amazon WorkMail administrators to control how their organization's mailboxes are accessed. For more information, see Working with access control rules in the *Amazon WorkMail Administrator Guide*.	February 12, 2020
Tagging an organization (p. 112)	Tag an Amazon WorkMail organization to differentiate	January 23, 2020

	between organizations in the AWS Billing and Cost Management console, or to control access to organization resources. For more information, see Tagging an organization in the *Amazon WorkMail Administrator Guide*.	
Enforce DMARC policies on incoming email (p. 112)	For more information, see Enforcing DMARC policies on incoming email in the *Amazon WorkMail Administrator Guide*.	October 17, 2019
Retrieving message content with Lambda (p. 112)	Use the Amazon WorkMail Message Flow API with AWS Lambda to retrieve message content. For more information, see Retrieving message content with Lambda in the *Amazon WorkMail Administrator Guide*.	September 12, 2019
Logging Amazon WorkMail email events (p. 112)	Enable email event logging in the Amazon WorkMail console to track email messages for your organization. For more information, see Tracking messages in the *Amazon WorkMail Administrator Guide*.	May 13, 2019
Route 53 DNS record insertion (p. 112)	When setting up a domain that is managed in a Route 53 public hosted zone, Amazon WorkMail automatically inserts the DNS records for you. For more information, see Adding a domain in the *Amazon WorkMail Administrator Guide*.	February 13, 2019
Configuring Lambda for inbound email rule actions (p. 112)	Amazon WorkMail supports configuring Lambda functions to use with inbound email flow rules. For more information, see Managing email flows in the *Amazon WorkMail Administrator Guide*.	January 24, 2019
Configuring Lambda for Amazon WorkMail (p. 112)	Amazon WorkMail supports configuring Lambda functions to use with outbound email flow rules. For more information, see Configuring Lambda for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	November 19, 2018

SMTP routing (p. 112)	Amazon WorkMail supports configuring SMTP gateways to use with outbound email flow rules. For more information, see Configuring SMTP gateways in the *Amazon WorkMail Administrator Guide*.	November 1, 2018
Debugging tools for custom domains (p. 112)	Amazon WorkMail has added debugging tools for custom domains. For more information, see Adding a domain in the *Amazon WorkMail Administrator Guide*.	October 15, 2018
Support for Outlook 2019 (p. 112)	Amazon WorkMail supports Outlook 2019 for Windows and macOS. For more information, see Amazon WorkMail system requirements in the *Amazon WorkMail Administrator Guide*.	October 1, 2018
Various updates (p. 112)	Various updates to topic layout and organization.	July 12, 2018
Mailbox permissions (p. 112)	You can use mailbox permissions in Amazon WorkMail to grant users or groups the right to work in other users' mailboxes. For more information, see Working with mailbox permissions in the *Amazon WorkMail Administrator Guide*.	April 9, 2018
Support for AWS CloudTrail (p. 112)	Amazon WorkMail is integrated with AWS CloudTrail. For more information, see Logging Amazon WorkMail API calls with AWS CloudTrail in the *Amazon WorkMail Administrator Guide*.	December 12, 2017
Support for email flows (p. 112)	You can set up email flow rules for handling incoming email based on a sender's email address or domain. For more information, see Managing email flows in the *Amazon WorkMail Administrator Guide*.	July 5, 2017
Updates to Quick Setup (p. 112)	Quick Setup now creates an Amazon WorkMail directory for you. For more information, see Set up Amazon WorkMail with Quick Setup in the *Amazon WorkMail Administrator Guide*.	May 10, 2017

Support for a wider range of email clients (p. 112)	You can now use Amazon WorkMail with Microsoft Outlook 2016 for Mac and IMAP email clients. For more information, see System requirements for Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	January 9, 2017
Support for SMTP journaling (p. 112)	You can set up journaling to record your email communication. For more information, see Using email journaling with Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	November 25, 2016
Support for email redirection to external email addresses (p. 112)	You can set up email redirection rules by updating the Amazon SES identity policy for your domain. For more information, see Edit domain identity policies in the *Amazon WorkMail Administrator Guide*.	October 26, 2016
Support for interoperability (p. 112)	You can enable interoperability between Amazon WorkMail and Microsoft Exchange. For more information, see Interoperability between Amazon WorkMail and Microsoft Exchange in the *Amazon WorkMail Administrator Guide*.	October 25, 2016
General availability (p. 112)	The general availability release of Amazon WorkMail.	January 4, 2016
Support for reserving resources (p. 112)	Support for reserving resources, such as meeting rooms and equipment. For more information, see Working with resources in the *Amazon WorkMail Administrator Guide*.	October 19, 2015
Support for the email migration tool (p. 112)	Support for the email migration tool. For more information, see Migrating to Amazon WorkMail in the *Amazon WorkMail Administrator Guide*.	August 16, 2015
Preview release of Amazon WorkMail (p. 112)	The preview release of Amazon WorkMail.	January 28, 2015

AWS glossary

For the latest AWS terminology, see the AWS glossary in the *AWS General Reference*.