



Entwicklerhandbuch

# AWS IoT 1-Click



# AWS IoT 1-Click: Entwicklerhandbuch

Copyright © Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Die Marken und Handelsmarken von Amazon dürfen nicht in einer Weise in Verbindung mit nicht von Amazon stammenden Produkten oder Services verwendet werden, die geeignet ist, Kunden irrezuführen oder Amazon in irgendeiner Weise herabzusetzen oder zu diskreditieren. Alle anderen Marken, die nicht Eigentum von Amazon sind, sind Eigentum ihrer jeweiligen Inhaber, die mit Amazon verbunden oder nicht verbunden oder von Amazon gesponsert oder nicht gesponsert sein können.

---

# Table of Contents

Was ist AWS IoT 1-Click? .....	1
AWS IoT 1-Click Komponenten .....	1
Funktionsweise von AWS IoT 1-Click .....	4
AWS IoT 1-Click Geräte .....	4
Geräte beantragen .....	5
Projekte, Vorlagen und Platzierungen .....	5
Erste Schritte mit der AWS IoT 1-Click Konsole .....	9
Geräte beantragen .....	9
Erstellen eines Projekts .....	10
Beispiel: Projekt „Meeting Room Satisfaction“ (Korrekte Zuteilung von Meeting-Räumen) .....	10
AWS IoT 1-Click App .....	14
AWS IoT 1-Click-Programmiermodell .....	15
AWS IoT 1-Click Callback-Ereignisse .....	17
AWS IoT 1-Click ck-Ereignisse .....	17
AWS IoT 1-Click Health Ereignisse .....	18
Gerätemethoden .....	19
Überwachung mit CloudWatch Metriken .....	20
Protokollieren von AWS IoT 1-Click API-Aufrufen mit AWS CloudTrail .....	22
AWS IoT 1-Click Informationen in CloudTrail .....	22
Beispiel: Einträge AWS IoT 1-Click Protokolldateien .....	24
Integration von AWS CloudFormation .....	27
Authentifizierung und Zugriffskontrolle für AWS IoT 1-Click .....	28
AWS IoT 1-Click-Ressourcen und -Operationen .....	28
Verwenden von identitätsbasierten Richtlinien (IAM-Richtlinien) für AWS IoT 1-Click .....	29
AWS-verwaltete (vordefinierte) Richtlinien für &AWS IoT 1-Click; .....	29
Markierungen Ihrer AWS IoT 1-Click Ressourcen .....	33
Grundlagen zu Tags .....	33
Tag-Einschränkungen .....	34
Benutzerhandbuch AWS IoT Enterprise Button .....	35
AWS IoT 1-Click mit der AWS CLI verwenden .....	38
AWS IoT 1-Click Anhang .....	57
Unterstützte AWS IoT 1-Click Geräte .....	57
AWS IoT 1-Click Beschränkungen .....	60
Dokumentverlauf .....	61

---

AWS-Glossar .....	62
.....	lxiii

# Was ist AWS IoT 1-Click?

AWS IoT 1-Click erleichtert es Enterprise-Kunden, einfache IoT Geräte in ihre Arbeitsabläufe zu integrieren, ohne dass sie Geräte herstellen, Firmware schreiben oder sie für die sichere Konnektivität konfigurieren müssen. Unsere Fertigungspartner stellen Geräte her, die unmittelbar eine sichere Verbindung zum AWS IoT herstellen können. Diese Geräte können [AWS Lambda](#)-Funktionen, die in Sprachen wie Java, Python und C # geschrieben sind. Die Lambda Funktionen können eigene Geschäftslogik implementieren oder Aktionen in der AWS Cloud oder lokal auslösen.

AWS IoT 1-Click soll zur Vereinfachung des Internets of Things für Kunden dienen, indem es so viele Details wie möglich über die Gerätehardware und -firmware abstrahiert. Dadurch ist es möglich, in der Cloud gehostete AWS IoT 1-Click Geräte wie Software-Komponenten anzuzeigen, die in der AWS Cloud gehostet sind. Wie bei allen anderen Software-Komponenten sind diese Geräte konform zu wohldefinierten Schnittstellen. AWS IoT 1-Click verfügt über Schnittstellen, die pro Gerätetyp definiert sind. Sie können diese Schnittstellen verwenden, um Ihre Anwendungen zu erstellen und darauf zu stützen.

Mit AWS IoT 1-Click können Sie Geräte nach Funktion, Standort oder anderen Kriterien gruppieren. Diese logische Gruppe von Geräten wird als „Projekt“, in AWS IoT 1-Click. Sie können Projekte nutzen, um Gerätegruppen Lambda Funktionen für gewünschte Aktionen zuzuordnen.

Projekte enthalten Vorlagen, die angeben, welche Art von Geräten verwendet wird, welche Lambda Funktionen sie aufrufen, und welche optionalen Attribute für diese Geräte definiert sind, wie z. B. kontextbezogene Daten für Standort oder Funktion.

Sobald das Projekt erstellt wurde und Vorlagen definiert wurden, können Sie dem Projekt Platzierungen hinzufügen – die jeweils der Vorlage folgen und die tatsächlichen Geräte anhand ihrer Seriennummern und Attributwerte (Schlüssel/Wert-Paare) angeben, die sinnvoll für den spezifischen Standort oder die Funktion für diese spezielle Platzierung sind.

## AWS IoT 1-Click Komponenten

### Antrag

Bezieht sich auf den Prozess der Zuordnung eines AWS IoT 1-Click Geräts mit einem AWS-Konto über die AWS IoT 1-Click-Konsole, die AWS IoT 1-Click-App oder die AWS IoT 1-Click-API.

## Beantragungscodes

Ein Wert, der verwendet wird, um eine Reihe von AT&T LTE-M-Buttons gleichzeitig zu beantragen (also in der Masse). Sie können auch Geräte-IDs verwenden, um Geräte zu beantragen. Siehe den Eintrag Geräte-ID.

## Gerät

Ein physisches Gerät, z. B. die AWS IoT Enterprise Button oder die AT&T LTE-M-Taste.

## Geräteattribute

Standardmäßige oder benutzerdefinierte Daten, die einem bestimmten Gerät in Form von Schlüssel-Wert-Paaren zugeordnet sind. Standardattribute werden aus der Platzierung abgeleitet. Siehe den Eintrag Platzierung.

## Geräte-ID

Alle Geräte haben eine Geräte-ID, z. B. eine Geräteseriennummer (Device Serial Number, DSN). Eine Geräte-ID kann verwendet werden, um ein AWS IoT 1-Click Gerät bei AWS IoT 1-Click zu registrieren. Ein Antragscode ist nicht identisch mit einer Geräte-ID. Siehe den Eintrag Antragscode.

## Placement

Eine Gruppe von einer oder mehreren Vorlagen, die Geräte darstellen (z. B. ein Raum mit zwei auf Vorlagen beruhenden Buttons). Um eine Platzierung aufzufüllen, verwenden Sie die AWS IoT 1-Click Konsole oder die AWS IoT 1-Click-App, um auf Vorlagen beruhende Geräte auszuwählen.

## Placement-Name

Der Name der Platzierung, der häufig eine geografische Position oder eine Objekt-ID enthält (z. B. Room 217, North Dumpster oder Container 314).

## Projekt

Eine benannte Gruppe von null oder mehr Platzierungen (mit auf Vorlagen beruhenden Geräten).

## Project name

Ein sprechender Name für eine Gruppe von Platzierungen (z. B. „Besprechungsraum Kundenzufriedenheit“ oder „Charta Container-Abholung“).

## Vorlage

Wird verwendet, um ein Standardverhalten und Standardattribute für eine Gruppe von Geräten bereitzustellen. Ein physisches Gerät nutzt eine bestimmte Vorlage, um die Eigenschaften dieser

Vorlage zu erben: ihre Lambda Funktion und Standardgeräteattribute. Eine Vorlage definiert das Verhalten und Standardattribute für eine Klasse von Geräten in einer Platzierung. Ein Projekt kann mehr als eine Vorlage enthalten.

## Zurücknehmen

Der Prozess der Trennung eines AWS IoT 1-Click Geräts von einem AWS-Konto. Beispielsweise sollte eine Person, die ein AWS IoT 1-Click Gerät leihen möchte, zuerst die Verknüpfung des Geräts vom AWS-Konto trennen, damit der neue Benutzer das Gerät mit seinem eigenen AWS-Konto verknüpfen kann.

# Funktionsweise von AWS IoT 1-Click

AWS IoT 1-Click Workflow:

1. Wählen Sie ein Gerät aus einer Reihe unterstützter Geräte aus.
2. Verknüpfen Sie AWS Lambda Funktionen -Geräten, um Aktionen auszulösen. Sie können eine Ihrer eigenen Lambda Funktionen oder eine der vordefinierten Funktionen des Service verwenden.
3. Stellen Sie Ihre Geräte physisch bereit und verwenden Sie die AWS IoT 1-Click Konsole, die AWS IoT 1-Click App oder die AWS IoT 1-Click-API, um sie zu aktivieren.
4. Erhalten Sie Informationen über Gerätestatus und -nutzung an, indem Sie vorgefertigte AWS IoT 1-Click Berichte verwenden oder eigene erstellen.

## AWS IoT 1-Click Geräte

Von AWS IoT 1-Click-unterstützte Geräte:

- Sind gebrauchsfertig. Die Kunden müssen nicht entwickeln oder herstellen.
- Kann AWS IoT 1-Click Konten hinzugefügt werden, indem die [Funktion](#).
- Werden bereits beim Hersteller mit Zertifikaten ausgerüstet und so konfiguriert, dass sie eine sichere Verbindung zum AWS IoT herstellen. Sie müssen sich also nicht um die Installation von Zertifikaten für AWS IoT 1-Click-Geräte bemühen.
- Jedes AWS IoT 1-Click Gerät sendet Ereignisse in einem Standardformat aus, das von AWS IoT 1-Click definiert ist. Beispielsweise können alle AWS IoT 1-Click Geräte `buttonType` haben das gleiche Ereignisformat, unabhängig vom Hersteller.
- Haben einen Gerätetyp und einen Produkttyp. Der Gerätetyp gibt das Format der vom Gerät ausgegebenen Ereignisse und die unterstützten Gerätemethoden an. Weitere Informationen finden Sie unter [AWS IoT 1-Click-Programmiermodell](#). Der Produkttyp macht Angaben zum Hersteller und zur Marke. Wenn der Gerätetyp beispielsweise `button` und der Produkttyp `AT&T LTE-M Button` sein.

### Important

Von AWS IoT 1-Click-unterstützte Geräte werden im Werk konfiguriert, um eine Verbindung mit einem bestimmten [AWS-Region](#). Diese werden als Geräte-Regionen bezeichnet. Diese



Zuordnung eines Geräts zu einer Geräte-Region erforderlich, um sicherzustellen, dass Geräte eine sichere Verbindung zum AWS IoT herstellen, ohne dass zusätzliche Benutzereingaben erforderlich sind. Aus diesem Grund kann die Geräte-Region nicht geändert werden. Von AWS IoT 1-Click Geräten ausgegebene Ereignisse werden immer durch die vorkonfigurierte Geräte-Region geleitet, sodass Sie auf Geräte-bezogene Amazon CloudWatch Logs und AWS CloudTrail -Metriken in derselben AWS-Region zugreifen können. In der Geräte-Region werden auch aktivierte Geräte abgerechnet. Platzierungs-, Vorlagen- und Projektdaten werden in der AWS-Region gespeichert, die Ihrem Konto zugeordnet ist. Diese Region kann sich von der Geräte-Region unterscheiden.

Weitere Informationen zu von AWS IoT 1-Click unterstützten Geräten, einschließlich zum Kauf und zum Kauf von [Antrags](#)ie, siehe [AWS IoT 1-Click Anhang](#).

## Geräte beantragen

Wenn AWS IoT 1-Click Geräte das Werk verlassen, sind sie keinem AWS S-Kundenkonto zugeordnet. Kunden müssen einen Beantragungsprozess durchlaufen, um die Geräte in ihren Konten zu verwenden. Es gibt zwei Möglichkeiten, Geräte zu beantragen:

- Verwenden eines Einlöscodes: Wenn Sie einen Einlöscodes (im Format C-xxxxx) Sie können ihn in die AWS IoT 1-Click-Konsole oder die AWS 1-Click-App eingeben, um Geräte zu beantragen, die zu einem bestimmten Auftrag gehören. Nicht alle -Geräte, einschließlich der AWS IoT Enterprise Button, können unter Verwendung eines Antragscodes beantragt werden.
- Verwenden einer Geräte-ID: Sie können die Geräte-ID (die Geräteseriennummer, auch DSN genannt) verwenden, um Geräte über die AWS IoT 1-Click Konsole oder die AWS IoT 1-Click-App zu beantragen. Alle AWS IoT 1-Click Geräte können unter Verwendung einer Geräte-ID beantragt werden.

Weitere Informationen zum Beantragen von Geräten finden Sie unter [AWS IoT 1-Click Anhang](#) und [Geräte beantragen](#).

## Projekte, Vorlagen und Platzierungen

Geräte können nach Funktion, Standort oder anderen Kriterien gruppiert werden. Diese logische Gruppe von Geräten wird als „Projekt“ bezeichnet. Sie können Projekte nutzen, um Gerätegruppen Lambda Funktionen zuzuordnen.

Projekte enthalten Vorlagen, die angeben, welche Art von Geräten verwendet wird, welche Lambda Funktionen sie aufrufen, und welche Attributnamen verwendet werden, um kontextbezogene Daten zu speichern, wie z. B. Standort oder Funktion.

Nachdem das Projekt erstellt und Vorlagen definiert wurde, können Sie Platzierungen im Projekt hinzufügen. Platzierungen folgen der Vorlage und geben Geräte anhand ihrer Seriennummern und Attributwerte an, die für den jeweiligen Standort oder die Funktion dieser Platzierung sinnvoll sind.

Die folgenden Beispiele veranschaulichen die Verwendung von Projekten und Platzierungen.

#### Beispiel 1:

Im Projekt `SalesPersonNotification` erhalten 10 Kunden einen Button, den sie drücken können, um einen Vertriebsmitarbeiter zu kontaktieren. Es gibt 10 Platzierungen, eine für jeden Kunden. Jede Platzierung hat Werte für `CustomerName` (z. B. Mr. Jones), `SalesPersonPhoneNumber` (z. B. 1-555-555-1234) und die Seriennummer des Button (z. B. G030PM12345678). Die Gerätevorlage, `NotificationButton`, ist in der Platzierung enthalten. Die Attribute `CustomerName` und `SalesPersonPhoneNumber` werden für jede Platzierung definiert. Wenn ein Kunde auf die Schaltfläche klickt, ruft AWS IoT 1-Click auf `SendSMSLambda` mit dem `CustomerName` und `SalesPersonPhoneNumber`-Werte, die mit dieser Schaltfläche verknüpft sind. Die SMS wird basierend auf diesen Werten gesendet.

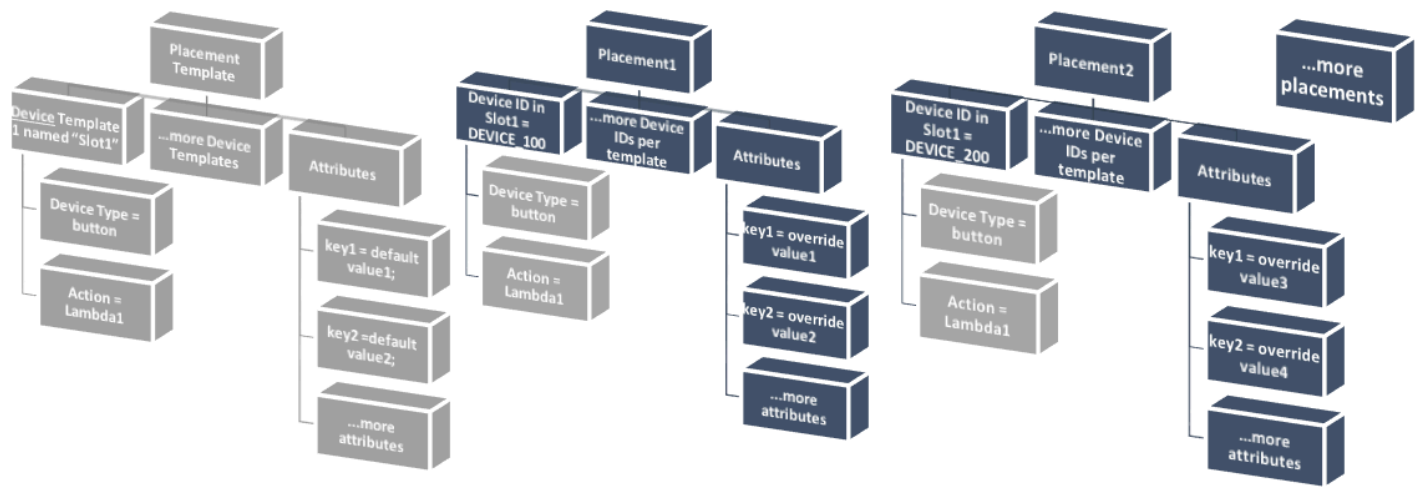
- Platzierungsvorlage:
  - Da jeder Kunde eine Schaltfläche erhält, um einen Vertriebsmitarbeiter zu benachrichtigen, wird die Gerätevorlage `NotificationButton` erstellt.
  - Die Gerätevorlage (in der Platzierung) gibt an, dass die Gerätevorlage `NotificationButton`, wenn darauf geklickt wird, ruft die `SendSMSLambdaLambda`-Funktion.
  - Für jede Platzierung sind die Attribute `CustomerName` und `SalesPersonPhoneNumber` definiert.
- Platzierungen: Es werden 10 Platzierungen erstellt, eine pro Kunde. Jede Platzierung hat spezifische Werte für `CustomerName` (z. B. „Mr. Jones“), `SalesPersonPhoneNumber` (z. B. 1-555-555-1234) und Button-Seriennummer (z. B. G030PM12345678).
- Operation: Wenn ein Kunde seinen Button klickt, ruft AWS IoT 1-Click auf `SendSMSLambda` mit dem `CustomerName` und `SalesPersonPhoneNumber`-Werte, die diesem speziellen Button zugeordnet sind — und abhängig von diesen Werten wird eine SMS gesendet.

#### Beispiel 2:

Im Projekt `MeetingRoomFeedback` wird die Benutzerzufriedenheit durch Drücken der Daumenhoch- und Daumen-runter-Buttons in jedem der 50 Konferenzräume nachverfolgt. Es gibt zwei Gerätevorlagen, `ThumbsUp` und `ThumbsDown`. Wenn der Button „Daumen hoch“ geklickt wird, wird die Funktion `PositiveFeedbackLambda` aufgerufen. Wenn der Button „Daumen runter“ geklickt wird, wird `NegativeFeedbackLambda` aufgerufen. `AMeetingRoomNumber` Das Attribut wird definiert, um die Raumnummer für jede Platzierung aufzunehmen. Es werden 50 Geräteplatzierungen erstellt, eine pro Konferenzraum. Jede Platzierung enthält den Schlüssel `MeetingRoomNumber`, der auf eine Raumnummer gesetzt ist (z. B. 1001), und zwei Buttons, die durch ihre eindeutigen Seriennummern identifiziert sind (z. B. `G030PM12345678` und `G030PM23456789`). Wenn in einem Besprechungsraum eine Schaltfläche geklickt wird, ruft AWS IoT 1-Click die `PositiveFeedbackLambda`-Funktion oder `NegativeFeedbackLambda`-Funktion mit dem `MeetingRoomNumber`-Wert. Feedback kann dann bearbeitet und tabuliert werden.

- Projektname: `MeetingRoomFeedback`
- Platzierungsvorlage:
  - Da jeder Raum zwei Schaltflächen erhält, werden zwei Gerätevorlagen erstellt, `ThumbsUp` und `ThumbsDown`.
  - Die Gerätevorlagen geben vor, dass die `ThumbsUp`-Schaltflächen beim anklicken `PositiveFeedbackLambda` aufrufen, und dass die `ThumbsDown`-Schaltflächen beim Anklicken `NegativeFeedbackLambda` aufrufen.
  - Das Attribut `MeetingRoomNumber` wird definiert, um die Raumnummer für jede Platzierung aufzunehmen.
- Platzierungen: 50 Geräteplatzierungen werden erstellt, eine Platzierung pro Raum. Jede Platzierung enthält den Schlüssel `MeetingRoomNumber`, der auf ein bestimmtes Raumnummernpaar gesetzt ist (z. B. 1001), und zwei Schaltflächen, die durch ihre eindeutigen Seriennummern identifiziert sind (z. B. `G030PM12345678` und `G030PM23456789`).
- Operation: Wenn in einem Besprechungsraum eine Schaltfläche geklickt wird, ruft AWS IoT 1-Click die `PositiveFeedbackLambda`-Funktion oder `NegativeFeedbackLambda`-Funktion mit dem `MeetingRoomNumber`-Wert — und Feedback kann bearbeitet und tabuliert werden.

Das folgende Diagramm zeigt diese Konzepte:



Weitere Informationen finden Sie unter [Erste Schritte mit der AWS IoT 1-Click Konsole](#).

# Erste Schritte mit der AWS IoT 1-Click Konsole

In den folgenden Themen wird beschrieben, wie häufige AWS IoT 1-Click Aufgaben ausgeführt werden.

Themen

- [Geräte beantragen](#)
- [Erstellen eines Projekts](#)

## Geräte beantragen

Das folgende Verfahren demonstriert, wie Sie ein oder mehrere von AWS IoT 1-Click unterstützte Geräte beantragen.

1. Melden Sie sich bei Ihrem AWS-Konto an. Wenn Sie noch kein AWS Konto besitzen, öffnen Sie <https://aws.amazon.com/>, wählen Sie Erstellen eines AWS-Kontos Folgen Sie den Onlineanweisungen.
2. Suchen Sie in der AWS Management Console nach „1-Click“ und wählen Sie AWS IoT 1-Click aus.
3. Wenn Sie einen oder mehrere AWS IoT Enterprise Buttons verwenden, installieren Sie die AWS IoT 1-Click App für iOS oder Android und richten eine Verbindung der Schaltflächen zu Ihrem lokalen WLAN-Netzwerk ein. Die AWS IoT 1-Click App ist im Onboarding der AWS IoT 1-Click Konsole. Dieser Schritt ist für den LTE-M-Button nicht erforderlich, da er das Mobilfunknetz verwendet.
4. Klicken Sie auf Onboarding Klicken Sie auf und danach auf Geräte beantragen.
5. Geben Sie mindestens eine aus [Geräte-IDs](#) (z. B. eine Geräteseriennummer) oder [Einlöscodes](#), durch Kommas getrennt, und wählen Sie Antrag. Wenn das Symbol Antrag Überprüfen Sie alle eingegebenen Werte.
6. Drücken Sie die Taste (n) auf Ihrem Gerät (en) und wählen Sie dann Fertig. Es sollte eine Liste aller bekannten Geräte angezeigt werden.

# Erstellen eines Projekts

Das folgende Verfahren demonstriert, wie Sie ein AWS IoT 1-Click Projekt für Ihre von AWS IoT 1-Click-unterstützten Geräte erstellen.

1. Melden Sie sich bei Ihrem AWS-Konto an und öffnen Sie die AWS IoT 1-Click-Konsole.
2. Klicken Sie auf **Onboarding** Klicken Sie auf und danach auf **Erstellen eines Projekts**.
3. Geben Sie einen Namen und eine optionale Beschreibung für das Projekt ein und wählen Sie dann **Weiter**.
4. Um eine oder mehrere Vorlagen für Ihre Platzierung zu definieren, wählen Sie unter **Programmieren einer Gerätevorlage**, wählen Sie **starten**.
5. Um eine Vorlage für ein beliebiges Button-Gerät zu definieren, wählen Sie **Alle Buttons**.
6. Geben Sie für Gerätevorlagenname einen beschreibenden Namen für Ihre Vorlage ein. Wählen Sie unter **Aktion** die Option **SMS senden** oder **E-Mail senden** aus. Sie können die Option **Benutzerdefinierte Aktion** mit einer **Lambda -Funktion** und wählen Sie eine Ihrer eigenen **Lambda Funktionen** aus. Geben Sie je nach Ihrer Wahl die Telefonnummer, die E-Mail-Adresse oder den Namen der **Lambda -Funktion** ein. Weitere Informationen zum Erstellen von **Lambda -Funktionen** finden Sie im [AWS Lambda Entwicklerhandbuch](#).
7. **UNTERHinzufügen** einer weiteren Gerätevorlage (wenn Sie mehrere Geräte pro Platzierung benötigen) Wählen Sie **Add**.
8. Geben Sie ein **Attribut-Schlüssel-Wert-Paar** ein. Bei Bedarf können Sie zusätzliche **Schlüssel-Wert-Paare** eingeben.
9. Wählen Sie **Create project (Projekt erstellen)**.

Im folgenden Abschnitt [Beispiel: Projekt „Meeting Room Satisfaction“ \(Korrekte Zuteilung von Meeting-Räumen\)](#) Hier finden Sie ein praktisches Beispiel für die Verwendung der AWS IoT 1-Click Konsole zum Erstellen eines Projekts.

## Beispiel: Projekt „Meeting Room Satisfaction“ (Korrekte Zuteilung von Meeting-Räumen)

Das folgende Beispiel hilft Ihnen, AWS IoT 1-Click Konzepte zu verstehen.

- Ein Projekt zum Nachverfolgen der Zufriedenheit für 50 Besprechungsräume (und die zugehörigen AV-Systeme) wurde erstellt und mit dem Namen `MeetingRoomSat` versehen.

- Jeder Besprechungsraum erhält zwei Geräte (Schaltflächen), physisch als „Satisfied“ (Zufrieden) und „Unsatisfied“ (Unzufrieden) beschriftet. Da es zwei Schaltflächen pro Raum gibt, werden zwei Vorlagen erstellt, Satisfied und Unsatisfied.
- Die Satisfied -Vorlage ist so konfiguriert, dass sie eine Lambda -Funktion namens SatLambda.
- Die Unsatisfied -Vorlage ist so konfiguriert, dass sie eine Lambda -Funktion namens UnsatLambda.
- Für beide Vorlagen wird ein Attribut (Schlüssel/Wert-Paar) mit dem Namen MeetingRoomNum erstellt, dessen Wert TBD ist (der TBD-Wert wird geändert, wenn sich beide Schaltflächen physisch in einem Raum befinden).
- Es werden 50 Platzierungen erstellt, eine für jede Raum. Jeder Platzierung sind die beiden Vorlagen zugeordnet (z. B. Satisfied und Unsatisfied).
- Zwei Schaltflächen sind physisch beschriftet und befinden sich in einem Raum. Dann werden mit AWS IoT 1-Click App oder der AWS IoT 1-Click Konsole und den Seriennummern der Schaltflächen die als „Zufrieden“ und „Unzufrieden“ gekennzeichneten Schaltflächen einer der 50 Platzierungen zugeordnet. Dieser Vorgang wird so lange fortgesetzt, bis alle verbleibenden Platzierungen bereitgestellt sind.
- Wird in einem Besprechungsraum eine Raumschaltfläche angeklickt, ruft AWS IoT 1-Click die SatLambda oder .UnsatLambda mit der -Funktion MeetingRoomNum-Wert — und das Feedback kann verarbeitet und in der Cloud gespeichert werden.
- Später kann dem Projekt eine weitere Vorlage hinzugefügt werden, sodass die 50 vorhandenen Platzierungen jetzt ein Platz für eine neue Schaltfläche enthalten, um beispielsweise anzugeben, dass mehr Handtücher oder andere Hygieneartikel in jedem Bad benötigt werden.

Im Folgenden finden Sie ein Beispiel für die Verwendung der AWS IoT 1-Click Konsole, um ein Projekt für die Überwachung der Zuteilung von Meeting-Räumen in einem Bürogebäude zu erstellen (als Teil einer Gruppe von Bürogebäuden).


Um die Zufriedenheit von Meeting-Räumen einschließlich ihrer Audio-/Video-Geräte zu überwachen, werden in jedem Besprechungsraum zwei AWS IoT Enterprise Buttons platziert, eine mit der Aufschrift „Zufrieden“, die andere mit der Aufschrift „Unzufrieden“. Dies ist ein Pilotprojekt, dessen Ergebnisse genutzt werden können, um die Kundenzufriedenheit mit Besprechungsräumen in anderen Gebäuden auf dem Campus zu verbessern.

Nach Abschluss einer Besprechung werden die Teilnehmer aufgefordert, den Button „Zufrieden“ oder „Unzufrieden“ zu drücken, um ihre allgemeine Zufriedenheit mit dem Besprechungsraum und den

zugehörigen Geräten aufzuzeichnen. Anhand dieser Daten werden dann Besprechungsräume mit nicht funktionierenden A/V-Geräten oder anderen Problemen identifiziert.

Die AWS IoT 1-Click Konsole kann verwendet werden, um dieses Projekt einzurichten:

1. Wählen Sie in der AWS IoT 1-Click Konsole die Option **Erstellen eines Projekts**.
2. Geben Sie für den Projektnamen **MeetingRoomSatisfaction** ein. Geben Sie für die Projektbeschreibung ein, **Project used to track customer meeting room satisfaction, including A/V equipment**. Klicken Sie auf **Weiter**.
3. **UNDER** Programmieren einer Gerätevorlage Wählen Sie **starten** Klicken Sie auf **und** danach auf **Alle Buttons**.
4. Geben Sie für Name der Gerätevorlage **Satisfied** ein. Dies ist die Vorlage, die für alle Buttons mit der Aufschrift „Zufrieden“ verwendet wird. Wählen Sie für Aktion die Option **E-Mail senden** aus.

 **Note**

Wenn das Pilotprojekt bezüglich eines gut geeigneten Besprechungsraums erfolgreich ist, können Sie unter Aktion die Option **Benutzerdefinierte Aktion** mit einer Lambda-Funktion auswählen. Diese benutzerdefinierte Lambda -Funktion kann eine E-Mail senden oder die Daten der Schaltfläche „Zufrieden“ in einer Amazon DynamoDB -Tabelle für eine spätere Analyse speichern. Weitere Informationen zum Erstellen von Lambda -Funktionen finden Sie im [AWS Lambda Entwicklerhandbuch](#).

5. **UNDER** Hinzufügen einer weiteren Gerätevorlage (wenn Sie mehrere Geräte pro Platzierung benötigen) Wählen Sie **Add** Klicken Sie auf **und** danach auf **Alle Buttons**. Geben Sie für Name der Gerätevorlage **Unsatisfied** ein. Dies ist die Vorlage, die für alle Buttons mit der Aufschrift „Unzufrieden“ verwendet wird. Wählen Sie für Aktion die Option **E-Mail senden** aus.
6. Geben Sie für **Erforderlicher E-Mail-Standardwert** eine E-Mail-Adresse ein. Geben Sie für **Erforderlicher Betreff Standardwert** **Meeting Room Feedback** ein. Geben Sie für **Erforderlicher Text-Standardwert** **Either positive or negative meeting room feedback has been provided** ein.
7. Geben Sie für **Attributsschlüssel** **Building** ein. Geben Sie unter **Standardwert** die Zeichenfolge **Headquarters** ein. Das Pilotprojekt zur Zufriedenheit mit den Meeting-Räumen findet im Hauptgebäude des Unternehmens statt. Wenn das Pilotprogramm erfolgreich ist, wird es auch in den anderen Gebäuden des Unternehmens eingesetzt. Daher ist es wichtig zu wissen, von welchem Gebäude aus die Meeting-Raum-Geräte Informationen bereitstellen.



8. In der zweiten Schlüssel-Wert-Paar-Zeile geben Sie für Attributsschlüssel **Room** ein. Geben Sie unter Standardwert die Zeichenfolge **TBD** ein. Die **TBD** wird in eine Nummer eines Besprechungsraums geändert, wenn die Schaltflächen dort platziert werden (entweder mit der AWS IoT 1-Click App oder der AWS IoT 1-Click-Konsole).
9. Wählen Sie Create project (Projekt erstellen).

Wenn in einem Besprechungsraum eine Schaltfläche „Zufrieden“ in der AWS IoT 1-Click App platziert wird, wird die Option Zufrieden-Vorlage zugeordnet, und die **TBD** wird durch die Nummer des Meetingraums ersetzt. Dasselbe gilt für den Fall, dass der Button „Nicht zufrieden“ in einem Besprechungsraum platziert wird.

# AWS IoT 1-Click App

Mit der AWS IoT 1-Click-App können Sie Folgendes ausführen:

- Um AWS IoT 1-Click Geräte im Feld bequem zu konfigurieren und zu überwachen, wobei eine Benutzeroberfläche ähnlich derjenigen der AWS IoT 1-Click Konsole verwendet wird.
- Für die Konfiguration von WLAN-Anmeldeinformationen für WLAN-verbundene AWS IoT 1-Click Geräte (z. B. AWS IoT Enterprise Button).

Die AWS IoT 1-Click App steht für Mobilgeräten für iPhone und Android-Mobilgeräten zur Verfügung. Um die App herunterzuladen, navigieren Sie zum [App Store](#) oder [Google Play](#) und suchen Sie nach AWS IoT 1-Click.

# AWS IoT 1-Click-Programmiermodell

Um Anwendungen mit AWS IoT 1-Click Geräten zu erstellen, verwenden Programmierer das [AWS IoT 1-Click Geräte-API](#) und die [AWS IoT 1-Click Projekte-API](#). Die Geräte-API interagiert mit der AWS IoT 1-Click Gerätekomponente und verarbeitet von den Geräten kommende Ereignisse. Diese Ereignisse umfassen die Aktivierung und Deaktivierung der Geräte und die Definition von Ereignisfunktionen und der Aktionen (Lambda-Funktionen), die sie auslösen. Die Geräte-API ist eng mit den AWS Komponenten in der Region verbunden, in der der Hersteller die Geräte registriert hat. Das ist der Grund, warum [AWS Geräteregionen](#) kann sich von der Region unterscheiden, in der der Kunde die Geräte verwendet. Die Projekte-API interagiert mit dem AWS IoT 1-Click Projekte-Service und wird verwendet, um AWS IoT 1-Click-Geräte aggregiert zu verwalten, was Folgendes ermöglicht:

- Geräte in Projekte zu gruppieren.
- Erstellen Sie Vorlagen, die zum Festlegen von Aktionen für alle Geräte im Projekt verwendet werden.
- Definieren Sie Attribute, die kontextbezogene Daten speichern, die für das Projekt relevant sind.

Sie können das AWS IoT 1-Click Programmiermodell verwenden, um einzelne Geräte unter Verwendung der Geräte-API zu programmieren. In diesem Fall nutzen Sie den AWS IoT 1-Click Gerätetyp. Das API definiert Standardereignisformate sowie eine Liste der Methoden, die die Programmierschnittstelle für alle Geräte dieses Typs bildet. Um Methoden für einen bestimmten Gerätetyp aufzurufen, kann ein Programmierer die [InvokeDeviceMethod-API](#) verwenden und die Gerätemethode als Parameter angeben.

Beispielsweise geben alle AWS IoT 1-Click Geräte mit dem Gerätetyp „Schaltfläche“ Ereignisse aus, die Klicks zugeordnet sind, und haben Methoden, um Callback-Funktionen aufzurufen, die aufgerufen werden, wenn das Gerät angeklickt wird. Weitere Informationen zur Button-Schnittstelle finden Sie unter [Schnittstellen nach Gerätetyp](#). Hier ist der Code, um diese Callback-Funktion zu setzen:

```
String methodParameters = mapper.writeValueAsString(
    SetOnClickCallbackRequestParameters.builder()
        .deviceId(deviceId)
        .callback(DeviceCallback.builder()
            .awsLambdaArn("arn:aws:lambda:us-
west-2:123456789012:MyButtonListener")
            .build())
        .build())
```

```
        .build());  
InvokeDeviceMethodRequest request = new InvokeDeviceMethodRequest()  
    .withDeviceMethod(new DeviceMethod()  
        .withDeviceType("button")  
        .withMethodName("setOnClickCallback"))  
    .withDeviceMethodParameters(methodParameters);
```

Sie verwenden das Projects API zur Programmierung einer Geräteflotte. Mit den APIs definieren Sie zunächst, wie die einzelnen Platzierungen aussehen, einschließlich der Gerätevorlagen und -attribute für jede Platzierung. Nachdem dies abgeschlossen ist, erstellen Sie Platzierungen mit spezifischen Gerät-IDs. Jede Platzierung folgt derselben Vorlage. Der Beispiel-Code dafür sieht wie folgt aus:

```
    final Map<String, String> callbacks = new HashMap<>();  
    callbacks.put("onClickCallback", "arn:aws:lambda:us-  
west-2:123456789012:MyButtonListener");  
    final DeviceTemplate item = DeviceTemplate.builder()  
        .withDeviceType("button")  
        .withCallbackOverrides(callbacks)  
        .build();  
    final Map<String, DeviceTemplate> deviceTemplateMap = new HashMap<>();  
    deviceTemplateMap.put("MyDevice", item);  
  
    final Map<String, String> placementDefaultAttributes = new HashMap<>();  
    placementDefaultAttributes.put("location", "Seattle")  
  
    request = CreateProjectRequest.builder()  
        .withProjectName("HelloWorld")  
        .withDescription("My first project!")  
        .withPlacementTemplate(PlacementTemplate.builder()  
            .withDefaultAttributes(placementDefaultAttributes)  
            .withDeviceTemplates(deviceTemplateMap)  
            .build())  
        .build();  
    projectsClient.createProject(request)
```

# AWS IoT 1-Click Callback-Ereignisse

AWS IoT 1-Click können Sie Geräteereignisse abonnieren, indem Sie Callbacks registrieren. Ein Beispiel für ein Callback ist eine AWS Lambda Funktion, die Ihnen, dem AWS IoT 1-Click Kunden, gehört und von Ihnen implementiert wird. Dieses Callback wird jedes Mal aufgerufen, wenn ein Ereignis zum Verbrauch zur Verfügung steht. Weitere Informationen zu Ereignissen und deren Nutzlasten finden Sie in den Abschnitten [AWS IoT 1-Click ck-Ereignisse](#) und [AWS IoT 1-Click Health Ereignisse](#).

## AWS IoT 1-Click ck-Ereignisse

Geräte des Typs `button` veröffentlichen beim jedem Anklicken ein Klick-Ereignis. Sie können dieses Ereignis durch folgende Maßnahme abonnieren:

- Aufrufen der `SetOnClickCallback`-Methode auf einem Gerät.
- Konfigurieren des zugeordneten Projekts entsprechend, wie im vorherigen Beispiel zum Erstellen von Projektcode gezeigt.

Beachten Sie im folgenden Beispiel, dass der Abschnitt `placementInfo` nur vorhanden ist, wenn es für das Gerät eine zugehörige Platzierung gibt. Weitere Informationen finden Sie unter [Projekte, Vorlagen und Platzierungen](#).

```
{
  "deviceEvent": {
    "buttonClicked": {
      "clickType": "SINGLE",
      "reportedTime": "2018-05-04T23:26:33.747Z"
    }
  },
  "deviceInfo": {
    "attributes": {
      "key3": "value3",
      "key1": "value1",
      "key4": "value4"
    },
    "type": "button",
    "deviceId": " G030PMXXXXXXXXXX ",
    "remainingLife": 5.00
  },
  "placementInfo": {
```

```
"projectName": "test",
"placementName": "myPlacement",
"attributes": {
  "location": "Seattle",
  "equipment": "printer"
},
"devices": {
  "myButton": " G030PMXXXXXXXXXX "
}
}
}
```

## AWS IoT 1-Click Health Ereignisse

Geräte veröffentlichen ein Health-Ereignis basierend auf den Health-Parametern, die vom AWS IoT 1-Click Dienst berechnet werden, Sie legen jedoch die entsprechenden Schwellenwerte fest. Das folgende Beispiel zeigt die JSON-Nutzlast eines Health-Ereignisses für Gerät G030PMXXXXXXXXXX mit einer Restlaufzeit von 10 % ("remainingLifeLowerThan":10 Schlüssel-Wert-Paar).

```
{
  "deviceEvent": {
    "deviceHealthMonitor": {
      "condition": {
        "remainingLifeLowerThan": 10
      }
    }
  },
  "deviceInfo": {
    "attributes": {
      "key2": "value2",
      "key1": "value1",
      "projectRegion": "us-west-2"
    },
    "type": "button",
    "deviceId": "G030PMXXXXXXXXXX",
    "remainingLife": 5.4
  }
}
```

## Gerätemethoden

AWS IoT 1-Click Gerätemethoden sind APIs, die von Geräten eines bestimmten Gerätetyps unterstützt werden, wie in der folgenden Tabelle gezeigt. Die vollständige Liste der von jedem Gerät unterstützten Gerätemethoden können durch den Aufruf von [GetDeviceMethods](#) abgerufen werden.

Gerätetyp	Methodenname	Beschreibung
device	<code>getDeviceHealthParameters</code>	Ruft die Zustandsparameter des Geräts ab, z. B. <code>remainingLife</code> .
device	<code>setDeviceHealthMonitorCallback</code>	Richtet einen Callback ein, der aufgerufen wird, wenn die Zustandsparameter für das Gerät unter einem bestimmten Schwellenwert liegen.
device	<code>getDeviceHealthMonitorCallback</code>	Ruft den konfigurierten Callback ab, der aufgerufen wird, wenn die Zustandsparameter unter einem bestimmten Schwellenwert liegen.
button	<code>setOnClickCallback</code>	Richtet einen Callback ein, der aufgerufen werden soll, wenn die Schaltfläche geklickt wurde.
button	<code>getOnClickCallback</code>	Ruft den konfigurierten Callback ab, die beim Klicken auf die Schaltfläche aufgerufen wird.

# Überwachen von AWS IoT 1-Click mit Amazon CloudWatch

AWS IoT 1-Click überwacht automatisch Geräte für Sie und meldet die Metriken über [Amazon CloudWatch](#). Diese Metriken werden in der Geräte-Region gemeldet, in der die Geräte des Herstellers registriert wurden. Weitere Informationen zu Geräte-Regionen finden Sie unter [Funktionsweise von AWS IoT 1-Click](#). Sie finden die Metriken im Amazon CloudWatch Dashboard unter dem `demIoT1clickNamespace`.

Mit Amazon CloudWatch Events können Sie Ihre AWS -Services automatisieren und automatisch auf Systemereignisse reagieren, z. B. bei Problemen mit der Anwendungsverfügbarkeit oder Ressourcenänderungen. Ereignisse von AWS -Services werden für CloudWatch Events nahezu in Echtzeit bereitgestellt. Sie können einfache Regeln schreiben, um anzugeben, welche Ereignisse für Sie interessant sind und welche automatisierten Aktionen ausgeführt werden sollen, wenn ein Ereignis mit einer Regel übereinstimmt. Die folgenden Aktionen, die ausgelöst werden können:

- Aufrufen einer AWS Lambda -Funktion
- Aufrufen des Amazon EC2 Ausführungsbefehls.
- Weiterleiten des Ereignisses an Amazon Kinesis Data Streams
- Aufrufen eines AWS Step Functions ctions-Zustandsautomaten
- Benachrichtigen eines Amazon SNS -Themas oder einer AWS SMS-Themas

AWS IoT 1-Click verfolgt und meldet die folgenden Metriken:

- `TotalEvents` verfolgt die Anzahl der von den Geräten ausgegebenen Ereignissen. Diese Metrik kann für Geräteereignisse, Projekte, Gerätetypen oder Produkttypen angezeigt und als Graph dargestellt werden.
- `RemainingLife` stellt den ungefähren Prozentsatz der verbleibenden Lebensdauer für ein Gerät dar. AWS IoT 1-Click meldet diese Zahl basierend auf der Bewertung des Geräts durch den Hersteller. Wurde ein Button beispielsweise so konzipiert, dass etwa 2000 Klicks möglich sind, und 500 Klicks wurden bereits aufgezeichnet, wird der `RemainingLife`-Wert als 75 % angegeben. Die `RemainingLife`-Metrik kann für Projekte, Gerätetypen oder Produkttypen angezeigt und als Graph dargestellt werden. Kunden können die `RemainingLife`-Metrik nutzen, um Alarme einzurichten, die ausgelöst werden, wenn Geräte unter eine bestimmte Schwelle fallen. Die Kunden können dann die `RemainingLife`-Metrik für Geräte unter Verwendung der Methode `GetDeviceHealthParameters` abfragen, um Geräte zu identifizieren, deren `RemainingLife`-Werte niedrig sind.



- `CallbackInvocationErrors` verfolgt Ausfälle beim Aufruf der Callbacks (Lambda-Funktionen), wenn das Gerät ein Ereignis ausgibt. Die `CallbackInvocationErrors`-Metrik kann nach aufgerufenen Callbacks (Lambda-Funktion-ARNs, die als Callbacks eingerichtet sind) oder nach Projekten angezeigt und als Graph dargestellt werden. Kunden können Alarme für die `CallbackInvocationErrors` mit können Sie benachrichtigt werden, wenn AWS IoT 1-Click Ereignisse nicht von ihren Geräten zu ihren konfigurierten Lambda Funktionen weiterleiten konnte.

Weitere Informationen finden Sie im [Amazon CloudWatch Events-Benutzerhandbuch](#).

# Protokollieren von AWS IoT 1-Click API-Aufrufen mit AWS CloudTrail

AWS IoT 1-Click ist in AWS CloudTrail integriert, einen Service, der die Aktionen eines Benutzers, einer Rolle oder eines AWS-Services in AWS IoT 1-Click protokolliert. CloudTrail erfasst API-Aufrufe für AWS IoT 1-Click als Ereignisse. Zu den erfassten Aufrufen gehören Aufrufe von der AWS IoT 1-Click Konsole und Code-Aufrufe an die AWS IoT 1-Click-API-Operationen. Wenn Sie einen Trail erstellen, können Sie die kontinuierliche Bereitstellung von CloudTrail -Ereignissen an einen Amazon S3 Bucket aktivieren, einschließlich Ereignissen für AWS IoT 1-Click. Auch wenn Sie keinen Trail konfigurieren, können Sie die neuesten Ereignisse in der CloudTrail-Konsole in Event history (Ereignisverlauf) anzeigen. Mit den von CloudTrail erfassten Informationen können Sie die an AWS IoT 1-Click gestellte Anfrage, die IP-Adresse, von der die Anfrage gestellt wurde, den Zeitpunkt der Anfrage und zusätzliche Angaben bestimmen.

Weitere Informationen über CloudTrail, einschließlich Konfiguration und Aktivierung, finden Sie im [AWS CloudTrail-Benutzerhandbuch](#).

## AWS IoT 1-Click Informationen in CloudTrail

CloudTrail wird beim Erstellen Ihres AWS-Kontos für Sie aktiviert. Die in AWS IoT 1-Click auftretenden Aktivitäten werden als CloudTrail -Ereignis zusammen mit anderen AWS - Serviceereignissen in Ereignisverlauf. Sie können die neusten Ereignisse in Ihr(em) AWS-Konto anzeigen, suchen und herunterladen. Weitere Informationen finden Sie unter [Anzeigen von Ereignissen mit dem CloudTrail -API-Ereignisverlauf](#).

Erstellen Sie einen Pfad für einen fortlaufenden Datensatz zu Ereignissen in Ihrem AWS-Konto, einschließlich Ereignissen für AWS IoT 1-Click. Ein Trail ermöglicht es CloudTrail, Protokolldateien in einem Amazon S3-Bucket bereitzustellen. Wenn Sie einen Pfad in der Konsole anlegen, gilt dieser für alle AWS-Regionen. Der Trail protokolliert Ereignisse aus allen Regionen in der AWS-Partition und stellt die Protokolldateien in dem Amazon S3-Bucket bereit, den Sie angeben. Darüber hinaus können Sie andere AWS-Services konfigurieren, um die in den CloudTrail-Protokollen erfassten Ereignisdaten weiter zu analysieren und entsprechend zu agieren. Weitere Informationen finden Sie unter:

- [Übersicht zum Erstellen eines Pfads](#)
- [Von CloudTrail unterstützte Services und Integrationen](#)

- [Konfigurieren von Amazon SNS-Benachrichtigungen für CloudTrail](#)
- [Empfangen von CloudTrail-Protokolldateien aus mehreren Regionen](#) und [Empfangen von CloudTrail-Protokolldateien aus mehreren Konten](#)

AWS IoT 1-Click [Geräte-API](#) unterstützt die Protokollierung der folgenden Aktionen als Ereignisse in CloudTrail -Protokolldateien:

- [ListDevices](#)
- [DescribeDevice](#)
- [GetDeviceMethods](#)
- [UpdateDeviceState](#)
- [InvokeDeviceMethod](#)

AWS IoT 1-Click [Projekte-API](#) unterstützt die Protokollierung der folgenden Aktionen als Ereignisse in CloudTrail -Protokolldateien:

- [CreateProject](#)
- [UpdateProject](#)
- [DescribeProject](#)
- [ListProjects](#)
- [DeleteProject](#)
- [CreatePlacement](#)
- [UpdatePlacement](#)
- [DescribePlacement](#)
- [ListPlacements](#)
- [DeletePlacement](#)
- [AssociateDeviceWithPlacement](#)
- [DisassociateDeviceFromPlacement](#)
- [GetDevicesInPlacement](#)

Jedes Event oder jeder Protokolleintrag enthält Informationen über den Ersteller der Anfrage. Anhand der Identitätsinformationen zur Benutzeridentität können Sie Folgendes bestimmen:

- Ob die Anforderung mit Stamm- oder AWS Identity and Access Management (IAM)-Benutzeranmeldeinformationen gesendet wurde
- Ob die Anforderung mit temporären Sicherheitsanmeldeinformationen für eine Rolle oder einen föderierten Benutzer ausgeführt wurde.
- Ob die Anforderung aus einem anderen AWS-Service gesendet wurde

Weitere Informationen finden Sie unter [CloudTrail userIdentity-Element](#).

## Beispiel: Einträge AWS IoT 1-Click Protokolldateien

Ein Trail ist eine Konfiguration, durch die Ereignisse als Protokolldateien an den von Ihnen angegebenen Amazon S3-Bucket übermittelt werden. CloudTrail-Protokolldateien können einen oder mehrere Einträge enthalten. Ein Ereignis stellt eine einzelne Anfrage aus einer beliebigen Quelle dar und enthält unter anderem Informationen über die angeforderte Aktion, das Datum und die Uhrzeit der Aktion sowie über die Anfrageparameter. CloudTrail-Protokolleinträge sind kein geordnetes Stack-Trace der öffentlichen API-Aufrufe und erscheinen daher in keiner bestimmten Reihenfolge.

Das folgende Beispiel zeigt den CloudTrail-Protokolleintrag, der die Aktion `DescribeDevice` demonstriert.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::012345678910:user/Alice",
    "accountId": "012345678910",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "userName": "Alice"
  },
  "eventTime": "2018-04-12T18:57:27Z",
  "eventSource": "iot1click.amazonaws.com",
  "eventName": "DescribeDevice",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "console.aws.amazon.com",
  "requestParameters": {
    "deviceId": "G030PM12345678"
  },
  "responseElements": null,
```

```

"requestID": "573c5654-3e83-11e8-9eac-c999bd01134e",
"eventID": "be323b62-082a-4352-929d-085d2a3249b0",
"readOnly": true,
"eventType": "AwsApiCall",
"recipientAccountId": "012345678910"
}

```

Das folgende Beispiel zeigt den CloudTrail-Protokolleintrag, der die Aktion `CreateProject` demonstriert.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::012345678910:user/Alice",
    "accountId": "012345678910",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "userName": "Alice"
  },
  "eventTime": "2018-04-12T20:31:02Z",
  "eventSource": "iot1click.amazonaws.com",
  "eventName": "CreateProject",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "127.0.0.1",
  "userAgent": "console.aws.amazon.com",
  "requestParameters": {
    "description": "",
    "placementTemplate": {
      "defaultAttributes": "****",
      "deviceTemplates": {
        "happyId": {
          "deviceType": "button",
          "callbackOverrides": {
            "onClickCallback": "arn:aws:lambda:us-
west-2:012345678910:function:rating_buttons_happy"
          }
        },
        "sadId": {
          "deviceType": "button",
          "callbackOverrides": {
            "onClickCallback": "arn:aws:lambda:us-
west-2:012345678910:function:rating_buttons_sad"
          }
        }
      }
    }
  }
}

```

```
}  
  }  
    }  
      }  
        }
```

# Integration von AWS CloudFormation

AWS IoT 1-Click ist in AWS CloudFormation integriert, eine gemeinsame Sprache, in der Sie alle Ihre Infrastrukturre Ressourcen in einer Cloud-Umgebung beschreiben und bereitstellen können (z. B. Amazon EC2, Auto Scaling, Amazon SNS usw.). Verwenden von AWS CloudFormation Diese Datei dient Ihrer Cloud-Umgebung als alleinige Informationsquelle. Weitere Informationen finden Sie im [Benutzerhandbuch für AWS CloudFormation](#) sowie die AWS IoT 1-Click Themen (z. B. [AWS::IoT1Click::Project](#)) in der Benutzerhandbuch für AWS CloudFormation.

# Authentifizierung und Zugriffskontrolle für AWS IoT 1-Click

Für den Zugriff auf AWS IoT 1-Click APIs sind Anmeldeinformationen erforderlich. Diese Anmeldeinformationen müssen über Berechtigungen für den Zugriff auf AWS-Ressourcen verfügen, wie beispielsweise ein AWS IoT 1-Click-Projekt oder -Gerät. In den folgenden Abschnitten wird beschrieben, wie Sie AWS Identity and Access Management (IAM) und AWS IoT 1-Click verwenden können, um den Zugriff auf Ihre Ressourcen zu schützen.

Jede AWS-Ressource ist Eigentum eines AWS-Kontos und die Berechtigungen für die Erstellung einer Ressource oder den Zugriff darauf werden durch Berechtigungsrichtlinien geregelt. Ein Kontoadministrator kann Berechtigungsrichtlinien an IAM-Identitäten (d. h. Benutzern, Gruppen und Rollen) anfügen. Manche Services (z. B. AWS Lambda) unterstützen auch das Anfügen von Berechtigungsrichtlinien an Ressourcen. Beim Erteilen von Berechtigungen entscheidet der Kontoadministrator, wer die Berechtigungen erhält, für welche Ressourcen die Berechtigungen gelten und welche Aktionen an diesen Ressourcen gestattet werden sollen.

## AWS IoT 1-Click-Ressourcen und -Operationen

In AWS IoT 1-Click sind die Primärressourcen Projekte und Geräte. In einer Richtlinie identifizieren Sie die Ressource, für welche die Richtlinie gilt, mithilfe eines Amazon-Ressourcennamens (ARN). Diesen Ressourcen sind einzigartige Amazon-Ressourcennamen (ARN) zugeordnet, siehe nachfolgende Tabelle.

Ressourcentyp	ARN-Format
Gerät	arn:aws:iot1click:region:account-id:devices/device-id
Projekt	arn:aws:iot1click:region:account-id:projects/project-name

AWS IoT 1-Click implementiert -APIs für die Arbeit mit AWS IoT 1-Click Ressourcen. Sie werden in IAM als Aktionen bezeichnet. Eine Liste der verfügbaren Operationen finden Sie in der Tabelle am Ende dieses Themas.



# Verwenden von identitätsbasierten Richtlinien (IAM-Richtlinien) für AWS IoT 1-Click

Dieses Thema enthält Beispiele zu identitätsbasierten Richtlinien, die verdeutlichen, wie ein Kontoadministrator IAM-Identitäten (d. h. Benutzern, Gruppen und Rollen) Berechtigungsrichtlinien zuweisen und somit Berechtigungen zur Durchführung von Operationen an AWS IoT 1-Click Ressourcen erteilen kann.

Hier ein Beispiel für eine Berechtigungsrichtlinie.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot1click:CreateProject"
      ],
      "Resource": "*"
    }
  ]
}
```

Die Richtlinie hat eine Anweisung — die Berechtigungen für eine AWS IoT 1-Click Aktion (`iot1click:CreateProject`) auf einer Ressource unter Verwendung des Amazon-Ressourcennamens (ARN) für die Anwendung. Der ARN gibt in diesem Fall einen Platzhalter (\*) an. Dies bedeutet, dass die Berechtigung für jede beliebige Ressource erteilt wird.

Eine Tabelle mit allen AWS IoT 1-Click -API-Operationen und den Ressourcen, für welche diese gelten, finden Sie hier: [AWS IoT 1-Click -API-Berechtigungen: Referenz zu Aktionen, Berechtigungen und Ressourcen](#).

## AWS-verwaltete (vordefinierte) Richtlinien für &AWS IoT 1-Click;

Durch die Bereitstellung von eigenständigen IAM-Richtlinien, die von AWS erstellt und administriert werden, deckt AWS viele häufige Anwendungsfälle ab. Diese von AWS verwalteten Richtlinien erteilen die erforderlichen Berechtigungen für viele häufige Anwendungsfälle, sodass Sie nicht mühsam ermitteln müssen, welche Berechtigungen erforderlich sind. Weitere Informationen finden Sie unter [AWS-verwaltete Richtlinien](#) im IAM Benutzerhandbuch.

Die folgenden AWS-verwalteten Richtlinien, die Sie Benutzern in Ihrem Konto zuweisen können, gelten für AWS IoT 1-Click und sind nach Anwendungsfallszenarien gruppiert:

- `AWSIoT1ClickFullAccess`: Erteilt vollen Zugriff auf AWS IoT 1-Click Ressourcen unter Verwendung der AWS-Managementkonsole. Die erteilten Berechtigungen enthalten alle AWS IoT 1-Click Aktionen zum Verwalten von Geräten und Projekten.
- `AWSIoT1ClickReadOnlyAccess`: Erteilt schreibgeschützten Zugriff auf AWS IoT 1-Click Ressourcen unter Verwendung der AWS-Managementkonsole. Dieser Zugriff ermöglicht einem Benutzer, AWS IoT 1-Click Geräte und -Projekte aufzulisten und die Projekt-Konfiguration anzuzeigen.

#### Note

Sie können diese Berechtigungsrichtlinien prüfen, indem Sie sich bei der IAM-Konsole (<https://console.aws.amazon.com/iam/>) und nach bestimmten Richtlinien suchen.

Sie können auch Ihre eigenen, benutzerdefinierten IAM-Richtlinien erstellen, um Berechtigungen für AWS IoT 1-Click Aktionen und -Ressourcen zu gewähren. Die benutzerdefinierten Richtlinien können Sie dann den IAM-Benutzern oder -Gruppen zuweisen, die diese Berechtigungen benötigen.

## AWS IoT 1-Click -API-Berechtigungen: Referenz zu Aktionen, Berechtigungen und Ressourcen

Wenn Sie die Zugriffskontrolle in der AWS Cloud einrichten und eine Berechtigungsrichtlinie für eine IAM-Identität (identitätsbasierte Richtlinie) verfassen, können Sie die folgende Tabelle als Referenz verwenden. In der Tabelle sind alle AWS IoT 1-Click -API-Operationen sowie die zugehörigen Aktionen und AWS Ressourcen, für die Sie Berechtigungen erteilen können, aufgeführt. Die Aktionen geben Sie im Feld `Action` und den Wert für die Ressource im Feld `Resource` der Richtlinie an.

Zum Formulieren von Bedingungen in Ihren AWS IoT 1-Click-Richtlinien können Sie die AWS-weiten Bedingungsschlüssel verwenden. Eine vollständige Liste der AWS-weiten Schlüssel enthält der Abschnitt [Verfügbare Schlüssel](#) im IAM Benutzerhandbuch.

**Note**

Um eine Aktion anzugeben, verwenden Sie das Präfix `iot1click:` gefolgt vom Namen der API-Operation (z. B. `iot1click:ListProjects`).

IoT 1-Click-Operationen	Erforderliche Berechtigungen (API-Aktionen)	Ressourcen
ListDevices	iot1click:ListDevices	*
DescribeDevice	iot1click:DescribeDevice	arn:aws:iot1click:region:account-id:devices/device-id
GetDeviceMethods	iot1click:GetDeviceMethods	arn:aws:iot1click:region:account-id:devices/device-id
UpdateDeviceState	iot1click:UpdateDeviceState	arn:aws:iot1click:region:account-id:devices/device-id
InvokeDeviceMethod	iot1click:InvokeDeviceMethod	arn:aws:iot1click:region:account-id:devices/device-id
ListDeviceEvents	iot1click:ListDeviceEvents	arn:aws:iot1click:region:account-id:devices/device-id
InitializeDeviceClaim	iot1click:InitializeDeviceClaim	arn:aws:iot1click:region:account-id:devices/device-id
FinalizeDeviceClaim	iot1click:FinalizeDeviceClaim	arn:aws:iot1click:region:account-id:devices/device-id
UnclaimDevice	iot1click:UnclaimDevice	arn:aws:iot1click:region:account-id:devices/device-id
ClaimDeviceByClaimCode	iot1click:ClaimDeviceByClaimCode	*

IoT 1-Click-Operationen	Erforderliche Berechtigungen (API-Aktionen)	Ressourcen
CreateProject	iot1click:CreateProject	arn:aws:iot1click:region:account-id:projects/project-name
UpdateProject	iot1click:UpdateProject	arn:aws:iot1click:region:account-id:projects/project-name
DescribeProject	iot1click:DescribeProject	arn:aws:iot1click:region:account-id:projects/project-name
ListProjects	iot1click:ListProjects	*
DeleteProject	iot1click>DeleteProject	arn:aws:iot1click:region:account-id:projects/project-name
CreatePlacement	iot1click:CreatePlacement	arn:aws:iot1click:region:account-id:projects/project-name
UpdatePlacement	iot1click:UpdatePlacement	arn:aws:iot1click:region:account-id:projects/project-name
DescribePlacement	iot1click:DescribePlacement	arn:aws:iot1click:region:account-id:projects/project-name
ListPlacements	iot1click:ListPlacements	arn:aws:iot1click:region:account-id:projects/project-name
DeletePlacement	iot1click>DeletePlacement	arn:aws:iot1click:region:account-id:projects/project-name
AssociateDeviceWithPlacement	iot1click:AssociateDeviceWithPlacement	arn:aws:iot1click:region:account-id:projects/project-name
DissacociateDeviceFromPlacement	iot1click:DissacociateDeviceFromPlacement	arn:aws:iot1click:region:account-id:projects/project-name
GetDevicesInPlacement	iot1click:GetDevicesInPlacement	arn:aws:iot1click:region:account-id:projects/project-name

# Markierungen Ihrer AWS IoT 1-Click Ressourcen

Zur einfacheren Verwaltung Ihrer AWS IoT 1-Click Ressourcen können Sie den ARN-basierten Ressourcen optional eigene Metadaten mithilfe von Tags zuweisen. In diesem Kapitel werden Tags und deren Erstellung beschrieben.

## Grundlagen zu Tags

Mit Tags können Sie AWS IoT 1-Click-Ressourcen auf unterschiedliche Weise kategorisieren (z. B. nach Zweck, Eigentümer oder Umgebung). Dies ist hilfreich, wenn Sie viele Ressourcen desselben Typs haben. In diesem Fall können Sie basierend auf den zugewiesenen Tags schnell bestimmte Ressourcen suchen und identifizieren. Jedes Tag besteht aus einem Schlüssel und einem optionalen Wert, die Sie beide selbst definieren können. Sie können beispielsweise einen Satz von Tags für mehrere Buttons definieren, die einem bestimmten Manager oder Konto gehören. Sie können die Ressourcen auf Grundlage der hinzugefügten Tags filtern und danach suchen. Wir empfehlen die Verwendung von Tag-Schlüsseln, die die Anforderungen der jeweiligen Ressourcentypen erfüllen. Eine Anzahl einheitlicher Tag-Schlüssel vereinfacht das Verwalten der Ressourcen. Weitere Informationen finden Sie unter [AWS Tagging-Strategien](#).

Sie können Tags auch verwenden, um Ihre Kosten zu kategorisieren und zu verfolgen. Wenn Sie Tags auf Ressourcen anwenden, generiert AWS einen Kostenzuordnungsbericht als CSV-Datei mit Ihrer Nutzung und Kosten gemäß Ihren Tags. Sie können Tags anwenden, die geschäftliche Kategorien (wie Kostenstellen, Anwendungsnamen oder Eigentümer) darstellen, um die Kosten für mehrere Services zu organisieren. Weitere Informationen zur Verwendung von Tags für die Kostenzuordnung finden Sie unter [Verwenden von Kostenzuordnungs-Tags](#) im [AWS-Benutzerhandbuch für die Abrechnungs- und Kostenverwaltung](#).

Zur einfacheren Nutzung können Sie den Tag-Editor in der AWS Management Console verwenden. In diesem Fall können Sie Tags zentral und einheitlich erstellen und verwalten. Weitere Informationen finden Sie unter [Arbeiten mit dem Tag Editor](#) im [Erste Schritte mit der AWS Management Console](#).

Sie können auch mit der AWS-CLI und den AWS IoT 1-Click-Geräte- und -Projekt-APIs mit Tags arbeiten. Sie können Tags AWS IoT 1-Click Projekten und -Geräten zuordnen, wenn Sie sie erstellen, indem Sie die Tags-Feld in den folgenden Befehlen:

- [CreateProject](#) (Projekte-API)
- [Antrag abschließen](#) (Geräte-API)

Sie können Tags für vorhandene Ressourcen hinzufügen, ändern oder löschen. Verwenden Sie dazu die folgenden Befehle:

<p>AWS IoT 1-Click <a href="#">Projekte-API</a> (Projekt-ARNs)</p>	<p>AWS IoT 1-Click <a href="#">Geräte-API</a> (Verwendung von Geräte-ARNs)</p>
<p><a href="#">TagResource</a></p>	<p><a href="#">Tag</a></p>
<p><a href="#">ListTagsForResource</a></p>	<p>Siehe <a href="#">TagResource</a> (POST), <a href="#">TagListTagsForResource</a> (GET) und <a href="#">UntagResource</a> (DELETE).</p>
<p><a href="#">UntagResource</a></p>	

Sie können Tag-Schlüssel und Werte bearbeiten und Tags jederzeit von einer Ressource entfernen. Sie können den Wert eines Tags zwar auf eine leere Zeichenfolge, jedoch nicht Null festlegen. Wenn Sie ein Tag mit demselben Schlüssel wie ein vorhandenes Tag für die Ressource hinzufügen, wird der alte Wert mit dem neuen überschrieben. Wenn Sie eine Ressource löschen, werden alle der Ressource zugeordneten Tags ebenfalls gelöscht.

## Tag-Einschränkungen

Die folgenden grundlegenden Einschränkungen gelten für Tags:

- Maximale Anzahl von Tags pro Ressource: 50
- Maximale Schlüssellänge — 127 Unicode-Zeichen in UTF-8
- Maximale Wertlänge: 255 Unicode-Zeichen in UTF-8
- Bei Tag-Schlüsseln und -Werten muss die Groß-/Kleinschreibung beachtet werden.
- Verwenden Sie in Tag-Namen oder -Werten nicht das Präfix `aws`. Dieses ist für die Verwendung durch AWS reserviert. Sie können keine Tag-Namen oder Werte mit diesem Präfix bearbeiten oder löschen. Tags mit diesem Präfix werden nicht als Ihre Tags pro Ressourcenlimit angerechnet.
- Wenn Ihr Markierungsschema für mehrere Services und Ressourcen verwendet wird, denken Sie daran, dass andere Services möglicherweise Einschränkungen für zulässige Zeichen haben. Im allgemeinen zulässige Zeichen: Buchstaben, Leerzeichen und Zahlen, die in UTF-8 darstellbar sind, sowie die folgenden Sonderzeichen: `+ - = . _ : / @`

# Benutzerhandbuch AWS IoT Enterprise Button

Der AWS IoT Enterprise Button ist eine einfache und unkompliziert zu konfigurierende WLAN-basierte Schaltfläche. Sie wurde für Unternehmen und Entwickler konzipiert, um eine einfache Integration in vorhandene Business-Workflows und Systeme unter Verwendung von AWS IoT 1-Click zu unterstützen.

Der AWS IoT Enterprise Button unterstützt drei Arten von Klicks:

- Einzel
- Double
- Langes Drücken

Um eine ordnungsgemäße Funktionsweise zu gewährleisten, müssen Sie die WLAN-Verbindung unter Verwendung der AWS IoT 1-Click App (iOS oder Android) konfigurieren. In der App können Sie die WLAN-Verbindung des Button konfigurieren, indem Sie sich bei Ihrem AWS-Konto anmelden oder indem Sie rechts oben in der App auf das WLAN-Symbol tippen, um die Anmeldung zu überspringen.

Nachdem die Verbindung konfiguriert und über die mobile App oder Konsole beantragt wurde, sollte der Button grün blinken, wenn ein einzelnes, doppeltes oder langes Drücken erfolgt.

Wenn Sie vermuten, dass nach der Konfiguration ein Problem mit dem Button vorliegt, kann diese Tabelle Ihnen bei der Fehlerbehebung helfen.

Farbe	Status	Empfehlung
Weiß blinken	Verbindung mit dem WLAN, die IP-Adresse wird abgerufen, oder die Verbindung mit AWS IoT wird hergestellt.	–
Durchgängig grün	Es wurde erfolgreich eine Verbindung mit dem WLAN hergestellt und eine Nachricht an AWS IoT gesendet.	–

Farbe	Status	Empfehlung
Blau blinken	Die Schaltfläche befindet sich im Konfigurationsmodus.	Warten Sie, bis der Konfigurationsprozess abgeschlossen ist.
Durchgängig orange	Das WLAN ist nicht konfiguriert.	Verwenden Sie die AWS IoT 1-Click App, um WLAN zu konfigurieren.
Rot: kurz, kurz, kurz	Bei der Verbindung zu dem konfigurierten WLAN ist ein Fehler aufgetreten.	Überprüfen Sie, ob alle Netzwerkeinstellungen geändert wurden, oder ob die Schaltfläche zu weit vom WLAN-Router entfernt ist.
Rot: kurz, kurz, lang	Beim Abruf einer IP-Adresse vom WLAN ist ein Fehler aufgetreten.	Überprüfen Sie, ob WLAN-Probleme vorliegen.
Rot: kurz, lang, kurz	Bei der Suche nach einem Hostnamen ist ein Fehler aufgetreten.	Überprüfen Sie, ob WLAN-Probleme vorliegen.
Rot: kurz, lang, lang	Verbindung mit AWS IoT kann nicht hergestellt werden.	Überprüfen Sie, ob WLAN-Probleme vorliegen. Wenn keine Netzwerkprobleme vorliegen und das Problem weiterhin besteht, wenden Sie sich bitte an <a href="#">AWS Support-Center</a> . Geben Sie die Geräteseriennummer (DSN) an. Sie finden sie auf der Rückseite des Button.



Farbe	Status	Empfehlung
Rot: lang, kurz, kurz	Es kann keine sichere Verbindung mit dem Server eingerichtet werden.	Verwenden Sie die AWS IoT 1-Click iOS - oder Android-Apps, um zu überprüfen, ob Sie über die neueste Firmware verfügen.
Rot: lang, kurz, lang	Es ist ein HTTP 403 Forbidden-Fehler aufgetreten.	So kontaktieren Sie <a href="#">AWS Support-Center</a> Geben Sie die DSN an. Sie finden sie auf der Rückseite des Button.
Rot: nach 15 Sekunden Tastendruck	Taste zurücksetzen.	Sie können die WLAN-Konfiguration von AWS IoT Enterprise Button zurücksetzen, indem Sie die Taste 15 Sekunden lang drücken.

# AWS IoT 1-Click mit der AWS CLI verwenden

Um die Verwendung der AWS Command Line Interface (AWS CLI) zu demonstrieren, stellen Sie sich das Szenario eines Entsorgungsunternehmens vor, das seinen Müllabholservice mit AWS IoT 1-Click optimieren möchte.

In diesem Szenario wird jeder Müllcontainer mit einer AWS IoT Enterprise Button gekoppelt. Wenn ein Müllcontainer voll ist, muss der Kunde nur den zugehörige Button drücken, um den Austausch des Müllcontainers anzufordern.

## Note

Alle AWS IoT Enterprise Button Geräte-IDs beginnen mit „G030PM“.

Die folgenden Schritte werden vom Müllbeseitigungsunternehmen verwendet, um die AWS IoT Enterprise Button für die Verwendung durch den Kunden vorzubereiten.

So bereiten Sie die AWS IoT Enterprise-Schaltfläche für die Verwendung durch den Kunden vor

1. Die einzige Möglichkeit, Wi-Fi für einen AWS IoT Enterprise Button einzurichten, besteht in der Verwendung der AWS IoT 1-Click mobile App. Informationen zur Installation der App finden Sie unter [AWS IoT 1-Click App](#). Nachdem Sie die App installiert haben, wählen Sie nicht Melden Sie sich bei Ihrem AWS-Konto an (wie Sie dies normalerweise tun würden). In dieser Übung möchten wir demonstrieren, wie Sie die AWS CLI verwenden. Wenn Sie Melden Sie sich bei Ihrem AWS-Konto an wählen, werden die Befehle `initiate-device-claim` und `finalize-device-claim` für Sie aufgerufen. Wir möchten dies „manuell“ mithilfe der Befehlszeilenschnittstelle tun, wie in den folgenden Schritten gezeigt.
2. Für die AWS CLI-Demonstrationszwecke, anstatt Anmeldung bei AWS Konto Wählen Sie das kleine runde WLAN-Symbol in der rechten obere Ecke aus. Wählen Sie als Nächstes WLAN konfigurieren. Scannen oder geben Sie die Geräte-ID ein, und befolgen Sie die restlichen Anweisungen der mobilen App.
3. Falls die AWS CLI nicht installiert ist, befolgen Sie die Anleitungen unter [Installieren der AWS-CLI](#). Führen Sie die folgenden beiden Befehle aus, um die verfügbaren AWS IoT 1-Click AWS CLI-Befehle aufzulisten.

```
aws iot1click-projects help
```

```
aws iot1click-devices help
```

- Um den jetzt mit Wi-Fi verbundenen AWS IoT Enterprise Button dem AWS-Konto des Müllbeseitigungsunternehmens zuzuordnen, führen Sie den folgenden Befehl mit der Geräte-ID Ihres Geräts aus.

```
aws iot1click-devices initiate-device-claim --device-id G030PM0123456789
{
  "State": "CLAIM_INITIATED"
}
```

Drücken Sie die Taste am Gerät. Nach unregelmäßigem Blinken von weißem Licht sollten Sie etwa eine Sekunde lang durchgehendes grünes Licht sehen. Wenn Sie dies nicht tun, wiederholen Sie das vorherige WLAN-Verbindungsverfahren.

- Nachdem Sie im vorherigen Schritt das durchgehend grüne Licht gesehen haben, führen Sie den folgenden Befehl aus (mit dem ID-Wert Ihres Geräts).

```
aws iot1click-devices finalize-device-claim --device-id G030PM0123456789
{
  "State": "CLAIMED"
}
```

Die "State": "CLAIMED" Die Antwort gibt an, dass das Gerät erfolgreich beim AWS IoT 1-Click -Service registriert wurde.

#### Note

Wenn der Gerätehersteller einen Antragscode bereitgestellt hat, der mit „C-“ beginnt, können Sie nur den Befehl `aws iot1click-devices claim-devices-by-claim-code` verwenden, um ein oder mehrere Geräte mit einem einzigen Antragscode zu beantragen, wie im folgenden Beispiel gezeigt.

```
aws iot1click-devices claim-devices-by-claim-code --claim-code C-123EXAMPLE
{
  "Total": 9
  "ClaimCode": "C-123EXAMPLE"
}
```

In diesem Beispiel fehlen bei der Aktion "Total": 9 Gibt an, dass die neun Geräte mit dem Antragscode verknüpft sind C-123EXAMPLE Vom AWS IoT 1-Click -Service erfolgreich beantragt wurden.

- Als Nächstes bereiten Sie das Erstellen eines geeigneten AWS IoT 1-Click Projekts für die Entsorgungsfirma vor, indem Sie eine JSON-Textdatei mit dem Namen `create-project.json`. Die Datei enthält Folgendes.

```
{
  "projectName": "SeattleDumpsters",
  "description": "All dumpsters in the Seattle region.",
  "placementTemplate": {
    "defaultAttributes": {
      "City" : "Seattle"
    },
    "deviceTemplates": {
      "empty-dumpster-request" : {
        "deviceType": "button"
      }
    }
  }
}
```

Die Schlüssel-Wert-Paare `placementTemplate` und `deviceTemplates` sind die Attribute, die auf alle Buttons angewendet werden, die Teil des `SeattleDumpsters`-Projekts sind. Um dieses Projekt zu erstellen, führen Sie den folgenden Befehl aus (wobei davon ausgegangen wird, dass `create-project.json` befindet sich in der [Aktuelles Arbeitsverzeichnis](#) der AWS CLI-Eingabeaufforderung).

```
aws iot1click-projects create-project --cli-input-json file://create-project.json
```

Führen Sie den folgenden Befehl aus, um das neu erstellte Projekt anzuzeigen.

```
aws iot1click-projects list-projects
{
  "projects": [
    {
      "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
```

```

        "projectName": "SeattleDumpsters",
        "createdDate": 1563483100,
        "updatedDate": 1563483100,
        "tags": {}
    }
]
}

```

Führen Sie den Befehl `describe-project` wie folgt aus, um weitere Details zu erhalten.

```

aws iot1click-projects describe-project --project-name SeattleDumpsters
{
  "project": {
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
    "projectName": "SeattleDumpsters",
    "description": "All dumpsters in the Seattle region.",
    "createdDate": 1563483100,
    "updatedDate": 1563483100,
    "placementTemplate": {
      "defaultAttributes": {
        "City": "Seattle"
      },
      "deviceTemplates": {
        "empty-dumpster-request": {
          "deviceType": "button",
          "callbackOverrides": {}
        }
      }
    },
    "tags": {}
  }
}

```

7. Mit dem für die Region Seattle erstellten Projekt können Sie als Nächstes Platzierung für einen bestimmten Müllcontainer (für Kunde 217) wie folgt erstellen. Die Escape-Anführungszeichen sind für Windows erforderlich.

```

aws iot1click-projects create-placement --project-name SeattleDumpsters --placement-name customer217 --attributes "{\"location\": \"1800 9th Ave Seattle, WA 98101\", \"phone\": \"206-123-4567\"}"

```

Führen Sie den folgenden Befehl aus, um die neu erstellte Platzierung anzuzeigen.

```
aws iot1click-projects list-placements --project-name SeattleDumpsters
{
  "placements": [
    {
      "projectName": "SeattleDumpsters",
      "placementName": "customer217",
      "createdDate": 1563488454,
      "updatedAt": 1563488454
    }
  ]
}
```

Führen Sie den Befehl `describe-placement` wie folgt aus, um weitere Details zu erhalten.

```
aws iot1click-projects describe-placement --project-name SeattleDumpsters --
placement-name customer217
{
  "placement": {
    "projectName": "SeattleDumpsters",
    "placementName": "customer217",
    "attributes": {
      "phone": "206-123-4567",
      "location": "1800 9th Ave Seattle, WA 98101"
    },
    "createdDate": 1563488454,
    "updatedAt": 1563488454
  }
}
```

8. Obwohl das Gerät jetzt mit dem AWS IoT 1-Click Konto des Entsorgungsunternehmens verknüpft ist, ist es nicht mit der Platzierung verknüpft. Bestätigen Sie dies mit dem folgenden Befehl.

```
aws iot1click-projects get-devices-in-placement --project-name SeattleDumpsters --
placement-name customer217
{
  "devices": {}
}
```

Führen Sie den folgenden Befehl aus, um das Gerät der Platzierung zuzuordnen.

```
aws iot1click-projects associate-device-with-placement --project-name
  SeattleDumpsters --placement-name customer217 --device-template-name empty-
  dumpster-request --device-id G030PM0123456789
```

Führen Sie zum Bestätigen des vorherigen Befehls `get-devices-in-placement` erneut aus.

```
aws iot1click-projects get-devices-in-placement --project-name SeattleDumpsters --
  placement-name customer217
{
  "devices": {
    "empty-dumpster-request": "G030PM0123456789"
  }
}
```

Führen Sie für weitere Details den Befehl `describe-device` folgendermaßen aus (beachten Sie den Wechsel von `iot1click-projects` zu `iot1click-devices`).

```
aws iot1click-devices describe-device --device-id G030PM0123456789
{
  "DeviceDescription": {
    "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/G030PM0123456789",

    "Attributes": {
      "projectRegion": "us-west-2",
      "projectName": "SeattleDumpsters",
      "placementName": "customer217",
      "deviceTemplateName": "empty-dumpster-request"
    },
    "DeviceId": "G030PM0123456789",
    "Enabled": false,
    "RemainingLife": 99.9,
    "Type": "button",
    "Tags": {}
  }
}
```

Da derzeit nur ein Gerät vorhanden ist, liefert der folgende Befehl ähnliche Ergebnisse.

```
aws iot1click-devices list-devices --device-type button
{
  "Devices": [
    {
      "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789",
      "Attributes": {
        "projectRegion": "us-west-2",
        "projectName": "SeattleDumpsters",
        "placementName": "customer217",
        "deviceTemplateName": "empty-dumpster-request"
      },
      "DeviceId": "G030PM0123456789",
      "Enabled": false,
      "RemainingLife": 99.9,
      "Type": "button",
      "Tags": {}
    }
  ]
}
```

9. Führen Sie den folgenden Befehl aus, um zu überprüfen, ob das Gerät ordnungsgemäß funktioniert. Passen Sie die Zeitstempel entsprechend an, die im [ISO 8061-Format](#) vorliegen.

```
aws iot1click-devices list-device-events --device-id G030PM0123456789 --from-time-
stamp 2019-07-17T15:45:12.880Z --to-time-stamp 2019-07-19T15:45:12.880Z
{
  "Events": [
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\"clickType\": \"SINGLE\",
\\\"reportedTime\\\": \\\"2019-07-18T23:47:55.015Z\\\", \\\"certificateId\\\":
\\\"fe8798a6c97c62ef8756b80eeefdcf2280f3352f82faa8080c74cc4f4a4d1811\\\",
\\\"remainingLife\\\": 99.850000000000001, \\\"testMode\\\": false}"
    }
  ]
}
```



Hier sehen wir, dass ein Single-Click-Ereignis (`"clickType": "SINGLE"`) am 2019-07-18T23:47:55.015Z aufgetreten ist. Doppelklicken Sie nun das Gerät (zwei schnelle Tastendrucke hintereinander) und führen Sie den Befehl erneut aus. Beachten Sie nun das Doppelklick-Ereignis (`"clickType": "DOUBLE"`), ähnlich dem folgenden.

```
aws iot1click-devices list-device-events --device-id G030PM0123456789 --from-time-stamp 2019-07-17T15:45:12.880Z --to-time-stamp 2019-07-19T15:45:12.880Z
{
  "Events": [
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\"clickType\": \"SINGLE\"",
      \"reportedTime\": \"2019-07-18T23:47:55.015Z\", \"certificateId\":
      \"fe8798a6c97c62ef8756b80eeefdcb2280f3352f82faa8080c74cc4f4a4d1811\",
      \"remainingLife\": 99.85000000000001, \"testMode\": false}"
    },
    {
      "Device": {
        "Attributes": {},
        "DeviceId": "G030PM0123456789",
        "Type": "button"
      },
      "StdEvent": "{\"clickType\": \"DOUBLE\"",
      \"reportedTime\": \"2019-07-19T00:14:41.353Z\", \"certificateId\":
      \"fe8798a6c97c62ef8756b80eeefdcb2280f3352f82faa8080c74cc4f4a4d1811\",
      \"remainingLife\": 99.8, \"testMode\": false}"
    }
  ]
}
```

10. Jeder Gerätetyp verfügt über eine Reihe von aufrufbaren Gerätemethoden. Führen Sie den Befehl `get-device-methods` wie folgt aus, um die verfügbaren Methoden für Ihren Gerätetyp aufzulisten.

```
aws iot1click-devices get-device-methods --device-id G030PM0123456789
{
  "DeviceMethods": [
```

```

    {
      "MethodName": "getDeviceHealthParameters"
    },
    {
      "MethodName": "setDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "getDeviceHealthMonitorCallback"
    },
    {
      "MethodName": "setOnClickCallback"
    },
    {
      "MethodName": "getOnClickCallback"
    }
  ]
}

```

Um eine der verfügbaren Methoden aufzurufen, verwenden Sie den Befehl `invoke-device-method`, wie als Nächstes gezeigt.

```

aws iot1click-devices invoke-device-method --cli-input-json file://invoke-device-
method.json
{
  "DeviceMethodResponse": "{\"remainingLife\": 99.8}"
}

```

Hier enthält `invoke-device-method.json` Folgendes.

```

{
  "DeviceId": "G030PM0123456789",
  "DeviceMethod": {
    "DeviceType": "device",
    "MethodName": "getDeviceHealthParameters"
  }
}

```

#### Note

Die `get`-Methoden (z. B. `getDeviceHealthParameters`) erwarten keine Parameter. Daher kann die Zeile `"DeviceMethodParameters": ""` innerhalb

der JSON-Datei nicht verwendet werden (dies führt dazu: An error occurred (InvalidRequestException) when calling the InvokeDeviceMethod operation: A request parameter was invalid.)

11. Wenn Sie `aws iot1click-devices list-devices --device-type button` ausführen, können Sie sehen, dass der Standardwert für `Enabled` `false` ist. Der folgende Befehl setzt diesen Schlüssel auf `true`.

```
aws iot1click-devices update-device-state --device-id G030PM0123456789 --enabled
```

Führen Sie den vorherigen Befehl erneut mit dem Argument `--no-enabled` aus, um ihn auf `false` zurückzusetzen.

12. Wenn sich die Kundendaten ändern, können Sie die Platzierungsinformationen eines Geräts wie im nächsten Schritt aktualisieren (beachten Sie den Wechsel von `iot1click-devices` zu `iot1click-projects`). Führen Sie den folgenden Befehl aus, um die aktuellen Informationen von `customer217` anzuzeigen (siehe `attributes`).

```
aws iot1click-projects describe-placement --project-name SeattleDumpsters --
placement-name customer217
{
  "placement": {
    "projectName": "SeattleDumpsters",
    "placementName": "customer217",
    "attributes": {
      "phone": "206-123-4567",
      "location": "1800 9th Ave Seattle, WA 98101"
    },
    "createdDate": 1563488454,
    "updatedAt": 1563488454
  }
}
```

Führen Sie als Nächstes den folgenden Befehl aus, um die Telefon- und Standortattribute des Kunden zu aktualisieren.

```
aws iot1click-projects update-placement --cli-input-json file://update-
placement.json
```

Hier enthält `update-placement.json` Folgendes.

```
{
  "projectName": "SeattleDumpsters",
  "placementName": "customer217",
  "attributes": {
    "phone": "206-266-1000",
    "location": "410 Terry Ave N Seattle, WA 98109"
  }
}
```

Führen Sie, wie hier gezeigt, `describe-placement` erneut aus, um dieses Update zu überprüfen.

```
aws iot1click-projects describe-placement --project-name SeattleDumpsters --
placement-name customer217
{
  "placement": {
    "projectName": "SeattleDumpsters",
    "placementName": "customer217",
    "attributes": {
      "phone": "206-266-1000",
      "location": "410 Terry Ave N Seattle, WA 98109"
    },
    "createdDate": 1563488454,
    "updatedAt": 1563572842
  }
}
```

13. Verwenden Sie den Befehl `update-project`, um Projektinformationen zu aktualisieren. Ein Projekt enthält in der Regel mehrere Kundenplatzierungen. Hier sind die vorhandenen `SeattleDumpster`-Projektinformationen.

```
aws iot1click-projects describe-project --project-name SeattleDumpsters
{
  "project": {
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/
SeattleDumpsters",
    "projectName": "SeattleDumpsters",
    "description": "All dumpsters in the Seattle region.",
    "createdDate": 1563483100,
    "updatedAt": 1563483100,
    "placementTemplate": {
      "defaultAttributes": {
```

```
        "City": "Seattle"
      },
      "deviceTemplates": {
        "empty-dumpster-request": {
          "deviceType": "button",
          "callbackOverrides": {}
        }
      }
    },
    "tags": {}
  }
}
```

Um „Alle Müllcontainer in der Region Seattle“ in „Alle Müllcontainer (HofWasterFühren Sie den folgenden Befehl aus.

```
aws iot1click-projects update-project --project-name SeattleDumpsters --description
  "All dumpsters (yard waste, recycling, garbage) in the Seattle region."
```

Sie können sehen, dass der Wert des "description"-Schlüssels für alle SeattleDumpsters-Platzierungen aktualisiert wurde.

```
aws iot1click-projects describe-project --project-name SeattleDumpsters
{
  "project": {
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
    "projectName": "SeattleDumpsters",
    "description": "All dumpsters (yard waste, recycling, garbage) in the Seattle region.",
    "createdDate": 1563483100,
    "updatedDate": 1563819039,
    "placementTemplate": {
      "defaultAttributes": {
        "City": "Seattle"
      },
      "deviceTemplates": {
        "empty-dumpster-request": {
          "deviceType": "button",
          "callbackOverrides": {}
        }
      }
    }
  }
}
```

```
    },
    "tags": {}
  }
}
```

14. Sie können Tags verwenden, um Metadaten wie folgt auf Projektressourcen (iot1click-projects) und Platzierungsressourcen (iot1click-devices) anzuwenden.

```
aws iot1click-projects tag-resource --cli-input-json file://projects-tag-resource.json
```

Hier enthält `projects-tag-resource.json` Folgendes.

```
{
  "resourceArn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
  "tags": {
    "Account": "45215",
    "Manager": "Tom Jones"
  }
}
```

Führen Sie die folgenden Schritte aus, um die Tags für die Projektressource aufzulisten.

```
aws iot1click-projects list-tags-for-resource --resource-arn "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters"
{
  "tags": {
    "Manager": "Tom Jones",
    "Account": "45215"
  }
}
```

Führen Sie die folgenden Schritte aus, um die Projekt-Tags im Kontext anzuzeigen.

```
aws iot1click-projects describe-project --project-name SeattleDumpsters
{
  "project": {
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
    "projectName": "SeattleDumpsters",
```

```

    "description": "All dumpsters (yard waste, recycling, garbage) in the
Seattle region.",
    "createdDate": 1563483100,
    "updatedAt": 1563819039,
    "placementTemplate": {
      "defaultAttributes": {
        "City": "Seattle"
      },
      "deviceTemplates": {
        "empty-dumpster-request": {
          "deviceType": "button",
          "callbackOverrides": {}
        }
      }
    },
    "tags": {
      "Manager": "Tom Jones",
      "Account": "45215"
    }
  }
}

```

Führen Sie die folgenden Schritte aus, um Amazon-Ressourcennamen (ARN) von Geräten zu ermitteln.

```

aws iot1click-devices list-devices
{
  "Devices": [
    {
      "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789",
      "Attributes": {
        "projectRegion": "us-west-2",
        "projectName": "SeattleDumpsters",
        "placementName": "customer217",
        "deviceTemplateName": "empty-dumpster-request"
      },
      "DeviceId": "G030PM0123456789",
      "Enabled": true,
      "RemainingLife": 99.7,
      "Type": "button",
      "Tags": {}
    }
  ]
}

```

```
]
}
```

Führen Sie folgende Schritte aus, um dem vorherigen Gerät Tags hinzuzufügen.

```
aws iot1click-devices tag-resource --cli-input-json file://devices-tag-
resource.json
```

Hier enthält `devices-tag-resources.json` Folgendes (beachten Sie die erforderliche Schreibweise von `ResourceArn` und `Tags`).

```
{
  "ResourceArn": "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789",
  "Tags": {
    "Driver": "John Smith",
    "Driver Phone": "206-123-4567"
  }
}
```

Führen Sie die folgenden Schritte aus, um die Tags für die Gerätesource aufzulisten.

```
aws iot1click-devices list-tags-for-resource --resource-arn "arn:aws:iot1click:us-
west-2:012345678901:devices/G030PM0123456789"
{
  "Tags": {
    "Driver Phone": "206-123-4567",
    "Driver": "John Smith"
  }
}
```

Um die Geräte-Tags im Kontext anzuzeigen, führen Sie `list-devices` aus.

```
aws iot1click-devices list-devices
{
  "Devices": [
    {
      "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789",
      "Attributes": {
        "projectRegion": "us-west-2",
```



```
        "projectName": "SeattleDumpsters",
        "placementName": "customer217",
        "deviceTemplateName": "empty-dumpster-request"
    },
    "DeviceId": "G030PM0123456789",
    "Enabled": true,
    "RemainingLife": 99.7,
    "Type": "button",
    "Tags": {
        "Driver Phone": "206-123-4567",
        "Driver": "John Smith"
    }
}
]
```

15. An dieser Stelle können Sie eine Aktion mit einem Tastendruck des Geräts verknüpfen, z. B. das Auslösen einer AWS Lambda Funktion oder das Senden einer Amazon SNS S-Nachricht. Sie können dies einfach über die AWS IoT 1-Click Konsole (die [AWS IoT 1-Click-Programmiermodell](#) ist auch eine Option). Nachdem die entsprechenden Aktionen mit dem Gerät verknüpft wurden, können Sie das Gerät an den Standort des Kunden bringen und es mit dem WLAN-Netzwerk verbinden, indem Sie das gleiche Verfahren verwenden, das in den Schritten 1 und 2 beschrieben wird.

## Teardown AWS IoT 1-Click Gerät

In den folgenden Schritten wird beschrieben, wie Sie die vorherigen Schritte rückgängig machen (rückgängig).

1. Führen Sie den folgenden Befehl aus, um die Tags einer Projektressource aufzuheben.

```
aws iot1click-projects untag-resource --resource-arn "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters" --tag-keys "Manager"
```

Dadurch wird das `Manager`-Tag des Projekts entfernt, wie es als Nächstes gezeigt wird.

```
aws iot1click-projects describe-project --project-name SeattleDumpsters
{
  "project": {
    "arn": "arn:aws:iot1click:us-west-2:012345678901:projects/SeattleDumpsters",
```

```

    "projectName": "SeattleDumpsters",
    "description": "All dumpsters (yard waste, recycling, garbage) in the
Seattle region.",
    "createdDate": 1563483100,
    "updatedAt": 1563819039,
    "placementTemplate": {
      "defaultAttributes": {
        "City": "Seattle"
      },
      "deviceTemplates": {
        "empty-dumpster-request": {
          "deviceType": "button",
          "callbackOverrides": {}
        }
      }
    },
    "tags": {
      "Account": "45215"
    }
  }
}

```

2. Führen Sie den folgenden Befehl aus, um die Kennzeichnung einer Gerätesource aufzuheben.

```

aws iot1click-devices untag-resource --resource-arn "arn:aws:iot1click:us-
west-2:012345678901:devices/G030PM0123456789" --tag-keys "Driver Phone" "Driver"

```

Dadurch werden die Tags des Geräts entfernt, wie als Nächstes gezeigt (beachten Sie die leere Liste "Tags": {}).

```

aws iot1click-devices list-devices
{
  "Devices": [
    {
      "Arn": "arn:aws:iot1click:us-west-2:012345678901:devices/
G030PM0123456789",
      "Attributes": {
        "projectRegion": "us-west-2",
        "projectName": "SeattleDumpsters",
        "placementName": "customer217",
        "deviceTemplateName": "empty-dumpster-request"
      }
    }
  ]
}

```

```
    },
    "DeviceId": "G030PM0123456789",
    "Enabled": true,
    "RemainingLife": 99.7,
    "Type": "button",
    "Tags": {}
  }
]
}
```

3. Führen Sie den folgenden Befehl aus, um die Zuordnung eines Geräts von einer Platzierung zu trennen.

```
aws iot1click-projects disassociate-device-from-placement --project-name
  SeattleDumpsters --placement-name customer217 --device-template-name empty-
  dumpster-request
```

Wie Sie im Folgenden sehen können, ist mit der Platzierung `customer217` kein Gerät mehr verknüpft.

```
aws iot1click-projects get-devices-in-placement --project-name SeattleDumpsters --
placement-name customer217
{
  "devices": {}
}
```

4. Führen Sie den folgenden Befehl aus, um eine Platzierung aus einem Projekt zu löschen.

```
aws iot1click-projects delete-placement --project-name SeattleDumpsters --
placement-name customer217
```

Wie Sie im Folgenden sehen können, hat das Projekt `SeattleDumpsters` keine Platzierungen, da die Platzierung `customer217` die einzige Platzierung innerhalb von `SeattleDumpsters` war.

```
aws iot1click-projects list-placements --project-name SeattleDumpsters
{
  "placements": []
}
```

5. Führen Sie den folgenden Befehl aus, um ein Projekt zu löschen.

```
aws iot1click-projects delete-project --project-name SeattleDumpsters
```

Wie im Folgenden ersichtlich, werden alle Projekte entfernt, da `SeattleDumpsters` war das einzige Projekt, das mit Ihrem AWS IoT 1-Click Konto verknüpft ist.

```
aws iot1click-projects list-projects
{
  "projects": []
}
```

Wenn Sie z. B. einen Freund Ihr Gerät mit seinem AWS-Konto ausprobieren lassen möchten, müssen Sie zunächst die Beantragung Ihres Geräts von Ihrem AWS IoT 1-Click-Konto wie folgt aufheben.

```
aws iot1click-devices unclaim-device --device-id G030PM0123456789
{
  "State": "UNCLAIMED"
}
```

Das Gerät kann jetzt mit jedem AWS IoT 1-Click Konto verwendet werden.

# AWS IoT 1-Click Anhang

Dieser Abschnitt enthält weitere AWS IoT 1-Click Informationen, wie nachfolgend beschrieben.


## Unterstützte AWS IoT 1-Click Geräte

Produkt	Gerätetyp	Präfix der Geräte-ID	So beantragen Sie das Gerät	Kauf-Link	Geräte-Region <sup>†</sup>
IoT -Button (USA, EU und Japan)	Button	P5SJVQ (die ersten 6 Stellen der Geräte-ID)	Geben Sie in der mobilen AWS IoT 1-Click App die Geräte-ID ein, um das WLAN zu konfigurieren und das Gerät zu beanspruchen.	<a href="#">Seeed Studio Bazaar</a>	US West (Oregon)
Sercomm IoT -Taste (nur USA)	Button	7VT4EQ (die ersten 6 Stellen der Geräte-ID)	Geben Sie in der AWS IoT 1-Click App oder in der AWS IoT 1-Click Konsole den Einlöschungscode ein, den Sie beim Kauf der Geräte erhalten	<a href="#">Sercomm MobileIoT Produkte Online Shop</a>	US West (Oregon)

Produkt	Gerätetyp	Präfix der Geräte-ID	So beantragen Sie das Gerät	Kauf-Link	Geräte-Region <sup>†</sup>
			haben. Sie können das Gerät auch in Anspruch nehmen, indem Sie die Geräte-ID in der AWS IoT 1-Click Mobil-App eingeben.		
SORACOM LTE-M-Button (nur Japan)	Button	7MF6JK (die ersten 6 Stellen der Geräte-ID)	Geben Sie in der AWS IoT 1-Click App die Geräte-ID ein, um das Gerät zu beantragen.	<a href="#">SORACOM</a>	US West (Oregon)
AWS IoT Enterprise Button (USA, EU und Japan)	Button	G030PM (die ersten 6 Stellen der Geräte-ID)	Geben Sie in der AWS IoT 1-Click App die Geräte-ID ein, um das WLAN zu konfigurieren und das Gerät zu beantragen.	Nicht mehr angeboten	US West (Oregon)

Produkt	Gerätetyp	Präfix der Geräte-ID	So beantragen Sie das Gerät	Kauf-Link	Geräte-Region <sup>†</sup>
AT&T LTE-M-Taste (nur USA)	Button	B9GHXT (die ersten 6 Stellen der Geräte-ID)	Geben Sie in der mobilen AWS IoT 1-Click App oder in der AWS IoT 1-Click Konsole den Einlöschungscode ein, den Sie beim Kauf der Geräte erhalten haben. Sie können das Gerät auch in Anspruch nehmen, indem Sie die Geräte-ID in der AWS IoT 1-Click Mobil-App eingeben.	Nicht mehr angeboten	US West (Oregon)

<sup>†</sup>Weitere Informationen zu Geräte-Regionen finden Sie unter [AWS IoT 1-Click Geräte](#).

 Note

AWS IoT 1-Click unterstützt keine AWS IoT -Buttons, deren Geräteseriennummern (DSN) mit G030JF, G030MD und G030PT beginnen. Weitere Informationen zum Herstellen einer

Verbindung mit diesen Schaltflächen mit der AWS IoT Cloud (ohne Verwendung von AWS IoT 1-Click) finden Sie unter [Cloud Programmierbare Strich-Taste](#).

## AWS IoT 1-Click Beschränkungen

- Sie können maximal 5 Gerätevorlagen pro Platzierungsvorlage verwenden. Dies entspricht 5 Geräten pro Platzierung.
- Maximal 512 AWS IoT 1-Click Projekte pro [AWS-Region](#) Pro Konto.
- Es gibt maximal 50 Tags pro AWS IoT 1-Click Ressource. Tags sind Schlüssel-Wert-Paare (Metadaten), die zum Verwalten einer Ressource verwendet werden können. Weitere Informationen finden Sie unter [AWS Tagging-Strategien](#).



# Dokumentverlauf für das -Entwicklerhandbuch

Die folgende Tabelle beschreibt die Dokumentation für diese Version von AWS IoT 1-Click.

- API-Version: aktuelle
- Letzte Aktualisierung der Dokumentation: 22. Oktober 2018

Änderung	Beschreibung	Datum
Veröffentlichung	Erste Veröffentlichung dieser Dokumentation.	14. Mai 2018
Redaktionell	Redaktionelle Verbesserungen.	31. Mai 2018
Redaktionell	Die Tabelle unterstützter Geräte wurde aktualisiert.	22. Oktober 2018

# AWS-Glossar

Die neueste AWS-Terminologie finden Sie im [AWS-Glossar](#) im AWS-Referenzhandbuch.

Die vorliegende Übersetzung wurde maschinell erstellt. Im Falle eines Konflikts oder eines Widerspruchs zwischen dieser übersetzten Fassung und der englischen Fassung (einschließlich infolge von Verzögerungen bei der Übersetzung) ist die englische Fassung maßgeblich.