

AWS 활용 IoT

[6] AWS IoT : Thing group



강사 : 고병화

Thing grous

Thing group : 사물그룹 이란?

사물 그룹을 사용하면 여러 사물을 그룹으로 분류하여 한 번에 관리할 수 있다.

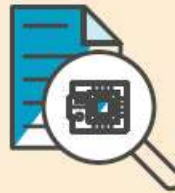
사물 그룹을 사용하면 사물 리소스를 구성하여 디바이스 플릿의 하위 집합을 찾고 작업을 수행할 수 있다. 정적 및 동적 사물 그룹을 생성하여 사용 할 수 있다.



Grouping and Searching for Devices



Organize
into logical
Hierarchies



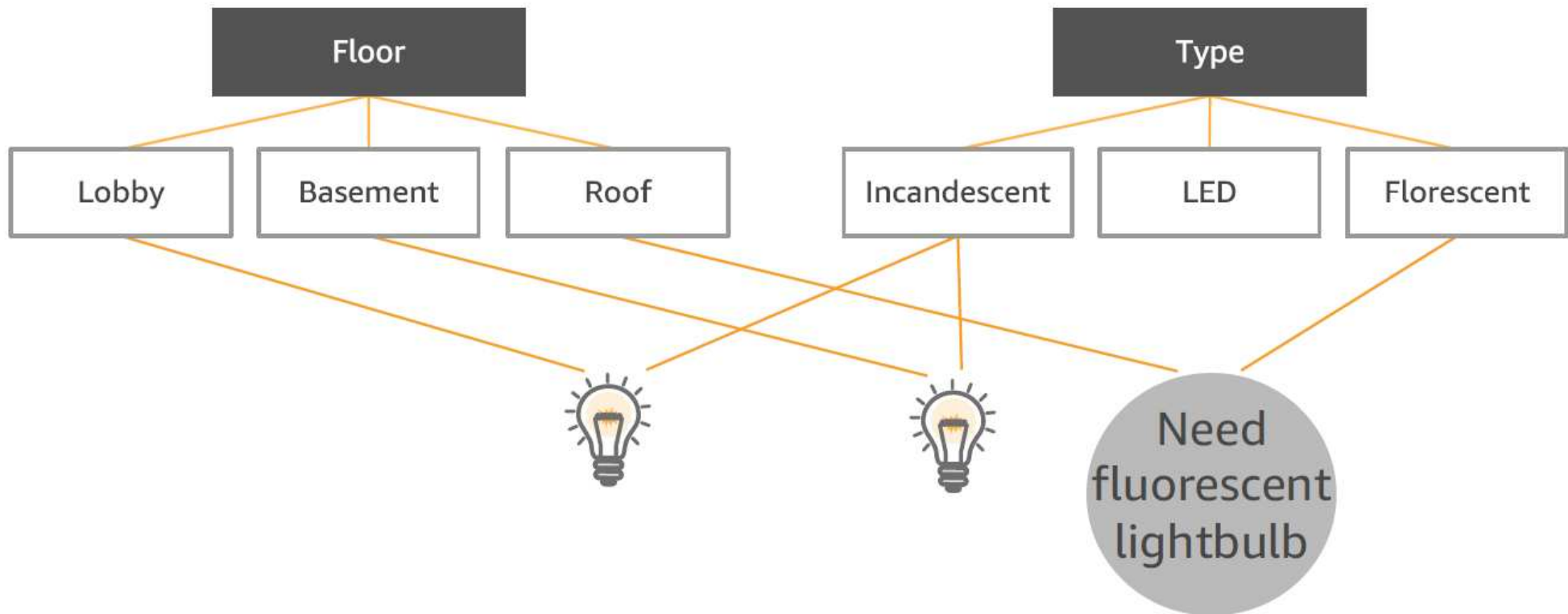
Search Both the
Registry and
Device Shadow



Notification of
Device Changes

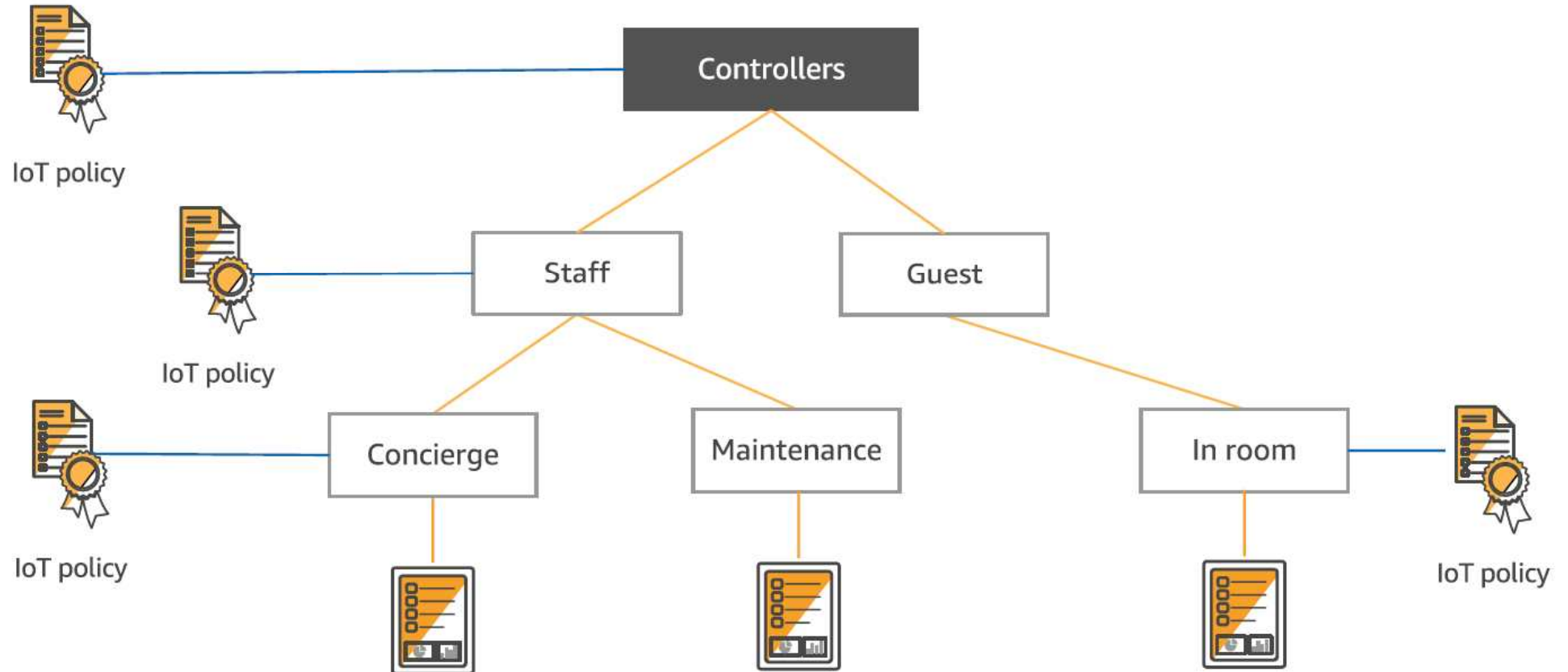


Thing Groups





Thing Group Policies





Thing Group Benefits

- Attaching Policies to groups facilitates device certificate rotation
- Assign jobs to groups instead to single devices
- Set logging level for a thing group. Useful for debugging because global log level remains unchanged

Thing group : 정적 사물 그룹

정적 사물 그룹을 사용하면 사물을 그룹별로 범주화하여 여러 사물을 한번에 관리할 수 있다. 정적 항목 그룹에는 콘솔, CLI 또는 API를 사용하여 관리되는 사물 그룹이 포함된다. 반면에 동적 사물 그룹에는 지정된 쿼리와 일치하는 사물이 포함된다.

정적 사물 그룹은 다른 정적 사물 그룹을 포함할 수 있으므로 그룹 계층 구조를 만들 수 있다. 정책은 상위 그룹에 연결할 수 있으며, 이렇게 연결된 정책은 하위 그룹을 비롯해 해당 그룹과 해당 그룹의 하위 그룹에 속한 모든 사물에게 상속된다. 따라서 사물의 수가 많더라도 권한을 쉽게 제어할 수 있다.

https://docs.aws.amazon.com/ko_kr/iot/latest/developerguide/thing-groups.html

Thing group : 정적 사물 그룹

Cloud9에서 아래 명령을 사용하여 사물 그룹을 생성한다

```
cd ~/provisioning
```

```
THING_GROUP_NAME=building-one
```

```
aws iot create-thing-group \  
  --thing-group-name $THING_GROUP_NAME >  
/tmp/create_group_response
```

```
GROUP_ARN=$(jq -r ".thingGroupArn"  
/tmp/create_group_response)  
ACCOUNT_ID=$(aws sts get-caller-identity | jq -r '.Account')
```

Thing group : 정적 사물 그룹

IoT 정책을 만든다

POLICY_NAME=SmartBuilding_Policy

```
aws iot create-policy --policy-name $POLICY_NAME \
```

```
--policy-document "{\
```

```
  \"Version\": \"2012-10-17\", \
```

```
  \"Statement\": [{ \
```

```
    \"Effect\": \"Allow\",
```

```
    \"Action\": [\"iot:Connect\"], \
```

```
    \"Resource\": [ \
```

```
\"arn:aws:iot:$REGION:$ACCOUNT_ID:client/${iot:ClientId}\" \
```

Thing group : 정적 사물 그룹

```
]\  
,\  
{\  
  "Effect": "Allow",\  
  "Action": ["iot:Publish"],\  
  "Resource": [
```

```
  "arn:aws:iot:$REGION:$ACCOUNT_ID:topic/telemetry/building/on  
e/${iot:ClientId}"\  
]\  
}\  
}"
```

Thing group : 정적 사물 그룹

사물 그룹에 정책을 연결한다

```
aws iot attach-policy \  
  --target $GROUP_ARN \  
  --policy-name $POLICY_NAME
```

정책이 연결되었는지 확인한다

```
aws iot list-attached-policies --target $GROUP_ARN
```

```
aicore0427:~/provisioning $ aws iot list-attached-policies --target $GROUP_ARN  
{  
  "policies": [  
    {  
      "policyName": "SmartBuilding_Policy",  
      "policyArn": "arn:aws:iot:us-east-1:844311781633:policy/SmartBuilding_Policy"  
    }  
  ]  
}
```

Thing group : 정적 사물 그룹

사물을 만들고 키와 인증서를 만든다

```
THING_NAME=group-member
```

```
aws iot create-thing --thing-name $THING_NAME
```

```
aws iot create-keys-and-certificate --set-as-active ₩  
--public-key-outfile $THING_NAME.public.key ₩  
--private-key-outfile $THING_NAME.private.key ₩  
--certificate-pem-outfile $THING_NAME.certificate.pem >  
/tmp/create_cert_and_keys_response
```

Thing group : 정적 사물 그룹

사물을 인증서와 연결한다

```
CERTIFICATE_ARN=$(jq -r ".certificateArn "  
/tmp/create_cert_and_keys_response)
```

```
CERTIFICATE_ID=$(jq -r ".certificateId "  
/tmp/create_cert_and_keys_response)
```

```
aws iot attach-thing-principal --thing-name $THING_NAME #  
--principal $CERTIFICATE_ARN
```

Thing group : 정적 사물 그룹

사물 그룹에 사물 추가

```
aws iot add-thing-to-thing-group \  
  --thing-name $THING_NAME \  
  --thing-group-name $THING_GROUP_NAME
```

list things in your group

```
aws iot list-things-in-thing-group \  
  --thing-group-name $THING_GROUP_NAME
```

```
{  
  "things": [  
    "group-member"  
  ]  
}
```

Thing group : 정적 사물 그룹

MQTT 테스트 클라이언트에서

"telemetry/building/one/#" topic으로 구독을 한다

메시지를 게시

```
mosquitto_pub --cafile ~/root.ca.bundle.pem \  
  --cert $THING_NAME.certificate.pem \  
  --key $THING_NAME.private.key -h $IOT_ENDPOINT -p 8883 \  
  -q 0 -t telemetry/building/one/$THING_NAME -i $THING_NAME --  
  tls-version tlsv1.2 \  
  -m "{\"group\": \"test\", \"date\": \"$(date)\"}" -d
```


Thing group : 정적 사물 그룹

MQTT 테스트 클라이언트에서 아래와 같은 메시지를 확인 할 수 있다

구독	telemetry/building/one/#
telemetry/building/one/#  	<div>▼ telemetry/building/one/group-member</div> <pre>{ "group": "test", "date": "Sat Sep 17 23:31:03 UTC 2022" }</pre>

Thing group : 정적 사물 그룹

MQTT 테스트 클라이언트에서 다음을 주제로 구독을 한다

```
" telemetry/building/one/${THING_NAME}_foo "
```

메시지를 게시

```
mosquitto_pub -cafile ~/root.ca.bundle.pem \  
  --cert $THING_NAME.certificate.pem \  
  --key $THING_NAME.private.key -h $IOT_ENDPOINT -p 8883 \  
  -q 0 -t telemetry/building/one/${THING_NAME}_foo -l  
$THING_NAME -s -version tlsv1.2 \  
-m " {\ " group\ " : \ " test\ " , \ " date\ " : \ " $(date)\ " } " -d
```

`${THING_NAME}_foo`는 등록되지 않았으므로 메시지가 전달되지 않는다

Thing group : 동적 사물 그룹

동적 사물 그룹은 검색 쿼리를 통해 그룹 멤버십을 업데이트한다. 동적 사물 그룹을 사용하여 연결, 레지스트리, 새도우 또는 Device Defender 위반 데이터에 따라 사물과 상호 작용하는 방식을 변경할 수 있습니다. 동적 사물 그룹은 플릿 인덱스에 연결되어 있기 때문에 **동적 사물 그룹을 사용하려면 플릿 인덱싱을 사용하도록 설정해야 한다.** 플릿 인덱싱 검색 쿼리를 사용해 그룹을 생성하기 전에 동적 사물 그룹의 사물을 미리 볼 수 있다.

동적 사물 그룹을 작업(job)의 대상으로 지정할 수 있다.

동적 사물 그룹을 정의하는 기준을 충족하는 사물만 작업을 수행한다.

Thing group : 동적 사물 그룹

동적 사물 그룹은 정적 사물 그룹과 다음과 같은 차이가 있다.

- 사물 멤버십이 명시적으로 정의되지 않는다. 동적 사물 그룹을 생성하려면 그룹 멤버십을 정의하는 [쿼리 문자열](#)을 정의해야 한다.
- 동적 사물 그룹은 계층 구조의 일부일 수 없다.
- **동적 사물 그룹에는 정책을 적용할 수 없다.**
- 동적 사물 그룹은 일련의 다른 명령을 사용해 생성, 업데이트 및 삭제한다. 하지만 그 외 모든 작업의 경우 정적 사물 그룹과 상호 작용하는 데 사용하는 것과 동일한 명령을 사용해 동적 사물 그룹과 상호 작용할 수 있다.

Thing group : 동적 사물 그룹

- 단일 계정에 포함될 수 있는 동적 그룹 수는 제한되어 있다.
- 사물 그룹 이름에 개인 식별 정보를 사용하면 안된다. 사물 그룹 이름은 암호화되지 않은 통신 및 보고서에 나타날 수 있다.
- 사물 그룹 이름에 콜론 문자(:)를 사용하면 안된다. 콜론 문자는 다른 AWS IoT 서비스에서 구분 기호로 사용되며 이로 인해 사물 그룹 이름이 있는 문자열을 잘못 구문 분석할 수 있다.

- 정적 사물 그룹 : 디바이스를 정적 그룹에 직접 등록하여 사용
- 동적 사물 그룹 : 디바이스 등록 시 속성 값으로 자동으로 동적으로 그룹에 속하도록 해서 사용. 플릿 인덱싱을 사용

https://docs.aws.amazon.com/ko_kr/iot/latest/developerguide/dynamic-thing-groups.html

Thing group : 동적 사물 그룹

Cloud9에서 아래 명령을 사용하여 "Dynamic_Building_Day_One" 이름의 동적 사물 그룹을 생성한다

```
aws iot create-dynamic-thing-group \  
  --thing-group-name "Dynamic_Building_Day_One" \  
  --query-string "attributes.building:Day_One"
```

```
{  
  "thingGroupName": "Dynamic_Building_Day_One",  
  "thingGroupArn": "arn:aws:iot:us-east-1:844311781633:thinggroup/Dynamic_Building_Day_One",  
  "thingGroupId": "d2748a64-781e-4c1e-9f1e-193c62cd90d6",  
  "indexName": "AWS_Things",  
  "queryString": "attributes.building:Day_One",  
  "queryVersion": "2017-09-30"  
}
```

Thing group : 동적 사물 그룹

동적 사물 그룹에 대한 정보를 얻는다

```
aws iot describe-thing-group --thing-group-name  
"Dynamic_Building_Day_One"
```

```
{  
  "thingGroupName": "Dynamic_Building_Day_One",  
  "thingGroupId": "d2748a64-781e-4c1e-9f1e-193c62cd90d6",  
  "thingGroupArn": "arn:aws:iot:us-east-1:844311781633:thinggroup/Dynamic_Building_Day_One",  
  "version": 1,  
  "thingGroupProperties": {},  
  "thingGroupMetadata": {  
    "creationDate": "2022-09-18T04:01:17.452000+00:00"  
  },  
  "indexName": "AWS_Things",  
  "queryString": "attributes.building:Day_One",  
  "queryVersion": "2017-09-30",  
  "status": "ACTIVE"  
}
```

Thing group : 동적 사물 그룹

동적 사물 그룹으로 쿼리 될 사물을 생성한다

```
aws iot create-thing --thing-name "dynamic-group-member1" \
  --attribute-payload "{\"attributes\": {\"building\": \"Day_One\"}}"
```

동적 사물 그룹에 속한 사물을 얻기

```
aws iot list-things-in-thing-group --thing-group-name
"Dynamic_Building_Day_One"
```

```
aicore0427:~ $ aws iot list-things-in-thing-group --thing-group-name "Dynamic_Building_Day_One"
{
  "things": [
    "dynamic-group-member1"
  ]
}
aicore0427:~ $ █
```


Thing group : 동적 사물 그룹

사물 생성

```
aws iot create-thing --thing-name "dynamic-group-member2" \  
  --attribute-payload "{\"attributes\": {\"building\": \"Day_Two\"}}"
```

```
aws iot list-things-in-thing-group --thing-group-name  
"Dynamic_Building_Day_One"
```

"dynamic-group-member2" 의 속성 값이 같지 않으므로 동적 그룹에 나타나지 않는다

```
aicore0427:~ $ aws iot list-things-in-thing-group --thing-group-name "Dynamic_Building_Day_One"  
{  
  "things": [  
    "dynamic-group-member1"  
  ]  
}
```

Thing group : 동적 사물 그룹

"dynamic-group-member2" 의 속성 값을 변경해보자

```
aws iot update-thing \  
  --thing-name "dynamic-group-member2" \  
  --attribute-payload "{\"attributes\": {\"building\": \"Day_One\"}}"
```

```
aws iot list-things-in-thing-group --thing-group-name  
"Dynamic_Building_Day_One"
```

```
{  
  "things": [  
    "dynamic-group-member1",  
    "dynamic-group-member2"  
  ]  
}
```

“dynamic-group-member2”의 속성이 동적 그룹의 값으로 변경 되어 동적 그룹의 사물 목록으로 보여진다 (다시 바꾸면 사라진다)

Thing group : 동적 사물 그룹

RaspberryPi의 속성 값을 변경해 본다

```
aws iot update-thing \  
  --thing-name "RaspberryPi" \  
  --attribute-payload "{\"attributes\": {\"building\": \"Day_One\"}}“
```

```
aws iot list-things-in-thing-group --thing-group-name  
"Dynamic_Building_Day_One"
```

```
aicore0427:~ $ aws iot list-things-in-thing-group --thing-group-name "Dynamic_Building_Day_One"  
{  
  "things": [  
    "RaspberryPi",  
    "dynamic-group-member1"  
  ]  
}
```

Thing group : 동적 사물 그룹

AWS IoT Core의 [원격 작업]→[작업]에서 [작업 생성]시 앞에서 만든 동적 사물 그룹을 작업 대상으로 선택해서 생성하면 같은 동적 사물 그룹내의 디바이스를 대상으로 Job을 수행할 수가 있게된다

이 작업을 실행할 사물 그룹

기존 사물 그룹 선택 ▲

Q |사물 그룹 필터링

☒ Dynamic_Building_Day_One

☐ building-one

☐ bulk-group

☐ fleet-provisioning-group

원

로 변환할 수 있습니다. AWS는 작업 템플릿 아래에 몇 가지 퍼블

대기됨

2

작업 실행 (2) 정보

현재 이 작업을 처리 중인 디바이스입니다.

Q 작업 실행 찾기

☐ 사물 이름 ▼

☐ dynamic-group-member1

☐ RaspberryPi

작업 대상 (1) 정보

이 작업이 대상으로 하는 사물 및 사물 그룹입니다.

Q 작업 대상 찾기

이름

Dynamic_Building_Day_One

The End