CookBook Agent

Upload PDF files

Drag and drop files here

Limit 200MB per file • PDF

Browse files

Build Vector Index

Load Vector Index

Enter your query:

provide steps for onboarding a new tenant Track A. Tenant name is appio

Generate

› Retrieved context (top-K)

Editor

Review/Edit (scripts or answers)

```
POLICY_FILE="./files/iot_policy.json"
IOT_POLICY_NAME="${THING_NAME}-policy"
AWS_PRIMARY_REGION="us-east-1"
AWS_PROFILE="default"
AWS_ACCOUNT_ID="247590354562"
KINESIS_STREAM="${TENANT_ID}-KS"
DDB_TABLE="${TENANT_ID}-DDBT"
S3_DATALAKE_BUCKET="${TENANT_ID}bucket"
SNS_TOPIC_NAME="${TENANT_ID}topic"

# create the policy (first time)
aws iot create-policy \
  --policy-name "$IOT_POLICY_NAME" \
  --policy-document file://"$POLICY_FILE" \
  --region "$AWS_PRIMARY_REGION" --profile "$AWS_PROFILE"

aws iot create-thing --thing-name "$THING_NAME"   --region "$AWS_PRIMARY_REGION" --
profile "$AWS_PROFILE"

aws iot create-keys-and-certificate --set-as-active \
  --certificate-pem-outfile "cert-${THING_NAME}.pem" \
  --public-key-outfile "public-${THING_NAME}.key" \
  --private-key-outfile "private-${THING_NAME}.key" \
  --region "$AWS_PRIMARY_REGION" \
  --profile "$AWS_PROFILE" \
  --output json \
  --query '{certificateArn:certificateArn,certificateId:certificateId}' \
   | tee cert-output.json

export CERT_ARN=$(jq -r '.certificateArn' cert-output.json)
export CERT_ID=$(jq -r '.certificateId' cert-output.json)

# lock down file perms
chmod 600 "private-${THING_NAME}.key"
chmod 644 "cert-${THING_NAME}.pem" "public-${THING_NAME}.key"



# attach the policy to your certificate
aws iot attach-policy \
  --policy-name "$IOT_POLICY_NAME" \
  --target "$CERT_ARN" \
  --region "$AWS_PRIMARY_REGION" --profile "$AWS_PROFILE"

aws iot attach-thing-principal --thing-name "$THING_NAME" --principal "$CERT_ARN" \
           $AWS_PRIMARY_REGION" --profile "$A

curl -fsSL -o AmazonRootCA1.pem
```

Save/OK

Deploy/Run

```
https://www.amazontrust.com/repository/AmazonRootCA1.pem
# Tenant data plane
aws kinesis create-stream --stream-name "$KINESIS_STREAM" --shard-count 1 \
  --region "$AWS_PRIMARY_REGION" --profile "$AWS_PROFILE" || true

aws dynamodb create-table --table-name "$DDB_TABLE" \
  --attribute-definitions AttributeName=pk,AttributeType=S AttributeName=sk,AttributeType=S \
  --key-schema AttributeName=pk,KeyType=HASH AttributeName=sk,KeyType=RANGE \
  --billing-mode PAY_PER_REQUEST --region "$AWS_PRIMARY_REGION" --profile "$AWS_PROFILE" || true

aws sns create-topic --name "$SNS_TOPIC_NAME" --region "$AWS_PRIMARY_REGION" --profile "$AWS_PROFILE"

# IoT Rule → Kinesis (+optional Lambda for scoring)
```

💾 Saved to `scripts/shell/artifact_20250826092004.sh`

Logs

🖥️ Executing script: scripts/shell/artifact_20250826092004.sh

🔧 STDOUT: { "policyName": "appio-device1-policy", "policyArn": "arn:aws:iot:us-east-1:247590354562:policy/appio-device1-policy", "policyDocument": "{\n "Version": "2012-10-17",\n "Statement": [\n {\n "Effect": "Allow",\n "Action": [\n "iot:Connect",\n "iot:Publish",\n "iot:Subscribe",\n "iot:Receive"\n],\n "Resource": [\n "arn:aws:iot:us-east-1:247590354562:client/*",\n "arn:aws:iot:us-east-1:247590354562:topic/*",\n "arn:aws:iot:us-east-1:247590354562:topicfilter/*"\n]\n }\n]\n}\n", "policyVersionId": "1" } { "thingName": "appio-device1", "thingArn": "arn:aws:iot:us-east-1:247590354562:thing/appio-device1", "thingId": "7ebad6a4-62b7-46d2-9ad1-6edda15bcfe6" } { "certificateArn": "arn:aws:iot:us-east-1:247590354562:cert/2b438f57140d0b1540290b2c2a16a22b040ff1fffd829450ee93d2d85323e58c", "certificateId": "2b438f57140d0b1540290b2c2a16a22b040ff1fffd829450ee93d2d85323e58c" } { "TableDescription": { "AttributeDefinitions": [{ "AttributeName": "pk", "AttributeType": "S" }, { "AttributeName": "sk", "AttributeType": "S" }], "TableName": "appio-DDBT", "KeySchema": [{ "AttributeName": "pk", "KeyType": "HASH" }, { "AttributeName": "sk", "KeyType": "RANGE" }], "TableStatus": "CREATING", "CreationDateTime": "2025-08-26T09:20:16.198000+05:30", "ProvisionedThroughput": { "NumberOfDecreasesToday": 0, "ReadCapacityUnits": 0, "WriteCapacityUnits": 0 }, "TableSizeBytes": 0, "ItemCount": 0, "TableArn": "arn:aws:dynamodb:us-east-1:247590354562:table/appio-DDBT", "TableId": "d2ad691d-6ff9-422a-9750-1cc5dec36239", "BillingModeSummary": { "BillingMode": "PAY_PER_REQUEST" }, "DeletionProtectionEnabled": false } } { "TopicArn": "arn:aws:sns:us-east-1:247590354562:appiotopic" }

✅ Completed successfully.

Chat History (newest first)

⌄ provide steps for onboarding a new tenant Track A. Tenant name is appio — intent: answer → generic

```
7) Tenant Onboarding
7A) Track A — IoT Things (no Greengrass)
# Thing + certs
THING_NAME="${TENANT_ID}-device1"
aws iot create-thing --thing-name "$THING_NAME" --region "$AWS_PRIMARY_REGIC
aws iot create-keys-and-certificate --set-as-active \
 --certificate-pem-outfile cert-${THING_NAME}.pem \
 --public-key-outfile public-${THING_NAME}.key \
 --private-key-outfil
```

```
17.2) Validation of an onboarded tenant
Run these checks end-to-end. Do them once and save outputs in your onboardir
17.2.1) Connectivity & identity
# Thing & cert presence
aws iot describe-thing --thing-name "$THING_NAME" \
 --region "$AWS_REGION_PRIMARY" --profile "$AWS_PROFILE"
aws iot list-thing-principals --thing-name "$THING_NAME" \
 --region "$AWS_REGION_PRIMARY" --profile "$
```

```
Security & guardrails
• Verify the IoT policy only allows topics under tenants/${TENANT_ID}/... ar
= ${THING_NAME}.
• If Greengrass: ensure the role alias maps to an IAM role limited to requir
(S3/Kinesis only, least-priv).
• Tag resources: tenant=${TENANT_ID}, env=prod|stage.
DR readiness (optional in onboarding)
• If using dual-publish firmware: publish to DR endpoint t
```

```
3.2 AWS account & profile
aws configure --profile default
3.3 Inputs (set once per session)
Before deployment, set the cli environment using below script -
export AWS_PROFILE="default"
export AWS_PRIMARY_REGION="us-east-1"
export AWS_DR_REGION="us-west-2"
export AWS_ACCOUNT_ID="<12-digit>"
export TENANT_ID="tenant001"
# Common resource names (per-tenant patterns encouraged)
export KINESIS
```

⌄ what are the aws resources required for new tenant onboarding — intent: answer → generic

```
10) Security, Patching, Upgrades & Notifications
Edge devices: monthly OS updates via Ansible (apt/yum).
Containers: pull latest base images monthly; rebuild.
IoT policies: least privilege—restrict to tenants/${TENANT_ID}/** and
client/${THING_NAME}.
```

```
Notifications: SNS topics to Slack/PagerDuty; acknowledgement required.
Secrets: store in SSM/Secrets Manager; rotate on schedule.

17.2) Validation of an onboarded tenant
Run these checks end-to-end. Do them once and save outputs in your onboardir
17.2.1) Connectivity & identity
# Thing & cert presence
aws iot describe-thing --thing-name "$THING_NAME" \
 --region "$AWS_REGION_PRIMARY" --profile "$AWS_PROFILE"
aws iot list-thing-principals --thing-name "$THING_NAME" \
 --region "$AWS_REGION_PRIMARY" --profile "$
```

```
Index
1. Purpose, Scope, Non-Goals
2. Reference Architecture & Data Flow
3. Preconditions & Inputs (parameterized)
4. Golden Path (Day-0/1/2 overview)
5. Primary Deployment (CLI-first)
6. DR / Replica Deployment (CLI-first)
7. Tenant Onboarding
7A) Using IoT Things (no Greengrass)
7B) Using AWS IoT Greengrass (edge)
8. Post-Deployment Validation
9. Monitoring & Observability (SLOs, alerts)
10. Se
```

```
Security & guardrails
• Verify the IoT policy only allows topics under tenants/${TENANT_ID}/... ar
= ${THING_NAME}.
• If Greengrass: ensure the role alias maps to an IAM role limited to requir
(S3/Kinesis only, least-priv).
• Tag resources: tenant=${TENANT_ID}, env=prod|stage.
DR readiness (optional in onboarding)
• If using dual-publish firmware: publish to DR endpoint t
```

This app always searches the vector DB first. If no high-similarity match is found, it generates grounded artifacts using retrieved context. CLI execution is gated by a safety toggle.