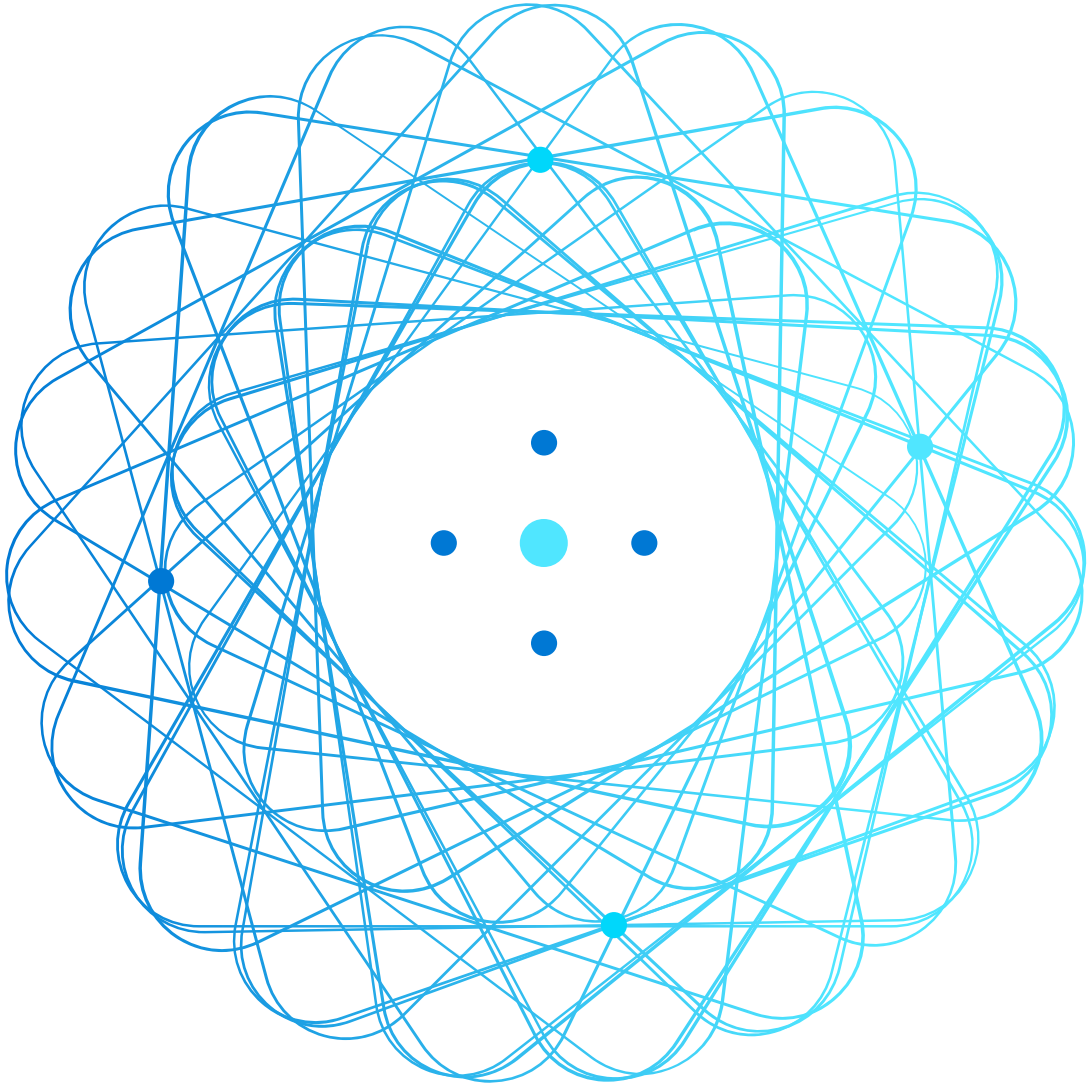
Microsoft Azure

AZ-220T01
Module 03: Device provisioning at scale

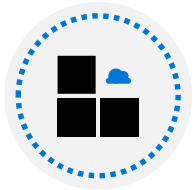


Lesson 1: Learning objectives

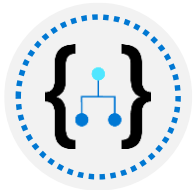
Module 3 – Learning objectives

Explain the process of device provisioning and the features of the Device Provisioning Service

Explain the security considerations associated with device provisioning and how they are managed



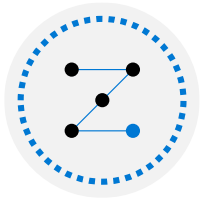
Implement the Device Provisioning Service SDKs



Manage the device enrollment process, including deprovisioning and disenrollment

Lesson 2: Device provisioning service terms and concepts

Devices and device provisioning



Provisioning Process:

Manufacturing Phase: When the device is created and prepared at the factory

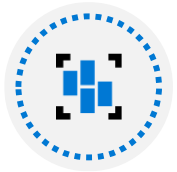
Cloud Setup Phase: When the Device Provisioning Service is configured for automatic provisioning

Registration vs. *Provisioning:*

Registration: adding a device to the cloud provider's device list (registry)

Provisioning: Configuring the device with a desired configuration after registration; DPS does this through an initial device twin that is copied to the identity registry on the associated IoT Hub

Features of the Device Provisioning Service (DPS)



Secure attestation support (X.509 and TPM-based identities)

A configurable, updatable enrollment list containing the complete record of devices/ groups of devices that may at some point register

Multi-hub support (including across subscriptions and regions), assigned by multiple allocation policies

Monitoring and diagnostics logging to make sure everything is working properly

Cross-platform support:

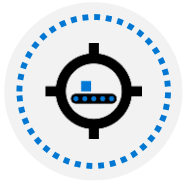
A variety of operating systems

SDKs across multiple languages

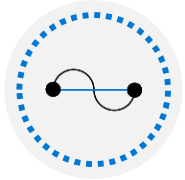
HTTPS, AMQP, and MQTT protocol support (Service SDK is HTTPS only)

When to use the Device Provisioning Service

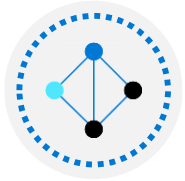
If you want any of these capabilities, the DPS is a good choice:



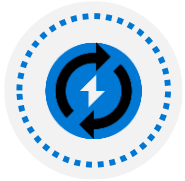
Zero-touch provisioning



Load balancing



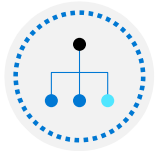
Connecting devices (multitenancy, solution isolation, geo-sharding)



Reprovisioning

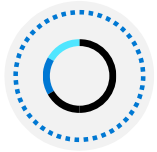
Rolling keys

Service concepts



Service operations endpoint – used for managing DPS and the enrollment list

Device provisioning endpoint – single address used for all provisioning, shared across all customers and DPS instances

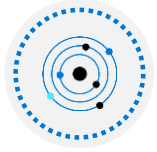


Linked IoT hubs – target Azure IoT Hub instances for the DPS

Allocation policy – as previously mentioned, the mapping of device to target Azure IoT Hub

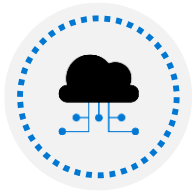
Enrollment – the record of a device or group of devices that may register against the DPS

Registration – the record of a successful registration/provisioning of a device



Operations – the billing unit for DPS; one successfully completed request

Device enrollment concepts



ID scope – differentiates different DPS instances and tenants at the fixed, shared target endpoints

Registration ID – uniquely identifies a device in the DPS instance

Device ID – uniquely identifies a device in the associated IoT Hub instance

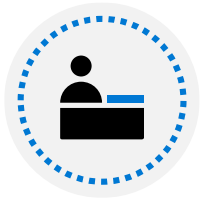
Attestation mechanism – the way a device proves its identity to the DPS:

X.509 Certificates

TPM nonce challenge

Symmetric key

Device enrollment types



Individual Enrollments:

An Individual enrollment is an entry for a single device that may register. Individual enrollments may use either X.509 certificates or SAS tokens (from a physical or virtual TPM) as attestation mechanisms

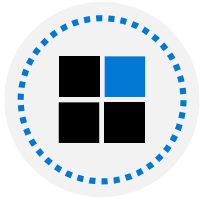
Group Enrollments:

An Enrollment group is an entry for a group of devices that share a common attestation mechanism of X.509 certificates, signed by the same signing certificate, which can be the root certificate or the intermediate certificate, used to produce device certificate on physical device

Security concepts: Certificates

X.509 certificates – digital identity based on private/public key pairs and a chain of trust

Issued by a certificate authority (CA)



Certificate rules for DPS:

Chain must be trusted

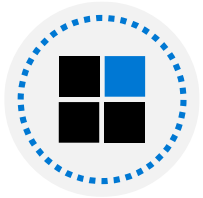
Group or individual enrollment

Individual overrides group

Security concepts: Hardware/TPM attestation

Hardware security module (HSM) – used for secure, hardware-based storage of device secrets

Trusted Platform Module (TPM) – a specification for storing keys or the interface for communicating with an HSM acting as a TPM



Two hardware keys for the TPM:

Endorsement key (EK) – unique identifier for the TPM; read-only, injected by the manufacturer

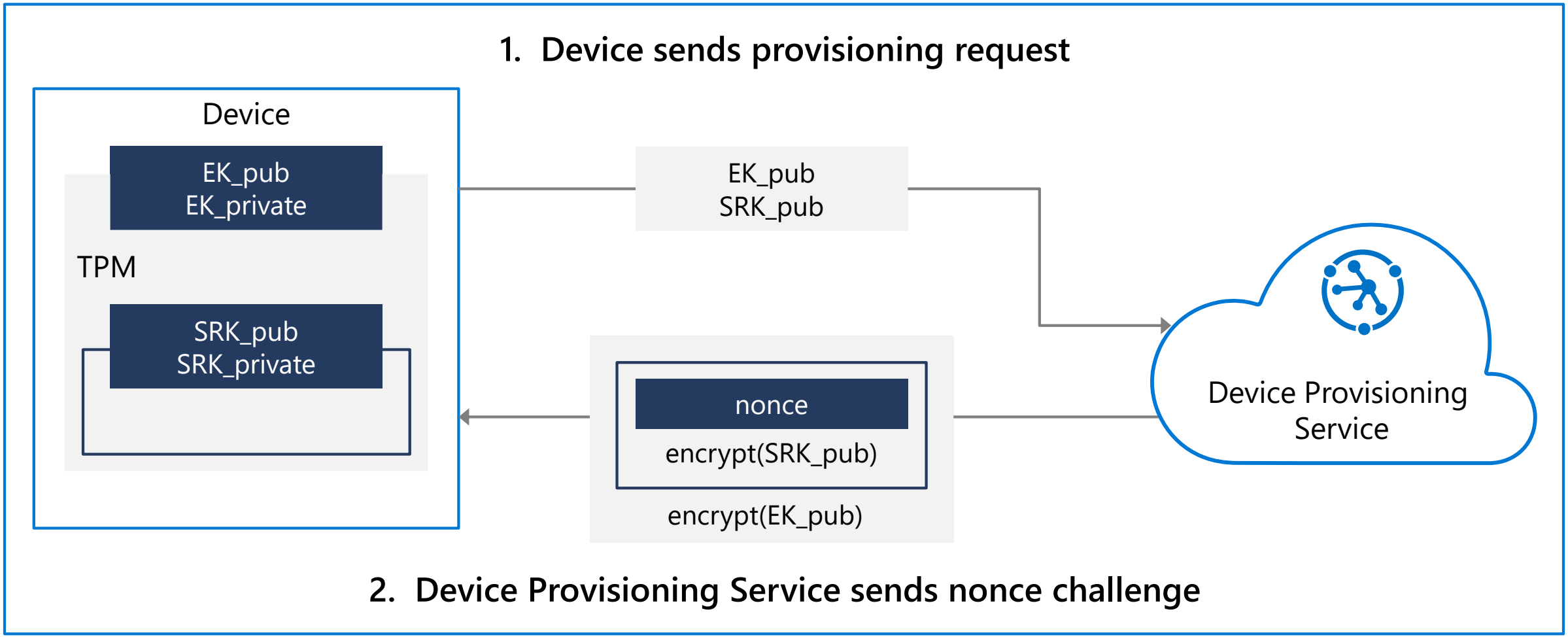
Storage root key (SRK) – protects the TPM secrets; generated when a user takes ownership of the TPM

TPM attestation

Attestation process overview – endorsement and storage root keys

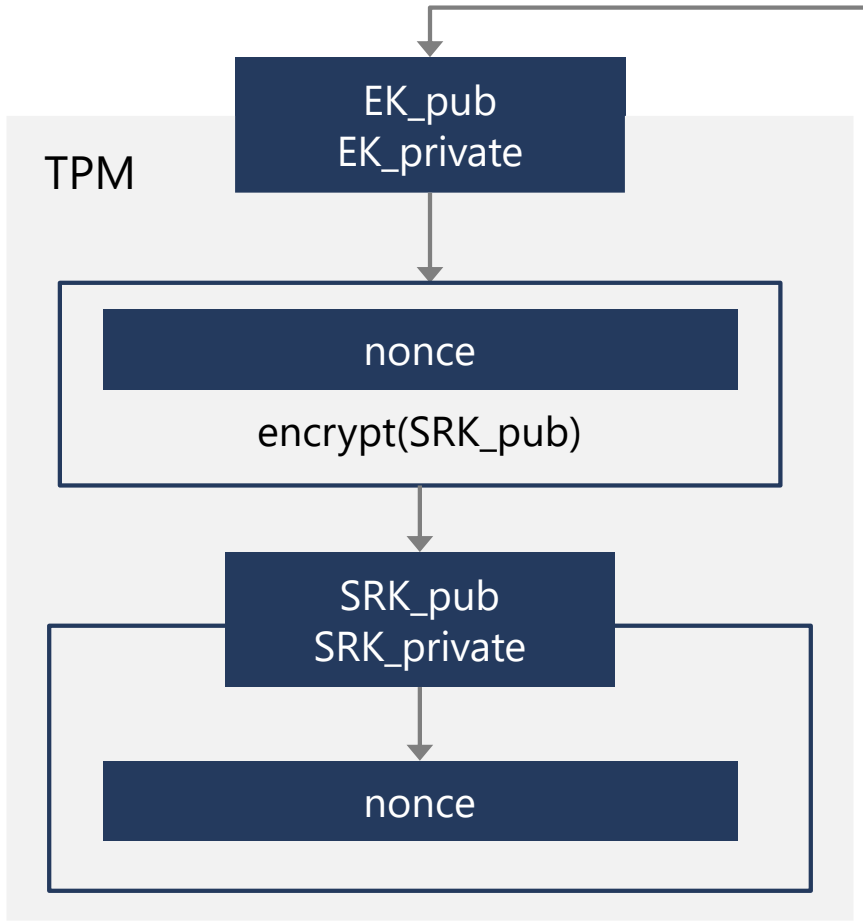
TPM attestation (step 1)

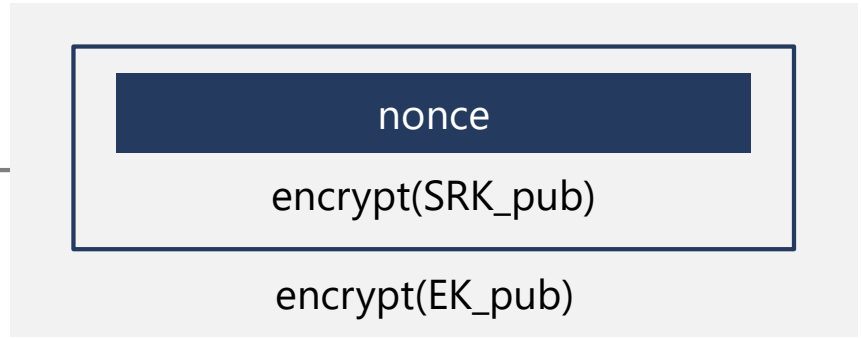
Attestation process details, Step 1 – Request provisioning



TPM attestation (step 2)

Attestation process details, Step 2 – Nonce challenge





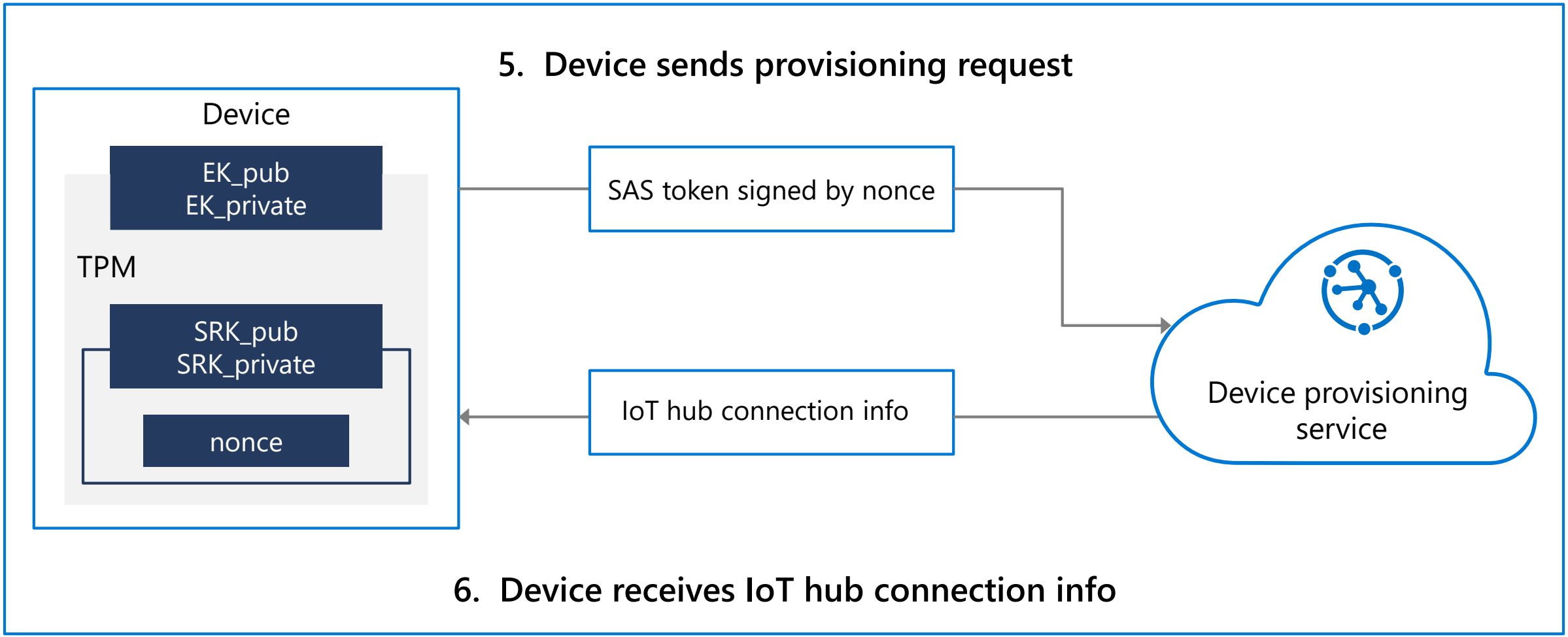
3. Decrypt outer wrapper using EK

4. Decrypt remaining wrapper using SRK

Nonce is now stored un TRM

TPM attestation (step 3)

Attestation process details, Step 3 – Validation

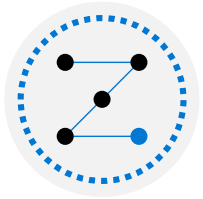


Security concepts: Symmetric key attestation

Symmetric Key Creation:

Automatic

Manual



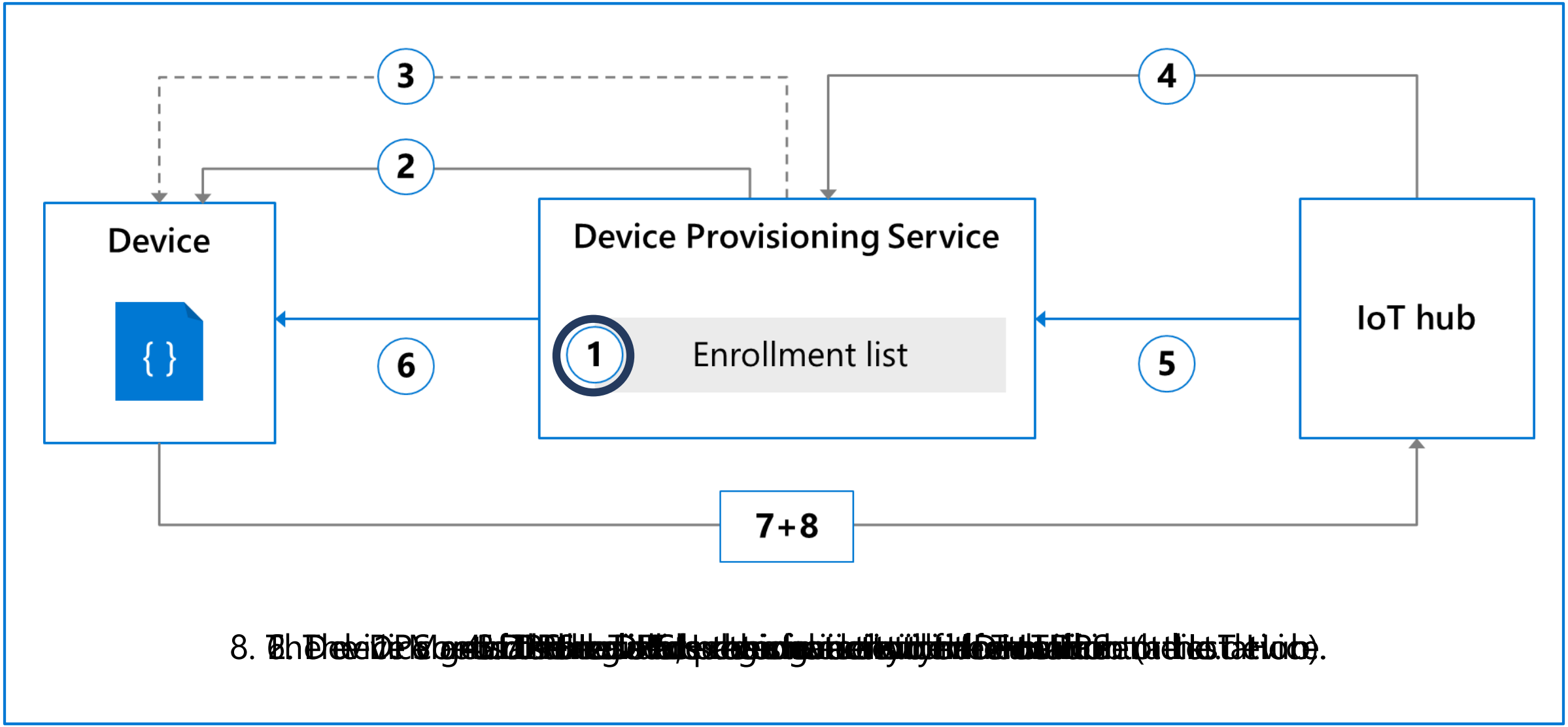
Attestation Process:

Tied to a SAS token based on the key

SDKs handle the SAS token when you supply the key

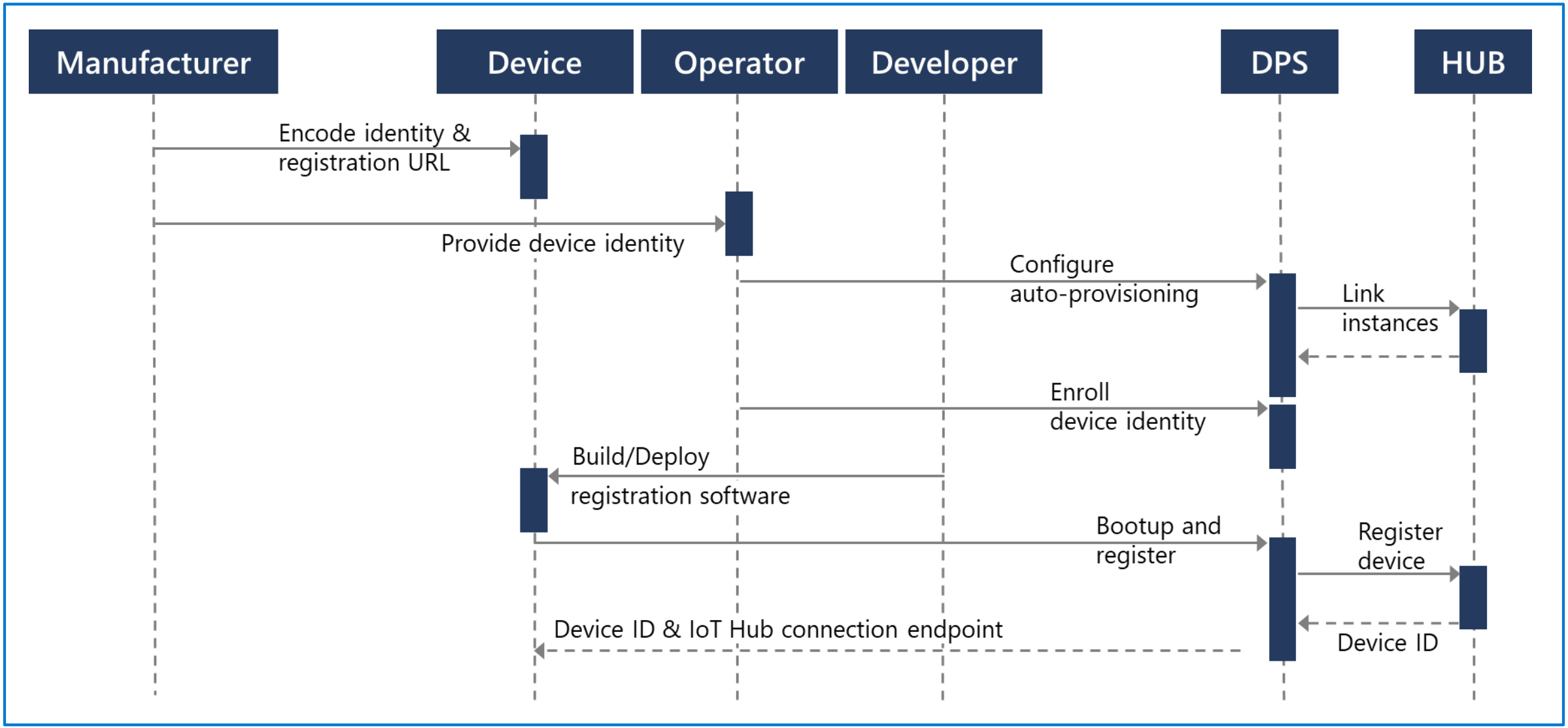
More details later around exactly how the SAS token works, because we'll see the concept again

DPS auto-provisioning behind the scenes



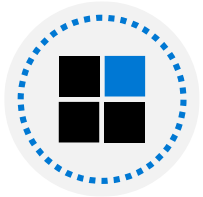
8. [illegible]

Sample auto-provisioning roles and responsibilities



Introduction to reprovisioning

Reprovisioning – the process of associating a device with a different Azure IoT Hub



Reasons for reprovisioning devices:

Geolocation/GeoLatency – a device is moved to a new location

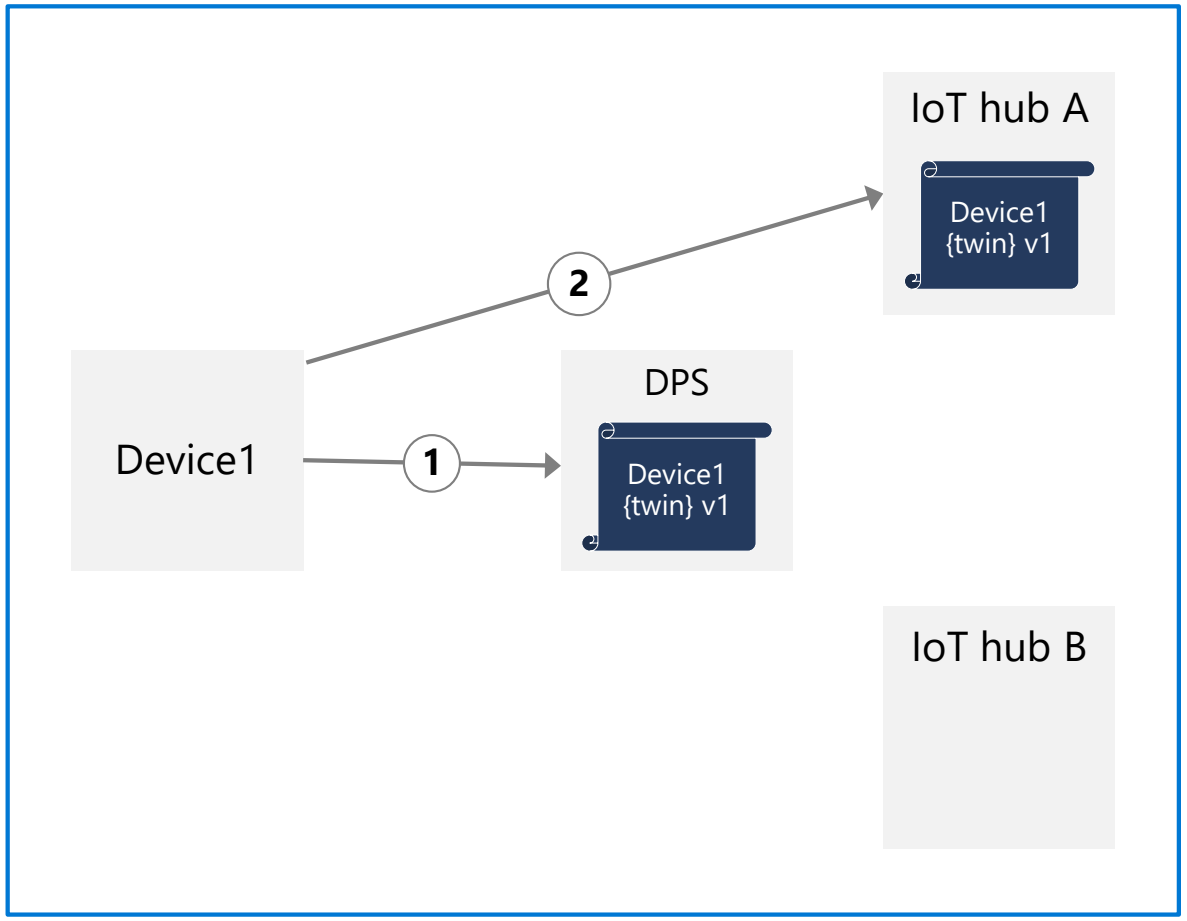
Multi-tenancy – Different customer, customer site, etc.

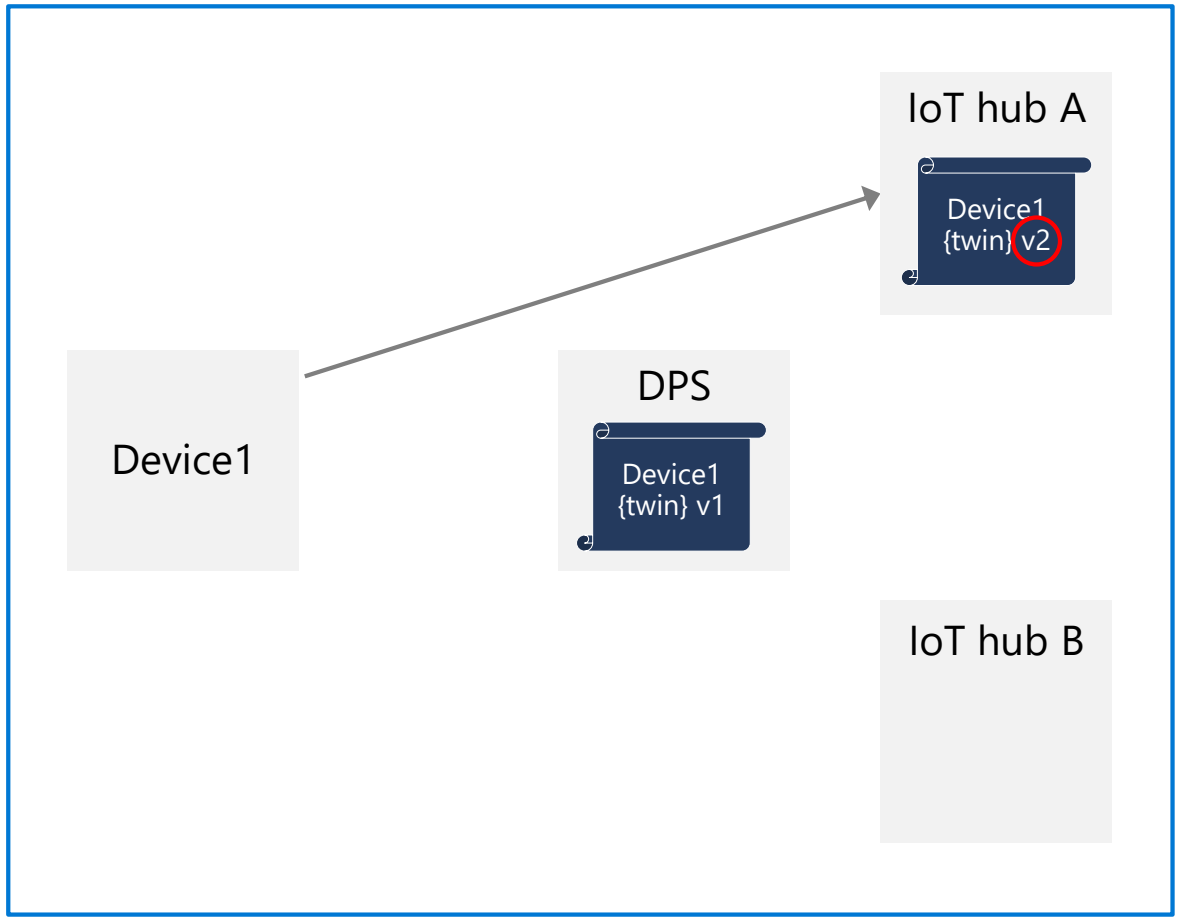
Solution change – choosing to move to a new solution, say with different back-end services connected

Quarantine – moving to an Azure IoT Hub that can fix the configuration or correct the compromise

Provisioning with device state in a Device Twin

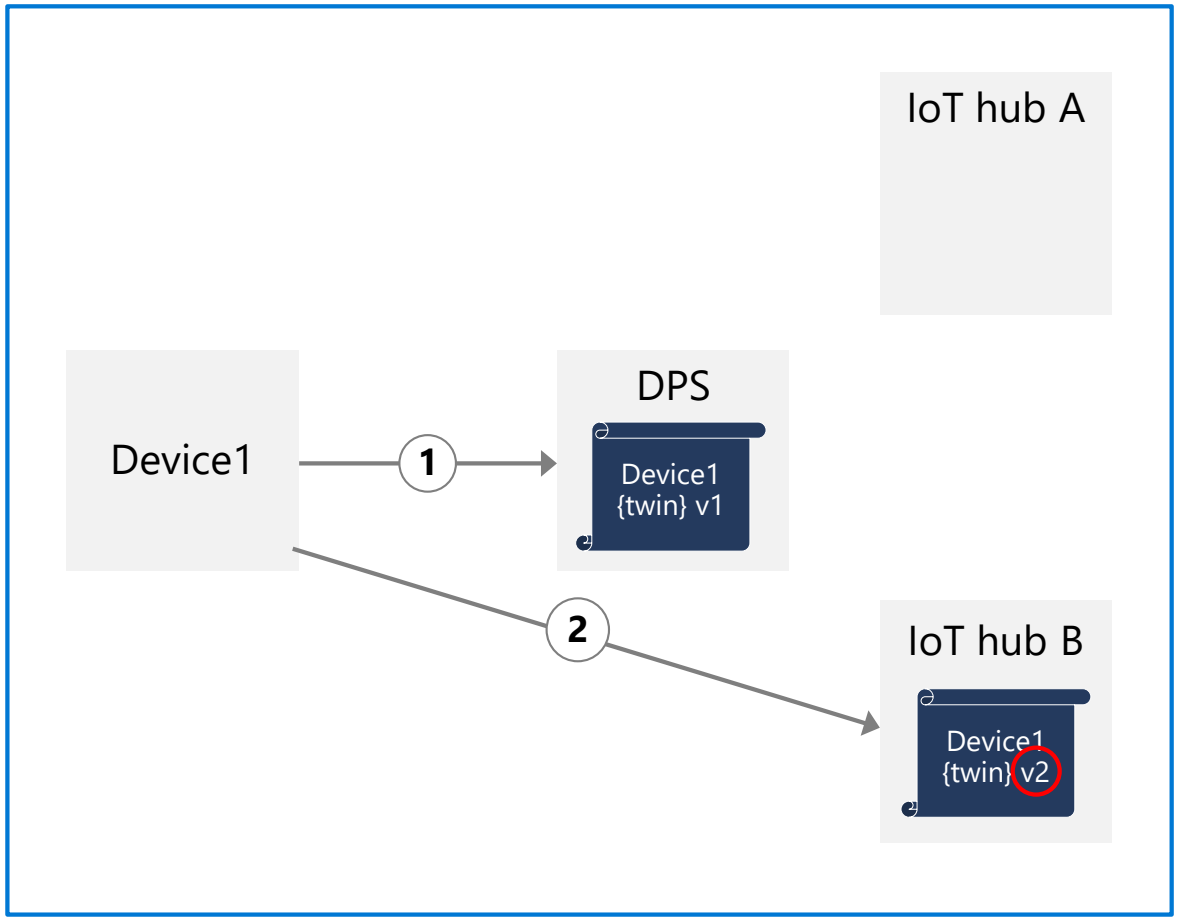
Device state data



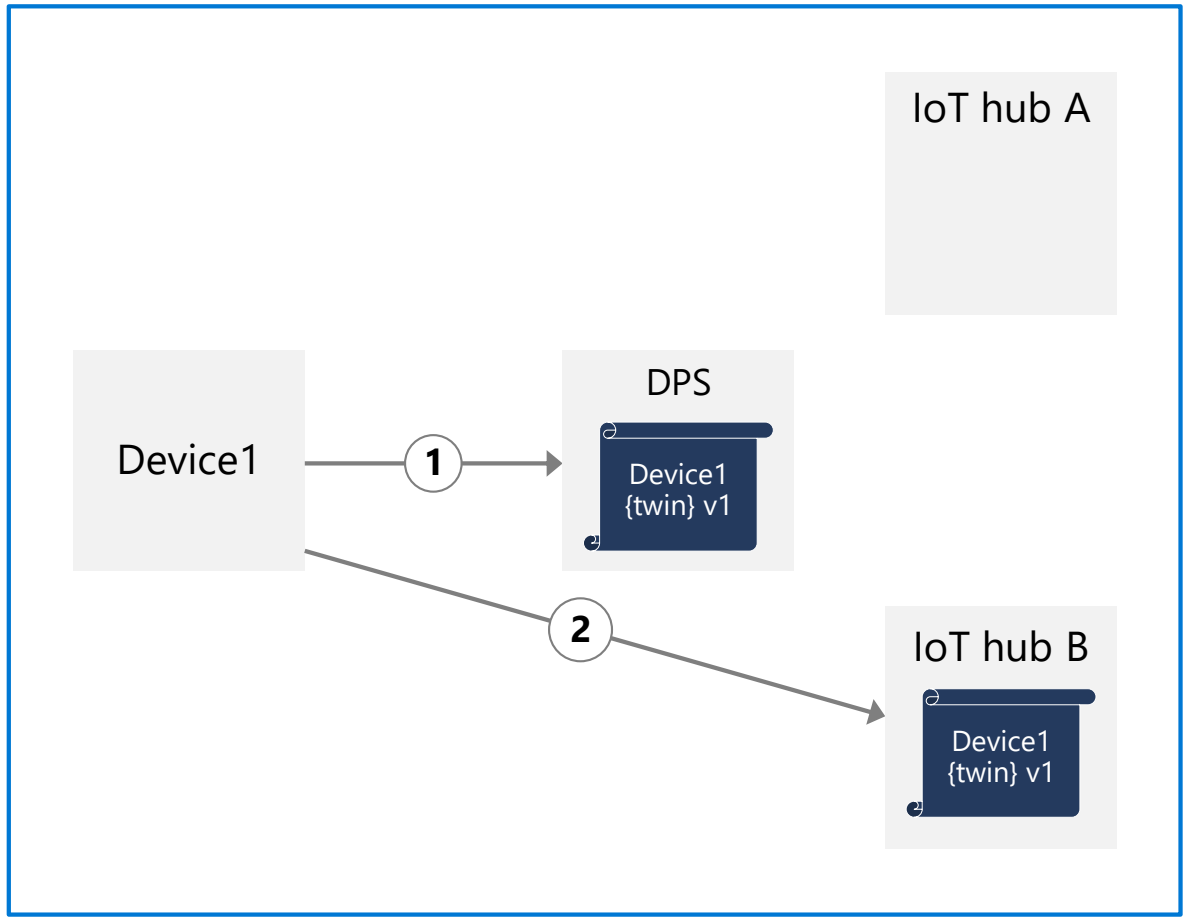


Reprovisioning

Reprovisioning policies



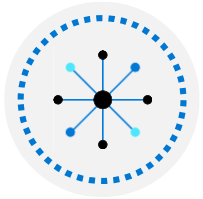
Reprovision and migrate data



Reprovision and reset to initial config

Lesson 3: Configure and manage the device provisioning service

Azure CLI support for device provisioning



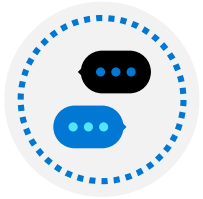
Azure CLI Commands for DPS:

Service Commands

Access Policy Commands

Certificate Commands

Linked Hub Commands

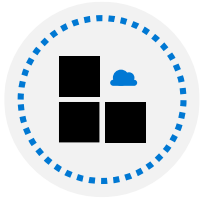


Using the DPS Service Commands:

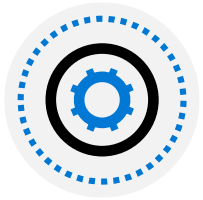
Create: `az iot dps create --name MyDps --resource-group MyResourceGroup`

Delete: `az iot dps delete --name MyDps --resource-group MyResourceGroup`

Device provisioning service SDKs



Provisioning Device Client SDK – enables you to build apps that run on your IoT devices to communicate with the Device Provisioning Service



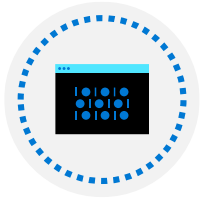
Provisioning Service Client SDK – enables you to build backend applications to manage your enrollments in the Device Provisioning Service

Additional Tools:

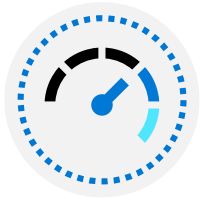
Trusted Platform Module (TPM) simulator

X.509 certificate generator

Control access to DPS



Access control and permissions – via authorization policies

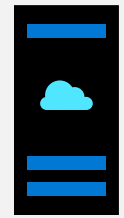


Authentication – using a SAS token against a named policy

Security tokens – previously mentioned structured URL components used for authentication

Device Provisioning Service permissions – rolled up into authorization policies

Lesson 04: Device provisioning tasks



Device enrollment processes and tools

Create – Adding enrollments to DPS

Update – Changing enrollments in DPS

Remove – Preventing future device registrations against that enrollment through DPS

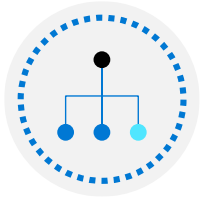
Portal

CLI

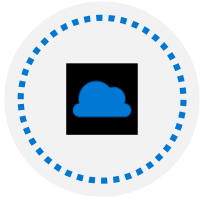
SDK

Configure verified CA certificates

When you register a CA certificate in DPS as a parent certificate for device certificates, DPS does not immediately trust the certificate

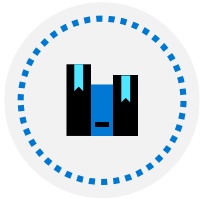


Certificate proof-of-possession is the process to enable DPS's trust



This works through having you issue a child certificate with a specific subject and signed with a specific value, chained to the certificate uploaded to DPS

DPS generates a random subject and signing value for you to use in the verification certificate – this is referred to as the *verification code*



The SDKs include sample scripts to help with this process

Rolling device certificates – reasons

Reasons to roll certificates:

Compromise – will cover more on this specific scenario later in the course

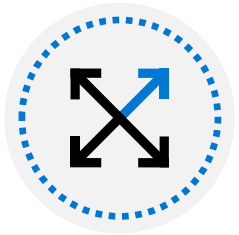
Expiration – a standard certificate management issue

Rolling device certificates – process

Process for rolling a device certificate:

1. Obtain new certificates – this will depend on your initial source for your certificates
2. Roll the certificate in the IoT hub – this will allow the device to be recognized with the new certificate
3. Roll the certificate in the DPS configuration
4. Roll the certificate on the device

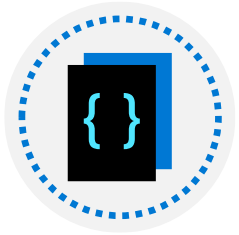
Deprovisioning process



Disenrollment

Deregister

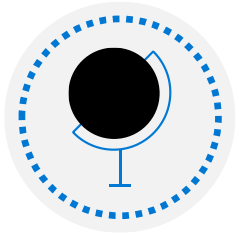
Manage disenrollment



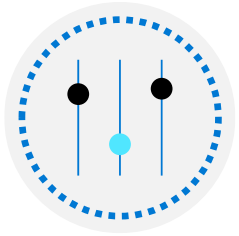
Block list individual devices

Block list an enrollment group

Provision for multitenancy

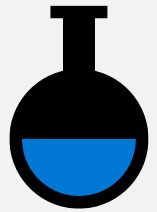


Geolocation/Geo-latency scenario

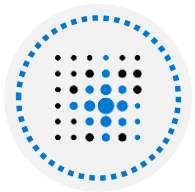


Multi-tenancy scenario

Lesson 5: Module 3 labs



Module 3 labs

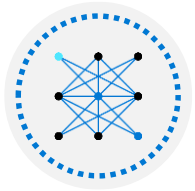


Lab 5: Individual Enrollment of Devices in DPS:

You will create a new individual enrollment using Symmetric key attestation

You will configure a simulated device using your individual enrollment

You will retire the device from both the Device Provisioning Service (DPS) and Azure IoT Hub



Lab 6: Automatic Enrollment of Devices in DPS:

You will generate an x.509 CA Certificate using OpenSSL within the Azure Cloud Shell

You will use the x.509 CA Certificate to configure the Group Enrollment within the Device Provisioning Service (DPS)

You will complete the automatic enrollment of a simulated device

You will retire the enrollment group

Lesson 6: Module 3 review questions

Module review: Question 3.1

A company will be using the IoT Hub Device Provisioning Service (DPS) to provision a large number of devices. They will begin by creating a plan for device enrollments.

What is device enrollment?

Answer A:
Device enrollment is a DPS-enabled process for automatically adding a single device or group of devices to the IoT Hub identity registry.

Answer B:
Device enrollment is a DPS-enabled process for creating a record of a single device or a group of devices that may at some point be registered with IoT Hub.

Answer C:
Device enrollment is a DPS-enabled process for creating a record of the devices connected to IoT Hub using the X.509 attestation mechanism.

Module review: Question 3.2

Which of the following answers correctly describes attestation and the attestation mechanisms supported by the IoT Hub Device Provisioning Service (DPS)?

Answer A:
Attestation is a process for linking DPS to one or more IoT hubs.

Answer B:
Attestation is a process for confirming a device's identity.

Answer C:
The attestation mechanisms supported by DPS are Symmetric Key and TPM.

Module review: Question 3.3

A company is investigating the IoT Hub Device Provisioning Service as an option for provisioning devices.

Which of the following answer choices accurately describes support provided by the Device Provisioning Service?

Answer A:
Provides multi-hub support that allows the Device Provisioning Service to assign devices to more than one IoT hub and region.

Answer B:
Provides enrollment lists for the devices that are registered with each specific IoT Hub for a subscription.

Answer C:
Provides support for removing IoT and IoT Edge devices from the IoT hub registry for each IoT hub in a subscription.

Module review: Question 3.4

Which of the following answer choices accurately describes the support provided by the IoT Hub Device Provisioning Service SDKs?

Answer A:

The Device Provisioning Service SDKs provide support for the following programming languages: C, Java, Python, Go, Swift

Answer B:

The Device Provisioning Service SDKs provide support for managing the provisioning service and for managing device provisioning.

Answer C:

The Device Provisioning Service SDKs provide support for enrollments using X.509 certificates but do not support Trusted Platform Module (TPM).

Module review: Question 3.5

Which of the following Azure CLI instructions should be used to create an instance of the IoT Hub Device Provisioning Service?

Answer A:

Enter: az dps create -name MyDps --resource-group MyResourceGroup --location westus2

Answer B:

Enter: az iotdps create -name MyDps --resource-group MyResourceGroup

Answer C:

Enter: az iot dps create -name MyDps --resource-group MyResourceGroup

Module review: Question 3.6

A company is in the process of deploying 700 devices when their plans change. Device enrollments have already been completed for all 700 devices. There are 500 devices that have been provisioned, but those devices must be moved to a different IoT Hub. The other 200 devices will not be provisioned due to a change in business requirements.

Which of the following answer choices describes the initial actions that must be taken?

Answer A:

Disenroll the devices that have not been provisioned. Leave the other enrollments unchanged.

Answer B:

Disenroll and deregister all 700 devices.

Answer C:

Disenroll all 700 devices. Deregister the 500 provisioned devices.

Module review: Question 3.7

A company will be using X.509 certificates during the provisioning process.

Under what conditions would they need to roll device certificates?

Answer A:
When they know a certificate is about to expire.

Answer B:
When they need to reprovision a device.

Answer C:
When they will be moving a device between hubs.

Module review: Question 3.8

The device enrollment process includes the option to specify the initial configuration of a device.

What mechanism is used to specify initial device settings?

Answer A:
The enrollment process instructs IoT hub to trigger a direct method when the initial device connected event is detected.

Answer B:
The enrollment process includes an initial device twin state field that enables you to specify desired properties for the device.

Answer C:
The enrollment process uses a security token to request device access permission from IoT hub and pushes the initial configuration to the device directly.

Module review: Question 3.9

When someone creates a device enrollment, they must specify how they want to assign devices to IoT hubs.

What is the default setting?

Answer A:
Evenly weighted distribution

Answer B:
Static configuration via the enrollment list

Answer C:
Lowest latency

Module review: Question 3.10

What needs to be completed in order to deprovision a single device from a group enrollment?

Answer A:
Disable or delete the device from the IoT hub's identity registry.

Answer B:
Create a disabled individual enrollment for the device's leaf certificate, disable or delete the device from the IoT hub's identity registry, and then delete the disabled individual enrollment for the device.

Answer C:
Create a disabled individual enrollment for the device's leaf certificate and then disable or delete the device from the IoT hub's identity registry.

Module review: Question 3.11

The Device Provisioning Service uses X.509 certificates to help ensure that device connections are secure.

What type of certificate is used to verify proof of possession for the root certificate?

Answer A:
A leaf certificate.

Answer B:
An intermediate certificate.

Answer C:
A verification certificate.