DEVICE AUTHORITY

KeyScaler Associate Level Certification Course

KeyScaler®

KeyScaler Foundations Course Overview

Course Objective:

This course provides foundational knowledge of Device Authority's KeyScaler platform. Participants will learn about KeyScaler's architecture, functionality, and core services, equipping them with the skills needed to manage IoT security effectively.

Course Outline:

1. Introduction to Device Authority Products

1. Overview of KeyScaler Platform
2. Overview of Device Agent - Credential Manager.
3. Key terminology and concepts.
4. KeyScaler services and their components.

2. Basic Device Registration & Authentication

1. Device registration processes.
2. Authentication methods (DDKG, PKI Signature+, mTLS).
3. Zero-Touch Provisioning and Certificate Signing Request (CSR).

3. Understanding KeyScaler Groups & Policies

1. Device groups and group policies.
2. Authentication methods for groups.
3. Managing conflicting policies within groups.

4. Device Registration and Management

1. Device lifecycle management (from deployment to decommissioning).
2. Key concepts: quarantine, blacklisting, deleting, device whitelisting and certificate provisioning.
3. The role of Service Connectors and API integration.

5. Using the KeyScaler Administrative Interface

1. Navigating the KeyScaler Control Panel.
2. Administrative functions for managing devices and policies.
3. Tenant account settings and notifications.

6. Managing Support Tickets

1. Overview of the Device Authority support ticketing system.
2. How to raise and prioritize support tickets effectively.
3. Best practices for managing and resolving tickets, including providing necessary details.

7. Assessment & Certification

1. Overview of the written certification test.
2. Key concepts recap and Q&A session.

Key Terminology and Concepts for KeyScaler

IoT (Internet of Things):

- Network of interconnected devices that communicate and exchange data.

Device Authentication:

- Process of verifying the identity of a device before issuing the necessary crypto credentials, ensuring secure communication with other authorized devices within the network.

Credential Management:

- Management of digital certificates, passwords, and cryptographic keys used to authenticate devices. Includes creation, distribution, renewal, and revocation of credentials.

Key Rotation:

- Regularly changing cryptographic keys to enhance security.

Policy Management:

- Creation and enforcement of security policies for devices and data. Policies include rules for key rotation, automated password management, and certificate provisioning.

Certificate Provisioning:

- Process of issuing digital certificates to devices for authentication and secure communications. Certificates are used to establish trusted connections between devices.

Automated Password Management:

- Automatic generation and rotation of device passwords to enhance security. Eliminates the use of default or static passwords.

Data Encryption:

- Process of converting data into a secure format that cannot be easily read by unauthorized individuals.

Lifecycle Management:

- Management of devices and their security from initial deployment to decommissioning.

DDKG (Dynamic Device Key Generation):

- Method of generating unique cryptographic keys directly on the device based on device-specific attributes.

Key Terminology and Concepts for KeyScaler

EPIC (Enhanced Platform Integration Connector):

- The framework within KeyScaler that enables integration with third-party platforms and services. Facilitates seamless integration and automation of security processes.

Device Registration:

- Process of adding a new device to the system and assigning it the necessary crypto credentials for secure communication.

Blacklisting:

- Process of denying communication privileges to specific devices that are considered compromised or unauthorized, during which their crypto credentials are permanently disabled to prevent secure system interactions.

Quarantine:

- Isolating a device that is suspected to be compromised or non-compliant.

HSM (Hardware Security Module):

- Physical device that provides secure management, processing, and storage of cryptographic keys.

API (Application Programming Interface):

- Set of protocols and tools for building software applications. KeyScaler APIs allow for automation of device registration, policy management, and other security tasks.

Tenant Account:

- Account that represents an organization or user within the KeyScaler platform. Manages device groups, policies, and credentials specific to that tenant.

Control Panel:

- Administrative interface of KeyScaler used to manage devices, policies, and credentials. Provides a centralized platform for overseeing IoT security.

Compliance:

- Adherence to industry standards and regulations for IoT security. KeyScaler helps organizations comply with these requirements through its security features

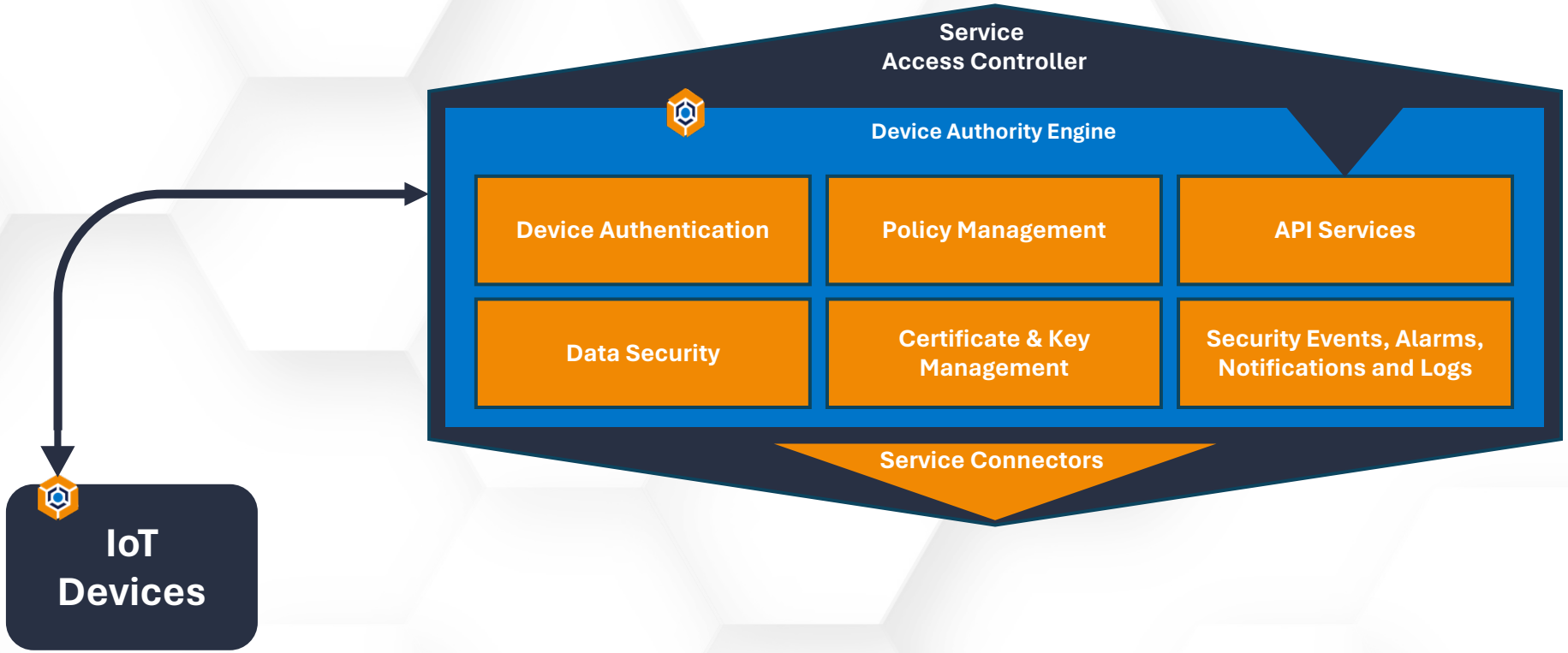
Introduction to Device Authority Products

Overview of KeyScaler and Credential Manager

Basic terminology and concepts

Device Authority Products

DA Platform : KeyScaler



DA Agent Software:
Credential Manager & DDKG

KeyScaler Overview

What is KeyScaler?

Purpose and functionality

Section 1

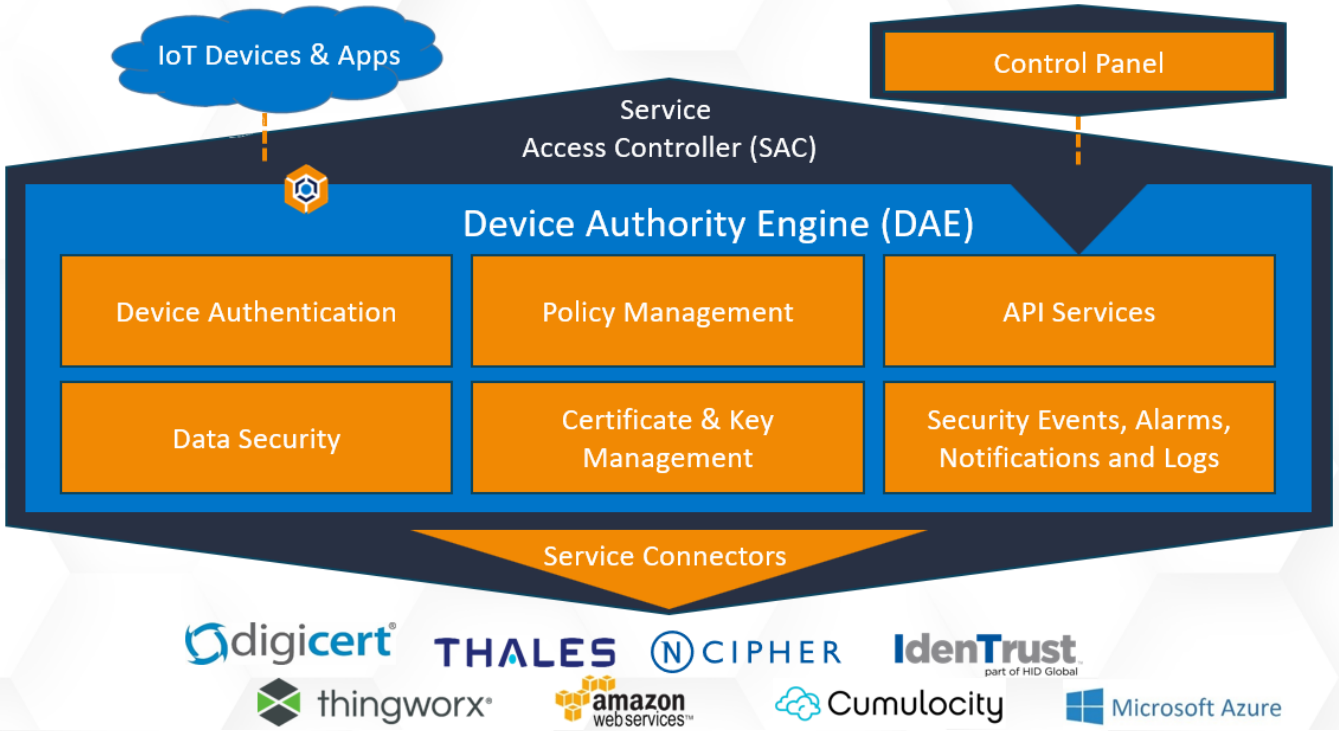
KeyScaler™ Platform

What is KeyScaler?

- **KeyScaler** is a powerful IoT security platform by Device Authority
- Designed to automate and simplify managing IoT device and application security.

What does it do

- **Verifies Devices are authenticated**
- **Secures Data via encryption**
- **Manages Device Credentials securely throughout their lifecycle.**



KeyScaler Overview

Key Functions of KeyScaler

Device Authentication:

- Process of verifying the identity of a device before issuing the necessary crypto credentials, ensuring secure communication with other authorized devices within the network.

Data Encryption:

- Protects transmitted data with advanced encryption, ensuring its integrity and confidentiality against interception or tampering.

Credential Management:

- Handles creation, distribution, renewal, and revocation of credentials (e.g., certificates, keys), automating Certificate provisioning to minimize human error.

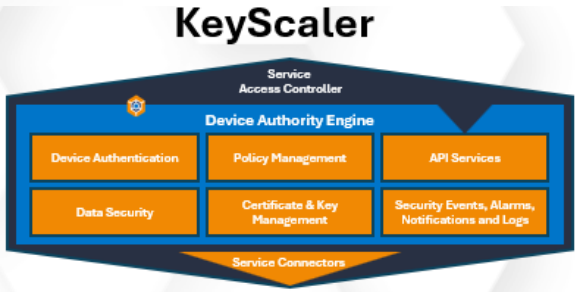
Policy Management:

- Allows admins to enforce security policies, like key rotation and automated password management, ensuring compliance and security.

Lifecycle Management:

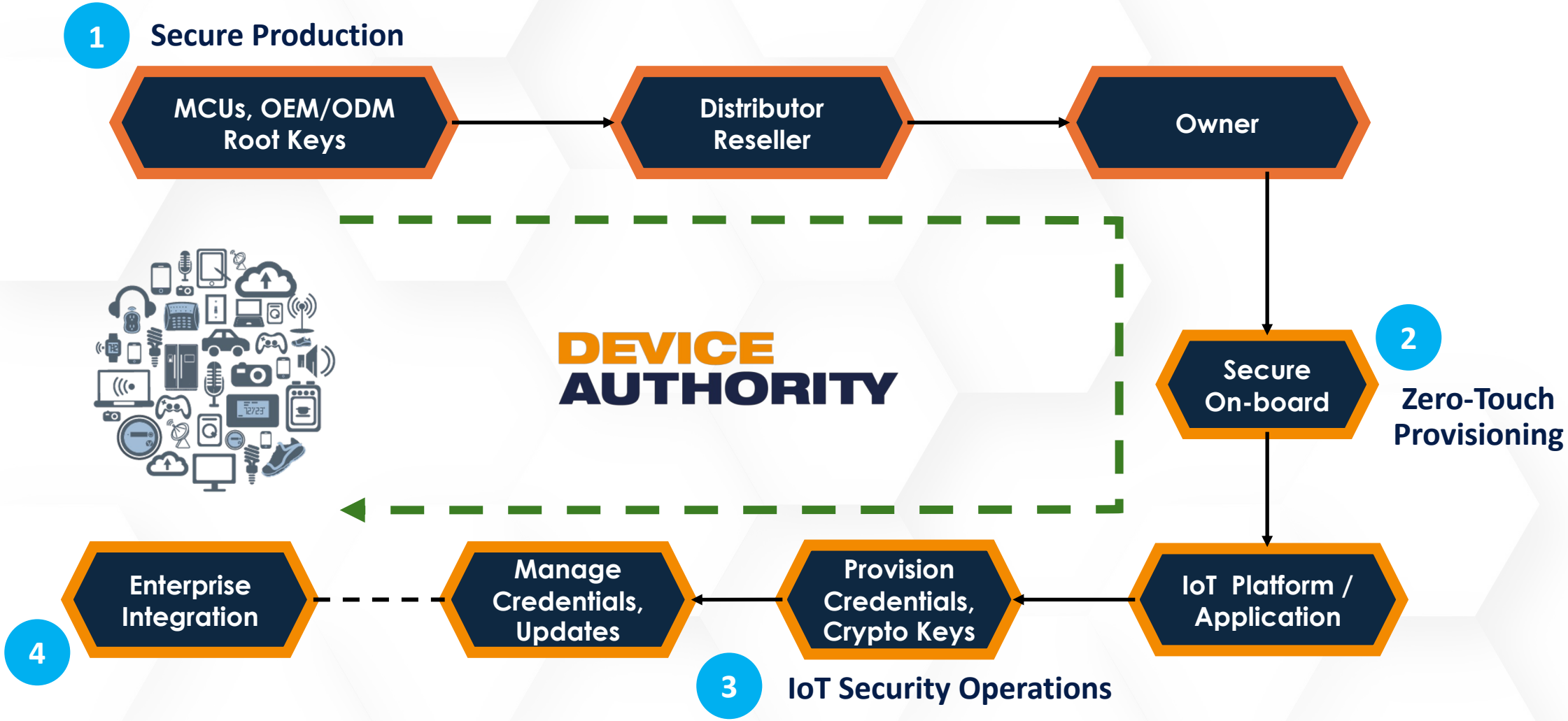
- Secures devices from deployment to decommissioning, maintaining consistent security measures throughout their lifecycle.

KeyScaler Services



Service Name	Service Description
Service Access Controller (SAC)	• Typically deployed to the DMZ • Provides HTTPS and WSS APIs for devices/endpoints • Acts as a “gateway” to receive requests from endpoints, and passes them on to the DAE • Prevents the DAE from being exposed to the internet
Device Authority Engine (DAE)	• Described as the “brain” of the platform • Processes device requests to register, authenticate, sign certificates, use HSM keys, etc. • Can be accessed via REST API
Key Management Service (KMS)	• Service provider for the tenant/CA keystore • Main module that handles key management; generation, signing etc. • Integrated with the HSM, where deployed
Control Panel (CP)	• Administrative UI • Uses the DAE REST APIs • View and manage devices, policies, CAs, service connectors, etc…
Enhanced Platform Integration Connector (EPIC)	• Frame-work to build integration connectors to 3rd Party systems (e.g. CA, HSM, IoT Platforms) • Provides a real-time API for receiving events from the KeyScaler system • Custom service connectors can be built to interact with other services

Let's look at a typical IoT device journey…



Credential Manager Overview

What is Credential Manager?

Purpose and functionality

Section 2

What is Credential Manager

- Credential Manager (CM) is a key component of the KeyScaler ecosystem. It is the device agent software that works in conjunction with the DDKG Library to automate and manage device identities, crypto credentials, and security policies across various IoT environments.

Credential Management Agent (CM) is a versatile component that can run on a variety of devices and platforms, including:

Linux-based Operating Systems:	• It supports popular Linux distributions, which are commonly used in industrial IoT and edge computing environments
Windows-based Systems:	• The CM is compatible with Windows OS, allowing it to be integrated into Windows-based IoT devices
Raspberry Pi:	• CM runs on Linux, and can be installed on Raspberry Pi devices, running **Raspbian** or **Ubuntu**, making it a good fit for smaller, resource-constrained IoT deployments
Edge Devices and Gateways:	• The CM is designed to run on IoT gateways and edge devices, providing secure credential management for devices in distributed environments
Cloud IoT Platforms:	• It integrates with cloud platforms such as **AWS IoT**, and **Microsoft Azure IoT**, offering seamless credential management for cloud-connected IoT ecosystems

Credential Manager Main features and functionalities

Automated Credential Management:

- The CM automates the process of creating, storing, rotating, and renewing credentials (e.g., certificates, keys) for IoT devices. This reduces the manual burden of managing credentials and minimizes the risks of expired or compromised certificates.

Secure Enrollment and Provisioning:

- The CM supports secure device enrollment into the KeyScaler platform. It ensures that devices are securely provisioned with the appropriate certificates and crypto credentials, preventing unauthorized devices from participating in secure communications.

Certificate Lifecycle Management:

- One of the key functions of the CM is to manage the entire lifecycle of certificates, including issuance, renewal, and revocation. This ensures that IoT devices always maintain valid certificates, preventing service disruptions and mitigating security threats.

Integration with Hardware Security Modules (HSM):

- The CM can work in conjunction with HSMs and Trusted Platform Modules (TPMs) to store keys and certificates securely on IoT devices. This integration helps maintain a high level of security, especially for sensitive data and transactions.

Policy-Based Credential Management:

- Device Authority's platform allows users to define policies for credential management. These policies determine how often credentials are rotated, when certificates should be renewed, and how to handle revoked or expired credentials.

Credential Manager Features and Functionalities continued

Credential Manager's role in Device Authentication and Credential Management

Device Authentication

- **Ensuring Secure Access:** Only authorized devices are permitted to communicate securely within the system..
- **Preventing Unauthorized Access:** Protects against unauthorized devices attempting to access the system.
- **Authentication Methods:** Employs various secure methods to verify device identity.

Credential Management

- **Lifecycle Management:** Oversees the entire lifecycle of credentials from creation to revocation.
- **Automated Provisioning:** Automates the provisioning of credentials to reduce human error.
- **Credential Rotation and Renewal:** Ensures credentials are regularly updated to maintain security.
- **Revocation:** Manages the revocation of credentials to prevent compromised devices from participating in secure communications.

Importance of Credential Manager

- The Credential Manager is essential for managing IoT device security. It handles device authentication and manages the entire credential lifecycle, ensuring secure device connectivity and protected communications. Understanding its fundamentals is key to maintaining a secure IoT ecosystem.

Client-side Integration Tools

Credential Manager

- Fully-built agents
- Provided for specific solutions
- e.g. Certificate management

KSClient SDK

- Wrapper for the DDKG library
- Combines KeyScaler REST APIs and DDKG interaction into single calls

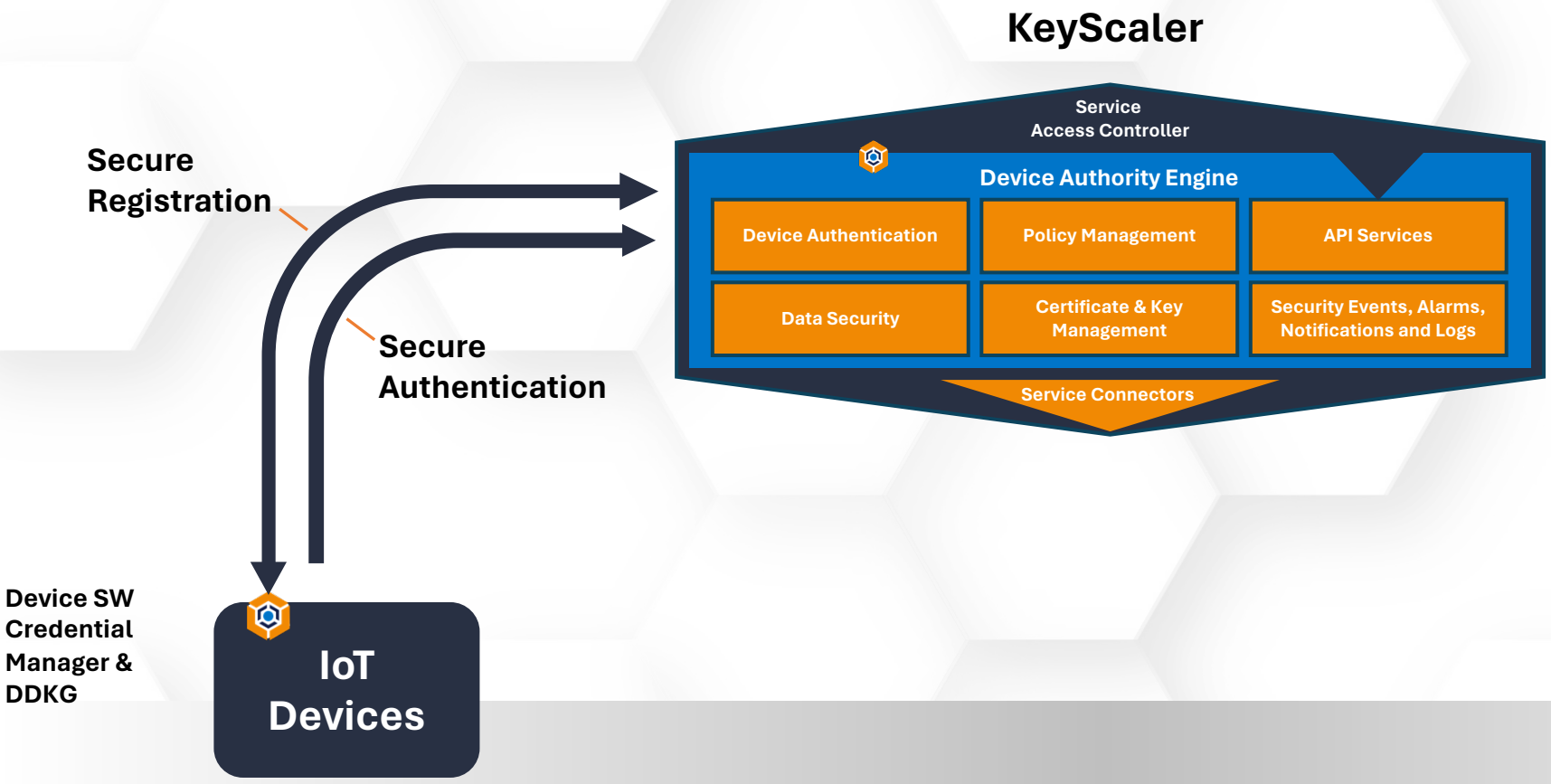
DDKG

- Core authentication library
- Challenge in, auth. key out

Basic Device Registration

Section 3

Basic Device Registration



Basic Device Registration

Device Registration Overview

Provisioning

Is the **Creation of a device registration record in KeyScaler** for a given device or application.

Authorization

The **Action of registering a device and associating it with an Authorization Identifier**.

Authorization Models

Supports multiple device authorization models to fit the needs of various applications.

Provisioning Model	Control Panel	API Library
Admin Initiated	YES	YES
Admin Approved	YES	YES

Basic Device Registration

Process Flows
A high-level overview of the process flow for each supported authorization model.

Administrator Initiated

Device Authorization begins when an administrator creates a registration record for either an application or Management Control Panel administrator.

For IoT devices

The application on the device initiates device registration, and if there is a valid registration record with registration controls the device can meet, the device is registered.

Admin Approved

Device Authorization begins with a registration request initiated by calling the appropriate API from an application (i.e.. cloud/web application).

The device registration record is created, but before a device can use that registration record, an administrator must approve the request from within the Management Control Panel.

To approve a registration request, navigate to the Manage Devices > Manage Pending Approval > Applications tab. Once approved, an email is sent with registration instructions.

Basic Device Provisioning

Section 4

Basic Device Provisioning

Create a Single Registration Record in KeyScaler

(Bulk Creation addressed in Level II Certification)

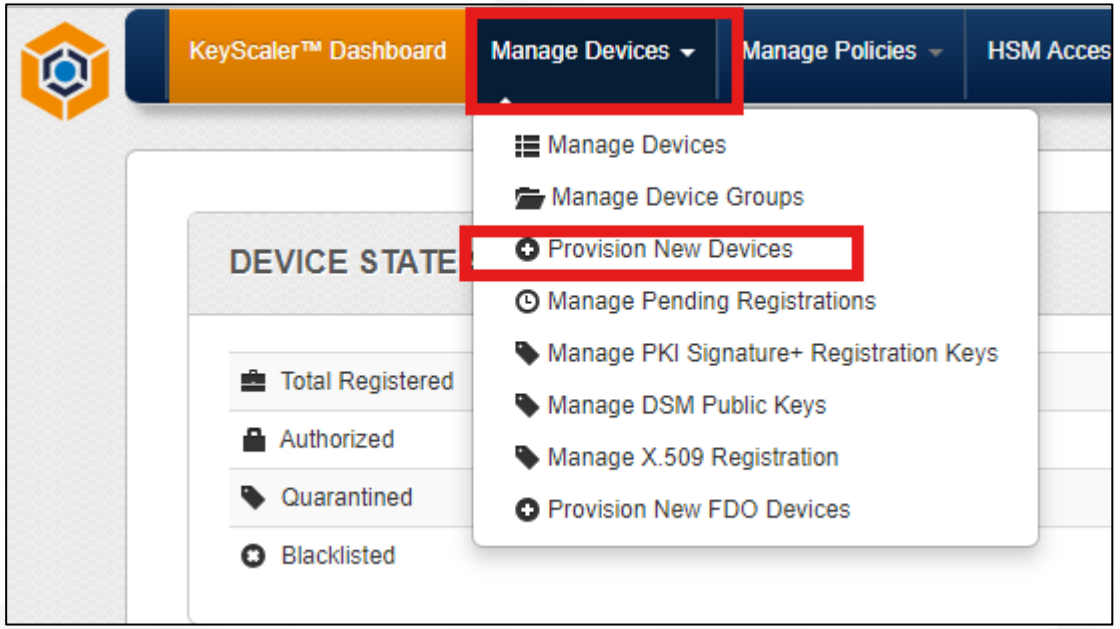
To create a new device provision record:

Navigate to :

Manage Devices Pull-Down

Select:

Provision New Devices



Provision Device Page

PROVISION DEVICES

Provision a Device | Bulk Provision Devices

DEVICE REGISTRATION CONTROLS

Restrict By

	Registration Method *	DDKG
☐	OS Type	Any
☐	Device Identifier	Device Identifier
☐	MAC Identifier	MAC Identifier
☐	IP Address	IPv4/IPv6 Address
☐	Application Identifier	Key / Value
☐	Valid From	
	Valid Until	
☐	Geofence	Latitude / Longitude / Radius meters

DEVICE AUTHORIZATION CONTROLS

Authorization Identifier	Authorization Identifier
Device Group(s) (optional)	Device Group(s)
Assign Device Name (optional)	Device Name
Send E-mail invitation to (optional)	E-mail address

Create Registration Record

VICE
THORITY

Basic Device Registration: DDKG

Select the Registration Method:

DDKG, PKI Signature+, or X.509. Input requirements will vary with your selection.

Basic Device Registration: Dynamic Device Key Generation (DDKG)

Registration method = DDKG

Device Registration Controls:

Complete the Device Registration Controls as desired.

If no Device Registration Controls are specified, ***any device*** will be able to consume the registration record created.

Using specific registration controls, like a MAC address, enhances security by allowing only the device with the matching MAC address to register using that particular registration record.

Basic Device Registration: DDKG

OS Type:

- Restrict device registration to a specific platform or OS. Default: "Any".

MAC Identifier:

- Restrict registration by MAC address (format: XX:XX:XX:XX:XX).

IP Address:

- Restrict registration by public IPv4 or IPv6 address. Truncation for ranges is supported.

Application Identifier:

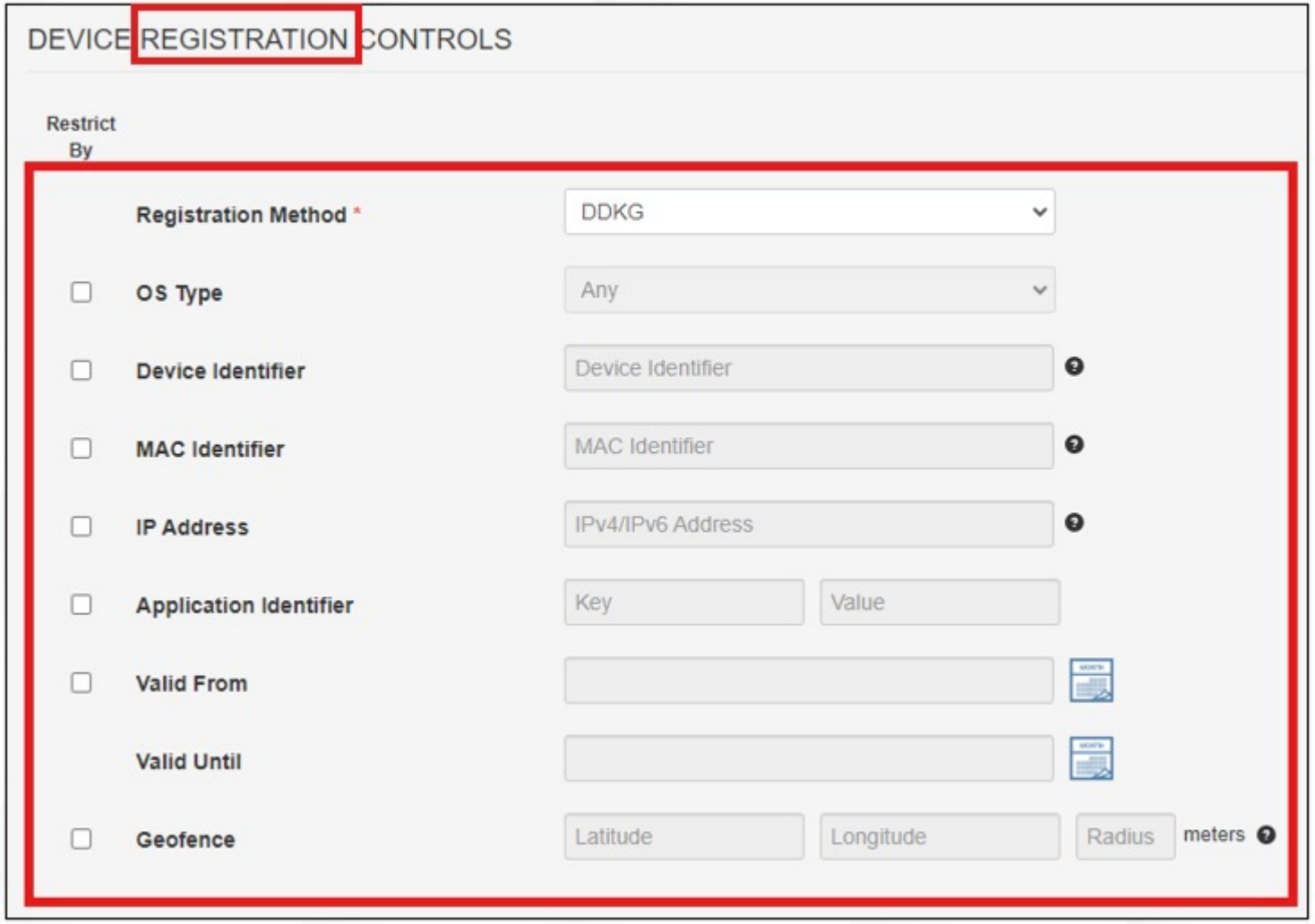
- Optional key/value pair for binding additional information during registration. Must match between registration record and initiating program. Not retrievable by DDKG library and unsupported by Crypto, ThingWorx Extension, and Credential Manager Agents.

Valid From/Until:

- Set a date/time range for registration validity. Default: 24 hours; minimum: 1 hour.

Geofence:

- Restrict registration to a geographical range using lat/long values and a specified radius. Device must support geo code retrieval.



Note: DDKG Level I is covered in this course. DDKG Level II is covered in Level II Certification

Basic Device Registration: DDKG

In the Device Authorization Controls section, complete the fields as desired:

Authorization Identifier

An extra control linking a device to an identifier, **allowing authentication only when the device is both authenticated and authorized for that identifier**. Authorization Identifiers are restricted to letters, numbers, and the following characters: @ : . # + ^ ' = _ ` ~ -

Device Group(s)

Option to add a device to specific groups during registration, with all available groups listed for selection.

Assign Device Name

Allows assigning a name to a device during registration.

If not specified, the name is based on the device's MAC address or hardware serial number.

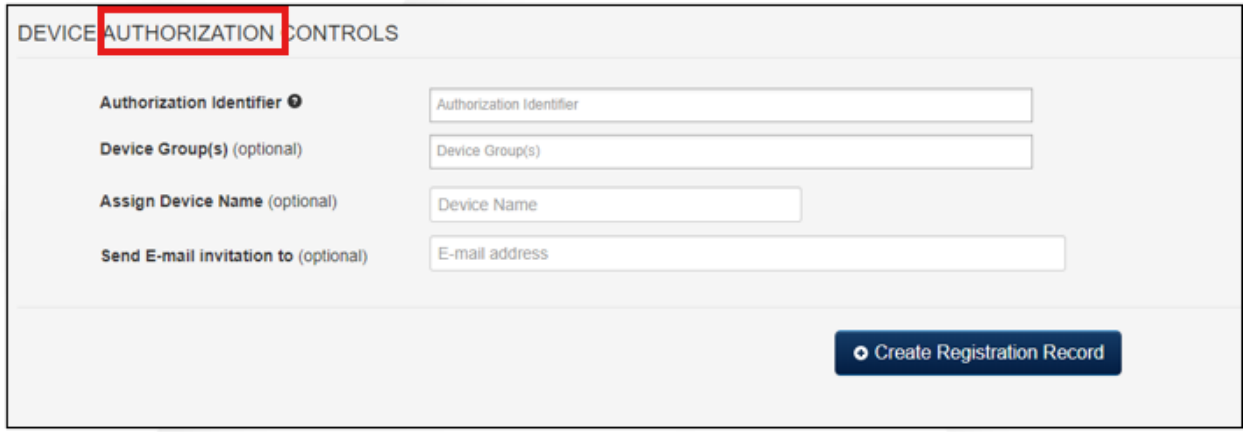
If this field is blank, the device name is determined by the device (devicename can be provided via API, or by CM configuration files).

If the registration record does not include device characteristics, the device name will be blank.

Send E-mail invitation to

If you would like to send an email notification with registration information at the time of registration record creation, complete this field.

Note that your system must be configured to send out emails for this feature to work. This feature is typically only used for end-user device registration.



Press the Create Registration Record button to generate a new registration record.

Basic Device Registration: PKI Signature+

Device Registration Controls

Complete the Device Registration Controls as appropriate for your device.

Device Identifier:

- Enter the unique identifier for your device, like the hardware serial number.

Public Key Registration :

- Choose the pre-defined Public Registration Key for your device in KeyScaler.

Valid From and Until:

- Set the date/time range for the registration record's validity; defaults to 24 hours if not specified.

Device Group(s):

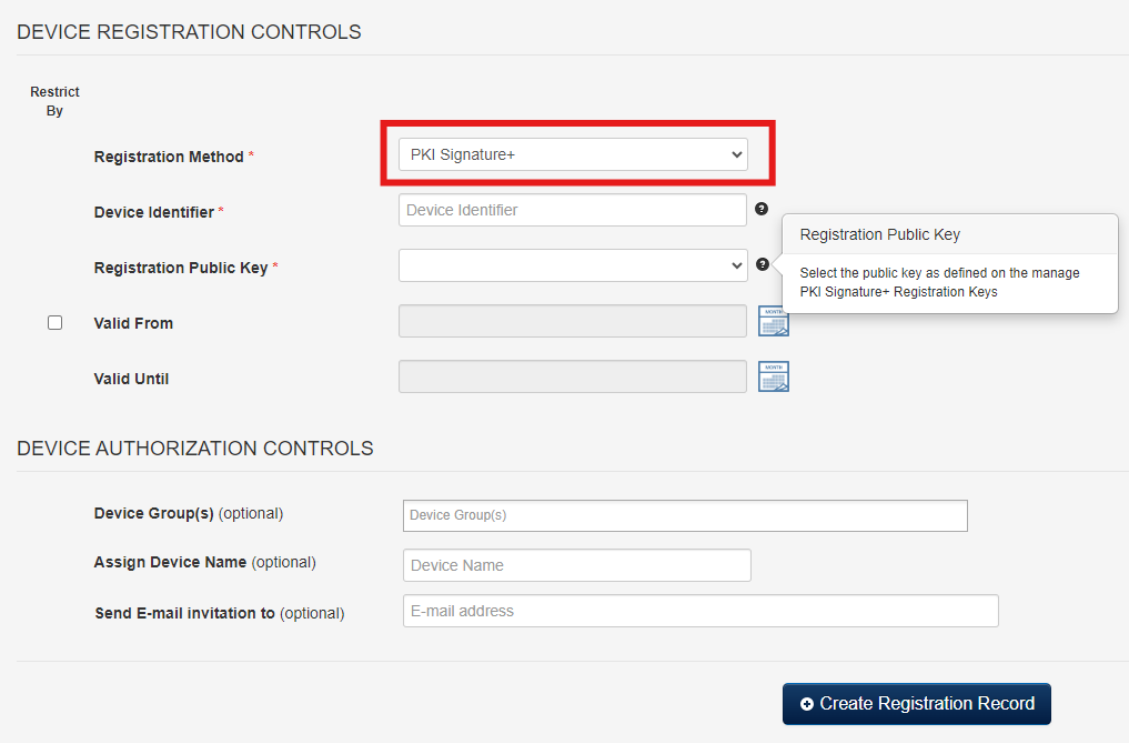
- Select groups to add the device to during registration.

Assign Device Name:

- Optionally assign a name to the device; if left blank, it's auto-determined based on the device's characteristics.

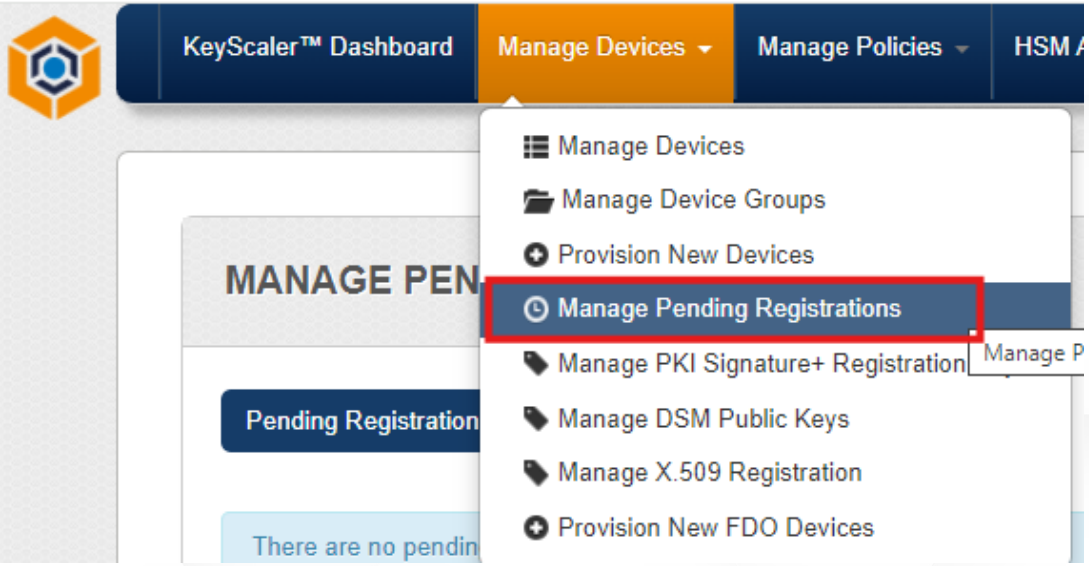
Send E-mail Invitation

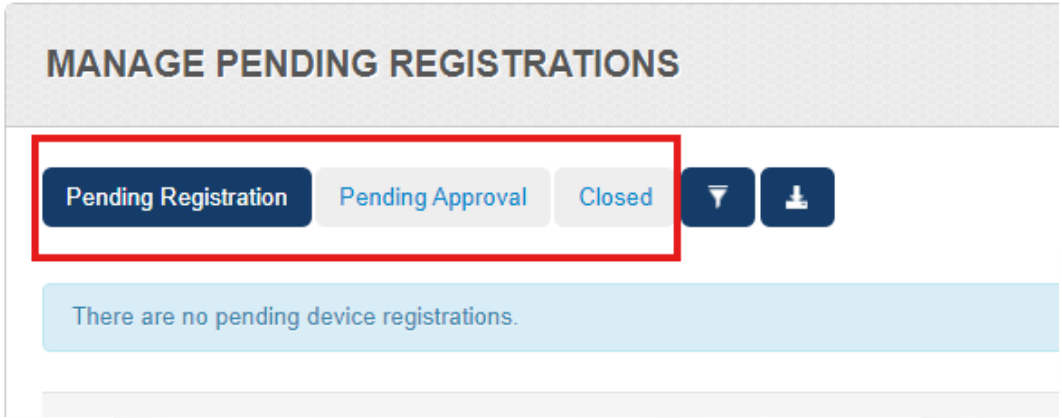
- Sends a registration email if configured.



Press the Create Registration Record button to generate a new registration record.

Manage Pending Registrations





Consuming Registration Records For **DDKG** Registration

1. Check for any registration records with a matching Authorization Identifier.

- Consider only those records and go to #2. If there are none, registration fails.

2. Using the registration records from #1

- Check for registration records with MAC address matching the incoming device. If some exist, consider only those records and go to #3. If none exist, go to #3 without eliminating any records from consideration.

3. Using the registration records from #2

- Check for registration records with IPv4/IPv6 matching the incoming device. If some exist, consider only those records and go to #4. If none exist, go to #4 without eliminating any records from consideration.

4. Can the device supply geo coordinates?

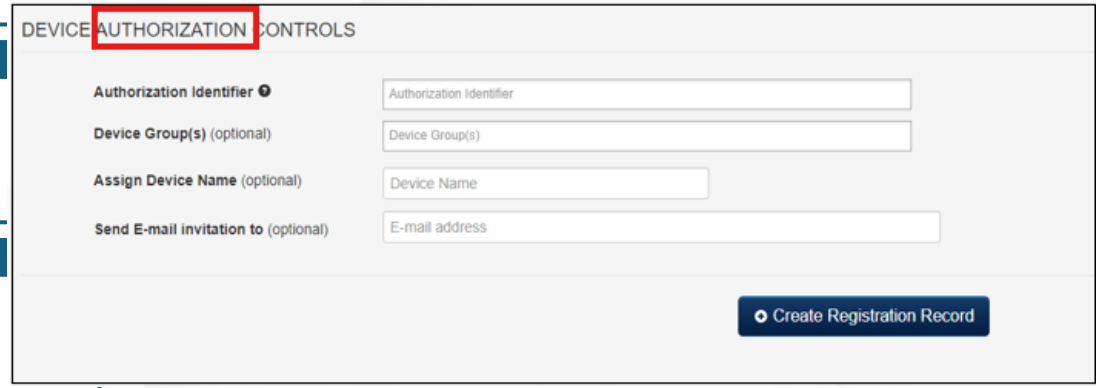
- If no, go to #5.
- If YES:
  - Using the registration records from #3, check for registration records with geo coordinates. If some exist, consider only those records and go to #5. If no, go to #5 without eliminating any records from consideration.

5. Using the registration records from #4

- Check for registration records with a match on Platform. If there is a match, use one of those records.
- If there is no match, see if there is a registration record with Platform=Any. If so, use that record.

6. Verify that all registration controls specified in the registration record match the device.

- If they do not, registration fails and the registration record is closed so it cannot be used again.
- If they do match, register the device.



Consuming Registration Records For PKI Signature+

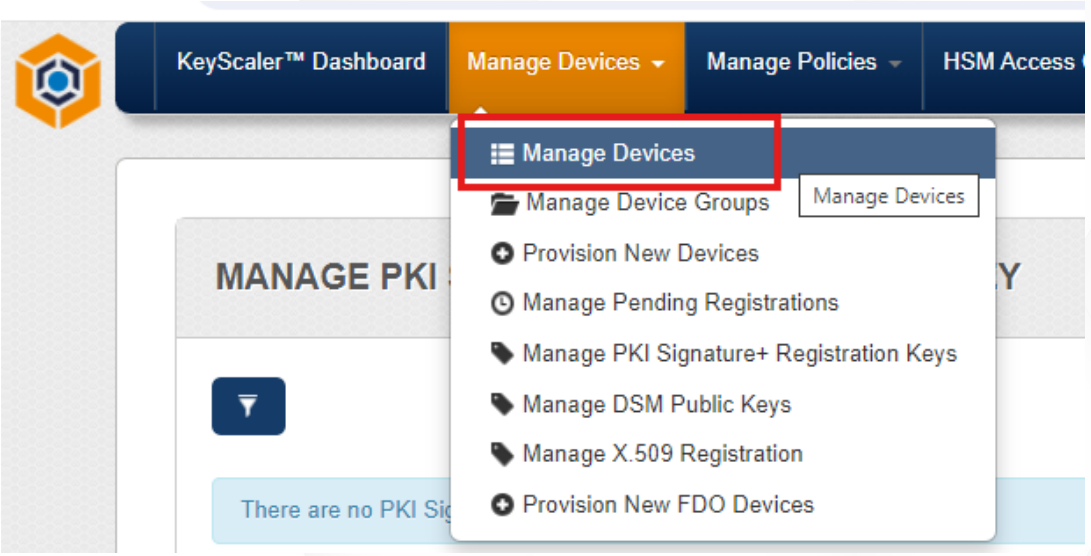
Check for any registration records with a matching Public Registration Key and matching Device Identifier.

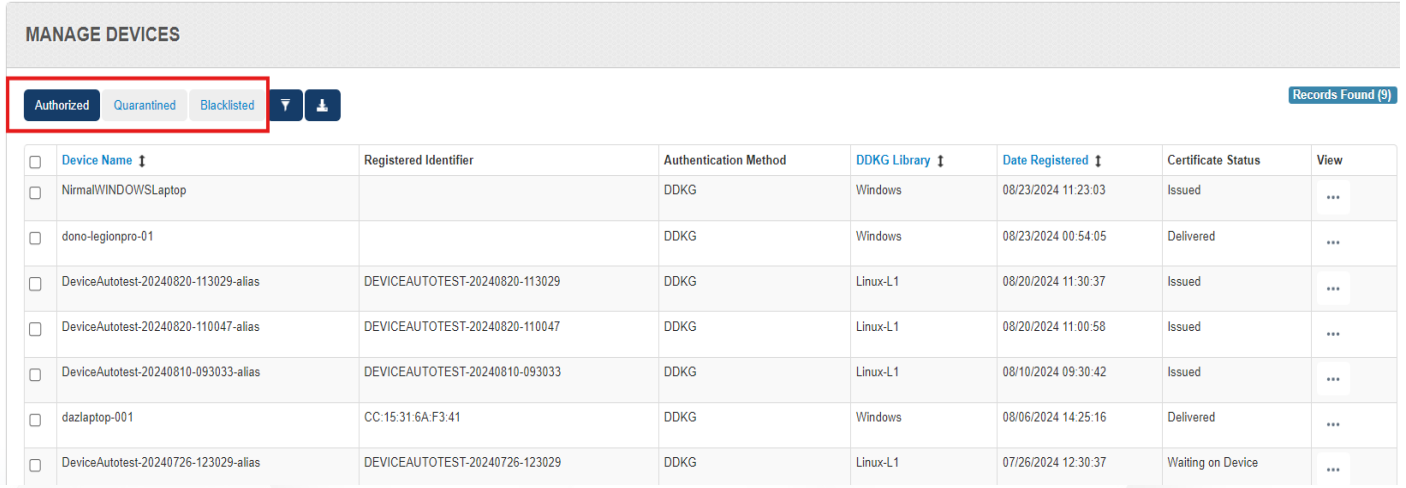
- If there are none, registration fails.
- If there are one or more records that do match, use one of the records and register the device.

Managing Devices

Section 5

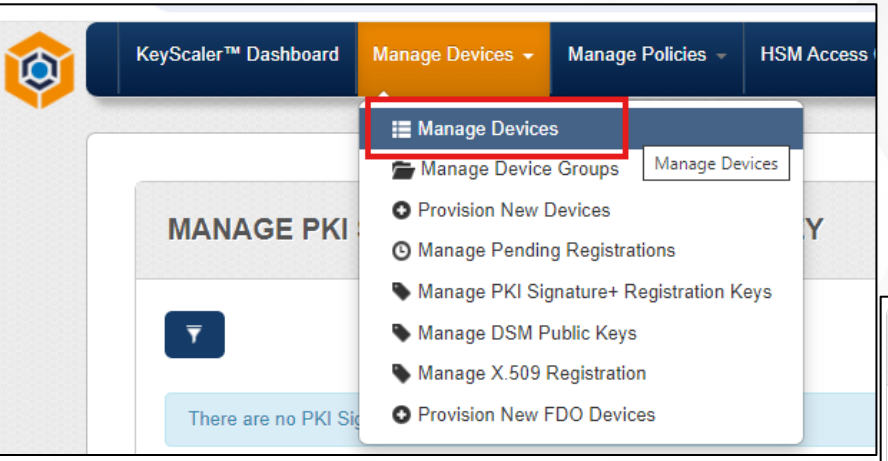
Manage Devices





Device Name	Registered Identifier	Authentication Method	DDKG Library	Date Registered	Certificate Status	View
NirmalWINDOWSLaptop		DDKG	Windows	08/23/2024 11:23:03	Issued	...
dono-legionpro-01		DDKG	Windows	08/23/2024 00:54:05	Delivered	...
DeviceAutotest-20240820-113029-alias	DEVICEAUTOTEST-20240820-113029	DDKG	Linux-L1	08/20/2024 11:30:37	Issued	...
DeviceAutotest-20240820-110047-alias	DEVICEAUTOTEST-20240820-110047	DDKG	Linux-L1	08/20/2024 11:00:58	Issued	...
DeviceAutotest-20240810-093033-alias	DEVICEAUTOTEST-20240810-093033	DDKG	Linux-L1	08/10/2024 09:30:42	Issued	...
dazlaptop-001	CC:15:31:6A:F3:41	DDKG	Windows	08/06/2024 14:25:16	Delivered	...
DeviceAutotest-20240726-123029-alias	DEVICEAUTOTEST-20240726-123029	DDKG	Linux-L1	07/26/2024 12:30:37	Waiting on Device	...

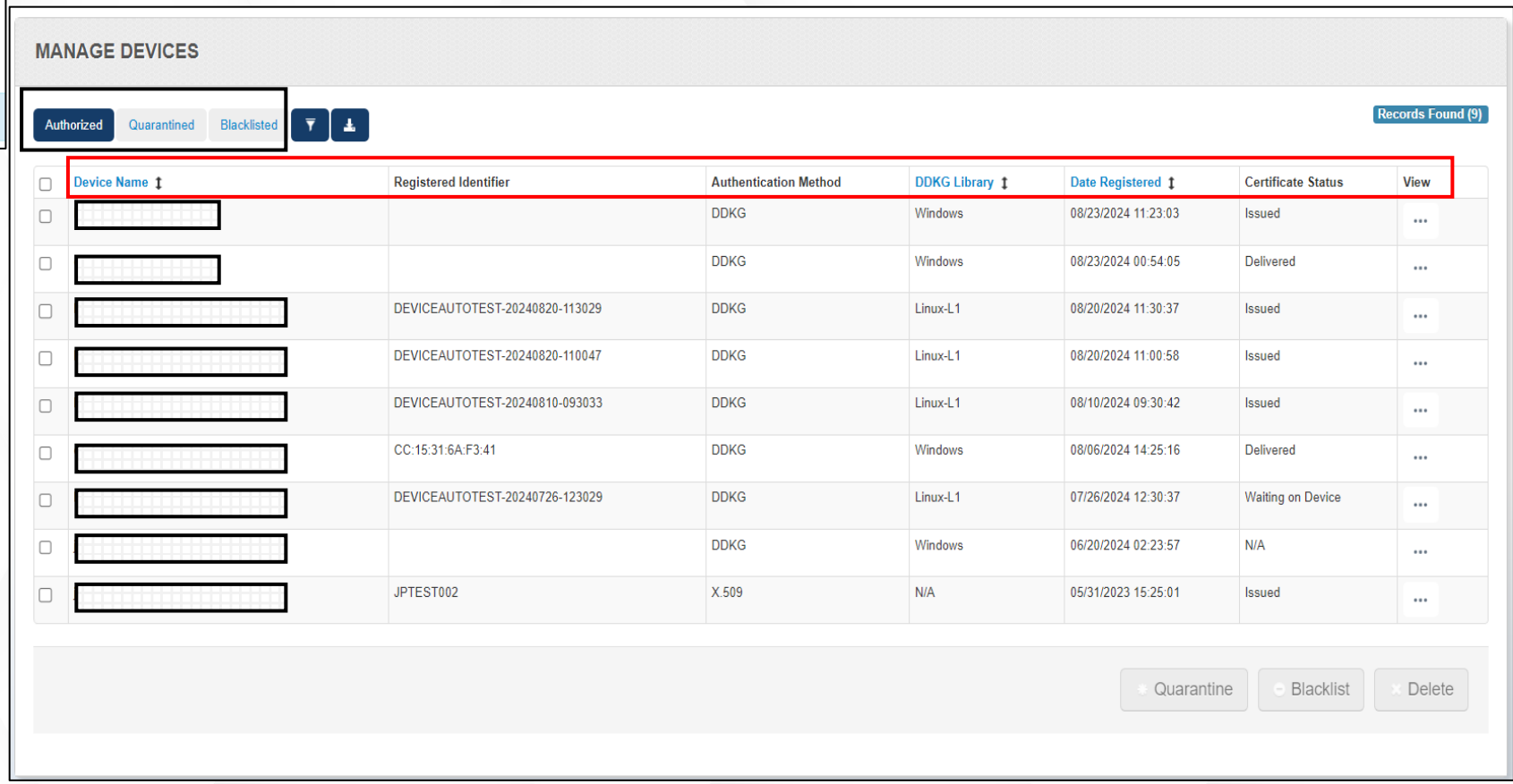
Provisioned Devices



Authorized: **Device is authorized for continued use**

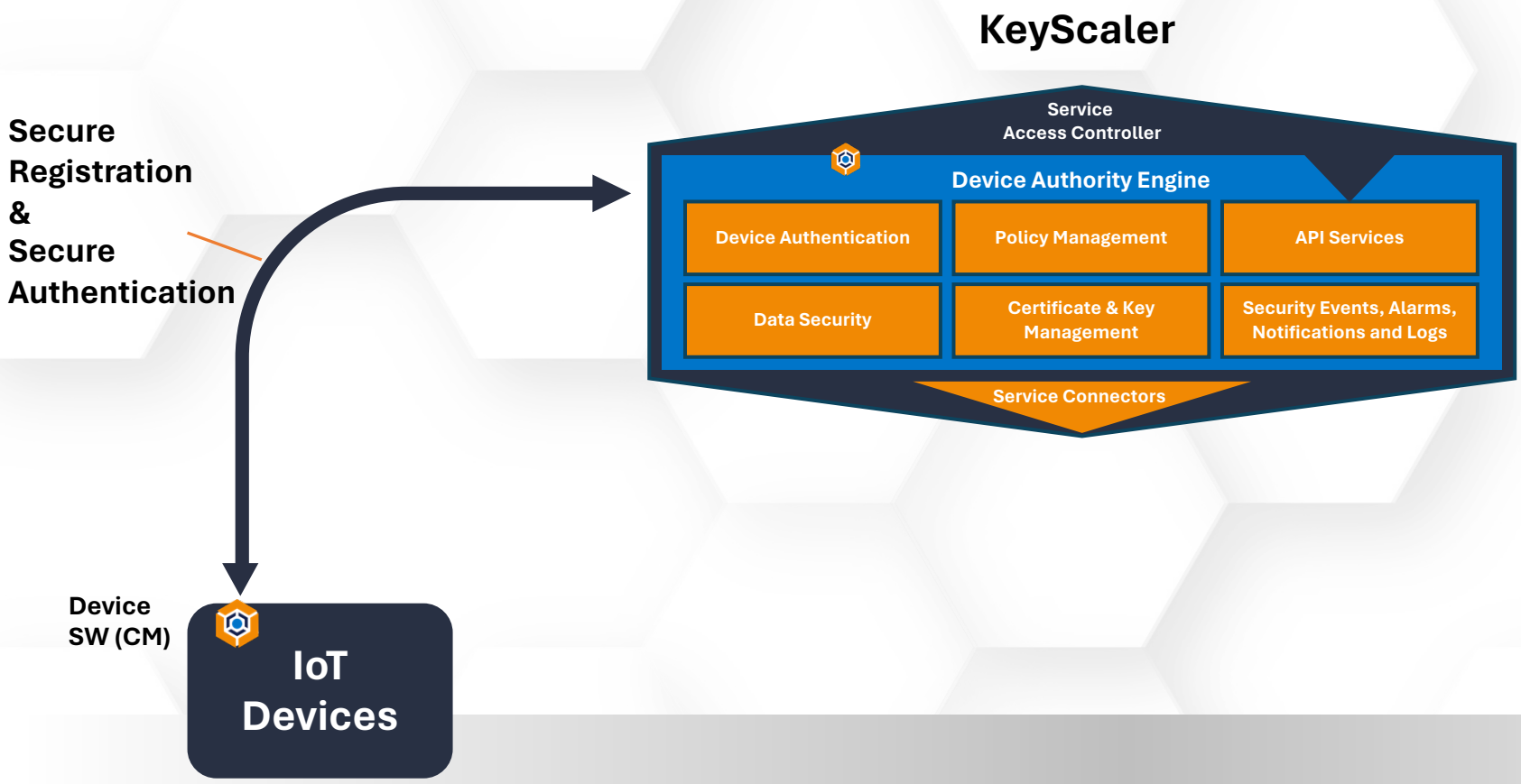
Quarantining: **It isolates the device for further investigation and can be reauthorized**

Blacklist: **revokes the device's access permanently**



Device Name	Registered Identifier	Authentication Method	DDKG Library	Date Registered	Certificate Status	View
[redacted]		DDKG	Windows	08/23/2024 11:23:03	Issued	...
[redacted]		DDKG	Windows	08/23/2024 00:54:05	Delivered	...
[redacted]	DEVICEAUTOTEST-20240820-113029	DDKG	Linux-L1	08/20/2024 11:30:37	Issued	...
[redacted]	DEVICEAUTOTEST-20240820-110047	DDKG	Linux-L1	08/20/2024 11:00:58	Issued	...
[redacted]	DEVICEAUTOTEST-20240810-093033	DDKG	Linux-L1	08/10/2024 09:30:42	Issued	...
[redacted]	CC:15:31:6A:F3:41	DDKG	Windows	08/06/2024 14:25:16	Delivered	...
[redacted]	DEVICEAUTOTEST-20240726-123029	DDKG	Linux-L1	07/26/2024 12:30:37	Waiting on Device	...
[redacted]		DDKG	Windows	06/20/2024 02:23:57	N/A	...
[redacted]	JPTEST002	X.509	N/A	05/31/2023 15:25:01	Issued	...

Device Registration and Authentication



Creating a KeyScaler Private Certificate Authority

Section 6

Private Certificate Authority (CA)

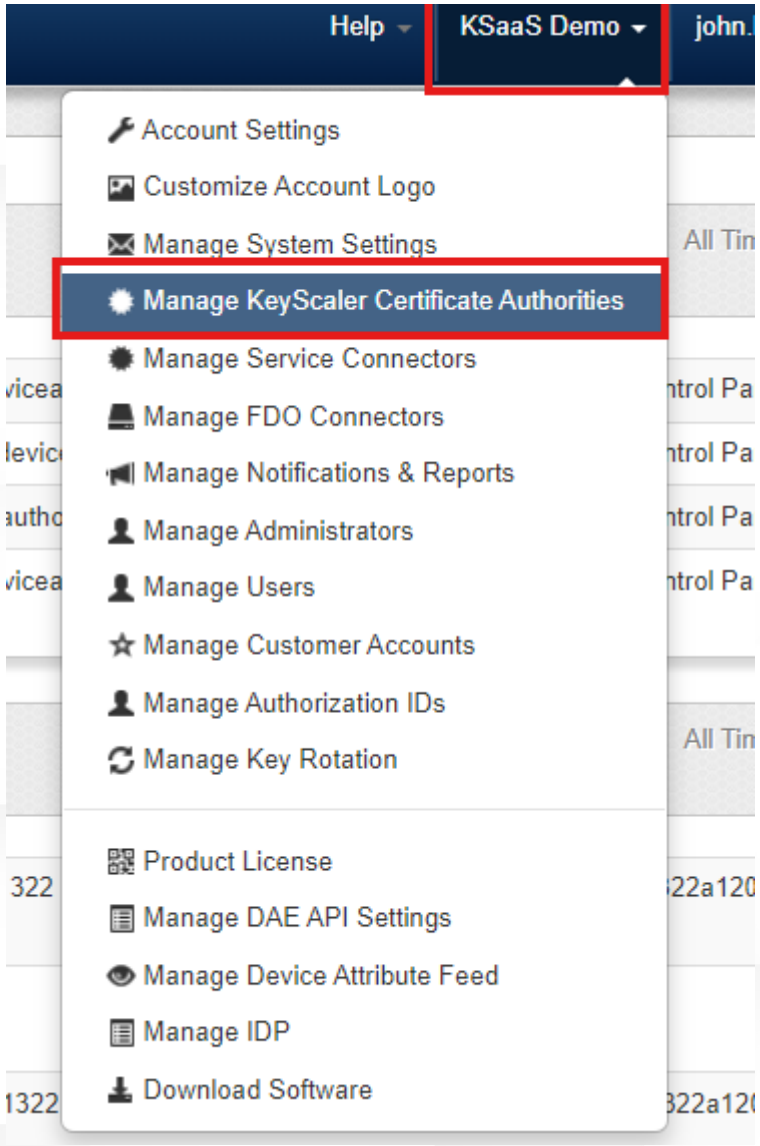
A Private Certificate Authority (CA)

- Is an internal system used by organizations to issue and manage digital certificates for securing communications within their private network.
- Unlike a public CA, which provides certificates that are trusted by external entities, a private CA's certificates are typically trusted only within the organization or by trusted external partners.
- This allows the organization to maintain control over the issuance and revocation of certificates and tailor security policies to their specific needs.

Private Certificate Authority (CA)

Create a New CA

- Click on the tenant name of your key scalar platform.
- Then click on manage KeyScaler certificate authority
- In the newly opened MANAGE KEYSCALER CERTIFICATE AUTHORITIES window click on “Add New”



Key Type:

- Specifies the type of cryptographic algorithm used for key generation, typically either RSA (Rivest-Shamir-Adleman) or EC (Elliptic Curve).
- **RSA or EC**:
  These are the two main types of cryptographic algorithms:
- **RSA**: A widely used public-key algorithm used for generating public-private key pairs
- **EC**: Elliptic Curve cryptography provides similar security to RSA but with shorter key lengths, making it more efficient.

Certificate Authority Name:

- The designated name for the certificate authority, **used to identify it in the certificate**.

Certificate Authority Common Name (CN):

- The **fully qualified domain name** (FQDN) that uniquely identifies the Certificate Authority.

Organization Unit (OU):

- Refers to a **specific department or division within an organization**, providing more detailed information about the certificate's ownership.

Organization Name (O):

- The **legal name of the organization** to which the certificate belongs.

Locality (L):

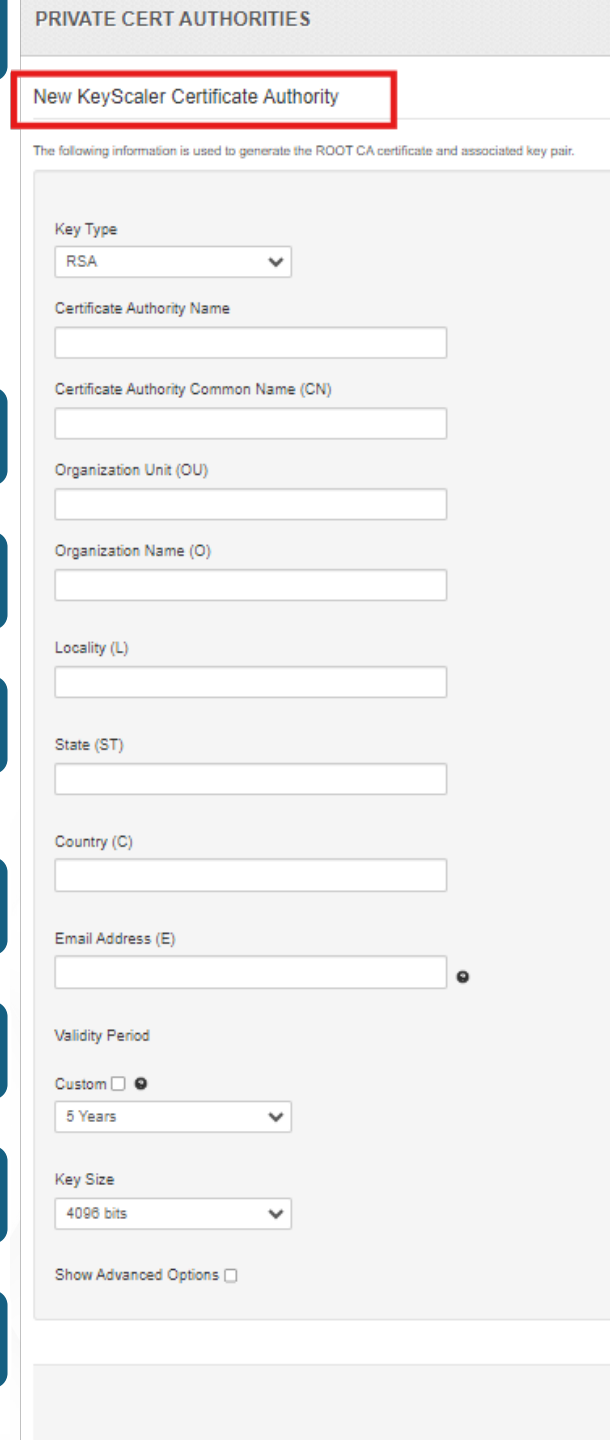
- The **city or town** where the organization is legally located.

State (ST):

- The **state or province** where the organization is legally registered.

Country (C):

- The **two-letter country code** representing the country where the organization is legally registered.



Email Address (E):

- **The email address** associated with the Certificate Authority, often used **for contact purposes**.

Validity Period (5, 7, or 10 years):

- The **duration for which the certificate is valid,** chosen from predefined options (5, 7, or 10 years).

Custom Validity Period:

- Allows specifying a **non-standard validity period** within the limits (e.g., between 1 and 45 years, as shown in the attached image). The expiry must be before 2070.

Key Size (2048 or 4096 bits):

- Defines **the length of the encryption key**. Larger keys (4096 bits, Keyscaler default) offer more security but are slower, while shorter keys (2048 bits) are faster but offer less security.

Show Advanced Options:

- Provides **additional configuration options** for advanced users when setting up the certificate authority.

Enable OCSP Responder:

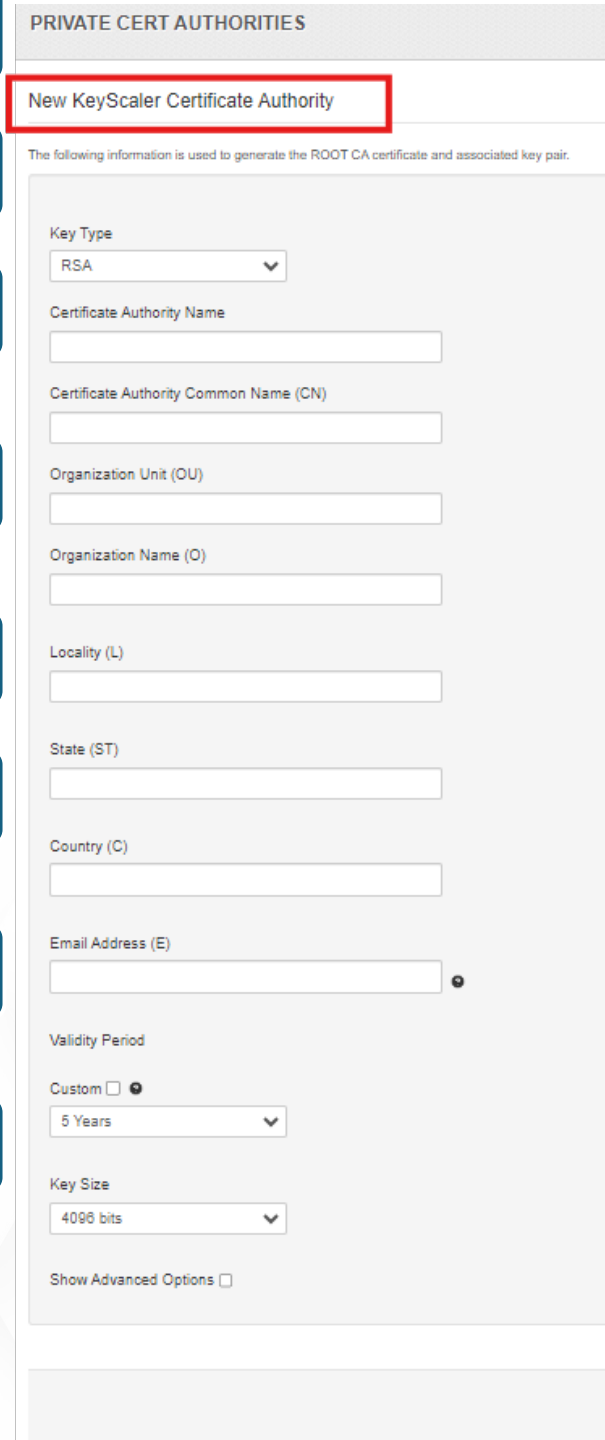
- Activates the Online Certificate Status Protocol (OCSP) responder, which allows for **real-time verification of certificate status**.

Sign Using External CA:

- Indicates that the **certificate should be signed by an external Certificate Authority** rather than the internal CA, typically used for establishing trust with external entities.

Save Key Externally:

- Allows you to **save the private key generated during the certificate creation process to an external location**, rather than storing it within the certificate authority system.
- Saving the key externally can be useful for backup purposes, secure storage in a hardware security module (HSM), or for use in another system or application.
- It ensures that the private key can be kept in a sepa ng the risk of unauthorized access or loss.



Key Points for Managing KeyScaler Certificate Authorities:

MANAGE KEYSCALER CERTIFICATE AUTHORITIES KSAAS DEMO

KeyScaler Certificate Authorities	Key Info	Date Created	Valid Until	Status	Actions
	RSA 4096 bits	08/09/2024 20:17:00	08/09/2069 20:17:00	Active	View | Export | Delete
	RSA 4096 bits	08/08/2024 19:08:03	08/08/2034 19:08:03	Active	View | Export | Delete
	RSA 4096 bits	08/06/2024 10:56:37	08/06/2029 10:56:37	Active	View | Export | Delete
	RSA 4096 bits	08/04/2024 12:25:32	08/04/2029 12:25:32	Active	View | Export | Delete

Overview:

- The interface **lists all configured Certificate Authorities** (CAs) with key details like key size, creation date, validity, and status.

Key Information:

- Displays the **cryptographic strength** (e.g., RSA 4096 bits) of each CA.

Validity:

- Shows when each CA **was created and when it will expire**, helping track active periods.

Status:

- Indicates **if the CA is "Active"** and currently issuing certificates.

Actions:

- Options to **View, Export, or Delete** the CA.

Importance:

- Regular review and management of CAs is essential for maintaining network security and trust.

KeyScaler Groups

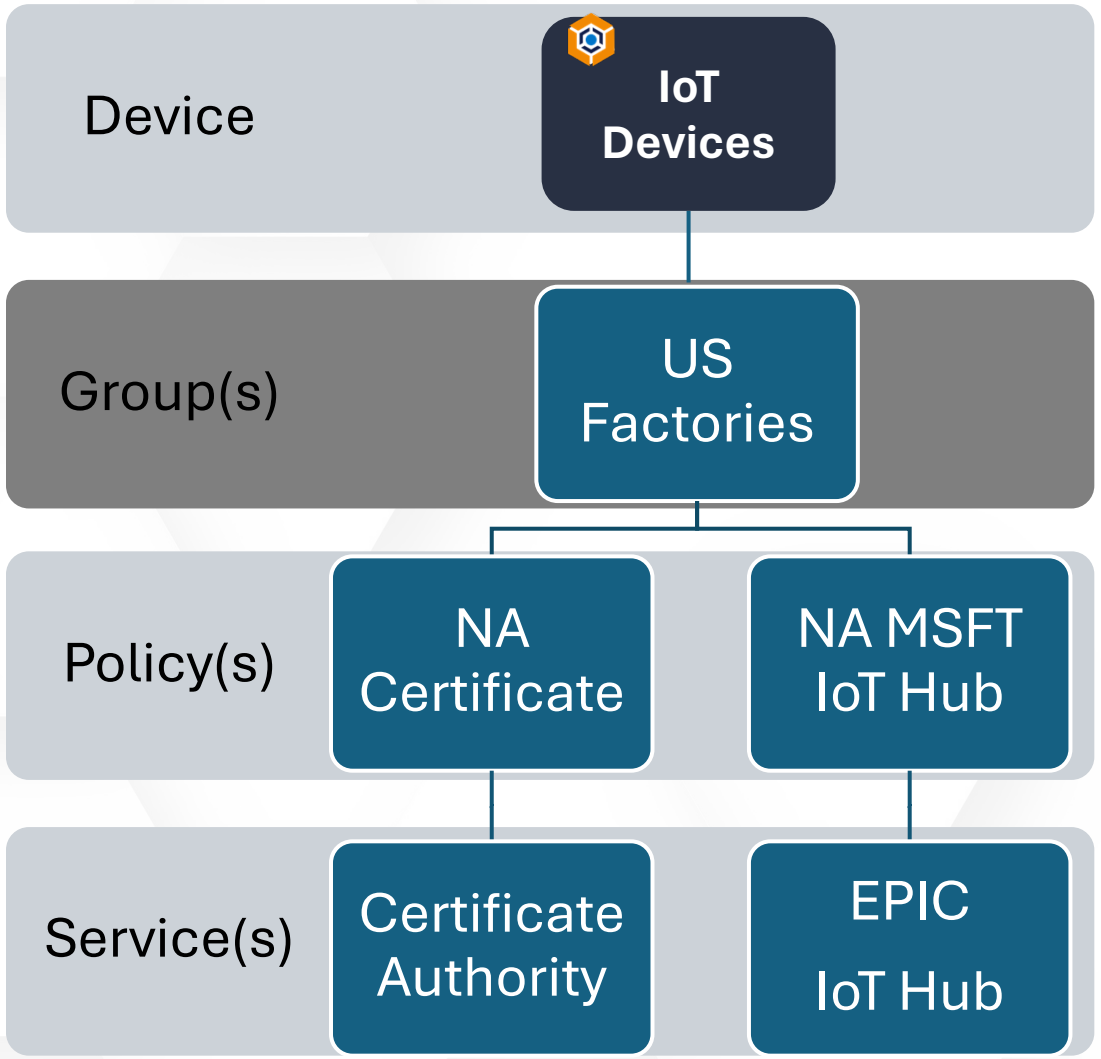
Section 7

KeyScaler Groups

- Groups are used to organize a common set of policies
  - Certificate Policies
  - Crypto Policies
  - Password Policies
  - 3rd Party Integrations

  And devices that use those policies.

  - If you are not using policies, defining groups is optional.



Understanding KeyScaler Groups

Important:

When using the DDKG authentication method

- When using multiple policies in a group it is also possible to create conflicting policies so proceed with caution.

PKI Signature+ authentication method

- Only supports the use of crypto policies.
- If you create a group that uses PKI Signature+ and Agent Crypto Policies, and also add KeyScaler Issued Cert Policies and/or Automated Password policies, your crypto keys may be deleted.

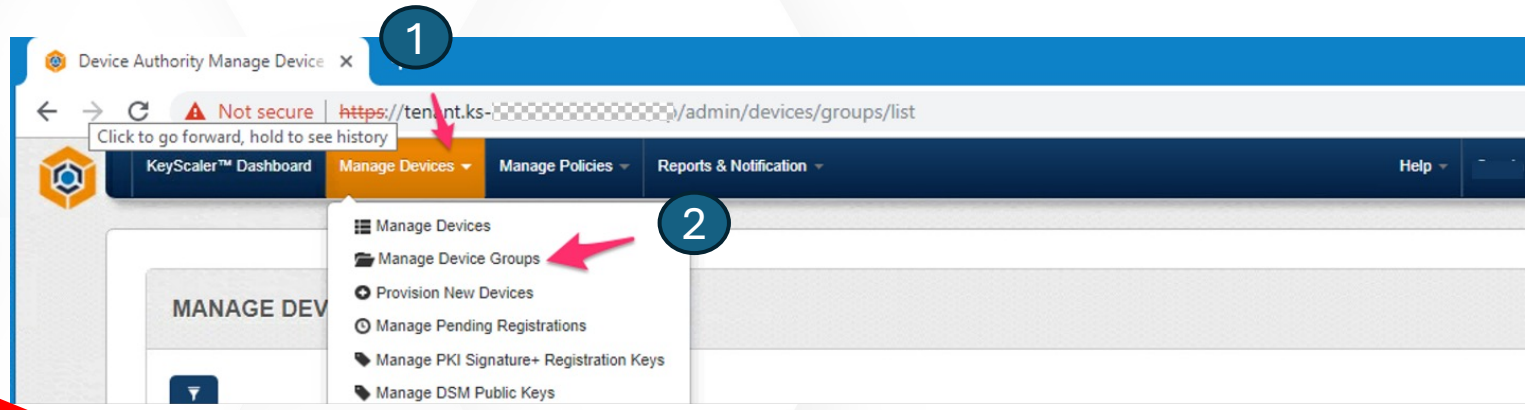
Adding a device to multiple groups

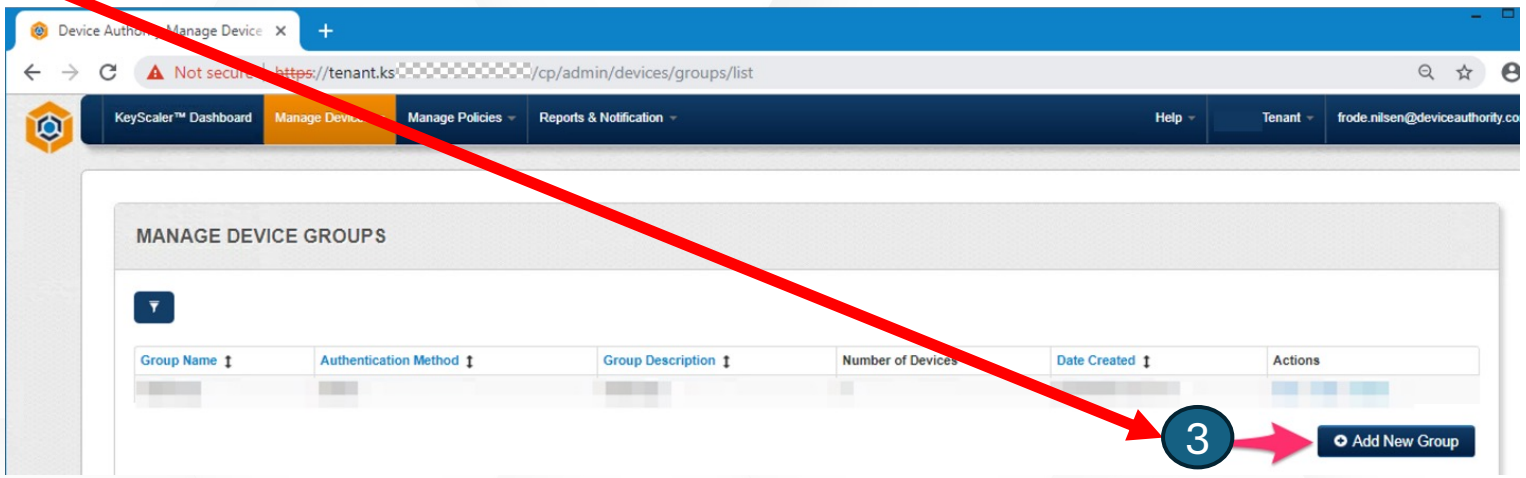
- It is also possible to assign a device to multiple groups, and if those groups have conflicting policies, unpredictable results may occur.

To create a new device group:

Navigate to the Manage Devices pull-down menu 1 and select Manage Device Groups. 2

Select the Add New Group button and complete the information 3





Applying Groups

The Group Name must be unique.

- Use the Description to clarify the **purpose or function of the devices in this group**.

Select the Device Authentication Method: DDKG or PKI Signature+.

- The **Authentication Method** cannot be changed after the group is created.

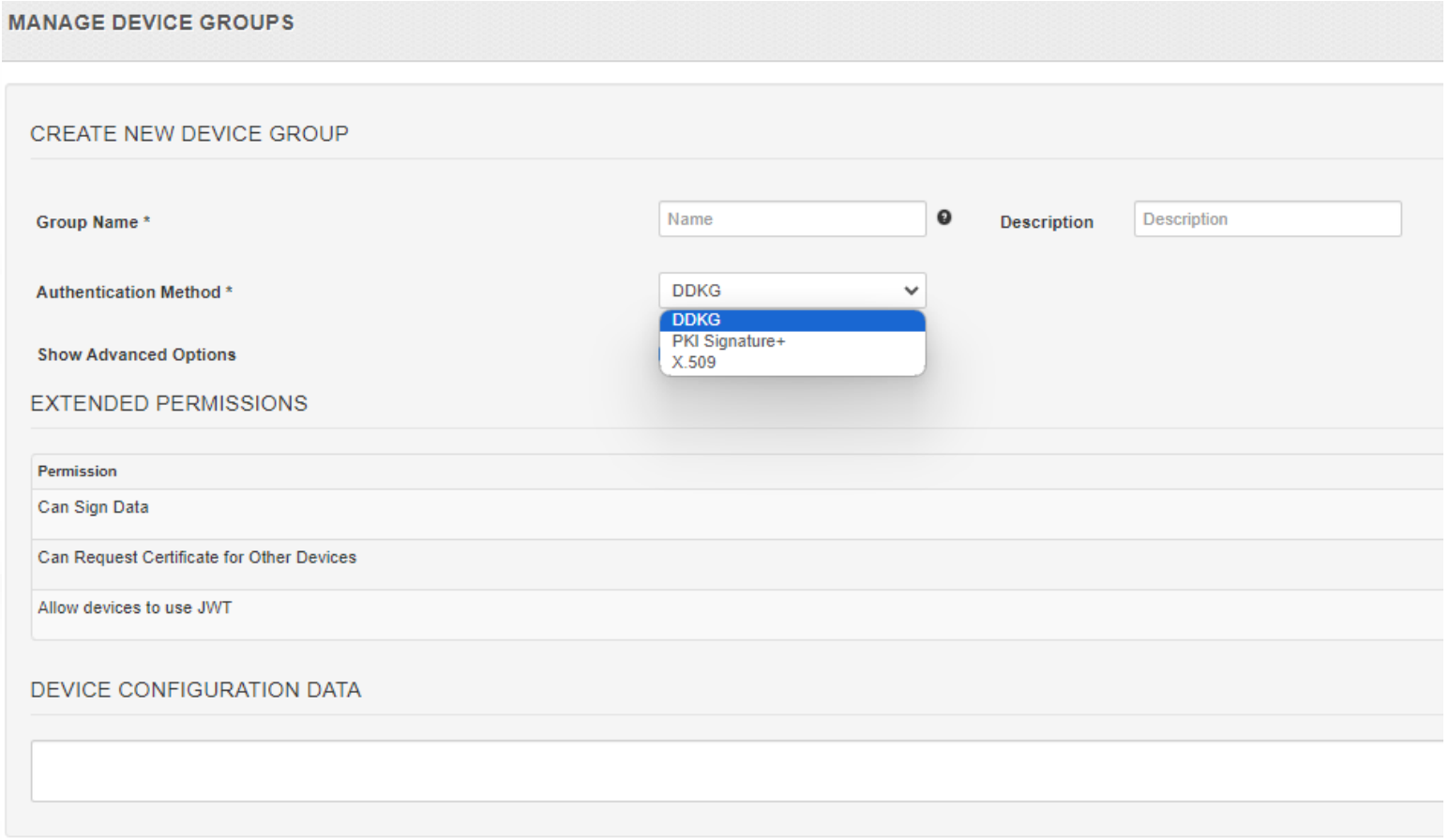
For general Device authentication or KeyScaler Credential Manager use, select DDKG.

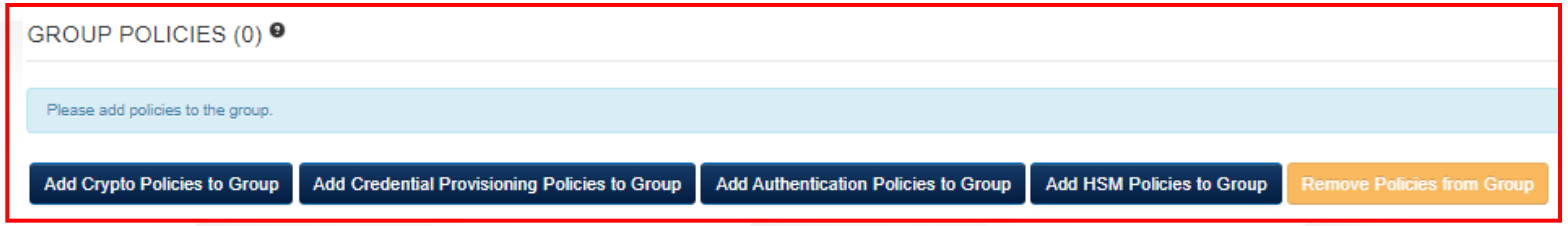
If using PKI Signature+ provide the Authentication Key Rotation Policy.

To add policies (Policies can be added during Group Creation or later by editing the Group.)

- Select the Add Policies to Group" option.
- The Group Policy section will show all available policies.

Check the box next to the desired policy, then select "Add Selection to Group."



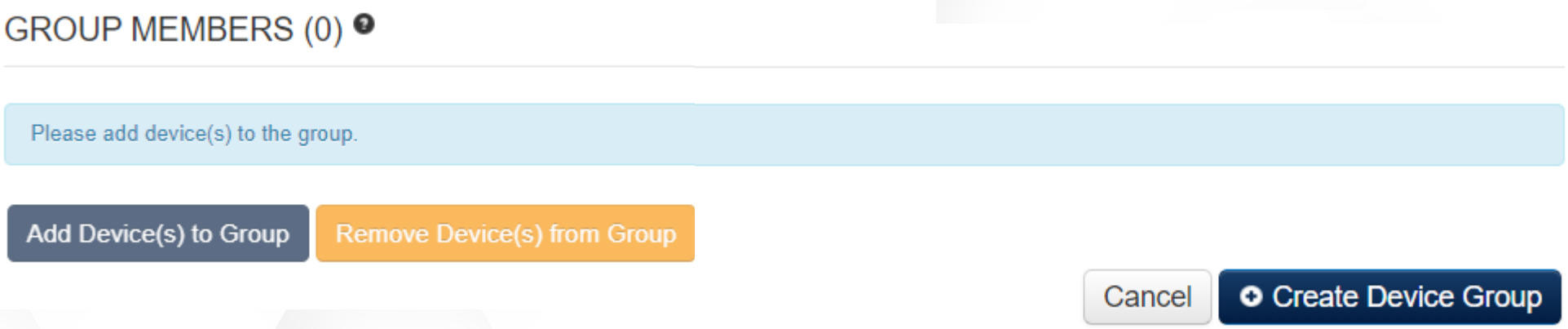


Applying Groups

Add Group Members, by selecting Add Device(s) to Group.

- A list of devices eligible to be added to the group will be displayed.
- Check the box next to the device to add to the group and then select Add Selection to Group.
- Note: The Manage Devices tab shows all devices (without their associated Authorization Id) eligible to be added to the group.

When your selections are complete, select Create Device Group.



Editing Groups

To Edit a group to add or delete policies and/or devices:

- a. Navigate to the Manage Devices pull-down menu and select Manage Device Groups. Select Edit for the group you wish to change.
- b. Add Policies - select the Add Policies to Group button. A list of available policies is shown. Select the desired policies to add and then click Add Selection to Group.
- c. Remove Policies - from the Edit Device Group page for the group you wish to modify, check the boxes next to the policies you wish to remove and then click Remove Policies from Group.
- d. Add Device(s) to Group - select the Add Devices to Group button. A list of devices eligible to be added to the group are shown. Select the desired devices to add and then click Add Selection to Group.
- e. Remove Devices - from the Edit Device Group page for the group you wish to modify, check the boxes next to the devices you wish to remove and then click Remove Devices from Group.

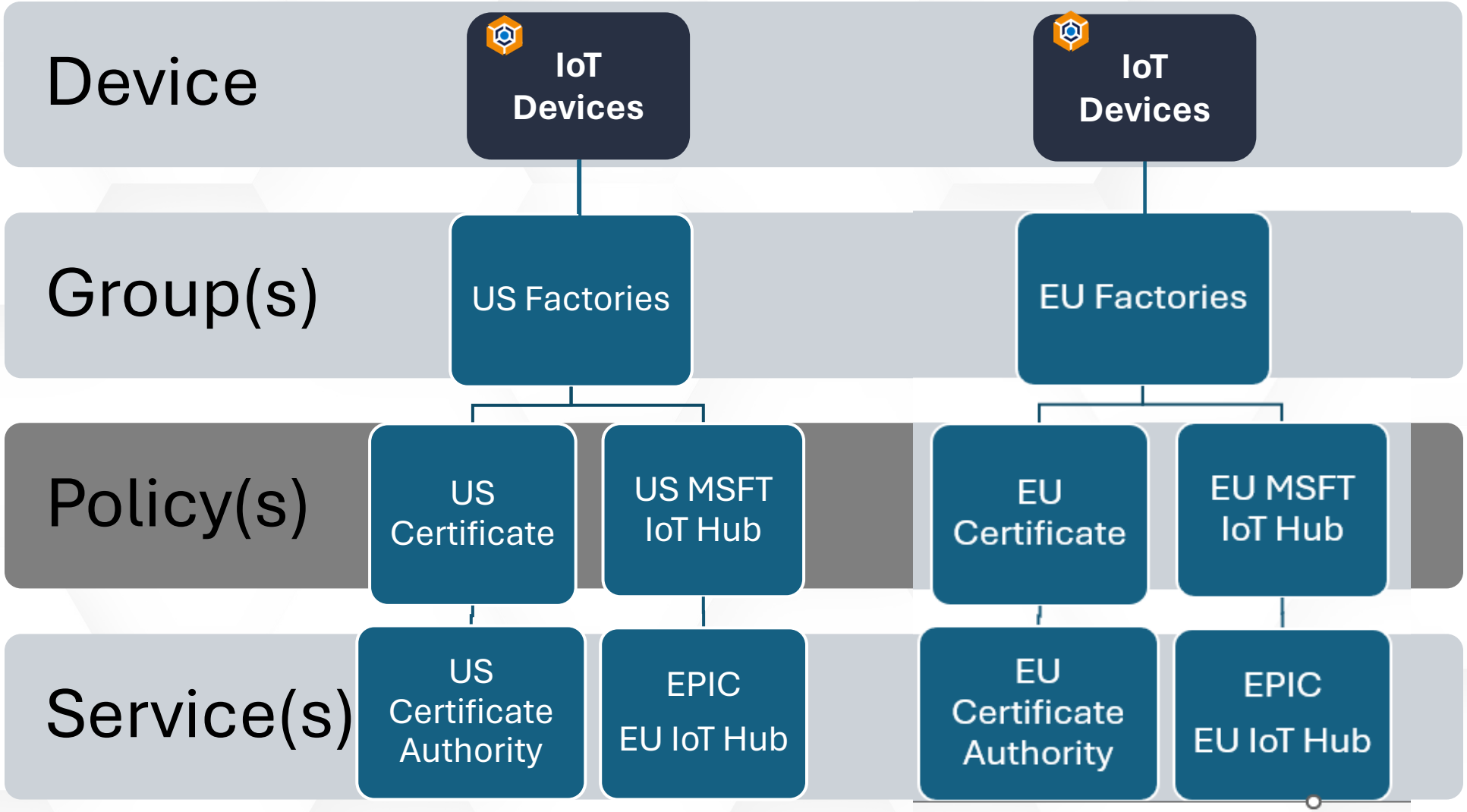
KeyScaler Policies

Section 8

KeyScaler Policies

A Policy is a set of security rules applied to any IoT device

KeyScaler Policies



Certificate Provisioning & Management

Secure Credential Delivery

- Lockdown certificate provisioning through granular authentication and access controls

Automated Certificate Rotation and Management

- Automate certificate renewal and revocation
- Full device security lifecycle management for the lifetime of the device

Supports ANY Certificate Authority

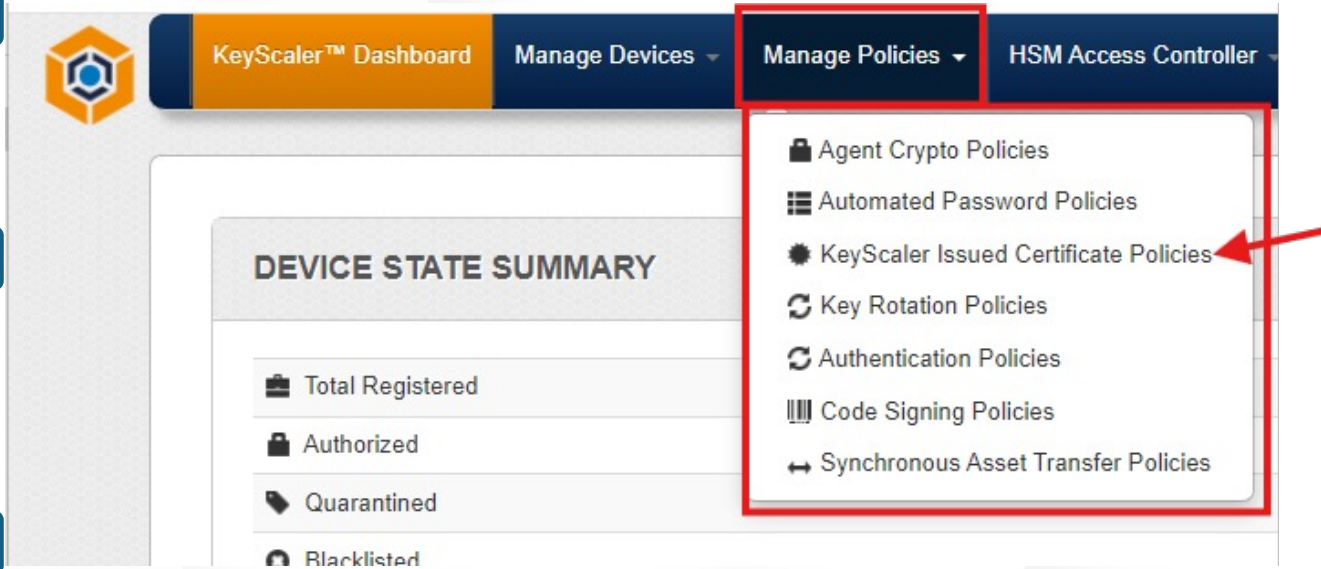
- Support for issuing certificates from both public and private CAs
- Rapidly add support for additional CAs using the EPIC framework

Supports ANY IoT platform

- Automate delivery of issued certificates to any backend platform
- Enables device onboarding and provisioning to 3rd party platforms at scale

Automated Certificate Binding and Authorization

- Automatically assign certificates to specific devices and ensure that they are only used when authorized



DEVICE AUTHORITY

Issued Certificate Policies

Certificate Policies Summary

- KeyScaler Issued Certificate Policies are used by the Device Authority Credential Manager Agent to deliver and manage PKI certificates or manage passwords for client devices.

Creating a New Certificate Policy:

Navigate to the Manage Policies Section:

- In the KeyScaler Control Panel, access the Manage Policies section to view and define all Certificate Policies.

Tabs Overview:

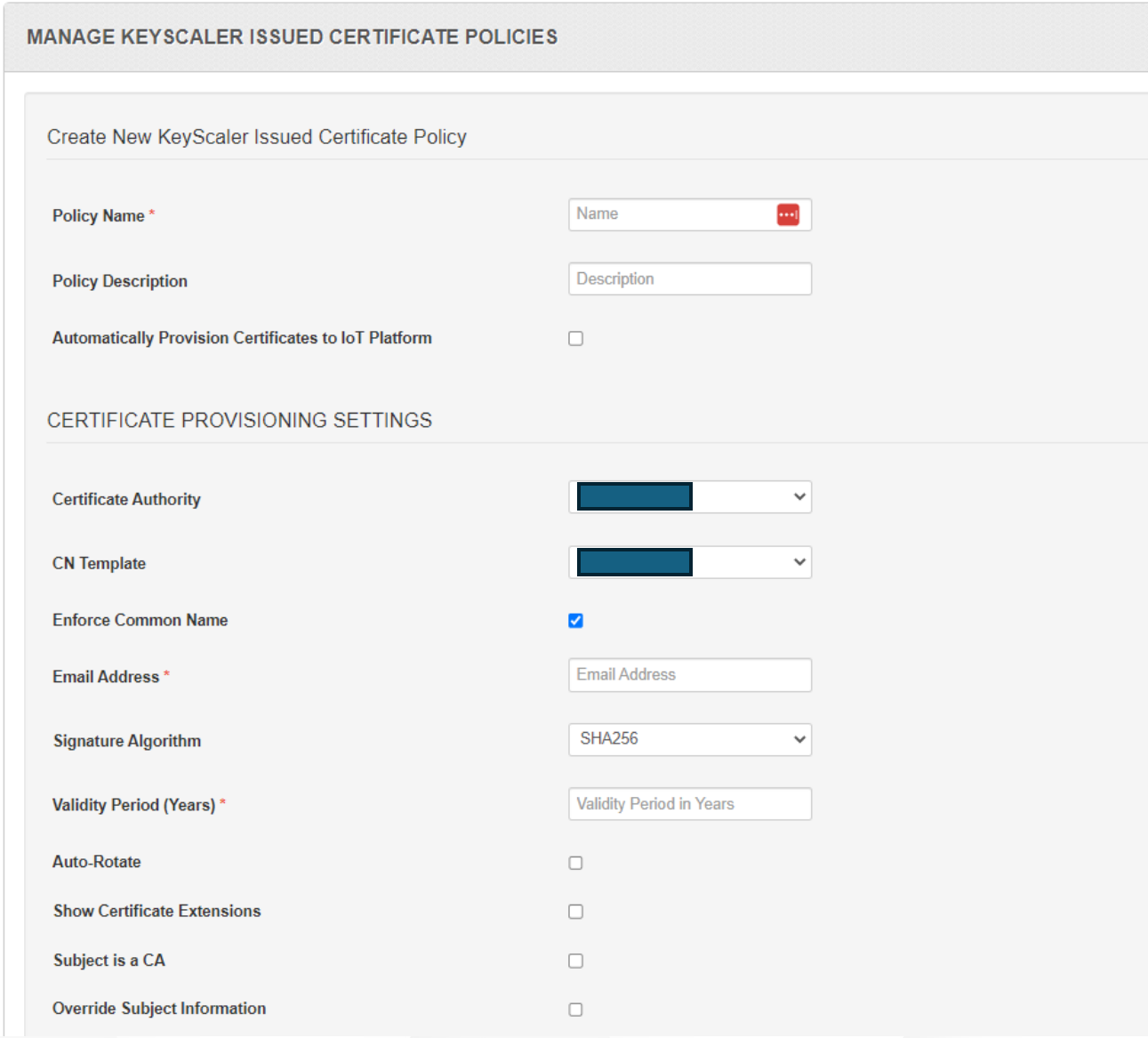
- **Policies Tab:** For managing policies.

Policies | CN Templates

- **CN Templates Tab (Common Name):** For managing CN templates. (Next topic)
- Both tabs work together when using the KeyScaler Issued Certificate Policies feature.

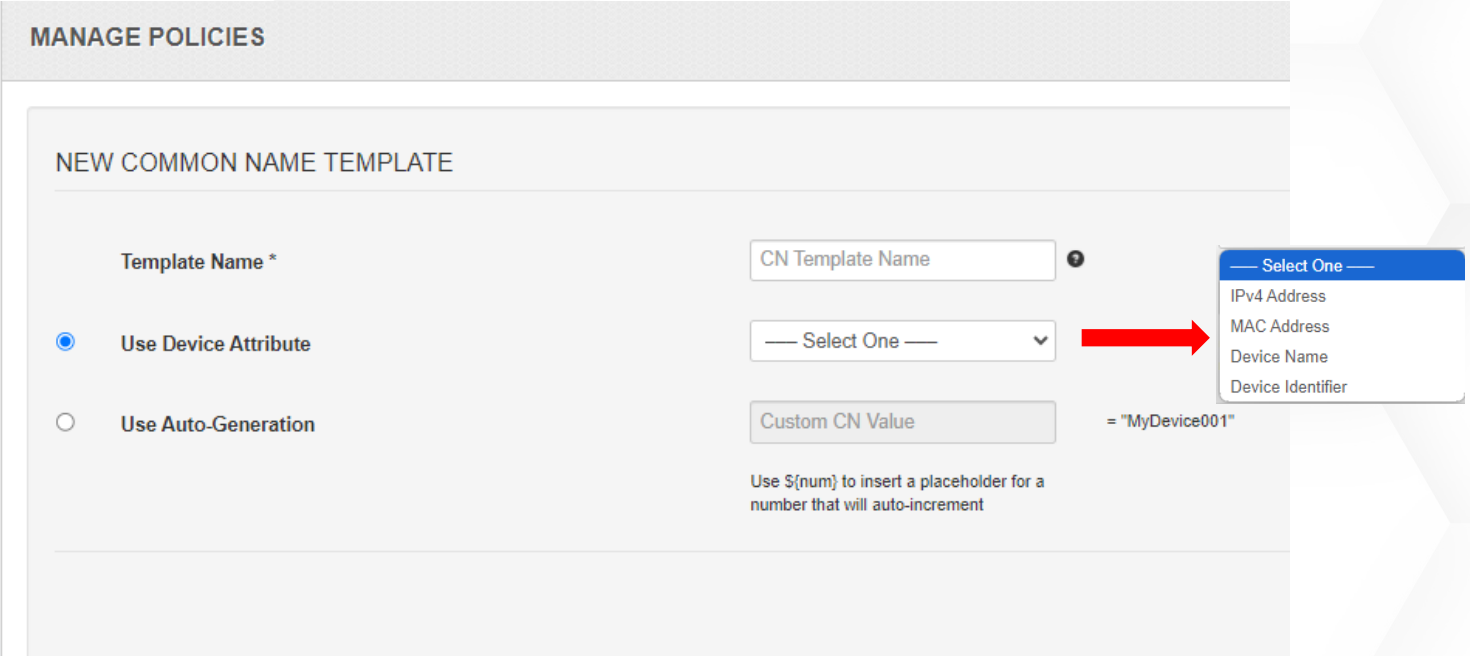
Creating and Editing Policies:

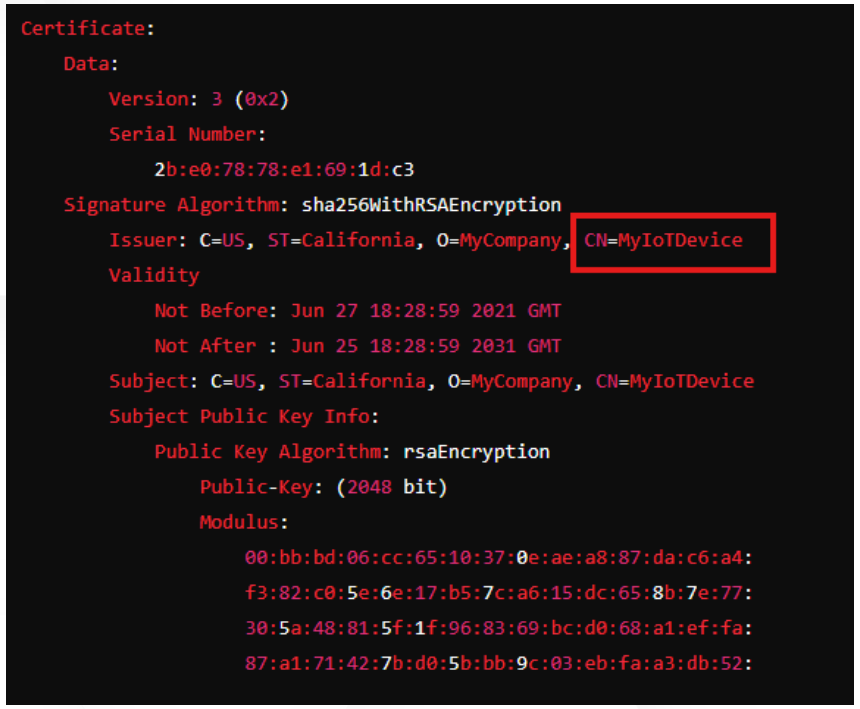
- Use the **New Policy** button to create a new policy.
- To modify an existing policy, use the **Edit** action.



Common Name (CN) Template

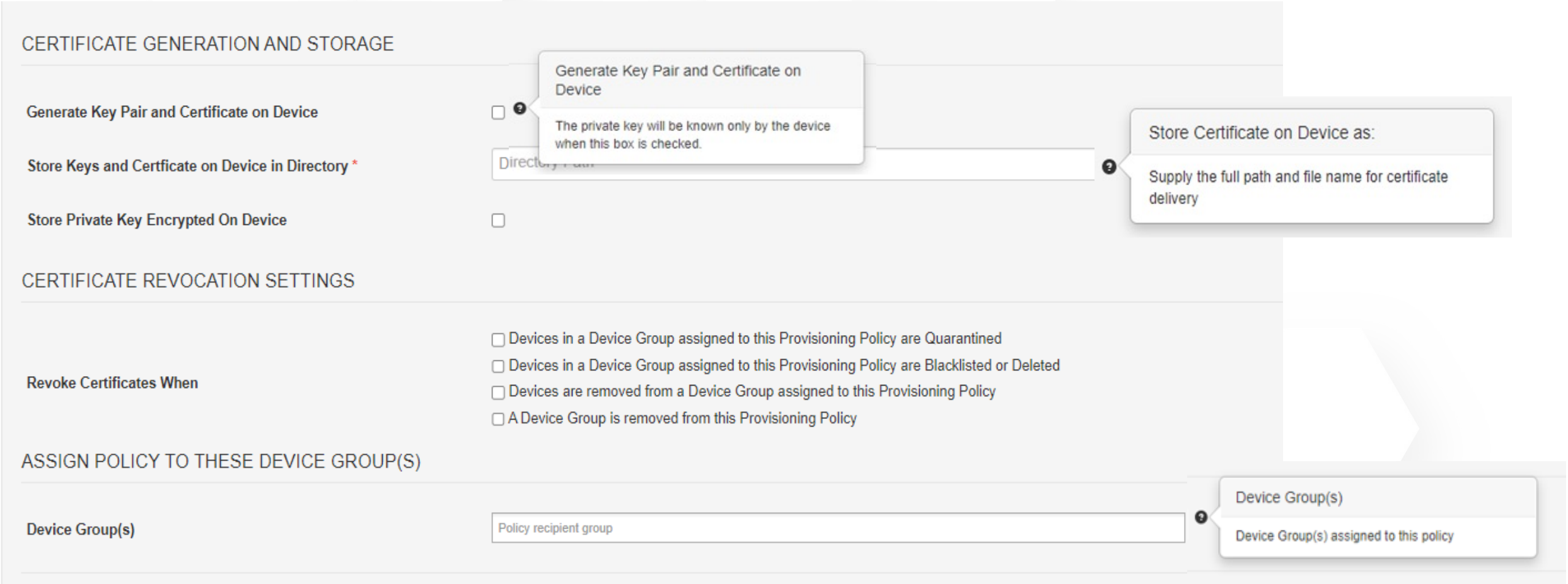
When a Certificate Policy is defined, it requires an existing CN Template to determine what value to use for the CN attribute in the certificates. This is a crucial attribute, as **it identifies your device to external platforms.**





The **Common Name (CN)** identifies the entity (e.g., domain name or individual) the certificate is issued to.

Adding Issued Certificate Policies



DDKG – The Basics

Section 9

Our Intellectual Property, Differentiation

10
US Patents

US 8,438,394
DEVICE-BOUND CERTIFICATE AUTHENTICATION

US 8,464,059
SYSTEM AND METHOD FOR DEVICE-BOUND PUBLIC KEY INFRASTRUCTURE

3
Foreign Patents

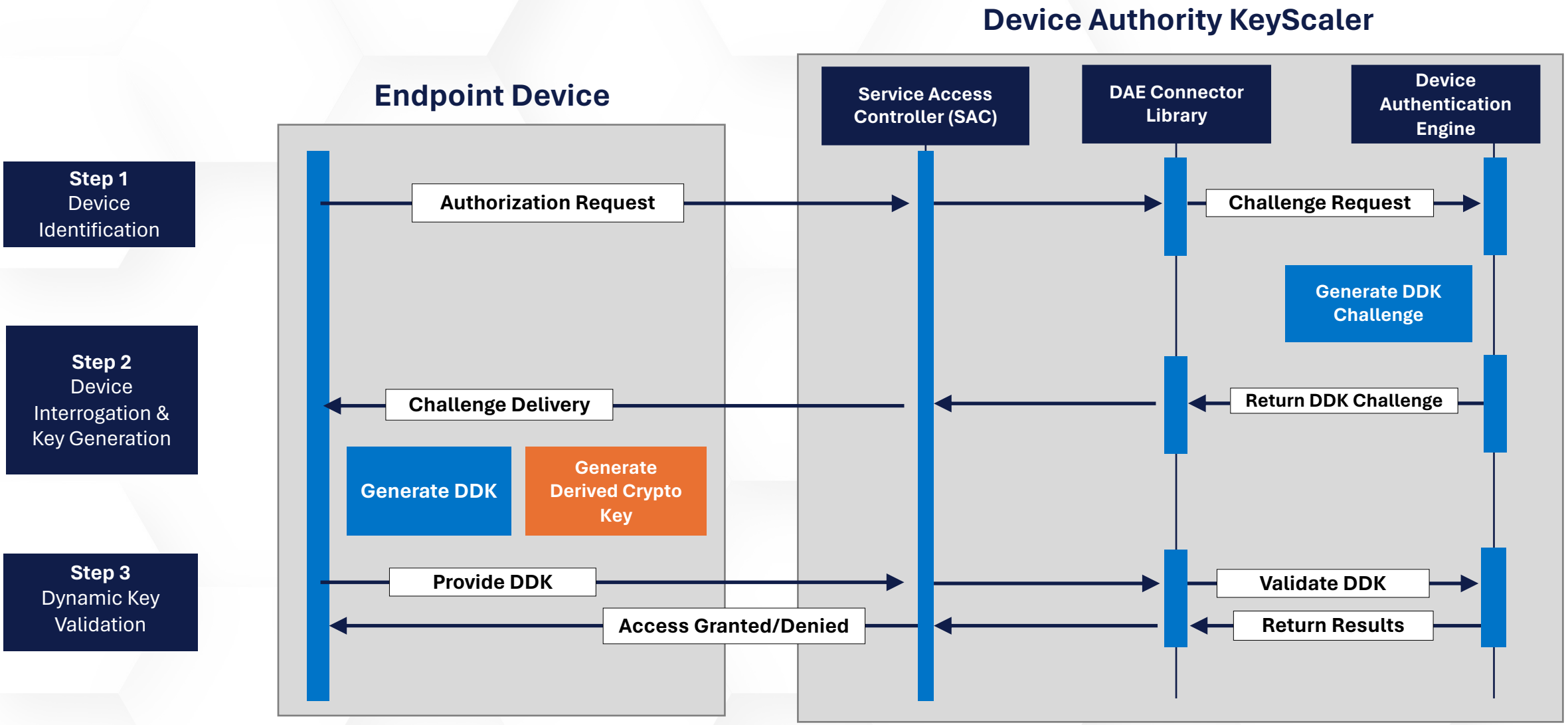
- Device Identity
- Device Authentication
- Device bound PKI
- Data Protection
- Secure Device Communication

Link to Patents Overview

Dynamic Device Key Generation

- The challenge and response mechanism utilizes inherent device entropy to query the physical properties of a device
- Device Authentication Keys are dynamically generated and unique to each device for each authentication session
- Rotating the synthetic key increases key entropy and identifies cloned devices.

Dynamic Device Key Generation



Dynamic Device Key Generation

Challenge & Response Authentication Process

- Patented technology leverages inherent entropy of device components
- Each authentication key is unique to a given device and session

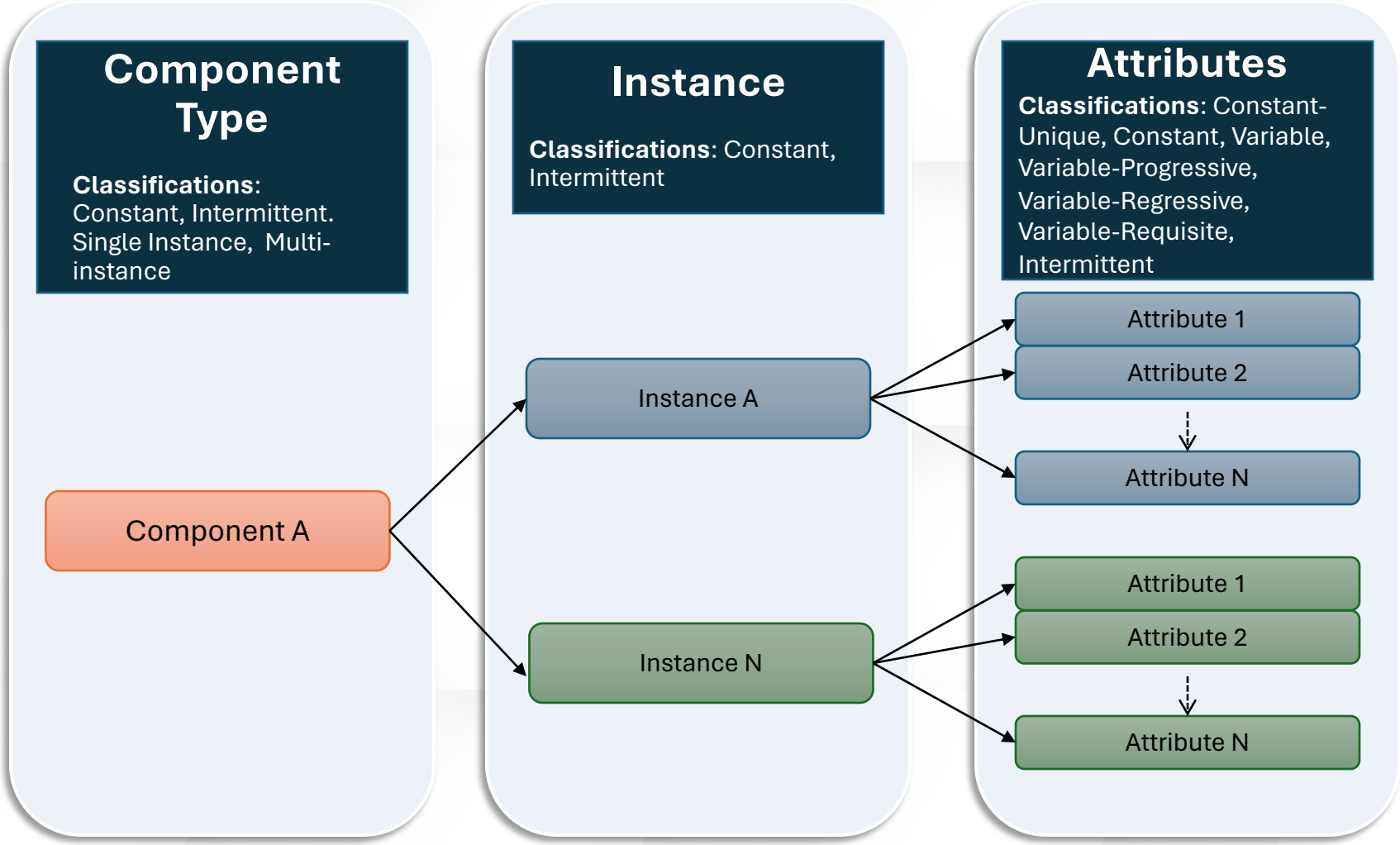
Each DDK Challenge includes:

- Triple-S recipe:
  - Attribute **S**election
  - Attribute **S**equencing
  - Byte **S**ampling

Synthetic Device Key Material

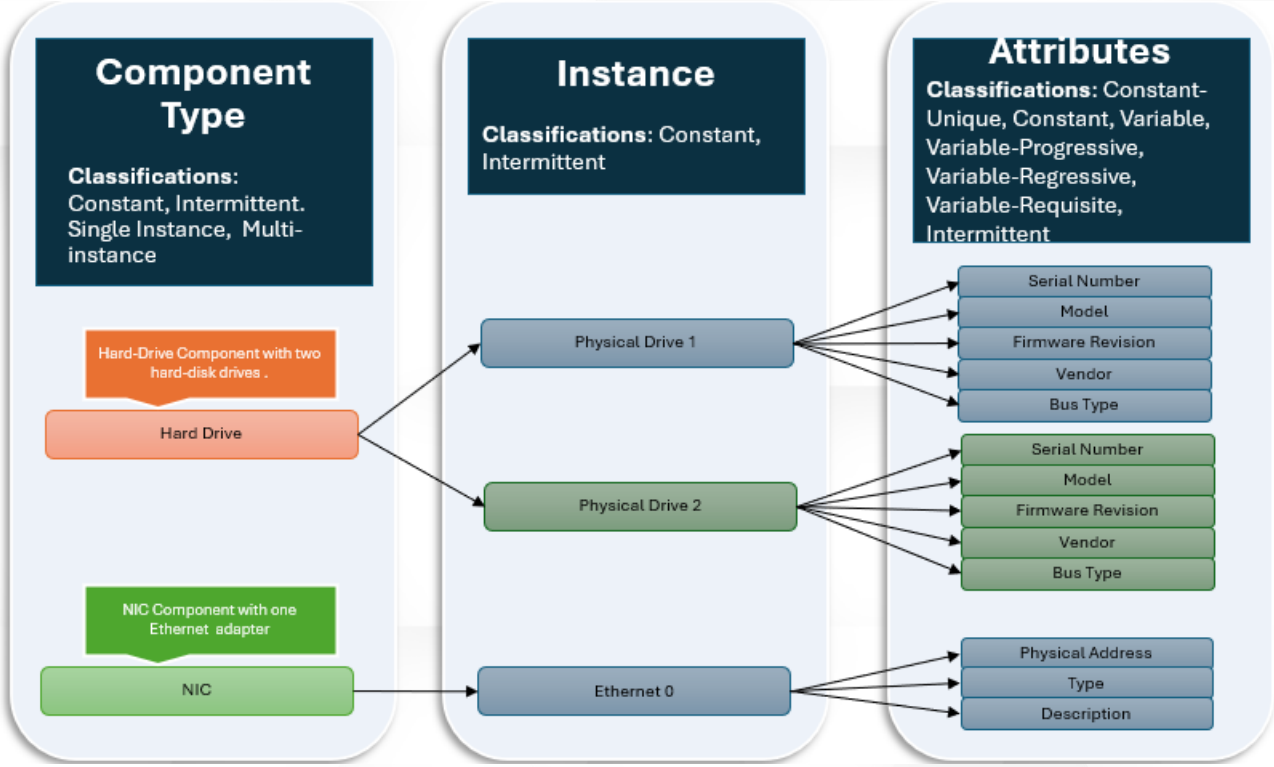
- Increases DDK attribute entropy
- Provides cloning/spoofing detection
- Provides DDK challenge protection
- Includes Hashing, device attribute salt and noncing

DDK Composition - Device Attribute Model



DDK Attribute Model Example

Why are Component type, Instance, and Attributes important to DDKG?



Being installed at the root level, DDKG can interrogate all components.
DDKG knows what is available, how many of each, and each of their attributes, such as serial number, model, version, etc.

All of this information is used to create the Dynamic Device Key. (DDK) Therefore, once a device has registered, if any of the component type, instances, or attributes change, its next attempt to authenticate to KeyScaler will fail!

Device Authority Add Device Pr

https://jamesdemo.cp.keyscaler.com/cp/admin/devices/addprovision

Provision a Device | Bulk Provision Devices | Applications

DEVICE REGISTRATION CONTROLS

Restrict By

	Registration Method *	DDKG
☑	OS Type	Linux
☑	Device Identifier	ABCD-EFGH-IJKL
☑	MAC Identifier	AA:BB:CC:DD:EE:FF
☑	IP Address	192.168.1.1
☑	Application Identifier	serial-number 1234567890
☑	Valid From	03/01/2019 17:44:18
	Valid Until	03/03/2019 17:44:18
☑	Geofence	37.554610 -121.976670 100 meters

Device Identifier

Your device unique identifier

MAC Address

XX:XX:XX:XX:XX:XX

IPv4/IPv6 Address

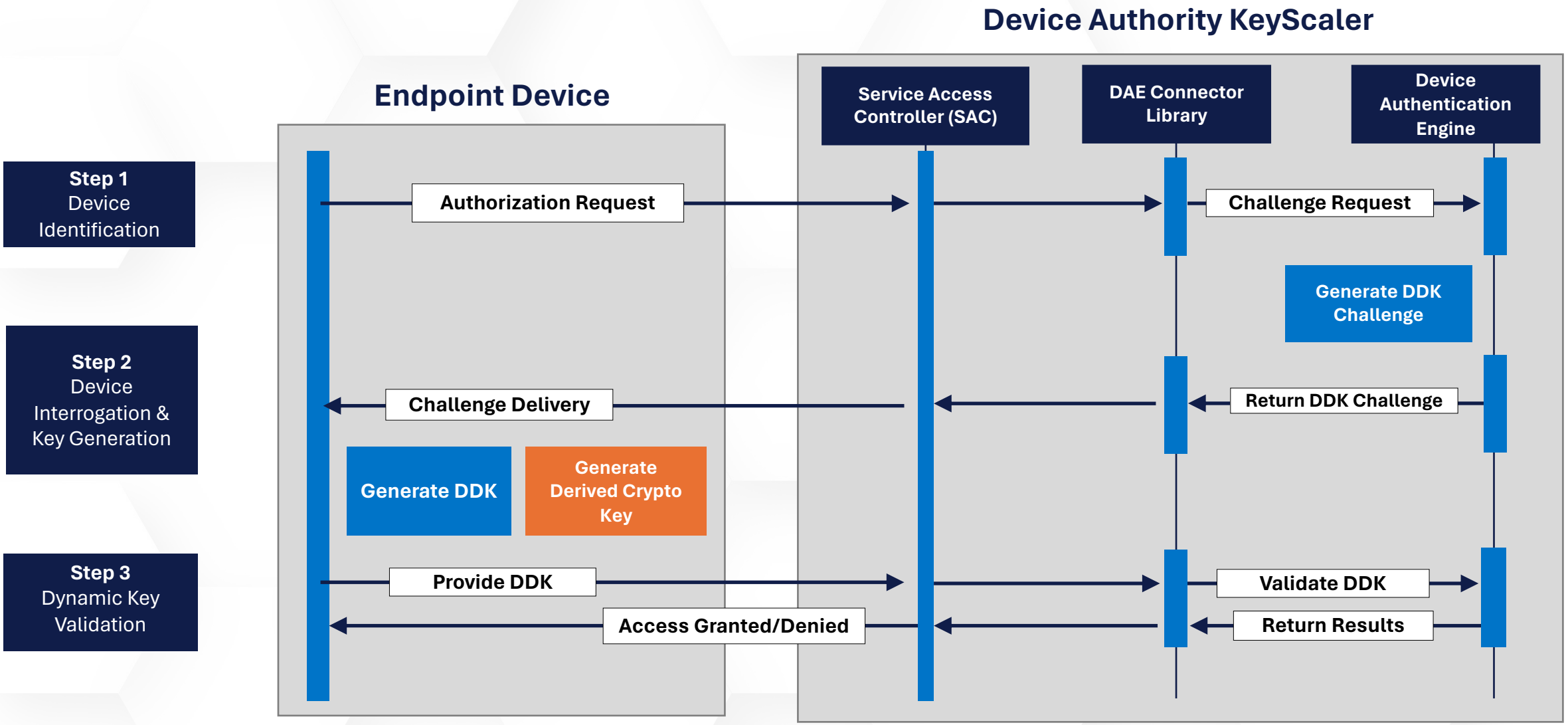
IPv4 - may be truncated
192.168.1.1
192.168
192

IPv6 - in compressed format; may be truncated
FE80::0202:B3FF:FE1E:8329
FE80::0202:B3FF:FE1E

DEVICE AUTHORIZATION CONTROLS

Authorization Identifier: MyDevice001

Dynamic Device Key Generation



KeyScaler Administrative Functions

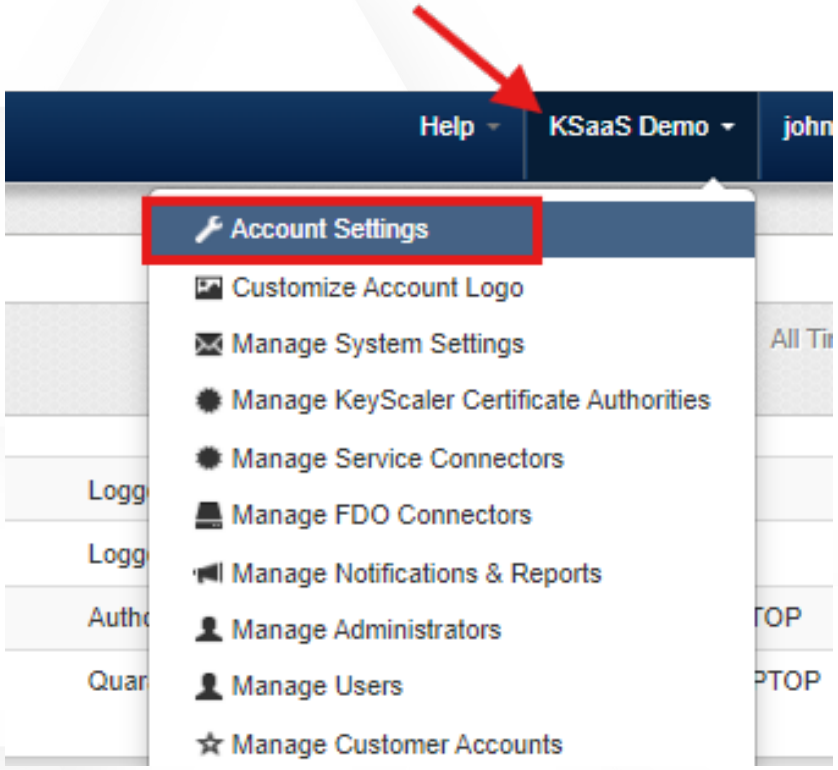
Section 10

Using the KeyScaler Administrative Interface

- Overview of the KeyScaler admin interface
- Navigating the interface

Manage Tenant Account Settings

- To manage Tenant Account settings, select "Account Settings" under the tenant company menu to navigate to the Account Settings from the top menu pull down under your organization name.



Account Settings Overview

Integration Info Section

- This read-only section provides tenant account information required for integrating device authentication using the DAE Connector, including the account number, participant ID, and secret for API calls.

General Settings Selection

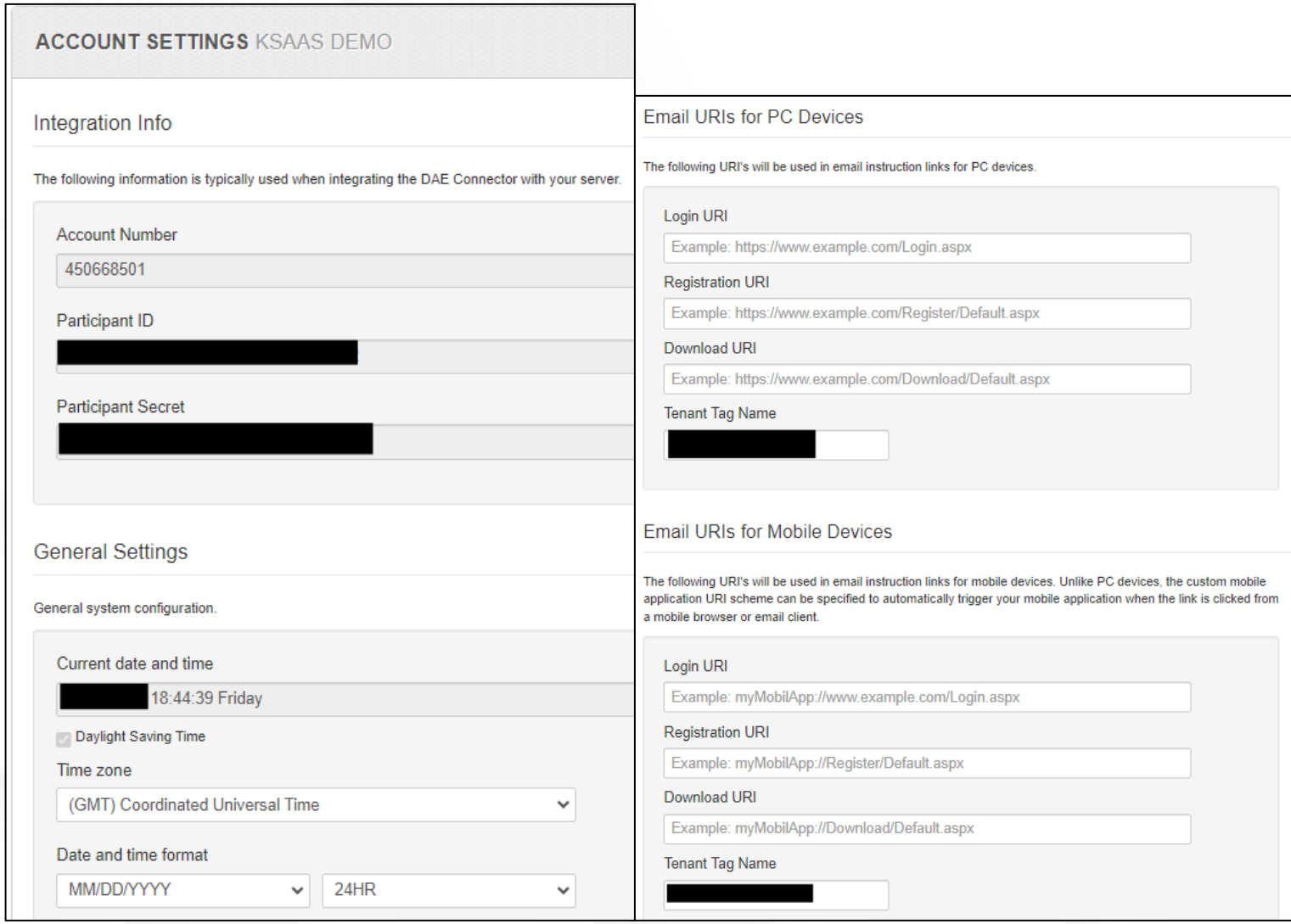
- Controls the format of date and time displayed on the Management Control Panel, allowing changes to time zone and format preferences.

Email URIs for PC Devices

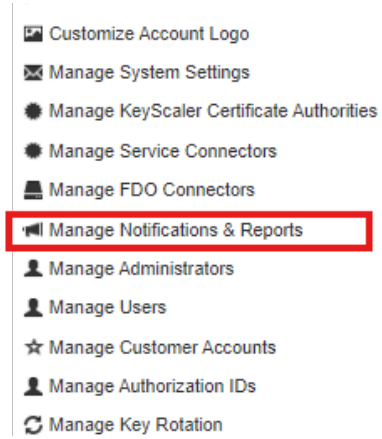
- Manages the URIs used in registration emails for non-mobile devices (PCs and Macs), including links for device registration, downloads, and authentication.

Email URIs for Mobile Devices

- Controls the URIs used in registration emails for mobile devices, providing links for downloading apps and device registration if mobile authentication is used.



Manage Control Panel Notifications

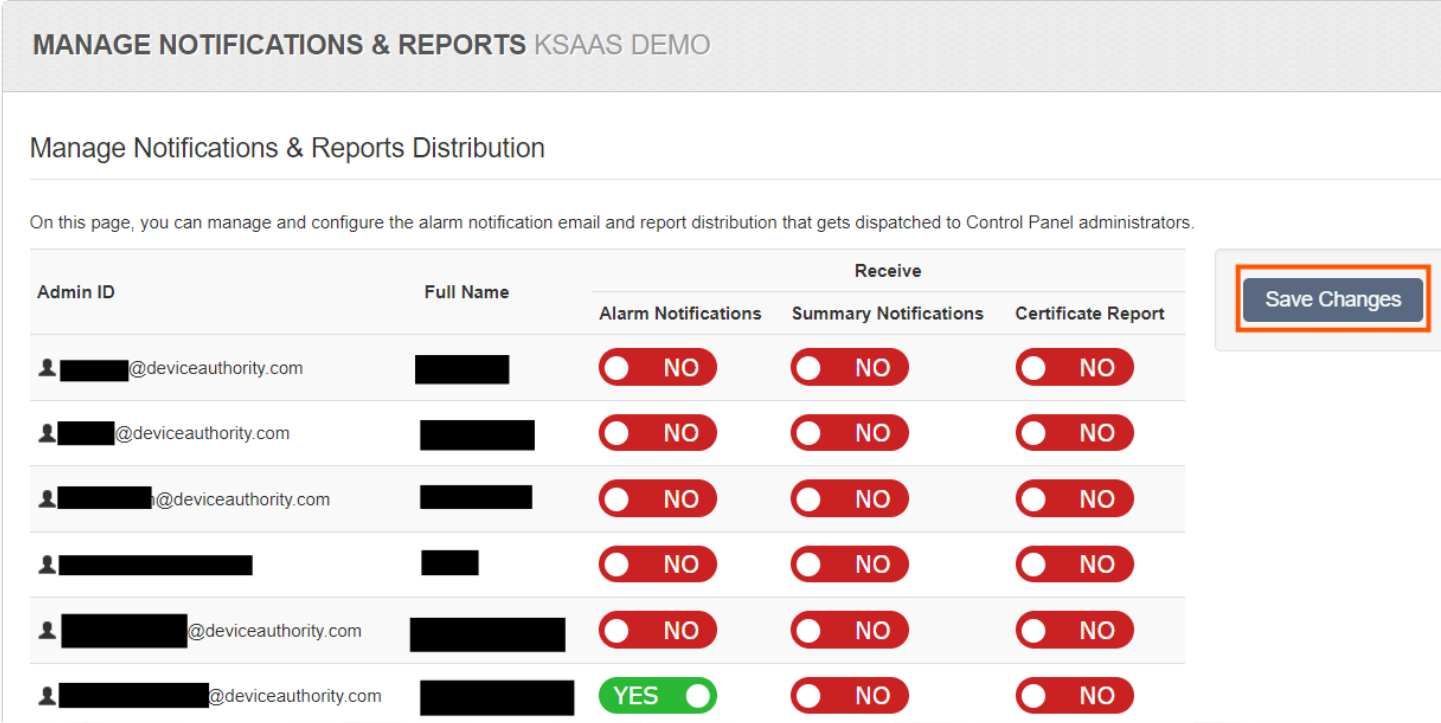


Alarm Notification

- Manage which administrators receive alarm notifications by enabling or disabling alerts for each account on the Alarm Notification Distribution page, then save your changes.

Email Notification

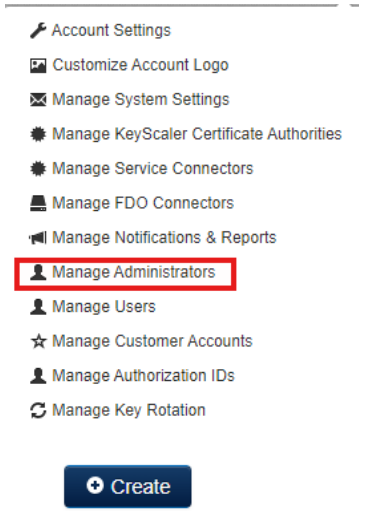
- Customize the content of email notifications for different events using the rich-text editor and system variables available under the "Show System Variables" option.

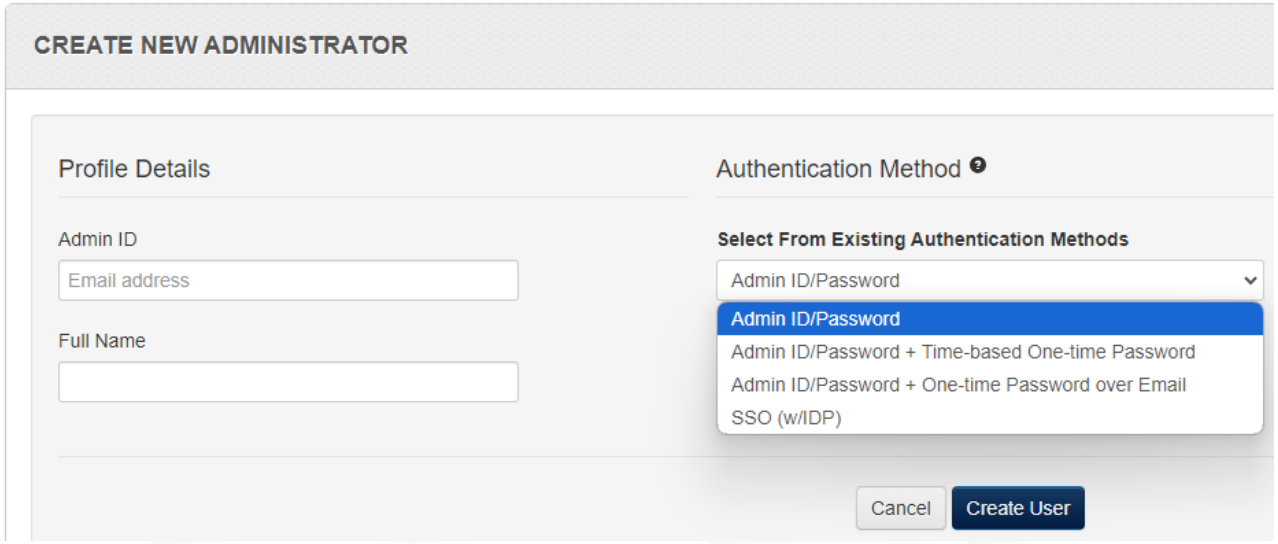


Add a New Administrator

To add a new Control Panel Administrator account:

1. Click the "Create" button on the right-hand side of the Manage Administrators page.
2. Fill in the "Create New Administrator" form with the email, initial password and full name of the administrator.
3. Select an Authentication Method from the pull-down menu. Provide details about the device the new administrator will use to access the Management Control Panel once they register their device.
   a. If *Username/Password + Time-based One-time Password* is chosen, the admin must install and configure Google Authenticator (available from the App store) with the shared secret delivered in the Admin Invite email.
   b. If *Username/Password + One-time Password over Email* is chosen, the admin will receive an email every time they log in. The email includes a one-time password that must also be entered for CP access. This one-time password is valid for 5 minutes only. (Note: If you need to extend this time period, there is a parameter for the cp.properties file that can be customized.
   Consult support@deviceauthority.com for instructions in customizing the otp.timetolive parameter.
   c. If *Username/Password + Device Authentication (DDKG)* is chosen, also select the platform type and time to live value for the device registration record. An Admin Invite email will be sent to the new admin along with instructions on how to register his/her device. Registration of the admin's device must occur within the time to live window.





Managing License Alarm Settings

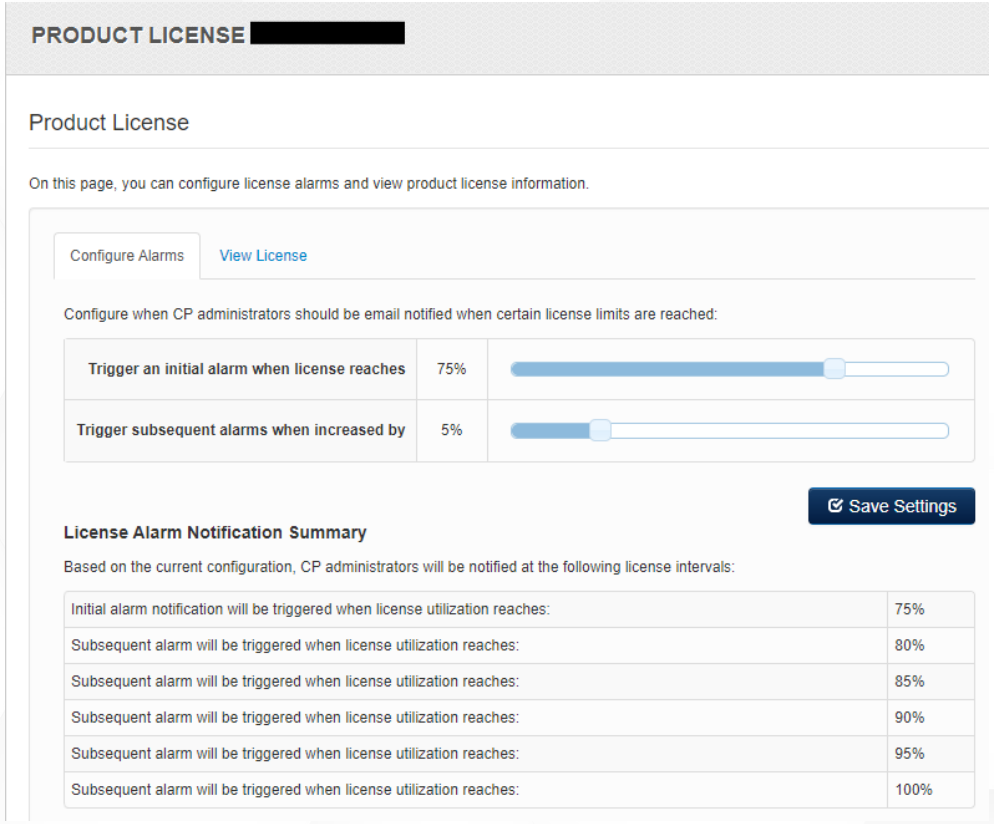
The Management Control Panel administrator can dispatch alarms when a certain license threshold is reached.
This feature can help plan license upgrades if an upsurge of device registrations occurs, consuming more licenses than initially anticipated.

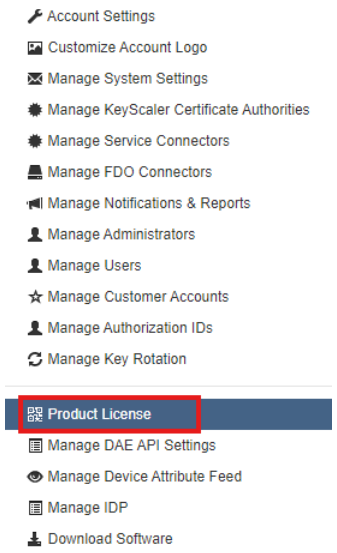
Configuring when alarms are dispatched is accomplished in two ways:

1. By selecting when the first alarm should be dispatched based on the overall percentage of licenses consumed.

2. By selecting when the subsequent alarms should be dispatched after a specific percentage of license is consumed from the initial threshold.

To do so, configure the initial and subsequent alarm percentages using the sliders and click the Save Settings button to apply the settings.





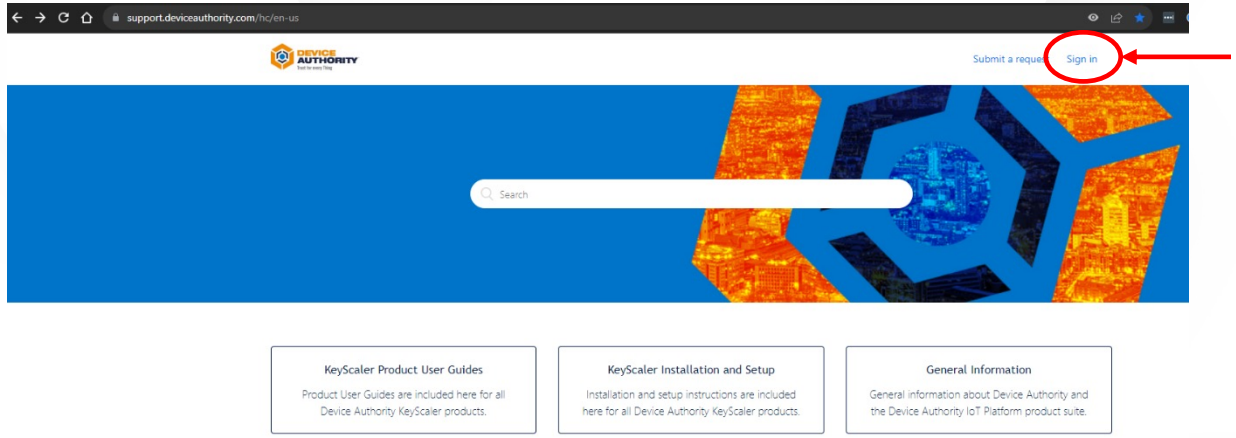
Submitting a Support Ticket

Section 11

Customer Portal Sign in

Steps:

1) Goto - https://support.deviceauthority.com/
2) Click Sign in

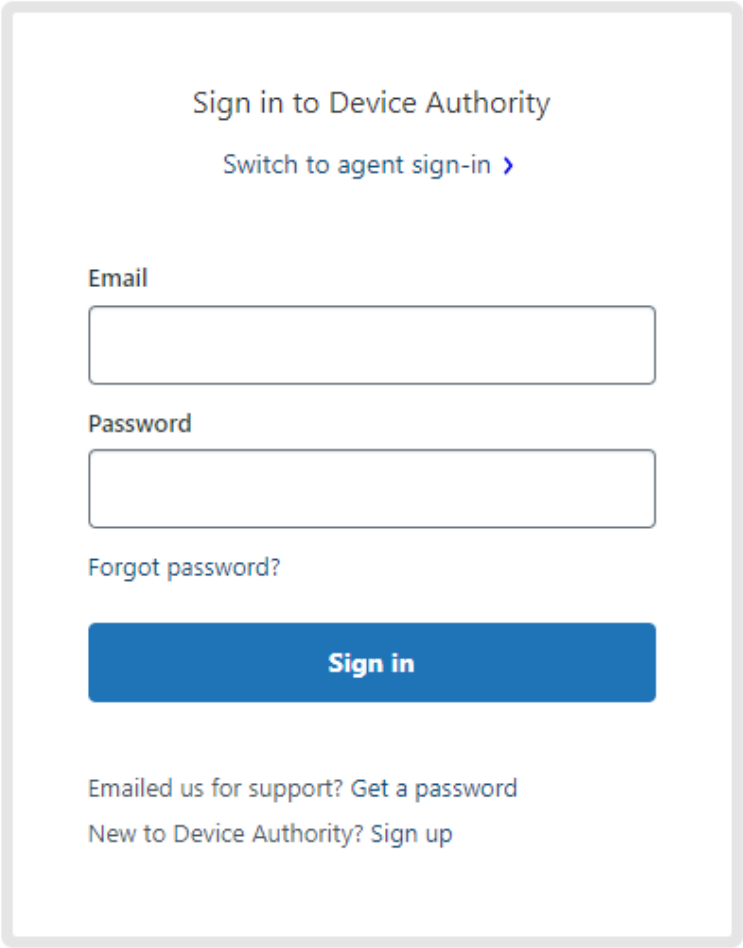


Open a new ticket to request users added to the Portal/Zendesk ticketing system

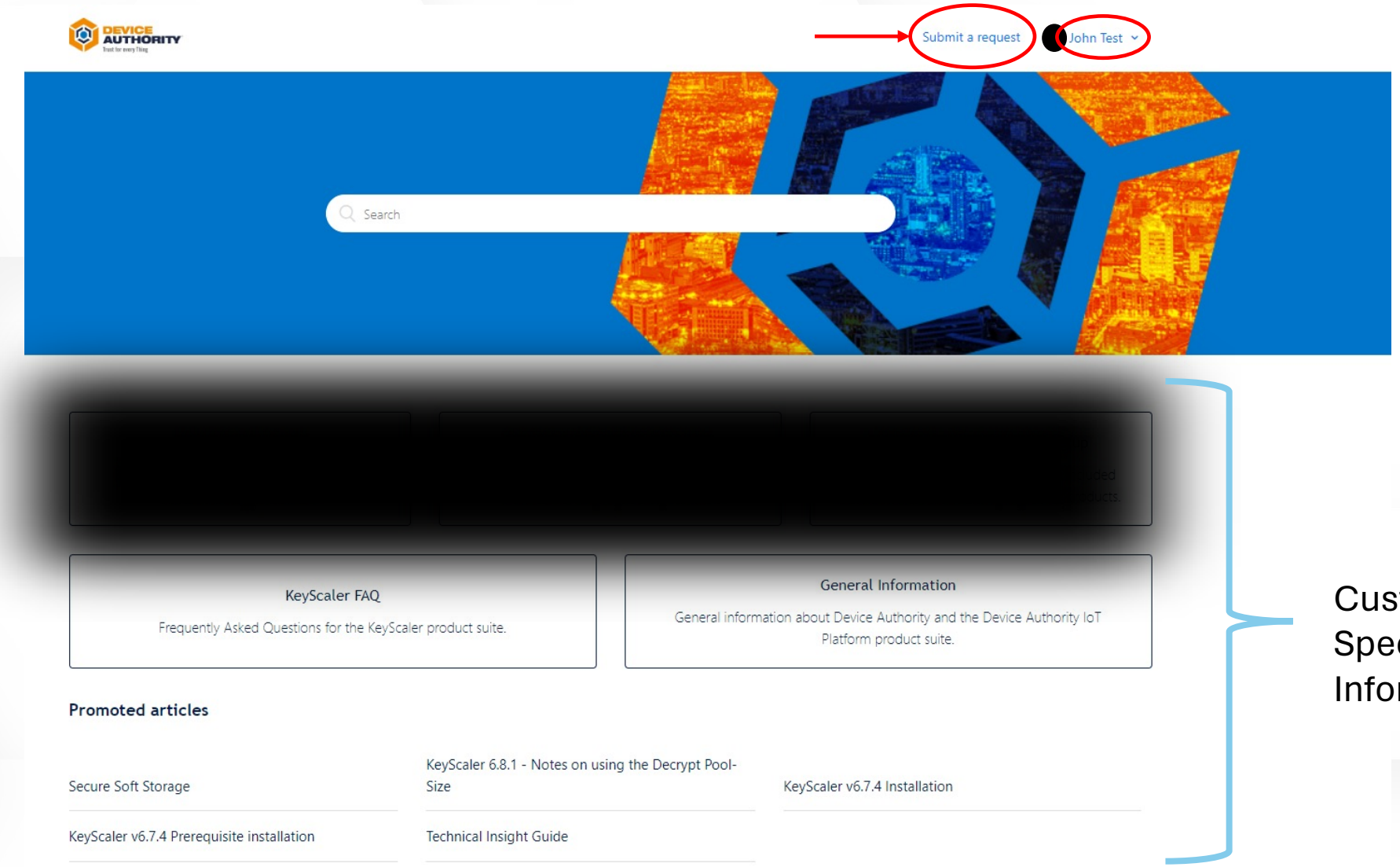
1) Include their Full Name and Email address
2) Can view All tickets or just their tickets
3) Same process to remove users

DEVICE AUTHORITY

Customer Portal Login Screen



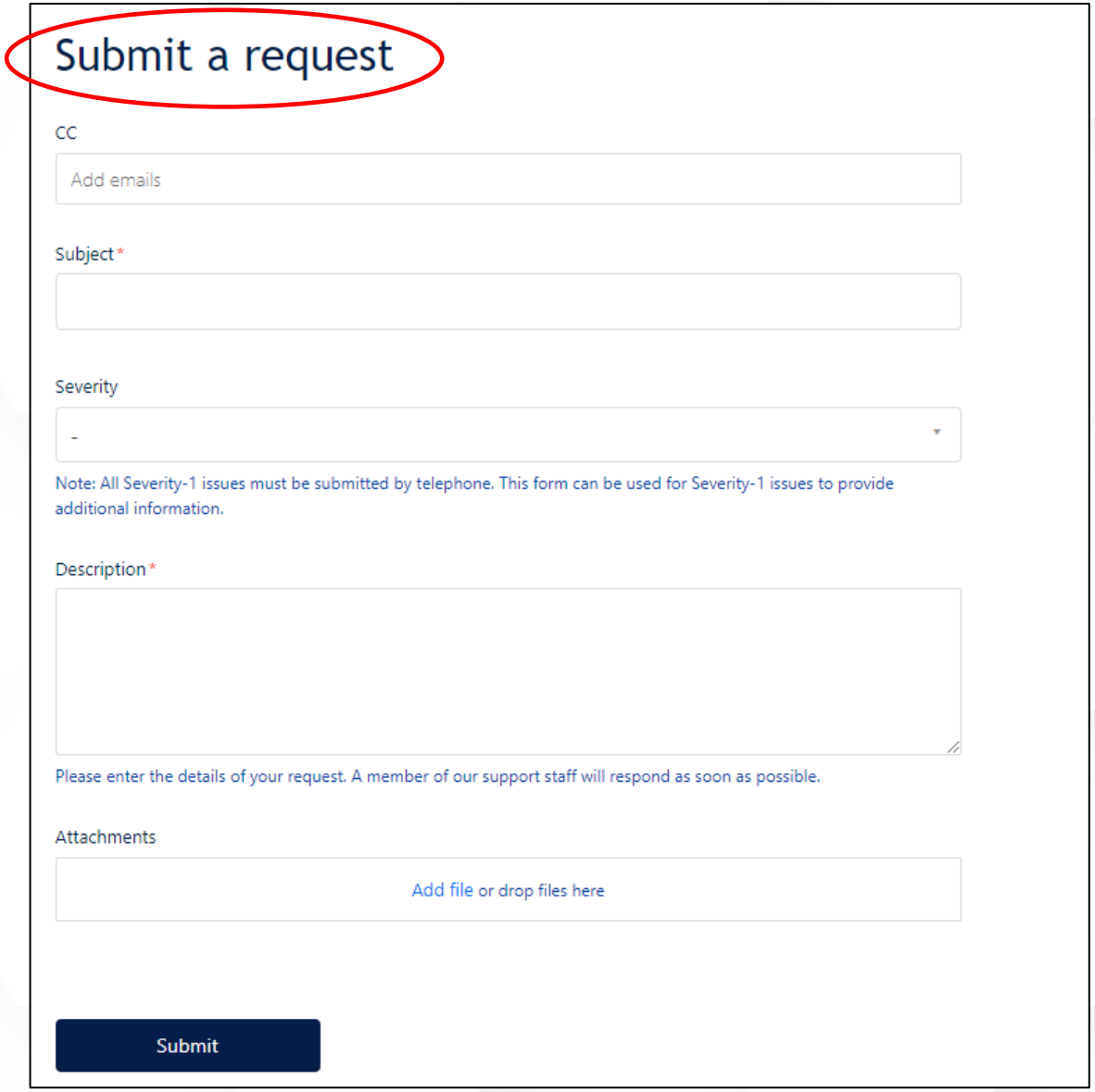
Submit a Request (ticket)



Customer Portal Specific Information

DEVICE AUTHORITY

Raise a New Ticket for Each Incident / Question or Task



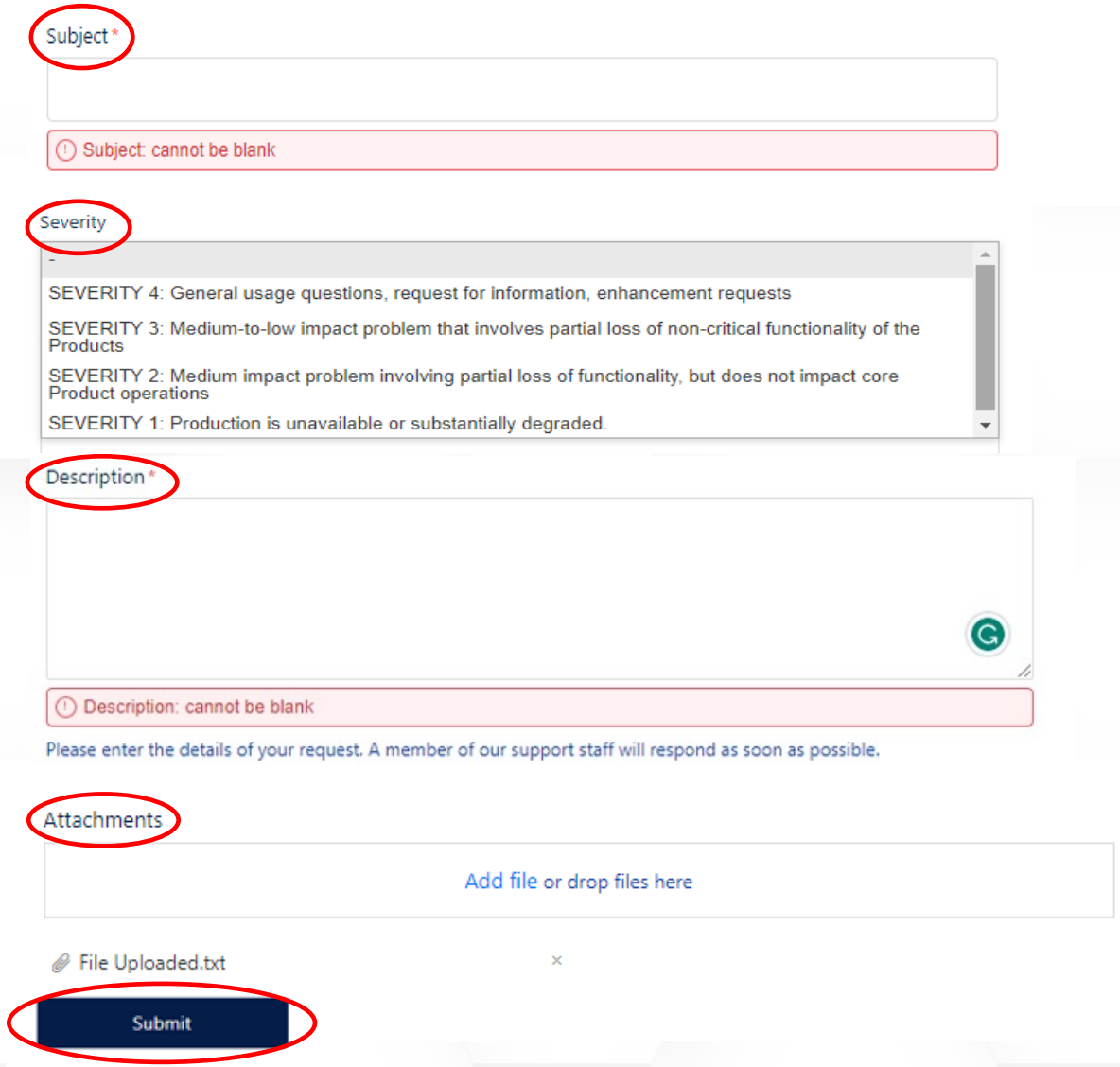
What to capture

- Cc (Not required)
  - Add others from your organization
- Subject Field (**Required**)
  - Summary of the description entered below
- Severity (**Required**)
  - Used to indicate the nature and urgency of the ticket
    - Ranging from a question to production-impacting
- Description (**Required**)
  - What were you trying to achieve?
  - What was the expected outcome?
  - What was the actual outcome?
- Attachments (Not required)
  - Upload all relevant files

- Enter as much information as possible
- Attach all the relevant Log files
- If the issue is repeatable, include the steps to re-create it
- Where the issue occurred.
  - KSaaS CP
  - Device
- Current CM Release

DEVICE AUTHORITY

Five steps to submit a ticket



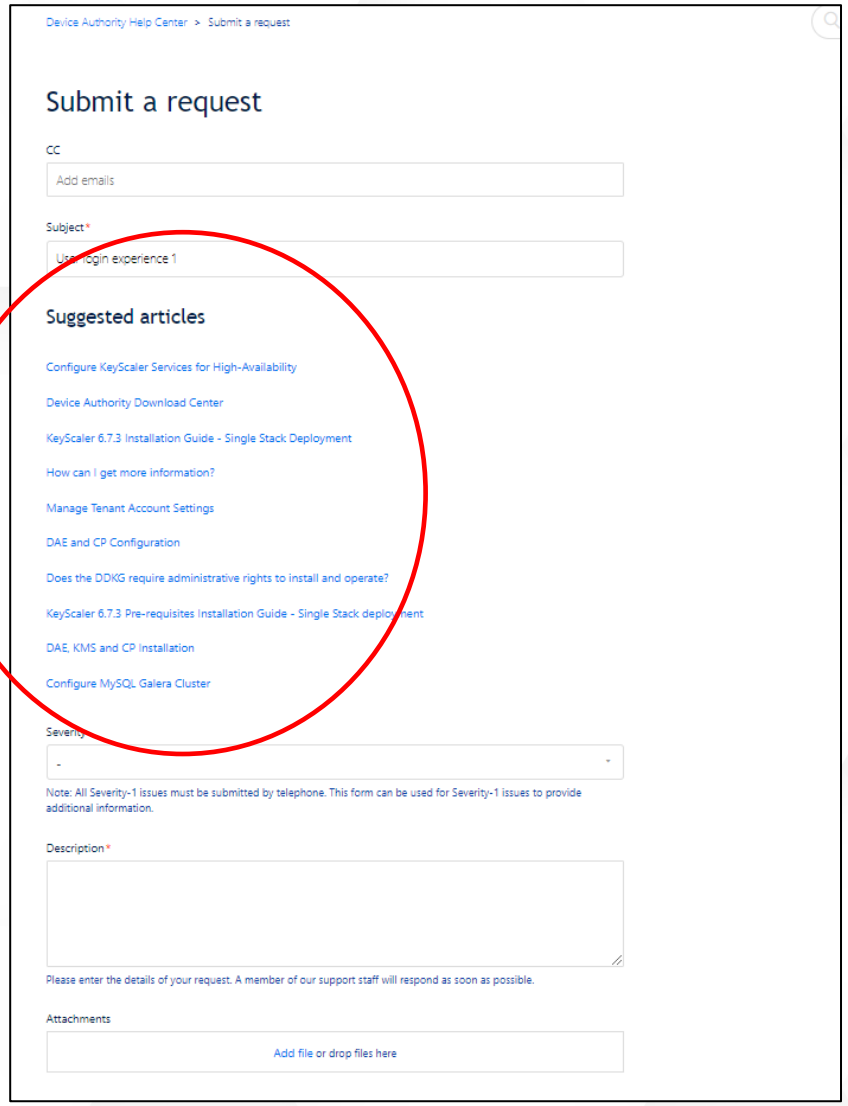
What to capture

- Cc (Not required)
  - Add others from your organization
- Subject Field (**Required**)
  - Summary of the description entered below
- Severity (**Required**)
  - Used to indicate the nature and urgency of the ticket
    - Ranging from a question to production-impacting
- Description (**Required**)
  - What were you trying to achieve?
  - What was the expected outcome?
  - What was the actual outcome?
- Attachments (Not required)
  - Upload all relevant files

- Enter as much information as possible
- Attach all the relevant Log files
- If the issue is repeatable, include the steps to re-create it
- Where the issue occurred.
  - KSaaS CP
  - Device
- Current CM Release

DEVICE AUTHORITY

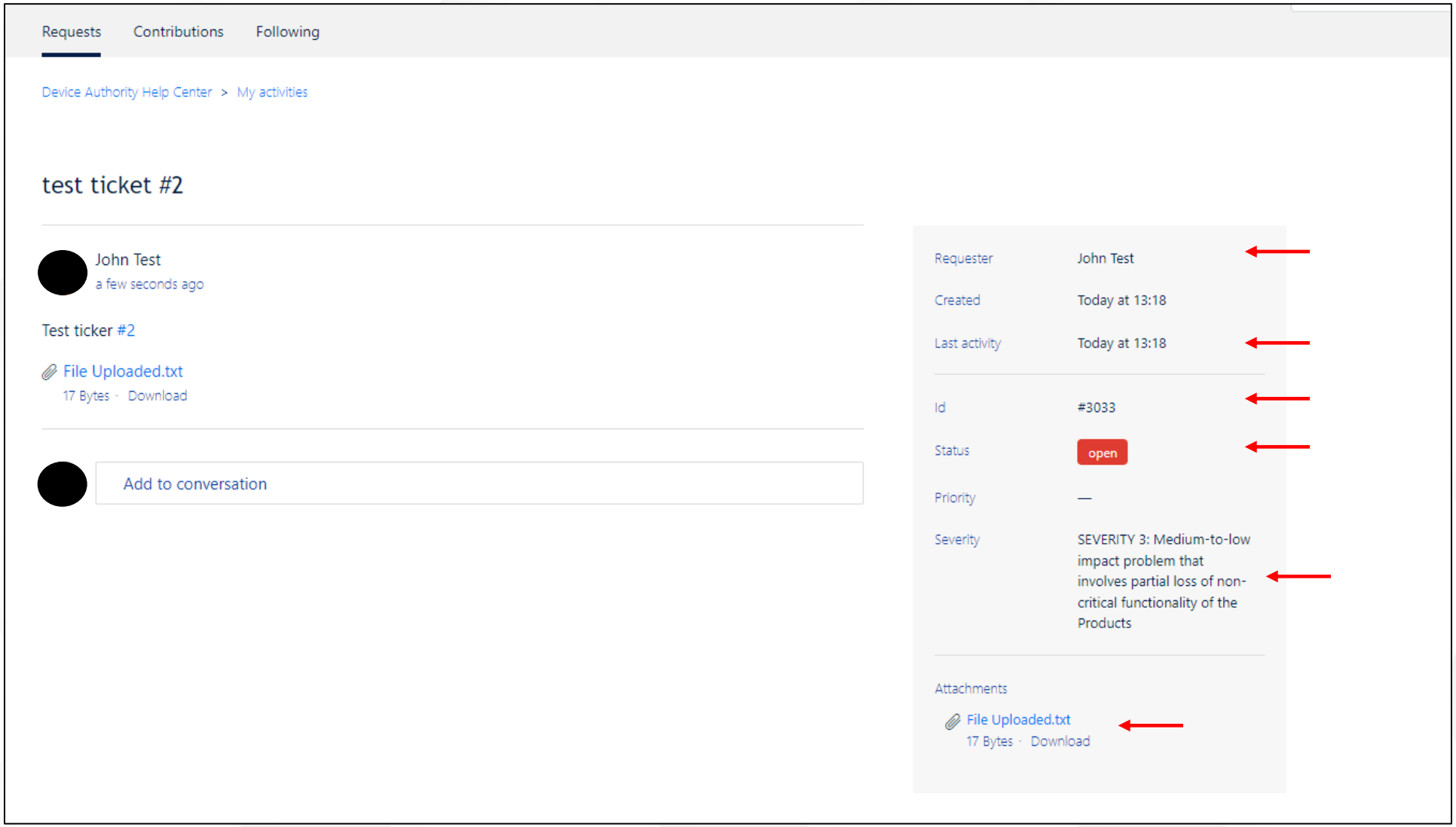
Suggestions for Ticket Submissions



- **When filling in the Subject, note the suggested articles**
- **Resolution may be found without submitting a ticket**

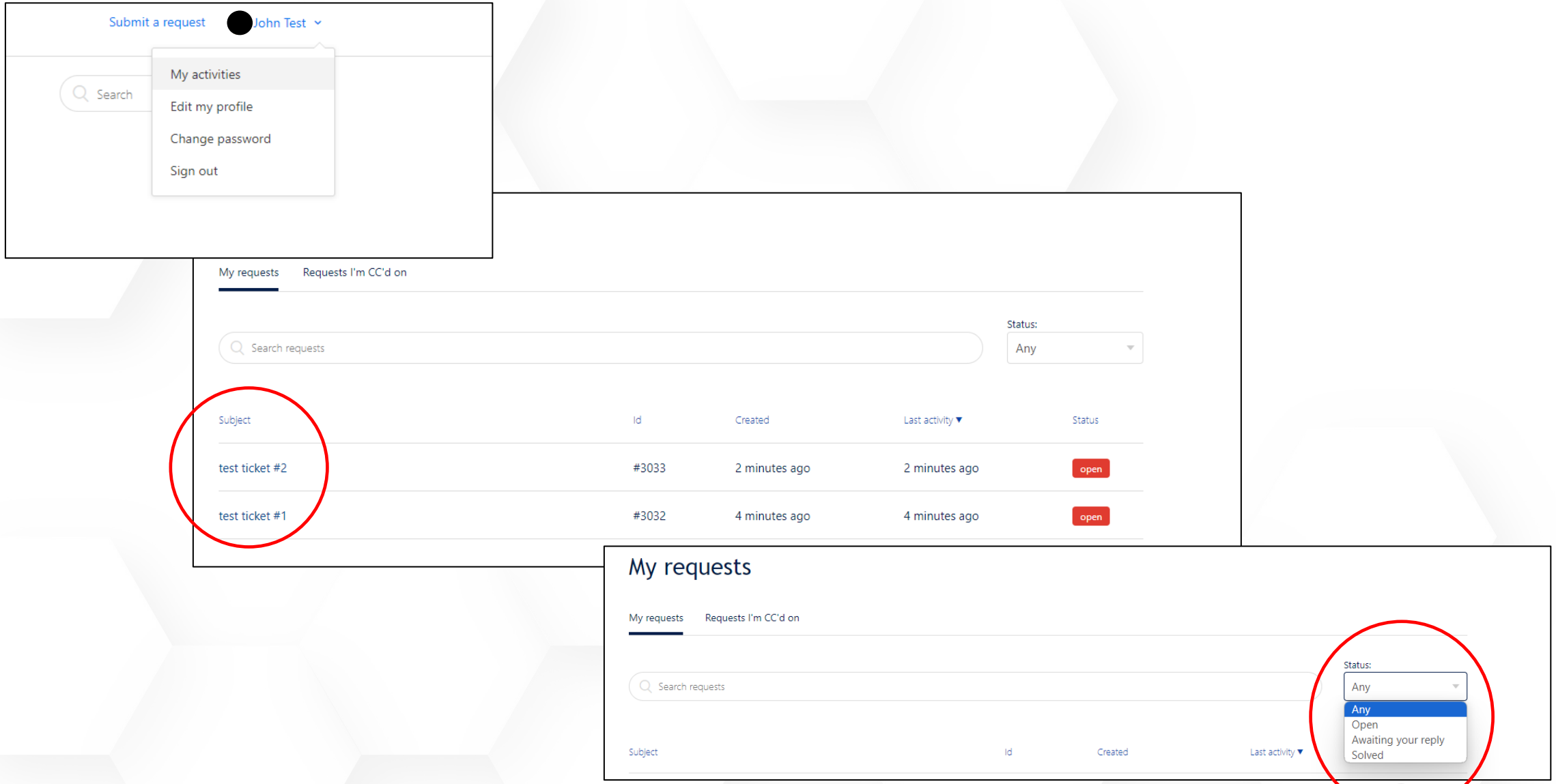
DEVICE AUTHORITY

Successful Ticket Submission

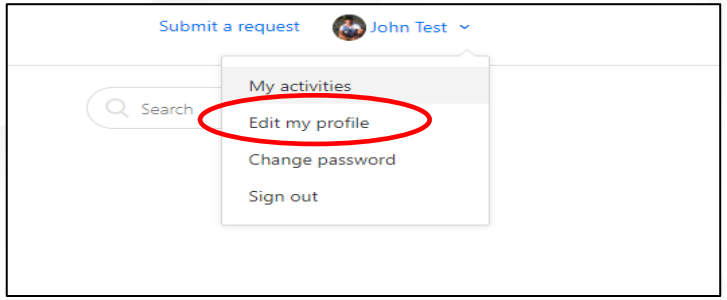


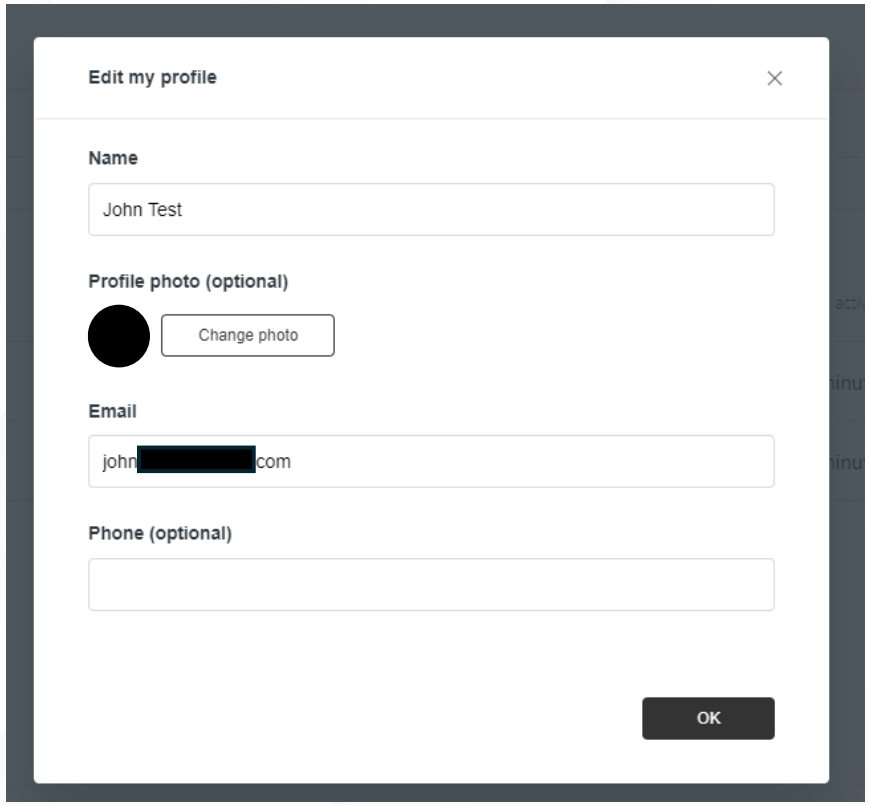
DEVICE AUTHORITY

Dropdown – My Activities

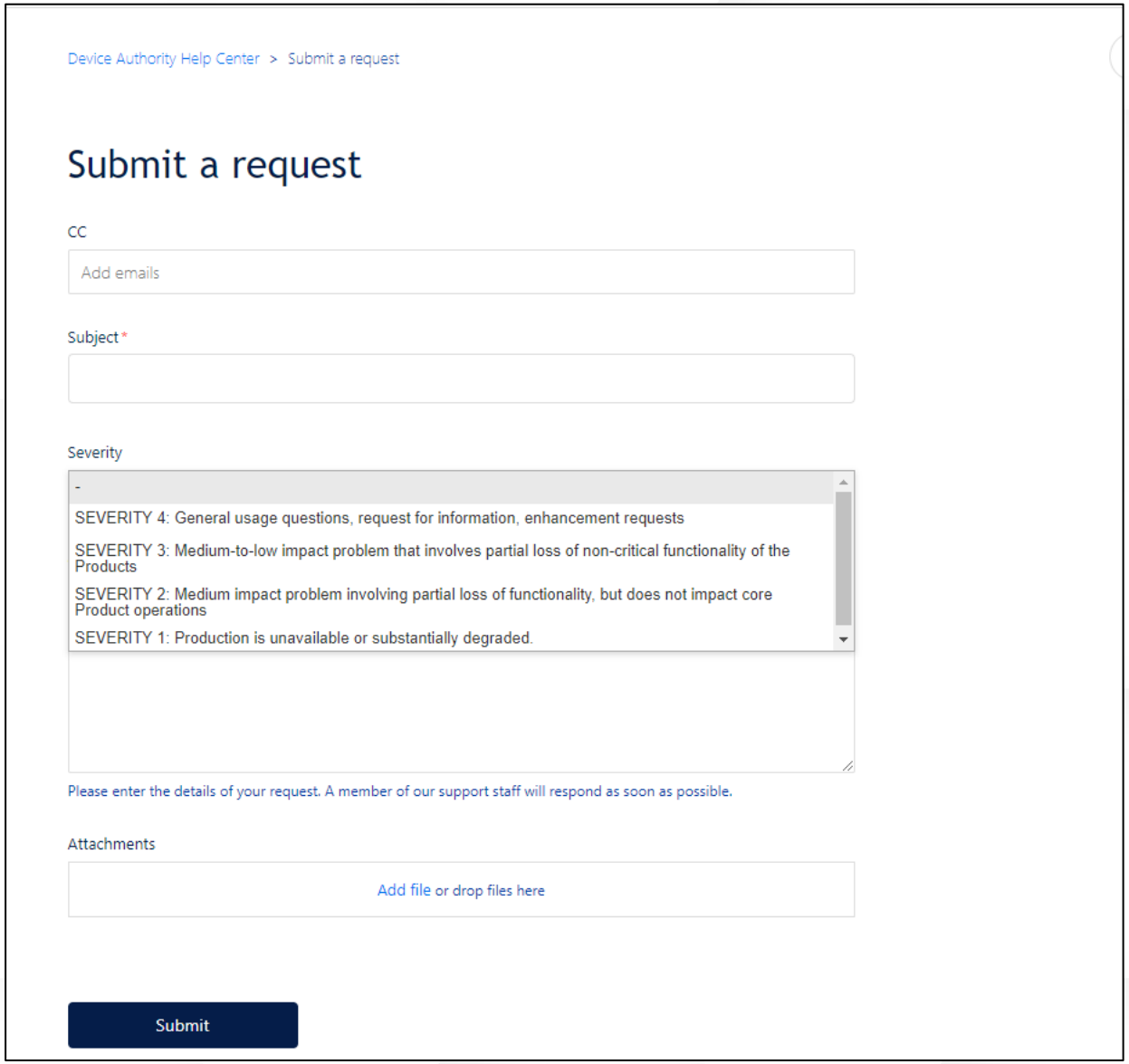


Dropdown – Edit my profile





Severity



- **Severity 1**: Production is unavailable or substantially degraded.
  - See next page
- **Severity 2**: Medium impact problem involving partial loss of functionality but does not impact core product operations.
  - See next page
- **Severity 3**: Medium to low impact problem that involves a partial loss of non-critical functionality of the product.
- **Severity 4**: General usage questions, requests for information, enhancement requests

Severity 1 & Severity 2 –

To ensure appropriate support the following must be included in the ticket

To open a **Severity 1 or Severity 2** ticket successfully, it must contain the following:

1. Ticket must explicitly state Production or Degradation impact by percentage.

2. Must contain primary technical contact and phone number (inc. country code)

3. All information required on New Ticket Requirements

4. All relative and accessible logs uploaded from devices and Screenshots from CP
   - Collect Credential Manager log

5. Did ANY maintenance or anomalies happen recently to impacted devices
   - Was it working? What changed? Any network, hardware, or software changes?

6. Are other devices working that are not impacted?

Required Information to include in all New Tickets

1. What were you trying to achieve?
2. What was the expected outcome?
3. What was the actual outcome?
4. Screenshots of the issue
5. The Date & Time the issue occurred
6. Zip & Upload Logs

Q&A

END OF COURSE

Thank you!

DEVICE AUTHORITY