
TRIDIUM
NS2024
POWER OF PARTNERSHIP

TRIDIUM

NS2024

POWER OF PARTNERSHIP

Defending the Cyber Castle

James Johnson
Tridium – Sales Engineer

TRIDIUM

Agenda

- PKI certificates
- HTML certificate management views
- Provisioning
- Certificate signing service
- Code signing and HSM
- Multi-tiered security dashboard
- Syslog service

Public Key Infrastructure (PKI)

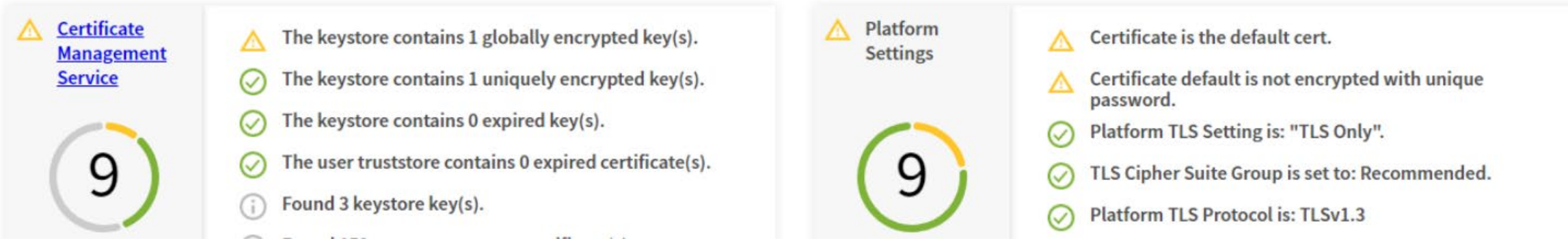
- An infrastructure that supports the distribution of certificates containing public identification keys that are used to both securely identify entities and provide confidentiality of transmissions.
- A Certificate Authority (CA) is an organization which issues and signs digital certificates.
- A digital certificate is an electronic document used to identify an entity, digitally signed by a trusted CA.

Public Key Cryptography

- Uses a **private** and **public key pair**, used together for encrypting and signing.
- Keys are asymmetric, meaning each key is unique but only two specific keys work together.
- The public key is not a secret and is available to everyone.
- Each participant keeps its private key a secret.
- A **sender encrypts data** with a **recipient's public key** and only the **recipient with the private key** can decrypt the data.
- A sender can **sign data** with **their private key**, and **everyone** can validate the sender signed the data using the **sender's public key**.

PKI Certificates 4.13+

- The global certificate password is unique to each Niagara instance and is used to encrypt a certificate's private key in the key store
- Previously only CA and code signing certificates required a unique password to encrypt the private key and to prevent unauthorized usage
- All new certificates may use a unique password to encrypt the private key
- Certificates with the private key encrypted using the global certificate password can be used or exported by any user with sufficient permissions.
- The security dashboard flags certificates with globally encrypted keys



Rivest – Shamir – Adleman (RSA)

- Invented in 1977
- Uses prime factorization method to achieve the one-way encryption of data
- Widely used today with most web sites and other applications
- Typical key sizes are 2048, 3072 or 4096 bits
- Relies on the fact that factorization of large prime numbers requires significant computing power to crack
- Compute power has increased significantly over time which requires larger keys to be used to provide sufficient encryption strength

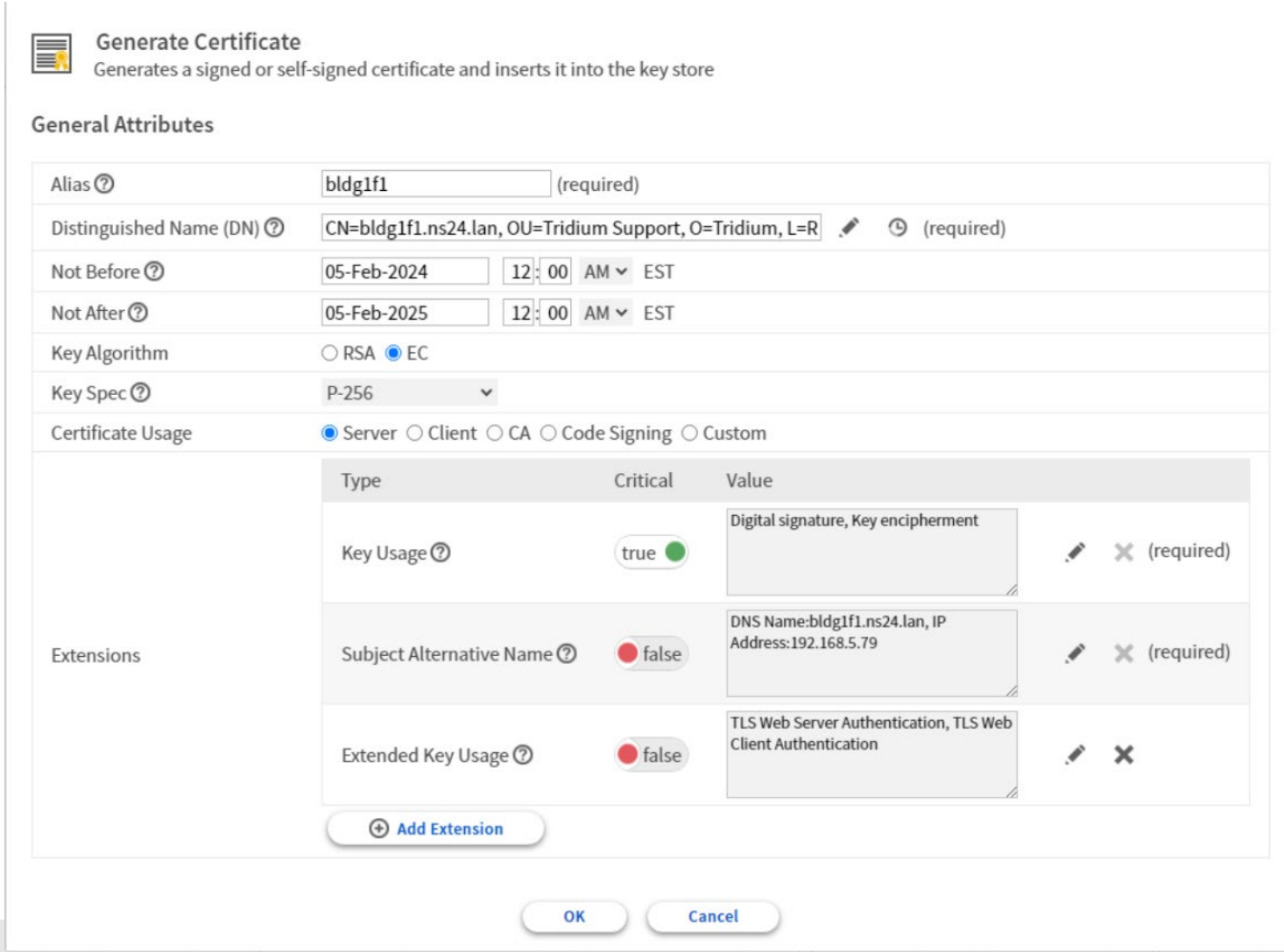
Elliptic Curve Cryptography (ECC)

- Invented in 1985 and mainstream usage began ~2005
- An approach to public key cryptography based on elliptic curves over finite fields.
- Provides the same cryptographic strength as an RSA-based key with much smaller key sizes.
- Smaller key sizes require fewer processing resources.

Security (bits)	RSA and DH Key Size (bits)	ECC Key Size (bits)
80	1024	160-223
112	2048	224-255
128	3072	256-383
192	7680	384-511
256	15360	521+

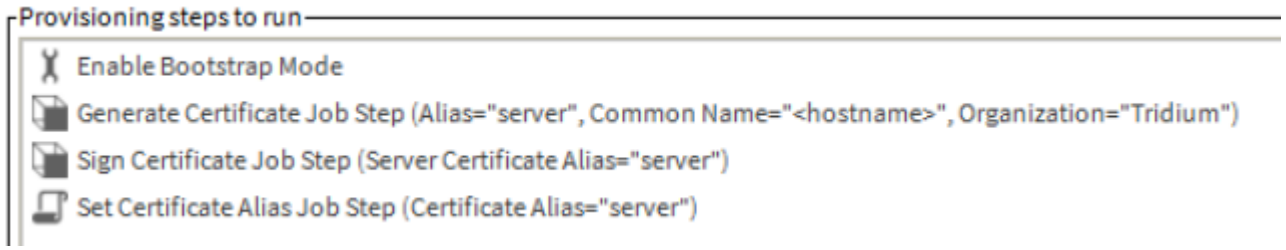
HTML Certificate Management 4.13+

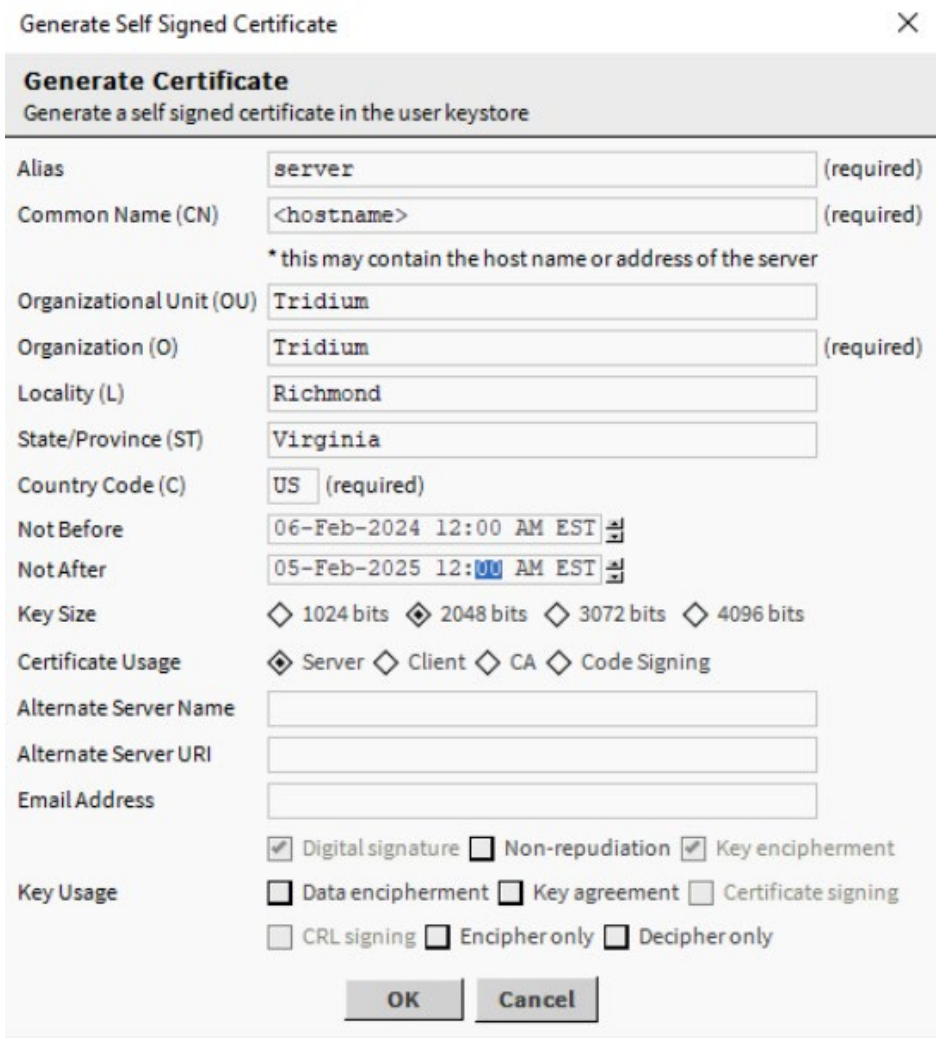
- Enables managing certificates without Workbench client
- More flexible interface for configuring certificate extensions
- Supports RSA and EC key algorithms



Provisioning 4.6+

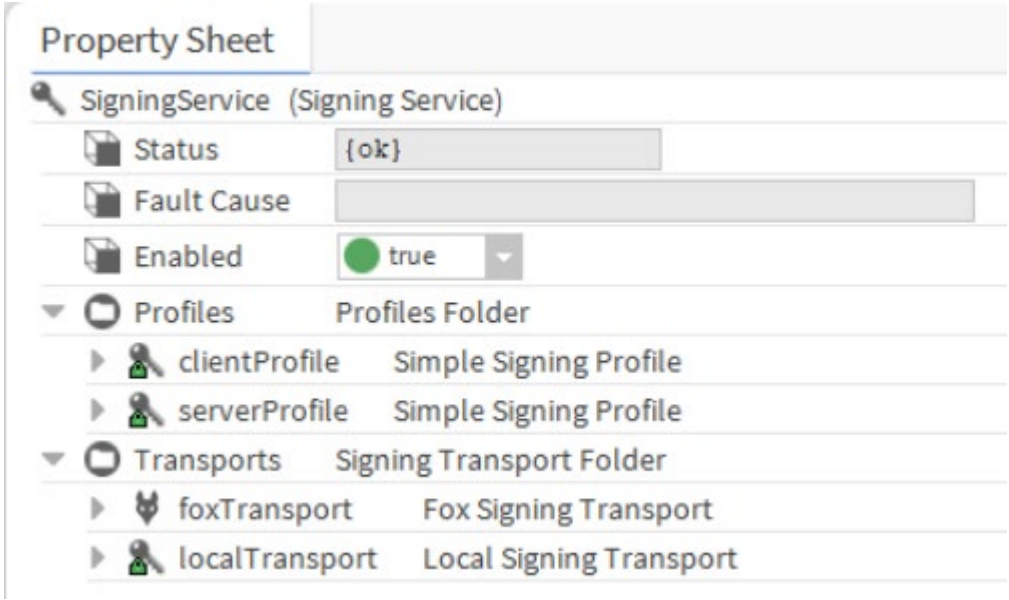
- Jobs can automate generating server certificates and signing using a CA certificate in the provisioning station's key store
- Bootstrap mode allows connectivity for initial certificate generation
- Unique Common Name (CN) and Subject Alternative Name (SAN) derived from <hostname> variable using the session connection details.





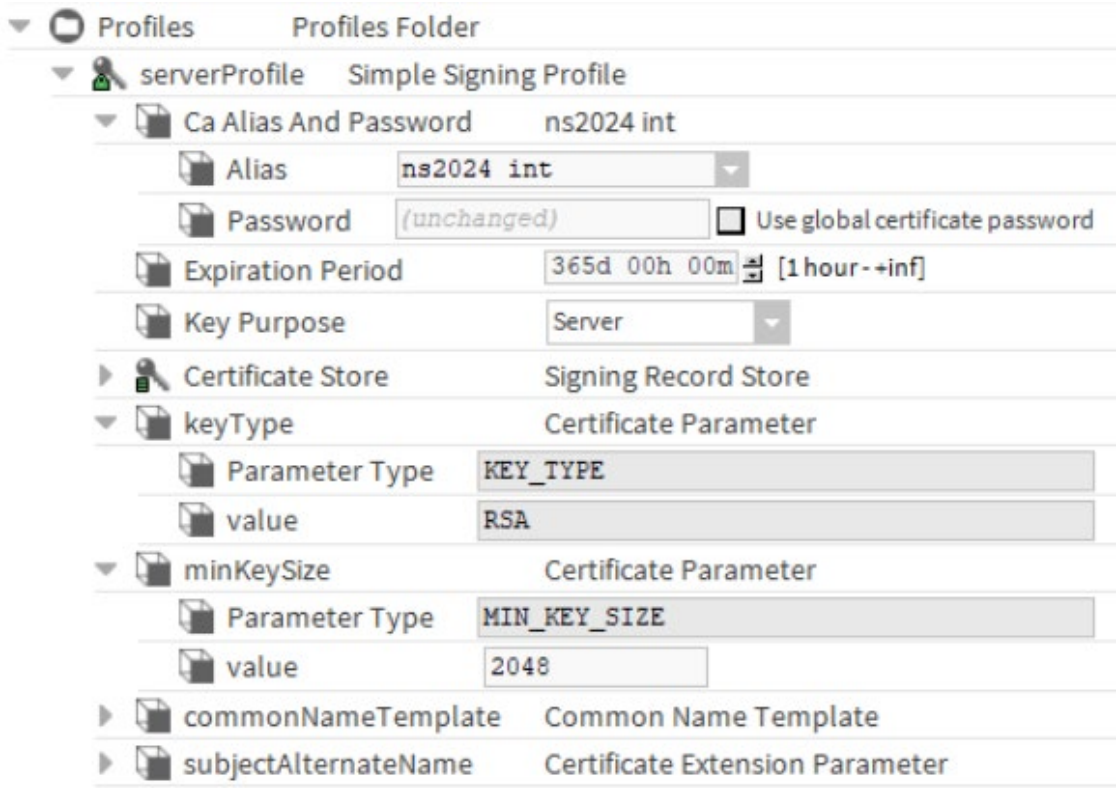
Certificate Signing Service 4.13+

- Intended primarily for use on a Niagara supervisor
- Allows components within the local or remote stations to make secure Certificate Signing Requests (CSR) to obtain signed X509 certificates
- Requires a CA certificate in the user key store of the station with the certificate signing service



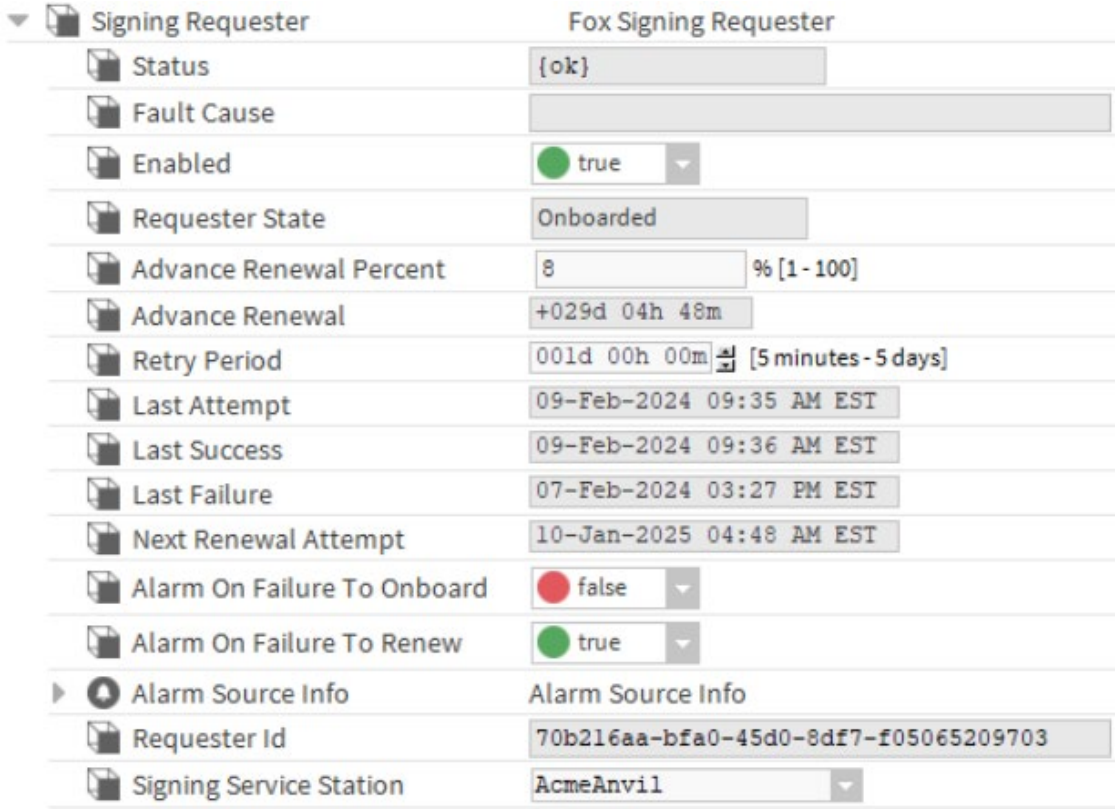
Simple Signing Profile

- Defines the desired CA certificate to utilize for signing requests
- Configures certificate properties
  - Expiration Period – 365 days default
  - Key Purpose – Client, Server, Cert Authority, Code Signing
  - Key Type – RSA, EC
  - Common Name
  - Subject Alternative Name



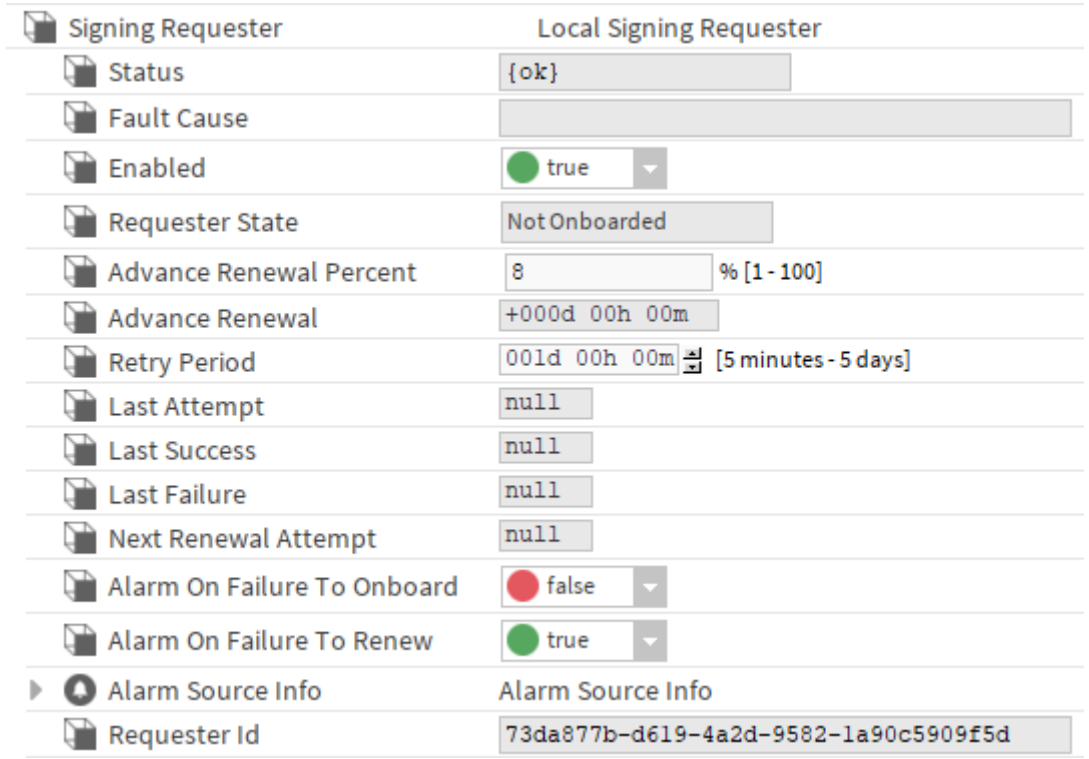
Fox Signing Requester

- Configures which Niagara network station is used to fulfill CSRs
- Initially invoke action to Onboard which must be approved in the signing service station
- Once onboarded, certificate is automatically renewed before expiry
- Action to manually renew if needed
- Configures certificate properties
  - Common Name
  - Subject Alternative Name



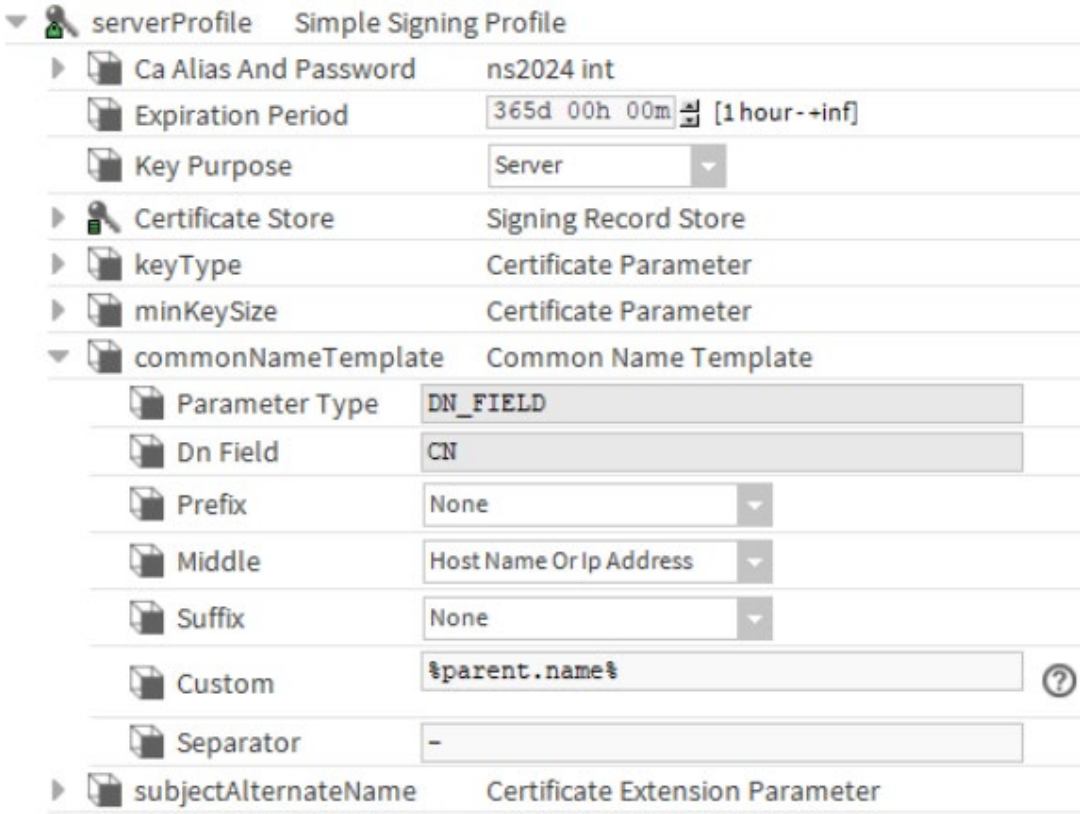
Local Signing Requester

- Used to fulfill CSRs for services in the local station with the Signing Service
- Initially invoke action to Onboard which is automatically approved
- Once onboarded, certificate is automatically renewed before expiry
- Action to manually renew if needed
- Configures certificate properties
  - Common Name
  - Subject Alternative Name



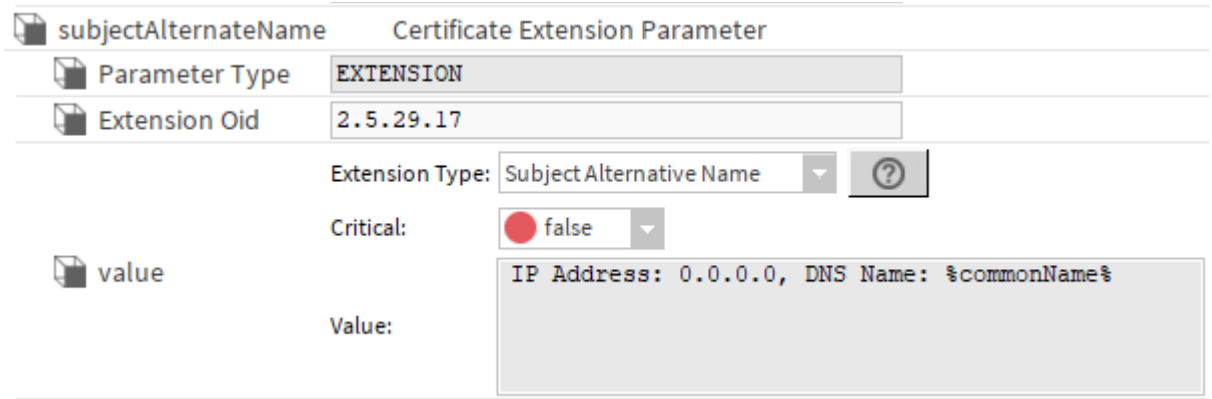
Common Name Template

- Configures the resolution of the certificat subject Common Name (CN)
- Options include station name, device nan network name, host name, UID, IP addres Niagara station address, or custom
- Custom property resolves BFormat agains the component in the station
- Prefix, middle and suffix properties allow flexible resolution
- Added to either the signing profile or the requester
- Signing profile configuration overrides the requester configuration



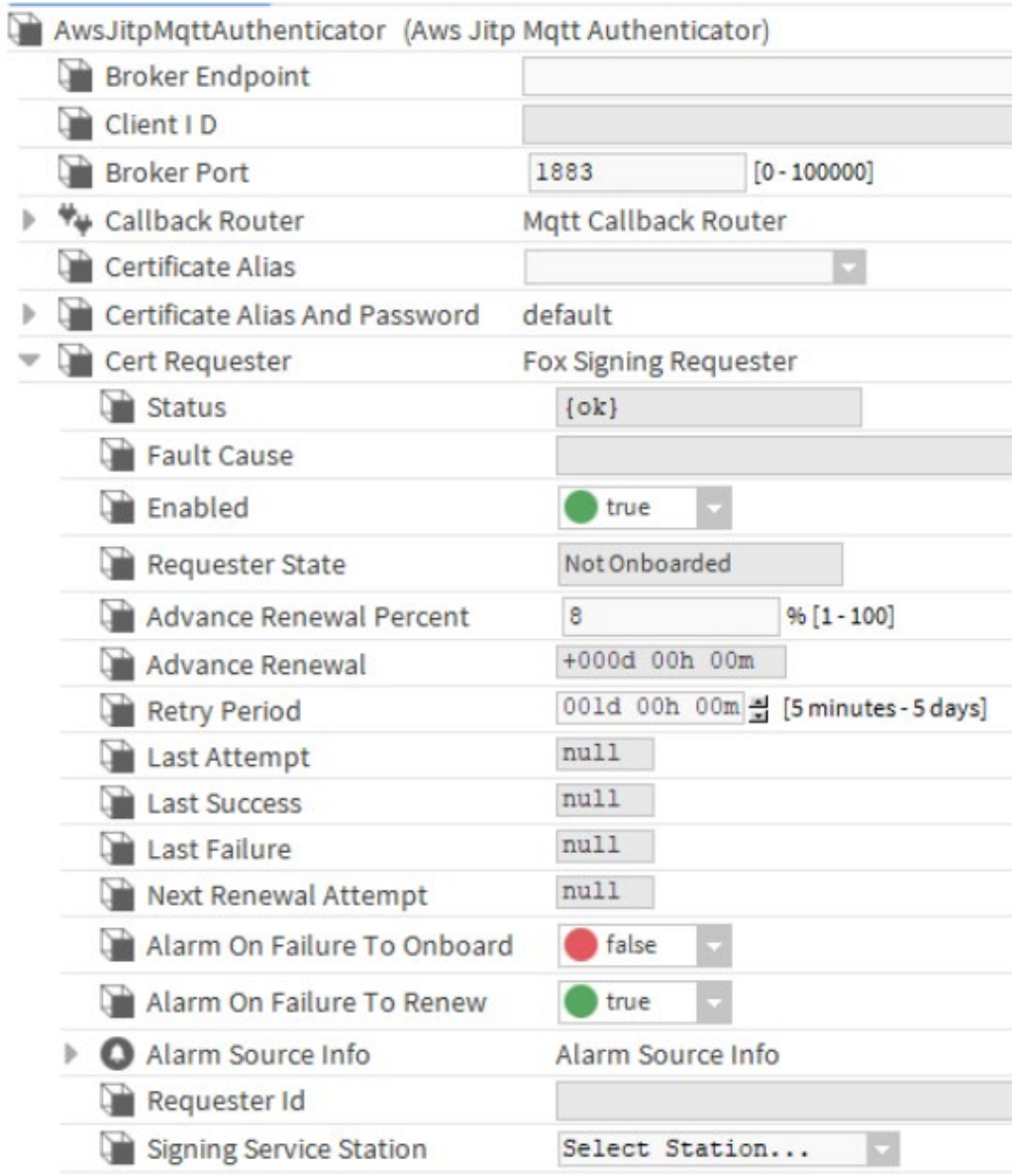
Subject Alternative Name

- Configures the certificate's Subject Alternative Name extension
- Options include DNS name, directory name, email, IP address, registered ID and URI
- BFormat options for values include
  - %commonName%
  - %hostnameOrIpAddress%
  - %ipAddress%
  - 0.0.0.0
  - %niagaraStationAddressOrHostname%
  - %niagaraStationAddressOrIpAddress%
  - %niagaraStationAddressOrHostnameOrIpAddress%
- Added to either the signing profile or the requester
- Signing profile and requester configurations are merged



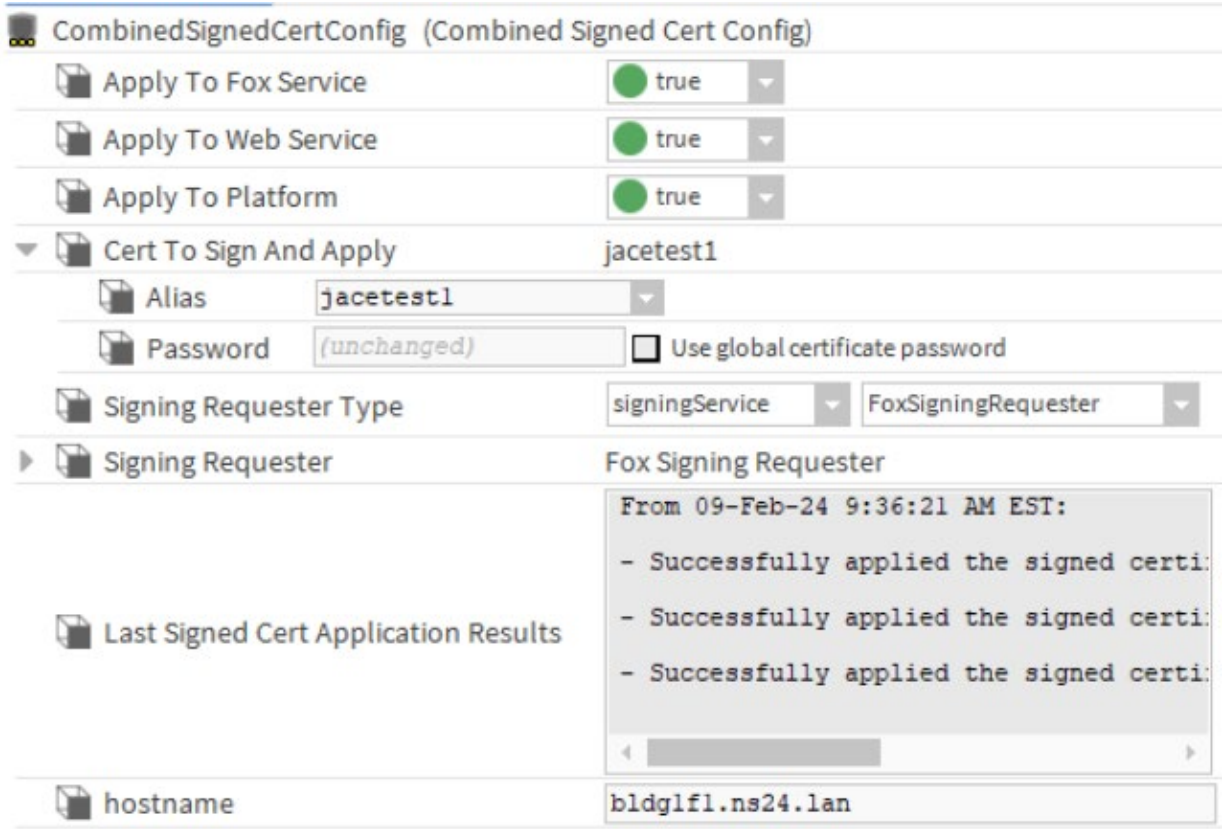
Signing Requester 4.13+

- Used to connect a Niagara station using MQTT to Amazon Web Services (AWS) utilizing Just In Time Provisioning (JITP)
- JITP allows a fleet of devices to automatically connect to AWS with auto-generated certificates as a means of authentication
- Signing requester handles requesting and renewing the certificate automatically



Combined Signed Cert Config 4.14+

- Added to the Security Service in the station
- Signing requester initiates CSR and updates the configured Fox, Web and Platform services
- Fox Signing Requester used in remote stations

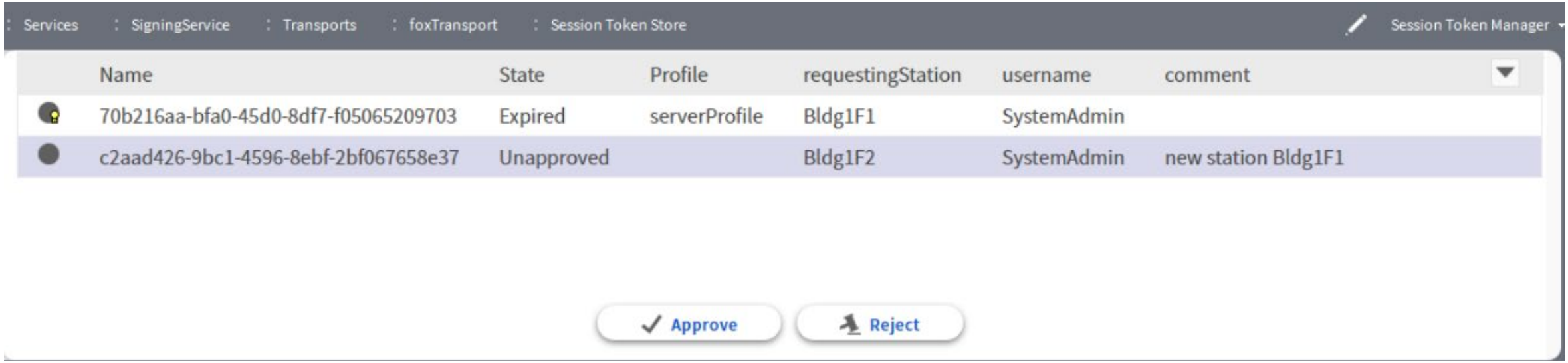


Individual Signed Cert Config 4.14+

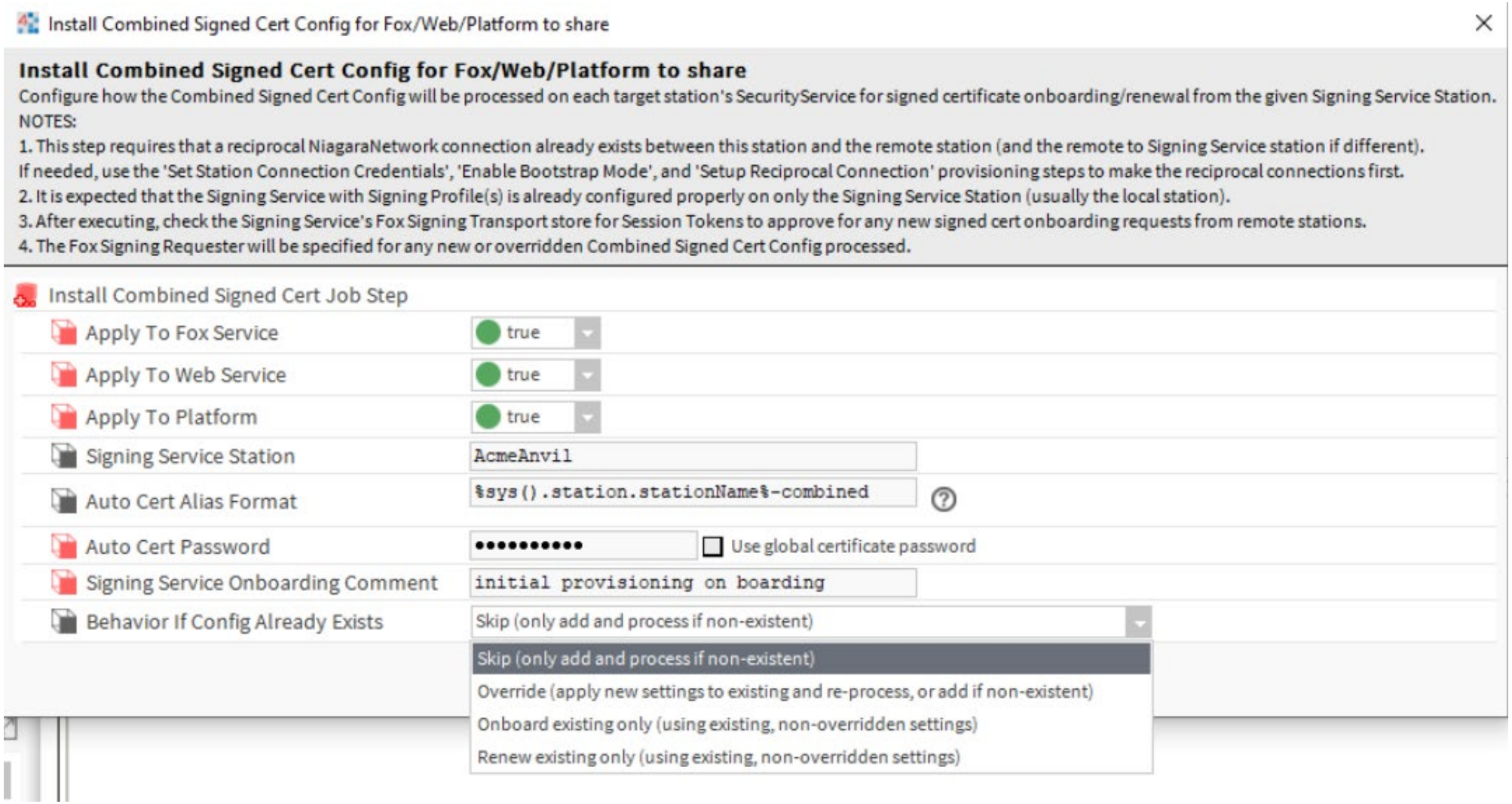
- Added to specific components such as the Fox Service, Web Service or Additional HTTPS Cert
- Enables the parent component to initiate and individual CSR

Session Token Manager

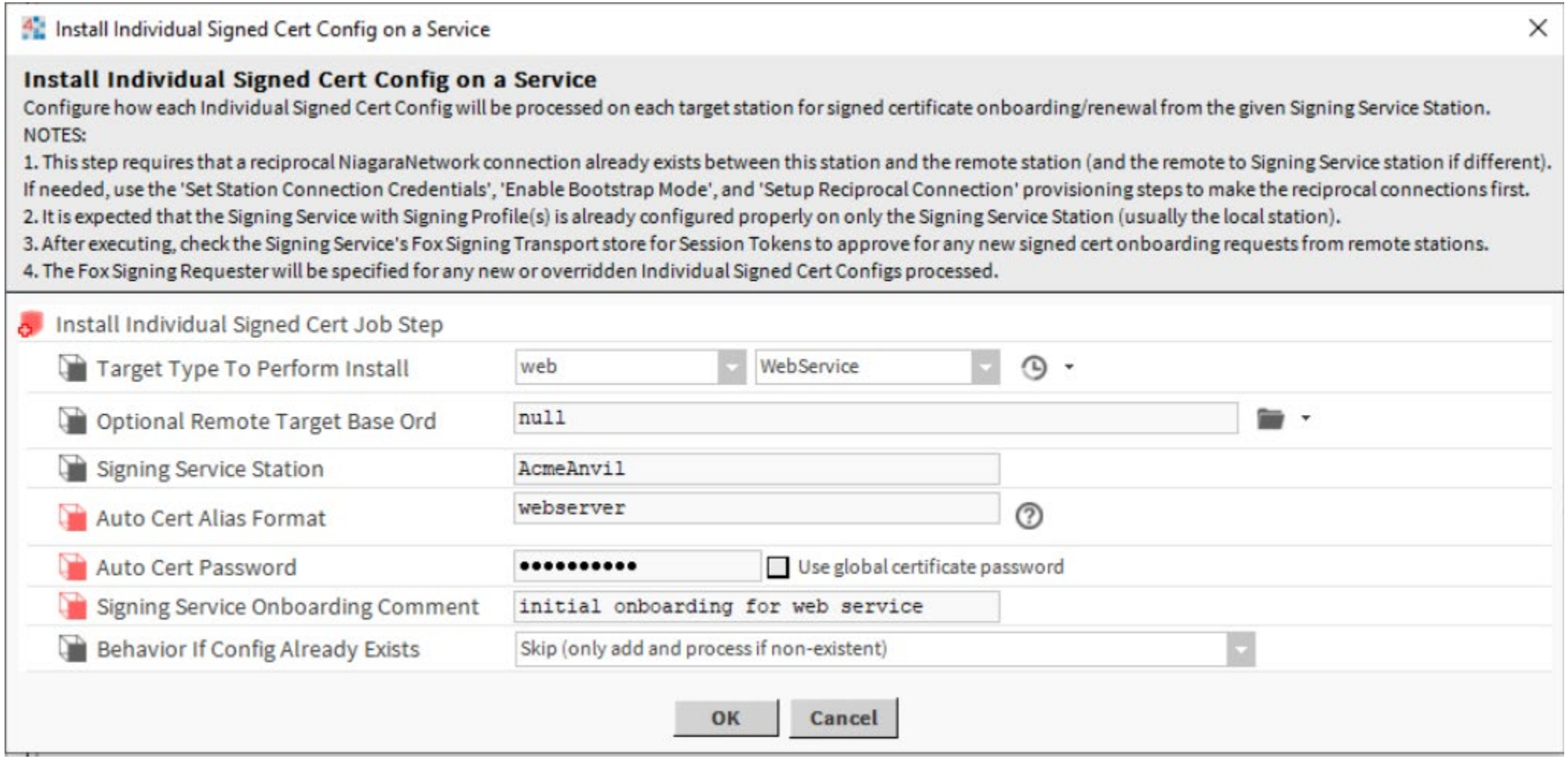
- Admin user approves or rejects session tokens for stations which have sent an onboarding request
- Once approved the remote station can send CSR to the signing service station



Provisioning Jobs



Provisioning Jobs



Code Signing

- The process of digitally signing executables and scripts to confirm the software author and guarantee the code has not been altered or corrupted since it was signed
- Trusted timestamping is the process of securely keeping track of the creation and modification times of a document
- Timestamping Authority (TSA) URL – online server which timestamps the code signature so a client can verify when the code was digitally signed

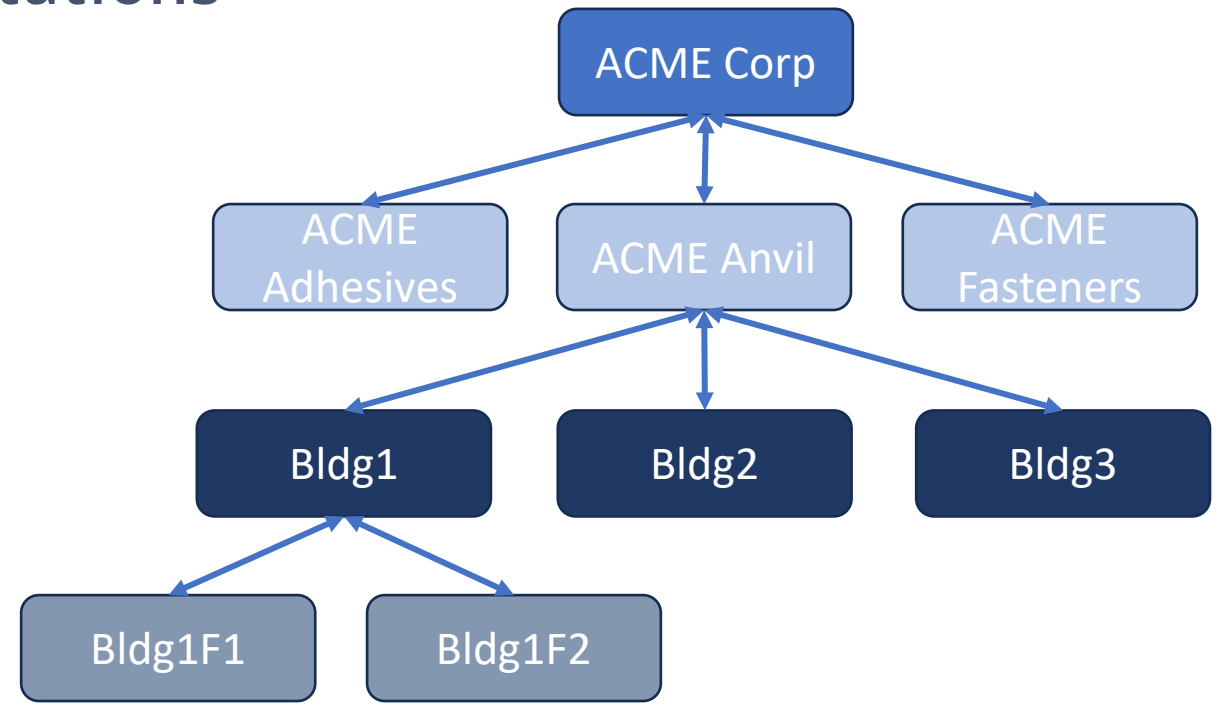
Hardware Security Module (HSM)

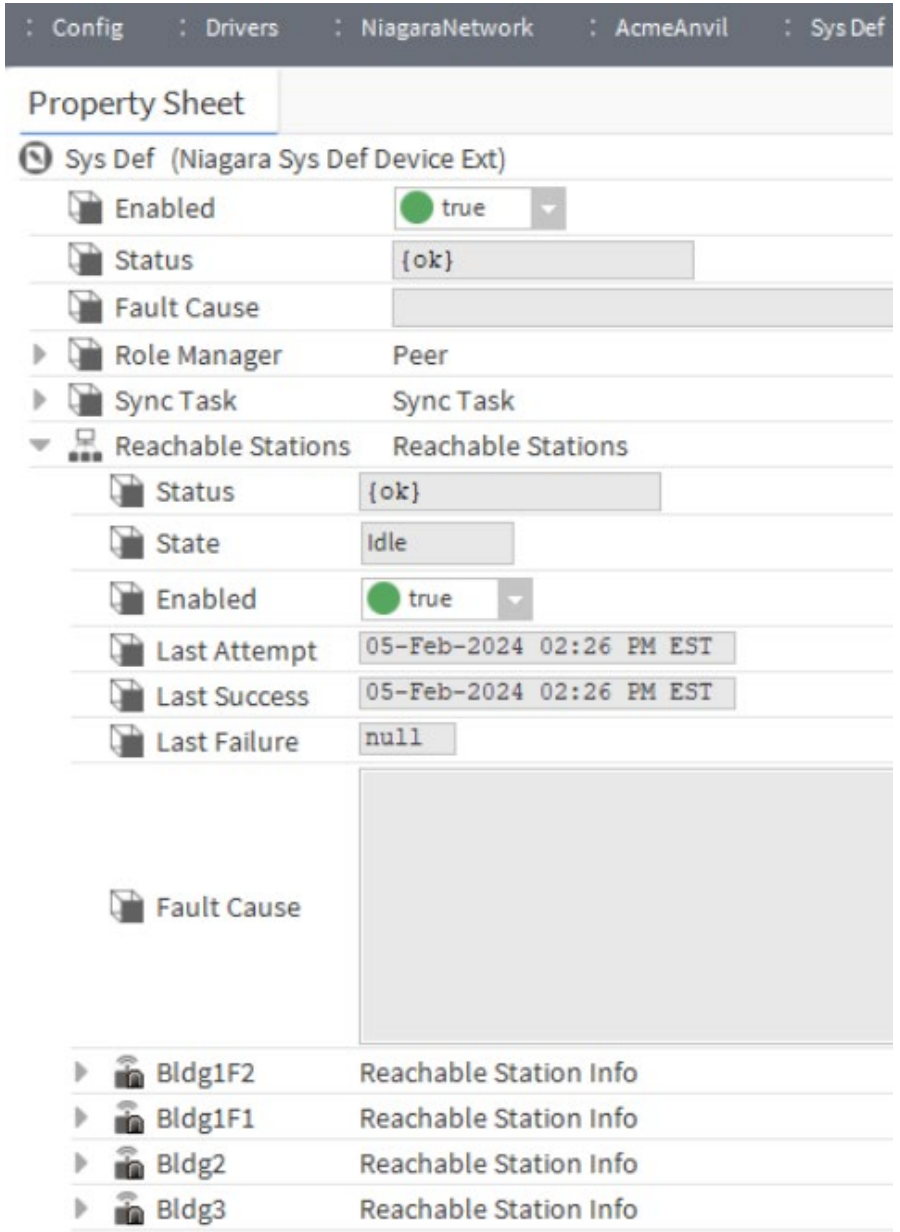
- HSMs are hardened, tamper-resistant hardware devices used to secure cryptographic processes
- HSMs could be a card inside of a PC, a rack mounted device, a USB device or part of a cloud service
- As of mid-2023, all commercial certificate authorities require the use of a HSM to protect the private key of any code signing certificate
- Starting in 4.14 version, the gradle scripts are updated to support code signing compiled modules using a physical HSM device that supports the PKCS11 standard

  https://www.niagara-community.com/s/article/Code-Signing-using-Hardware-Security-Modules

Reachable Stations 4.13+

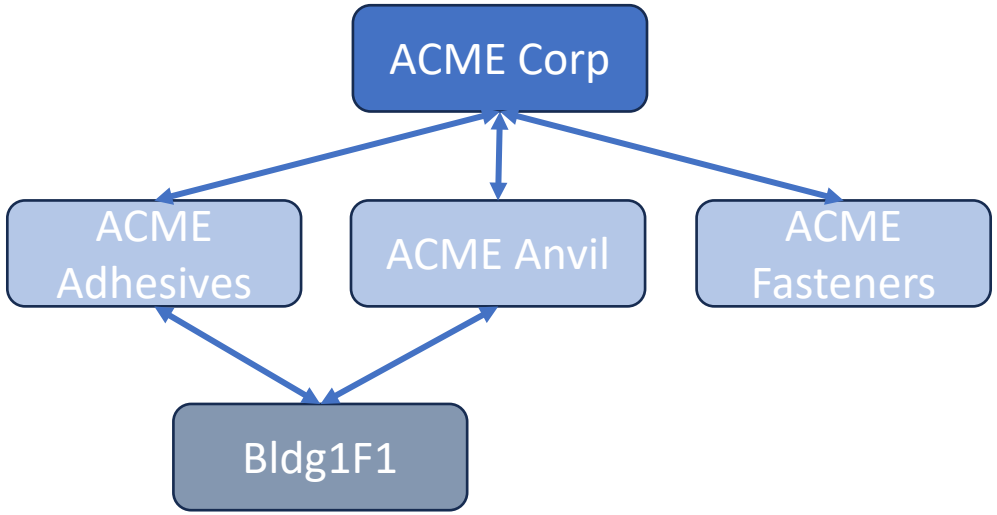
- Defines all downstream Niagara stations that are routable through this Niagara station
- System Database indexing (4.13+) and Security Dashboard (4.14+) may leverage reachable stations





Reachable Stations Info 4.13+

- Info and Spy view shows routes to each station
- If multiple routes exist to same station, route preference is utilized



Remote Station | niagaraNetwork | reachableStations | noForceUpdateIncludeUnoperationalInc

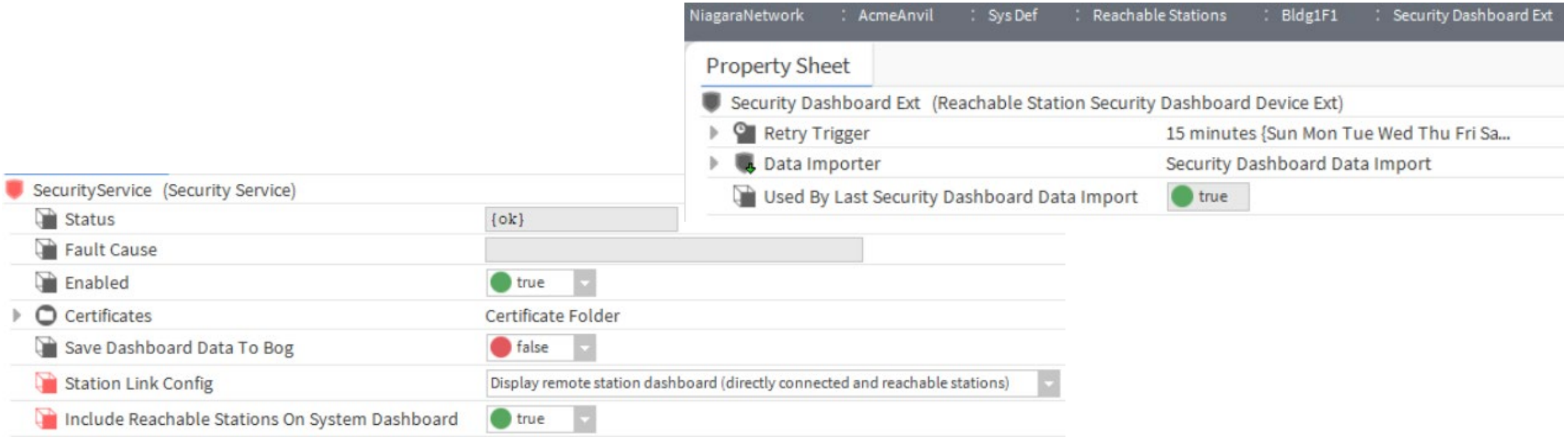
Reachable Station	Version	Route To Reachable Station	Route Preference (0 = primary)
Bldg1F1	4.14.0.118	AcmeAnvil -> Bldg1F1	0
Bldg1F1	4.14.0.118	AcmeAdhesives -> Bldg1F1	1
Bldg1F2	4.14.0.118	AcmeAnvil -> Bldg1F2	0
Bldg2	4.14.0.118	AcmeAnvil -> Bldg2	0
Bldg3	4.14.0.118	AcmeAnvil -> Bldg3	0

Bldg1F1 Reachable Station Info

Station Name	Bldg1F1
Station Version	4.14.0.118
Route To Station	AcmeAnvil;Bldg1F1
Min Version Along Route	4.14.0.118
Virtual Space Ord	station:|slot:/Drivers/NiagaraNetwork/AcmeAnvil/virtual|virt
Time To Reach	+00000h 00m 00.000s
Route Enabled	true

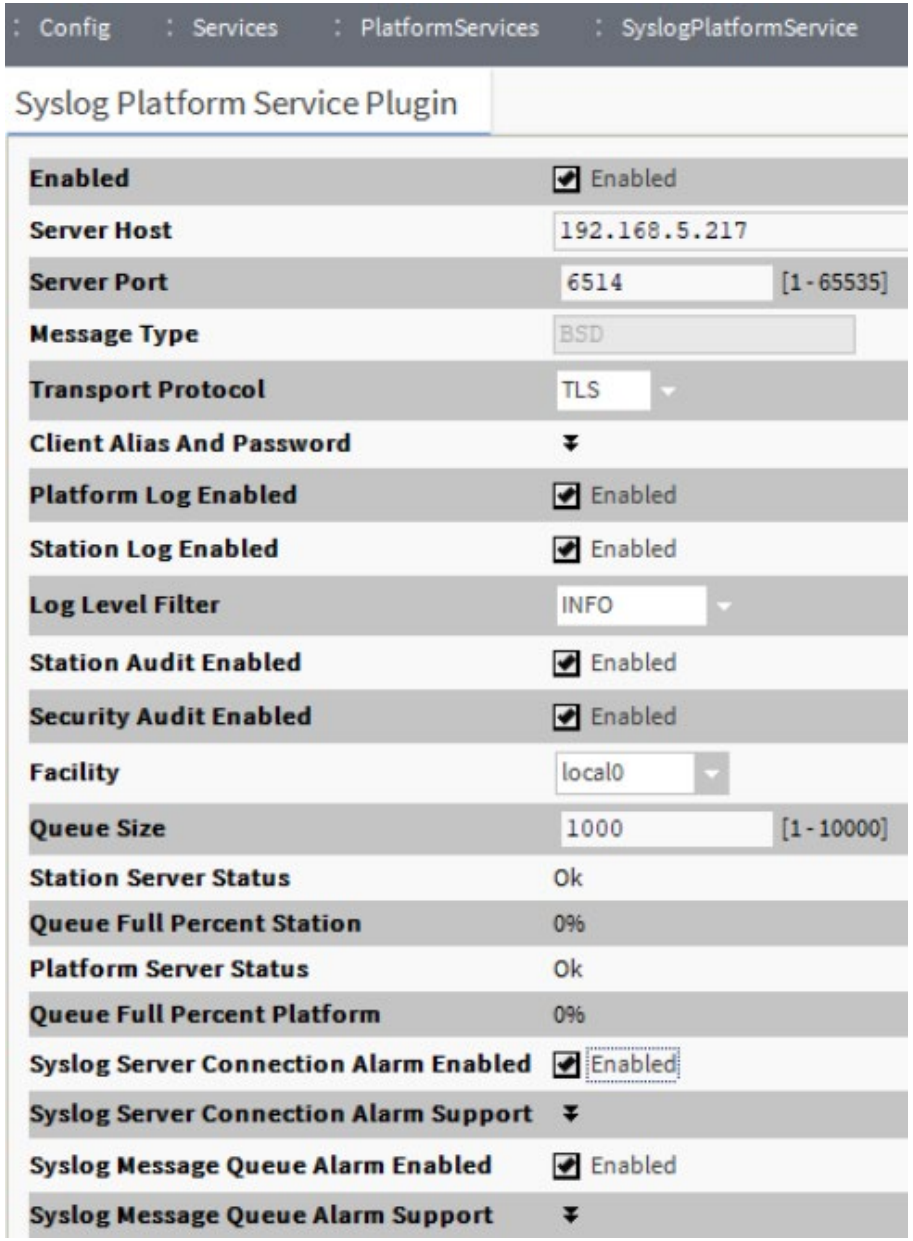
Multi-tiered Security Dashboard 4.14+

- When enabled the Security Dashboard in the top-level station can display information from all reachable stations
- Security Dashboard Extension displays information from that specific reachable station



Syslog Service

- Syslog is a standard protocol for message logging
- Using the BSD message format, messages can be sent over UDP, TCP or TLS
- Allows sending desired event messages from the Niagara station to an external application for analysis



Questions