4-Series™ Control Systems

Security Reference Guide

Crestron Electronics, Inc.

Original Instructions
The U.S. English version of this document is the original instructions.
All other languages are a translation of the original instructions.

Revision History

Rev	Date	Notes	Author(s)
A	July 8, 2021	Initial version	IH
B	January 7, 2022	Incorporated major updates from Military Unique version of document	IH, JD
C	May 20, 2022	Added information about default user roles and permissions	IH, JD
D	July 18, 2023	Added Domain-Joined Authentication and Authorization section Added Backup and Restore Functionality section Updated various sections in document to reflect RADIUS support Added "Configure MFA Support with RSA Authentication Manager" section to "Additional Instructions" topic	IH, JD, AP

Please send comments and change recommendations to:

SecurityDocs@crestron.com

Contents

Overview

This document describes the steps needed to harden a Crestron® installation with 4-Series™ control systems and assumes a basic understanding of security functions and protocols. This guide provides information about the system configuration used for 4-Series control systems firmware release 2.7000.00014 or later.

> **NOTE:** The term "device" is used in this document to refer to all applicable 4-Series control system models unless specified otherwise.

The information in this guide pertains to the following device models:

Model	Description
AV4	4-Series™ Control System
CP4	4-Series™ Control System
CP4N	4-Series™ Control System
DIN-AP4	4-Series™ DIN Rail Control System
MC4	4-Series™ Control System
MC4-I	4-Series™ Control System, International
PRO4	4-Series™ Control System
RMC4	4-Series™ Control System

Ports and Protocols

The following ports and protocols may be used by the device depending on the system design and configuration.

Crestron Control Devices

Function	Destination Port	From (Sender)	To (Listener)	Notes
Crestron-CIP	41794/TCP	Remote Device	Device	Crestron Internet Protocol
Crestron-SCIP	41796/TCP	Remote Device	Device	Secure Crestron Internet Protocol
HTTPS	49200/TCP	Remote Device	Device	Web API for Crestron HTML5 User Interfaces

Common Ports

Function	Destination Port	From (Sender)	To (Listener)	Notes
NTP	123/UDP	Device	NTP Server	Network Time Protocol (NTP)
SSH/SFTP	22/TCP	Admin Workstation	Device	Used for configuration, console, and file transfer
LDAP	3268/TCP	Device	LDAP Server	LDAP queries targeting global catalogs
HTTPS	443/TCP	Admin or End User Workstation	Device	Secure web configuration
HTTPS	443/TCP	Device	XiO Cloud® Service	For XiO Cloud services only and not required for device functionality. A persistent connection is made via AMQP over WebSockets. HTTPS services such as routing lookups and file transfers may be used.
DHCP	67/UDP	Device	DHCP Server	DHCP addressing
DHCP	68/UDP	DHCP Server	Device	DHCP addressing
HTTP	80/TCP	End User Workstation	Device	Redirect to Secure Web Configuration on port 443
Remote Syslog	Configurable	Device	Remote Syslog Server	Uses TLS
SNMP	161/UDP	SNMP Manager	Device	
SNMP Traps	162/UDP	Device	SNMP Manager	

Function	Destination Port	From (Sender)	To (Listener)	Notes
RADIUS	1812/UDP	Device	RADIUS server/client	RADIUS communications for Authentication and Authorization, for Multi-Factor Authentication (MFA) support.

Prerequisites

In order to perform a secure configuration, the following prerequisites must be met.

Operating Environment

Crestron assumes the following about the operating environment of its systems:

- If your organization's policy requires MFA, Remote Authentication Dial-In User Service (RADIUS) must be used for user authentication.
- Physical security is commensurate with the value of the system and the data it contains and is assumed to be provided by the environment.
- Administrators are trusted to follow and provide all administrator guidance.

Firmware Version

4-Series control systems must be running firmware version 2.7000.00014 or later.

Device Access

The administrator can access and configure the device by using a web browser or an SSH client. This document describes device configuration using an SSH client, which provides access to console commands. Some configuration capabilities can only be performed by issuing console commands. Additionally, some aspects of configuration can be performed via Crestron Toolbox™ software, or the XiO Cloud® service.

> **NOTE:** The SSH client that is used must be capable of connecting to the device using SSHv2 and must be compatible with FIPS 140-2 validated algorithms.

As an alternative to using an SSH client, the same console commands can be executed through the USB port.

Default Configuration Settings

In order to configure the device, it must first be placed in its factory default state. A device can be returned to this state by entering the following command on the console:

```
RESTORE
```

If you do not have access to the console (for example, the password has been lost), a factory reset may be performed as follows:

1. Press and release the **HW-R** button on the front panel of the control system.
2. Quickly press the **SW-R** button on the front panel of the control system 5 times, with less than a one-second gap between each press.
3. Wait 5 to 10 minutes for the self-recovery process to complete.
4. Proceed with the network configuration.

Required Configuration

The following sections describe the configuration changes required for the device for a secure deployment.

Configure the Network

The following sections provide information about the tasks necessary to configure the network.

DHCP or Static IP Address Configuration

To configure the device to communicate on the local LAN, the following changes must be made. If DHCP is available on the local network, then no additional configuration changes are necessary. If DHCP is not available or if the administrator wishes to manually set the network configuration, then the IP address, IP mask, default gateway, and DNS server settings must be set.

```
dhcp 0 off
```

Turns off DHCP so that the manually configured network information is used.

```
ipaddress 0 192.168.1.2
```

Sets the IP address of the device to the specified address.

```
ipmask 0 255.255.255.0
```

Sets the IP mask of the device to the specified mask.

```
defrouter 0 192.168.1.1
```

Sets the default network gateway to the specified IP address.

```
adddns 192.168.1.10
```

Sets the DNS server to use for DNS name lookups.

802.1X Authentication

802.1X is an IEEE network standard designed to enhance the security of both wireless and wired Ethernet networks. This device supports 802.1X on its primary wired Ethernet interface only. If the network requires 802.1X, the device must be configured for 802.1X before being put on the network. This configuration can be done through the USB port console or by attaching it to a temporary network which does not require 802.1X.

Before configuring 802.1X, perform the following tasks as necessary:

- Unless server authentication is going to be disabled, the trusted x.509 certificate or certificates that will be used to verify the 802.1X server's certificate must be loaded into the device. Use the certificate management commands to load the trusted certificates into the device. These may be Root or Intermediate certificates. Refer to the Required Configuration on page 6 section for instructions.
- If EAP-TLS authentication is going to be used, a client certificate will be needed and must be loaded into the device. Refer to the Required Configuration on page 6 section for instructions to load a client certificate into the "machine" store.

Once 802.1X configuration is complete, restart the device to activate those settings. The device will try to connect to the 802.1X network when it starts up.

802.1X Configuration

In order to configure and use 802.1X, various aspects of 802.1X will need to be configured, including enabling it, setting up server authentication, and selecting a client authentication method. The following commands are used for this configuration:

Enable 802.1X

To enable 802.1X, issue the following command:

```
8021xauthenticate [on/off]
```

- `on` - 802.1X is enabled
- `off` - 802.1X is disabled
- No parameter - Displays the current setting

Example: `8021xauthenticate on`

Set Trusted Server Certificates

Unless server validation will be disabled, the trusted certificates that 802.1X will use to verify the server's certificate must be indicated. The full list of trusted Root and Intermediate certificates loaded into the device is not used for 802.1X—only the specific certificates selected by the `8021xtrustedcas` command are used. As indicated earlier, the trusted certificates must first be loaded into the device using the standard Certificate Management commands.

The following commands can be used to list, add, and remove certificates from the list of certificates that will be used by 802.1X.

List Certificates

To list available certificates, issue the following command:

```
8021xtrustedcas [list|listn|listu]
```

- `list` - Shows all Root and Intermediate certificates for the device
- `listn` - Shows all Root and Intermediate certificates for the device and also includes identification number of each certificate
- `listu` - Shows Root and Intermediate certificates that are used by 802.1X

Example: `8021xtrustedcas listn`

This certificate list will show the name and UID of each certificate, along with an indication of whether or not it is being used by 802.1X.

Add Certificate to 802.1X Trust List

To add a certificate to the list of trusted certificates to be used by 802.1X, issue the following command:

```
8021xtrustedcas use [certificate number] [certificate name] [certificate UID]
```

- `certificate number` - Number that identifies the specific certificate to use
- `certificate name` - Name that identifies the specific certificate to use
- `certificate UID` - UID that identifies the specific certificate to use

Only one identifier (number, name, or UID) is needed. These identifiers can be determined by listing the certificates using the `8021xtrustedcas list` or `8021xtrustedcas listn` described above. Only the `listn` command will show the certificate number.

Remove Certificate from 802.1X Trust List

To remove a certificate from the list of trusted certificates to be used by 802.1X, issue the following command:

```
8021xtrustedcas dontuse [certificate number] [certificate name] [certificate UID]
```

- `certificate number` - Number that identifies the specific certificate to remove
- `certificate name` - Name that identifies the specific certificate to remove
- `certificate UID` - UID that identifies the specific certificate to remove

Only one identifier (number, name, or UID) is needed. These identifiers can be determined by listing the certificates using the `8021xtrustedcas list` or `8021xtrustedcas listn` described above. Only the `listn` command will show the certificate number.

Removing a certificate from 802.1X does not remove the certificate from the device. The certificate will still be present in the Root or Intermediate store.

Enable 802.1X Server Validation

Under most circumstances, validation of the 802.1X server should be enabled. By default, server validation is disabled on this device.

To enable 802.1X server validation, issue the following command:

```
8021xvalidateserver [off | on]
```

- `off` - 802.1X supplicant will not validate authentication server's certificate.
- `on` - 802.1X supplicant will validate authentication server's certificate.
- No parameter - Displays the current setting

Example: `8021xvalidateserver on`

Select 802.1X Client Authentication Method

802.1X requires that the device authenticate with the server before it will be allowed on the network. The device supports two client authentications methods: PEAPv0/EAP-MSCHAPv2 and EAP-TLS. PEAPv0/EAP-MSCHAPv2 requires a user name and password, and EAP-TLS requires a client certificate.

To select the 802.1X client authentication method, issue the following command:

```
8021xmethod [password/certificate]
```

- `password` - Uses PEAPv0/EAP-MSCHAPv2 authentication
- `certificate` - Uses EAP-TLS authentication

Example: `8021xmethod password`

If EAP-TLS is selected, a client certificate must be loaded into the device as explained earlier in the 802.1X instructions.

If PEAPv0/EAP-MSCHAPv2 is selected, the user name and password to use for authentication must be entered with the following commands:

```
8021xusername [username]
```

```
8021xpassword [password]
```

Additional 802.1X Options

Additional 802.1X options may need to be configured if required by the network to which the device is connected:

- If the 802.1X server requires that a specific domain name be included with the 802.1X authentication request, the domain name can be set by issuing the following command:

  ```
  8021xdomain [domain_name]
  ```

- If using PEAPv0/EAP-MSCHAPv2 authentication and the server requires the PEAP version to be sent as part of the authentication request, the PEAP version can be enabled with the `on` option in the following command:

  ```
  8021xsendpeapver [on/off]
  ```

Set Password Policy

To set the password policy, issue the following command:

```
setpasswordrule{-all|-none}|{-length:minpasswordlength}{-mixed}{-digit}{-special}
```

- `-all` - All password rules are applied.
- `-none` - No password rules are applied.
- `-length:` - Specifies the minimum password length. By default, the minimum password length is six characters.
- `-mixed` - Specifies that the password must contain a lower and upper case character.
- `-digit` - Specifies that the password must contain a numeric character.
- `-special` - Specifies that the password must contain a special character.

> **NOTE:** The `-length`, `-mixed`, `-digit`, and `-special` parameters cannot be combined with `-none`.

Example: `setpasswordrule -length:9 -mixed -digit -special`

> **NOTE:** The following special characters are allowed: ` ~ ! @ $ % ^ & * ( ) _ + = { } [ ] | ; " < > , .

All passwords that are created, updated, or reset for local users must follow the password rules set by this command to be considered valid.

As a security best practice, Crestron recommends setting the password policy to the following:

```
setpasswordrule -length:15 -all
```

Set Date and Time

All devices use NTP to synchronize their clock. To disable NTP synchronization and set the current date and time manually, issue the following commands:

```
sntp stop
```

```
timedate hh:mm:ss mm-dd-yyyy
```

> **NOTE:** Enter the current time (24-hour clock format, including minutes and seconds) and date.

By default, the time zone is set to EST (code 014). This is never changed automatically and must be changed manually if desired. To set the time zone, issue the following command:

```
timezone [list | zone]
```

- `list` - Returns a list of all time zones and codes
- `zone` - Enter the code of the time zone to be used

Example: `timezone 005`

By default, NTP is enabled and is configured to get the time from pool.ntp.org. The device supports using up to three NTP servers, including authentication servers. Issue the following command to configure custom NTP servers for time synchronization:

```
sntp [start|stop|sync|delete {server|server2|server3}|server {args}|server2
{args}|server3 {args}]
```

- `start` - Starts synchronization
- `stop` - Stops synchronization
- `sync` - Forces synchronization one time
- `delete {server|server2|server3}` - Deletes configuration for NTP server1, server2, or server3
- `server:{address} [optional args]` - Address of primary NTP server with optional arguments
- `server2:{address} [optional args]` - Address of secondary NTP server to synchronize with optional arguments

- `server3:{address} [optional args]` - Address of secondary NTP server to synchronize with optional arguments
- `optional args`:
  - `port:{1-65535}` - NTP port (default 123)
  - `auth:{mac}` - Secured NTP MAC authentication
  - `keytype:{md5(less secured)|sha1|sha256}` - Key type for MAC authentication only (default `sha1`)

    > **NOTE:** `md5` is not allowed when FIPSMODE is on.

  - `key:{shared key}` - Preshared key between NTP client and server (MAC authentication only)
  - `keyid:{1-65535}` - Preshared key index between NTP client and server (MAC authentication only)
- No parameter - Displays the current settings

Example: `SNTP SERVER:macntp.example.com AUTH:mac KEYID:1 KEY:e5fa44f2b31c1fb553b6021e7360d07d5d91ff5e`

> **NOTE:** NTP servers are configured into a particular slot. The server configured into the `SERVER` slot will be the primary server used for time synchronization. The servers configured into the `SERVER2` and `SERVER3` slots will be used as secondary servers.

Control Subnet

Certain 4-Series control system models provide support for a separate network called a Control Subnet and have one or more network ports specifically for connecting devices to the Control Subnet. If your device has a Control Subnet, it must be configured.

The Crestron AV4, CP4N, and PRO4 have a dedicated Control Subnet, which allows for dedicated communication between the control system and Crestron Ethernet devices without interference from other network traffic on the LAN. The AV4 and PRO4 provide four Control Subnet ports that also support PoE+ using the optional PW-5430DUS power supply.

> **CAUTION:** Do not connect the **CONTROL SUBNET** port to the LAN. The **CONTROL SUBNET** port must only be connected to Crestron Ethernet devices.

When using the Control Subnet, observe the following:

- The control system acts as a DHCP server to all devices connected to the Control Subnet and assigns IP addresses as needed.
- A DNS server is built in to the control system to resolve host names.
- Only connect Crestron Ethernet devices to the Control Subnet.

- The control system operates in isolation mode by issuing the `isolatenetworks on` command. When in isolation mode, the firewall is configured so that no communication can occur between the LAN and devices on the Control Subnet. Using this mechanism, customers can protect their corporate LAN from devices on the Control Subnet.
- When in isolation mode, devices on the Control Subnet do not have any resources on the LAN side. For example, if a touch screen with a SmartObjects® technology object requiring network access is installed on the Control Subnet, the object will not work.
- Devices on the LAN do not have access to any devices on the Control Subnet. Crestron Toolbox also does not have access to these devices when it is connected to the LAN. To configure devices on the Control Subnet with Crestron Toolbox (outside of runtime), the computer running Crestron Toolbox must be connected to the Control Subnet.
- Any NAT/port mapping rules that were previously created do not work when the control system is in isolation mode.

NOTE: If the control system is running in isolation mode, Crestron Ethernet devices requiring internet access should not be connected to the **CONTROL SUBNET** port (directly or indirectly) and should be instead connected to the LAN.

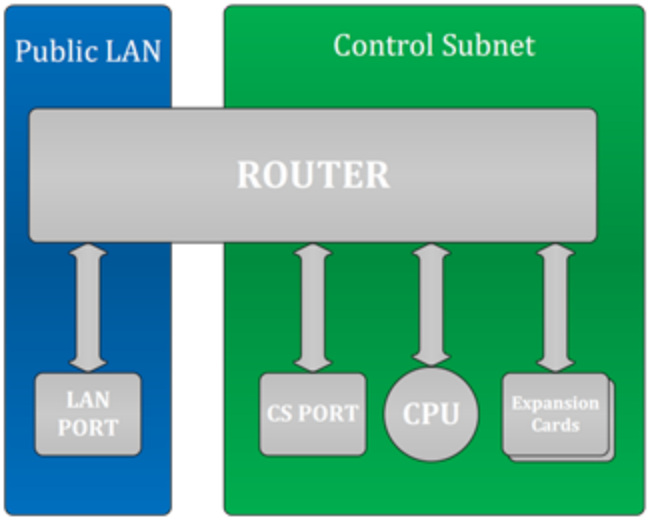
Control Subnet Architecture

Even if nothing is plugged into the **CONTROL SUBNET** port(s) on the back on the control system, the following are still present on the Control Subnet:

- Control System CPU (where AV programs run)
- Optional Expansion cards (PRO4 and AV4 only)

This design is in place to ensure that the Crestron CPU and optional expansion cards are protected from malicious packets on the LAN. Refer to the diagram below for more information on how these components work together.

Public LAN/Control Subnet Diagram



The firewall rules permit entry to only the traffic that is listened to by the CPU. As a result, a port scan will only show ports that are listened to by the CPU. Users can set up manual port forwarding rules to make custom connections to the devices on the Control Subnet.

Control Subnet Configuration

For increased security, the device supports a mode of operation called isolation mode. In isolation mode, the firewall is preconfigured to limit access to the Control Subnet, port mapping between the primary network and devices on the Control Subnet are blocked, and manual configuration of port forwarding is not available. To configure isolation mode, issue the following command:

`isolatenetworks [state]`

- `state` - {ON | OFF}
- No parameter - Displays the current setting

Example: `isolatenetworks on`

As a security best practice, the device should have its Control Subnet in isolation mode.

Control Subnet Router Configuration

By default, the Control Subnet router is configured to use 10.0.0.0/8 for the Control Subnet. If the device detects that the primary network is using that network address, the device will automatically switch to using 172.22.0.0/16 for the Control Subnet. To verify if the device is automatically choosing the Control Subnet address, issue the `CSINAutoMode` command. To confirm what address is being used, issue the `ipconfig` command to show the addresses assigned to the device's network interfaces.

If the control system will use a specific network address, that address can be configured using the following command:

`CSRouterPrefix [IP_Address/Prefix_Size]`

- `IP_Address` - The desired IP address
- `Prefix_Size` - The number of leading bits of the routing prefix
- No parameter - Displays the current Control Subnet configuration

Example: `CSRouterPrefix 192.168.0.0/24`

Control Subnet DHCP Configuration

By default, the device provides a DHCP server on the Control Subnet to issue IP addresses for anything connected to the Control Subnet. IP addresses that have been issued can be displayed by using the `DHCPLeases` command.

The `DHCPLeases` command will return a list of IP address that have been issued and information about them.

In addition, specific IP addresses can be assigned to specific devices on the Control Subnet. This can be done by issuing the following command:

`RESERVEDLeases [ADD | REM | CLEAR_ALL]`

- `ADD` - Adds an IP address to a device using the following syntax: `MAC_Address IP_Address Description`
  - `MAC_Address` - The device MAC address
  - `IP_Address` - The device IP address
  - `Description` - A description of the device
- `REM` - Removes a previously created IP address from the device using the following syntax: `MAC_Address`
  - `MAC_Address` - The device MAC address
- `CLEAR_ALL` - Clears all previous created IP addresses
- No parameter - Displays all current reserved DHCP leases in table format

Example: `RESERVEDLeases ADD 81:51:CA:48:73:24 10.0.0.9 TestDevice`

The MAC address should be in the format XX:XX:XX:XX:XX:XX on the command line.

Control Subnet Firewall Configuration

In isolation mode, the firewall is preconfigured to limit access to the Control Subnet and cannot be further configured. The firewall configuration in isolation Mode is as follows:

Control System Firewall Rules - Isolation Mode

Direction	Port(s)	Rule	Description
Inbound from LAN	22	To CPU	SSH
Inbound from LAN	80, 443	To CPU	Web server (if enabled)
Inbound from LAN	41794, 41796	To CPU	Crestron communication protocols
Inbound from LAN	Listen ports used by program	To CPU	Programmatic listeners
Inbound from LAN	49200	To CPU	Crestron HTML5 User Interface
Inbound from LAN	64000–64299	Blocked	In isolation mode, Crestron management tools cannot connect to any devices on the Control Subnet
Control Subnet Outbound to LAN	Any port	All other devices: Blocked	No outbound traffic is allowed
Inbound from LAN	User defined	Blocked	In isolation mode, no port forwarding can be managed by the user

Disable Auto Discovery

All devices support an autodiscovery feature which allows them to be detected, report basic information, and do some basic configuration remotely. This feature is not protected by any type of authentication. Disable auto discovery with the following command:

```
autodiscovery off
```

Disable Cloud Features

All devices connect to cloud services for remote monitoring and management. If your environment or policies do not permit communications with external services, disable cloud features by entering the following commands:

```
enablefeature cloudclient off
```

```
hydrogenenable off
```

Disable Wireless Communications

Certain control system models support infiNET EX® wireless communications. As a security best practice, this support should be disabled by issuing the following command:

```
rfgateway off
```

Enable User Account Locking

To prevent brute force attacks against a user's password, the device can automatically lock an account after a number of failed login attempts. This functionality operates independently and simultaneously with the device's User Login IP Blocking capability.

> **NOTE:** Access to an account over the USB port is never blocked.

Change Login Failure Count

To change the value for the login failure count, issue the following command:

```
setuserloginattempts [number]
```

- `number` - Number of login attempts a user can have before the console is blocked. A value of `0` indicates an infinite number of login attempts. A value of `-1` restores the default value.
- No parameter - Displays the current setting

Example: `setuserloginattempts 3`

As a security best practice, the failure count should be set to `3`.

Change Lockout Time

To change the duration that an IP address is blocked by the console, issue the following command:

```
setuserlockouttime [number]
```

- `number` - Number of hours (suffix `h`) or minutes (suffix `m`) to block a user. A value of `0` specifies an indefinite amount of time. The maximum amount of time is `750h` (hours) or `45000m` (minutes). A value of `-1` restores the default value.
- No parameter - Displays the current setting

Example: `setuserlockouttime 15m`

As a security best practice, the lockout time should be set to `15m`.

Display Last Logged-In Information

Devices do not display information about a user's last login or failed login attempts by default. To have this information displayed, issue the following command:

```
showlogininfo on
```

Enable Session Inactivity Timeout

> **NOTE:** The Enable Session Inactivity Timeout command affects both console and web sessions.

Devices do not terminate a user session due to inactivity by default. Configure the device to terminate inactive user sessions by issuing the following command:

```
setlogoffidletime 10
```

The number set with the `setlogoffidletime` command is the number of minutes after which the session will be terminated. The number can range from `1` to `9999`.

Enable Audit Logging

All devices have limited audit logging. Audit logging is turned off by default.

To configure audit logging, issue the following command:

```
auditlogging[on|off]{[all]|[none]|{[admin][prog][oper][user]}[remotesyslog]}
```

- `on` - Enables audit logging
- `-off` - Disables audit logging
- No parameter - Displays the current audit logging setting
- The following parameters are optional and are used to log commands by access level:
    - `admin` - Logs administrator-level commands
    - `prog` - Logs programmer-level commands

    - `oper` - Logs operator-level commands
    - `user` - Logs user-level commands
    - `all` - Logs all commands
    - `none` - Logs no commands
- `remotesyslog` - Writes to the remote syslog server only

Example: `auditlogging on admin oper`

Sample Log Output: `[2021-11-30T07:02:44-08:00]: EVENT: COMMAND(SHELL 172.30.255.255) USER: admin # AUDITLogging on all`

As a security best practice, full audit logging should be turned on by entering the following command:

```
auditlogging on all
```

Initial Login Process

A user name and password account must be created when the device is accessed for the first time. Using an SSH client, log in by entering `Crestron` and a blank password. To create the account, enter the desired user name and password (the password must be a minimum of 8 characters). Confirm the password by entering the password again. After the account is created, enter the user name and password to log in to the device.

> **NOTE:** Do not lose this information. The system cannot be accessed without it.

Disable Nonsecure CIP

By default, nonsecure incoming CIP connections are allowed. Disable these connections by issuing the following command:

```
securegatewaymode secureonly
```

Enable All Certificate Verifications

By default, outgoing TLS connections for some protocols will not perform a full set of verifications on the server certificate if it is presented. Enable these verifications by issuing the following command:

```
sslverify all
```

Load Default Server Certificates

The device requires a default server certificate for proper web server operation and to properly secure incoming CIP communications from other devices. Refer to the Required Configuration on page 6 section for instructions to load the default server certificate and any other needed certificates.

Optional Configuration

The following sections provide information about optional device configuration settings.

Enable or Disable Web Server

All devices have an active web server. If desired, disable the web server with the following command:

```
webserver off
```

To enable the web server, issue the following command:

```
webserver on
```

Enable User Login IP Blocking

To prevent distributed brute force attacks against user logins, the device can automatically block an IP address after a number of failed login attempts from that IP address. This functionality operates independently and simultaneously with the device's User Account Locking capability.

> **NOTE:** Access to an account over the USB port is never blocked.

Change Login IP Failure Count

To change the value for the logon failure count, issue the following command:

```
setloginattempts [number]
```

- `number` - Number of login attempts allowed before the console is blocked. A value of `0` enables unlimited attempts. The default value is `3`.
- No parameter - Displays the current setting

Example: `setloginattempts 3`

Change IP Blocked Time

To change the duration that an IP address is blocked by the console, issue the following command:

```
setlockouttime [number]
```

- `number` - Number of hours to block an IP address. A value of `0` blocks the IP address indefinitely. The maximum value is `255`. The default value is `24`.
- No parameter - Displays the current setting

Example: `setlockouttime 24`

Configure SNMP

The device supports SNMP v2x and v3. To configure an SNMP Manager to access SNMP on this device, it must be added with the `SNMPMANager` command and given access with the `SNMPAccess` command. When using SNMP v3, the SNMP Manager must support EngineID Discovery (RFC 5343) since there is no current way to display the EngineID being used by the device.

Enable or Disable SNMP

To enable or disable SNMP, issue the following command:

```
snmp [enable | disable | wipe]
```

- `enable` - Enables SNMP
- `disable` - Disables SNMP
- `wipe` - Clears the configuration and disables SNMP
- No parameter - Displays the current setting

Example: `snmp enable`

Add or Remove an SNMP Manager

Add information about an SNMP Manager that will be accessing the device or receiving notifications from the device. An SNMP Manager must be added even if the Manager will not be receiving notifications from the device. The Manager can be removed when no longer in use.

To add or remove an SNMP Manager, issue the following command:

```
snmpmanager [add/remove] [name] [community name] [address] [params]
```

- `params` - Specifies one of the following:

  ```
  noauthnopriv-v1
  noauthnopriv-v2
  noauthnopriv-v3
  authnopriv-v3
  authpriv-v3
  ```

  - `auth` = authentication
  - `priv` = privacy

Examples:

```
snmpmanager add testsitemanager testsitename 192.168.0.255 authpriv-v3
```

```
snmpmanager remove testsitemanager
```

For SNMPv2, the `community name` parameter is the SNMP community string. For SNMPv3, the `community name` parameter is used as the SNMPv3 user name. Entering the command with no parameters will list all the SNMP Managers that have been added.

As a security best practice, `authpriv-v3` (full SNMPv3) should be used.

Enable or Disable Unrestricted SNMP Access

By default, SNMP managers sending requests with a community string as the only authentication must send those requests from the IP address indicated when the manager was defined with the `SNMPMANager` command. The following command can be used to remove that restriction by changing the setting to `on`:

```
snmpallowall [on/off]
```

- `on` - Allows all managers
- `off` - Allows only permitted managers
- No parameter- displays current setting

By default, the command is set to `off`.

SNMPv3 requests are not affected by this command. SNMPv3 security is used to control access and does not check IP addresses.

Configure SNMP Access Information

This enables SNMP requests and provides the needed information for an SNMP Manager that has been created with the `SNMPMANager` command.

```
snmpaccess [community] [param] [-a:securitytype -p:password [-e:privacytype [-k:key] ] ]
```

- `param` = readonlyaccess, readwriteaccess, noaccess
- `securitytype` = MD5, SHA
- `privacytype` = DES, AES

The `-a` and `-p` options are required if the SNMP Manager was configured with `authnopriv-v3`,`authpriv-v3`.

The `-e` and `-k` options are required if the SNMP Manager was configured with `authpriv-v3`.

The string passed to the `-p` and `-k` options must be at least 8 characters long.

The `MD5` authentication type and `DES` privacy types are not available when the device is in FIPS 140-2 operation.

Example: `testsitename readwriteaccess -a:sha -p:secretstring1 -e:aes -k:secretstring2`

Enable or Disable SNMP Notifications

Notifications will be sent to all SNMP Managers that have been configured via the `SNMPMANager` and `SNMPAccess` commands. The device currently supports TRAP notifications and does not support INFORM notifications.

To enable or disable SNMP notifications, issue the following command:

```
snmptrap [on|off]
```

- `on` - Enables traps
- `off` - Disables traps

- No parameter- Displays the current setting

Example: `snmptrap on`

Domain-Joined Authentication and Authorization

This section describes domain-joined user authentication and authorization (A/A). This enhancement makes the original implementation of Active Directory® service user login more secure by restricting login-related interactions to the following:

- Crestron devices that are known by a domain.
- Domains that are known by Crestron devices.

When comparing domain-joined A/A with legacy Active Directory (AD), there are a few key similarities and differences:

- There is no change in how users and domain groups are configured with either approach.
- Legacy AD does not require any device-identifying or device-authenticating configuration on either the domain side or the device itself. Domain-joined A/A requires configuration of both.
- Legacy AD supports only Active Directory, while domain-joined A/A supports Active Directory and Remote Authentication Dial-In User Service (RADIUS). It also provides a framework for other A/A methods in the future.

Legacy AD has been deprecated since the introduction of domain-joined A/A, though Crestron devices will support both types for the foreseeable future. Only one of the two types can be active at any given time.

Definitions

The following terms are defined since they are used often throughout this section since they are applicable to domain configuration:

- **Authentication** is the process of validating that a user is who they assert themselves to be. Authentication is typically validated via a user ID and password. Domain-joined A/A also supports Multi-Factor Authentication (MFA). A password is something the user knows, so MFA (in general) satisfies the criteria of possession. For example, a user must have some type of authenticator application on their smart device in order to provide the correct One-Time Password (OTP) to the authentication system. Theoretically, no other user could provide the OTP when prompted.
- **Authorization** is the process of determining a user's access to some resource. Once the user's identity is known and confirmed, their access level (which is usually configured by an administrator) can be obtained. With respect to Crestron devices, a user's identity is either configured on the device itself (`adduser` and/or `addusertogroup`), or it is configured on the domain side. A user's authorization is determined based on their group membership(s) and the highest access level of these groups (Administrator, Operator, and so forth).

Activate or Deactivate Domain-Join Functionality

Domain-joined A/A is enabled only for those devices that support it and only after turning it on using the `authdomainmode` command. When a supported device is upgraded to a firmware version that contains this new functionality, the device will continue to use legacy AD following the upgrade. Once a restore is performed on the device, it will begin using domain-joined A/A.

To determine if a build contains this functionality and if it is supported by a particular device, issue the `authdomainmode` command. If this command does not exist, then that firmware version does not contain this feature. If domain-joined A/A is supported and available, this command displays the current authentication mechanism in use (legacy AD or domain-joined A/A).

Domains and Domain Types

In the context of domain-joined A/A, an authentication domain is a logical concept that is represented on the device by a domain name and a set of servers that will perform authentication and authorization for remote logins.

An authentication domain is separate from any network domain(s) assigned to the device (such as via DHCP), although they would likely have the same name in most cases.

There are two types of A/A domains supported at this time: Active Directory and RADIUS. These types refer to the protocols that are used for authenticating and authorizing users.

Active Directory Authentication Domains

An Active Directory authentication domain relies on Microsoft® Active Directory infrastructure to authenticate and authorize user logins on the device. Kerberos is used as the authentication protocol, while LDAP is used to obtain group membership information in the case of a domain group login attempt.

As with legacy AD, Active Directory authentication domains can allow both specific domain users and domain group members access to the device. Specific users are configured with the `addusertogroup` command. Domain groups are configured with `adddomaingroup`.

The authenticating mechanism between the device and domain uses a Kerberos keytab file. This file contains encrypted keys that authenticate the device to the domain controller. The keys in the keytab are also used to validate that a user has been authenticated by a known AD controller.

When configuring an Active Directory authentication domain on the device, the following information is required:

- The fully qualified name of the authentication domain (for example, "authdomain.com").
- Domain controller addresses, in name or numeric form.

  If a controller's address is specified by name, only the first part of the FQDN is needed. The name of the authentication domain is appended to the controller's name to obtain the FQDN. Refer to the following example valid controller addresses:

  - 192.168.1.100
  - 2607:f8b0:4000:819::200e
  - ad-controller1

- A keytab file. Keytab files are uploaded to the device's **/sys/** folder. If the keytab file name is not specified when adding an authentication domain, a default name is used.
- The domain-side computer account name for the device. An account must be created on the domain controller for each Crestron device.
- The name of the authentication realm over which the domain controller has authority. If not provided, this value defaults to the uppercase value of the authentication domain name. For the example domain name above, the default realm name would be "AUTHDOMAIN.COM".

Configuration is also necessary on the domain side (the Active Directory domain controller). There, each device requires:

- A computer account with a specific service principal name.
- A Kerberos keytab (the same one used on the device).

The following sections describe this configuration in more detail by showing how a sample device named "crestron-device-1" is added to an Active Directory domain called "crestron-domain.org". The domain controller host name in this case is "dc1".

Create the Domain-Side Computer Account

The first step to join a Crestron device to an AD domain is to create an account on the domain server for the device. The details of this process are outside the scope of this document, but in general, it is no different than adding a user account to the domain. For a computer account, the only required information at the time of creation is the computer name. This name should be something that uniquely identifies the Crestron device. In this example, the computer name assigned to the computer account is "crestron-device-1".

Add the Service Principal Name to the Computer Account

Each computer account that represents a Crestron device must have a Service Principal Name (SPN) called "host". This SPN represents a resource (the Crestron device) to which a user requests access.

One way to add the SPN to the computer account is to use the `setspn` command:

1. On the domain controller, open a Windows® command or PowerShell® prompt with Administrator credentials.
2. Execute the `setspn` command to add the "host" SPN to the computer account as follows:

   ```
   setspn -A host/<computer_name>.<domain_name>@<realm_name> <computer_name>
   ```

   For this example, the realm name is the same as the domain name in uppercase. This naming convention is typical. The exact command for this example is as follows:

   ```
   setspn -A host/crestron-device-1.crestron-domain.org@CRESTRON-DOMAIN.ORG crestron-device-1
   ```

3. Verify the SPN has been assigned as follows:

   ```
   setspn -L crestron-device-1
   ```

Set Encryption for the Computer Account

When a user accesses a Crestron device that belongs to an AD domain, the user's credentials are verified to ensure they were obtained from a known and valid domain controller. This is done by requesting a service ticket for the computer account using the credentials obtained via the user's authenticated password. The domain controller encrypts the service ticket before sending it to the device. Crestron devices support only AES encryption for service tickets.

Windows provides a PowerShell command that can be used to set the encryption type for the computer account:

```
Set-ADComputer <computer_account_sAMAccountName> -KerberosEncryptionType AES128, AES256
```

Using the example device above, the command to set the account encryption type to AES is as follows:

```
Set-ADComputer crestron-device-1$ -KerberosEncryptionType AES128, AES256
```

By default, the account name of the device consists of a $ character appended to the device name. This command should work with or without the $ character.

Generate the Keytab File

To authenticate with a domain controller, a Crestron device needs a password, and preferably one that is known only to itself and the domain controller. A keytab file contains pairings of SPNs and keys, which are derived from the shared password. The keytab file for a Crestron device can be generated on the domain controller by using the following command:

```
ktpass -princ host/<computer_name>.<domain_name>@<realm_name> -out <computer_name>.keytab -crypto
all -ptype KRB5_NT_SRV_HST -mapuser <computer_name>$@<domain_name> +setupn +rndPass +setpass
```

For the example in this document, the command would be as follows:

```
C:\> ktpass -princ host/crestron-device-1.crestron-domain.org@CRESTRON-DOMAIN.ORG -out crestron-
device-1.keytab -crypto all -ptype KRB5_NT_SRV_HST -mapuser crestron-device-1$@crestron-
domain.org +setupn +rndPass +setpass
```

This command generates and sets a random password for the crestron-device-1 computer account, then generates a keytab file in the current directory named "crestron-device-1.keytab", which contains keys for the "host" SPN.

Transfer the Keytab File to the Crestron Device

The keytab must be present on the device prior to configuring the authentication domain. Use a file transfer program to place the keytab file in the device's **/sys/** folder. In the next step, the name of the keytab file must be provided unless it is renamed to "domain.keytab" (the default keytab file name) when it is transferred.

It is not necessary to keep the keytab file on the Windows domain server after it is installed on the device. The file is used only on the device and should be removed from other locations.

For purposes of this example, the device's keytab file is uploaded to **/SYS/crestron-device-1.keytab/**.

Configure the Authentication Domain on the Crestron Device

The next step is to configure the authentication domain details on the device using the `addauthdomain` command. This can be done without an active network connection. By default, no interaction with the domain controller takes place during this operation.

Several parameters needed for domain configuration will use default values if not provided. The simplest method to add the authentication domain in this example is as follows (using the example "dc1" domain controller host name).

1. Open a console on the Crestron device and log in using a local administrator account.
2. Issue `addauthdomain` to add the "crestron-domain.org" authentication domain to the device:

```
ADDAUTHDOMAIN -N:crestron-domain.org -H:dc1 -D:crestron-device-1 -K:crestron-device-
1.keytab
```

3. Issue the `listauthdomains` command to show all configured authentication domains.

NOTE: Observe the following when using the `addauthdomain` command:

- In general, using a controller's host name ("dc1" in this example) is preferable to using a fixed IP address. This allows DNS to resolve the name and avoids the need to update configuration if the address changes.
- When contacting the domain controller, the device appends the domain name to the host name to get the FQDN. In this example, the combined name is "dc1.crestron-domain.org".
- The name of the authentication realm was not provided as part of the configuration, so it used the default of the uppercase domain name. In this example, the realm is set to "CRESTRON-DOMAIN.ORG".
- The authentication domain type was not specified and used the default type of "AD".

Validate the Authentication Domain

Crestron devices can validate that the authentication details configured using `addauthdomain` are correct before a user attempts to log in. This validation is not necessary prior to AD login attempts. However, it is good practice to validate an authentication domain prior to use, as it may help to troubleshoot failed login attempts in the future.

Validation can be performed at any point after the authentication domain is configured and may be repeated any number of times. A negative outcome does not prevent AD login attempts, but it is a good indication that any such attempts will fail. A positive result means that the device has successfully joined the domain.

Use the `validateauthdomain` command to check that domain configuration is correct on both the domain side and the device. Continuing with the current example, the domain added to the device in the previous step is validated as follows:

```
VALIDATEAUTHDOMAIN crestron-domain.org
```

Network connectivity between the device and domain controller is required for a validation attempt. For an AD-type authentication domain, validation consists of the device authenticating itself as the computer to which the computer account on the domain controller belongs. In other words, the device is logging in to the domain as itself. This results in the validation of the domain controller host address, the computer account, the SPN, and the keytab file.

Nothing is persistent about this login attempt: no new credentials are saved on the device, no authorization is granted to any specific resource, and no logout is necessary. The device maintains the outcome of the most recent validation attempt, but only as an indication that the authentication domain credentials were valid at some point in time.

RADIUS Authentication Domains

A RADIUS authentication domain uses the RADIUS protocol to authenticate and authorize user logins on the device. In RADIUS terms, the Crestron device is the Network Access Server (NAS) and sends A/A requests to a configured RADIUS server. Specifics regarding configuration of a RADIUS server are outside the scope of this document, but the server must have access to a user database, and it must provide some means of controlling which users may access the device.

Both specific domain users and domain group members may be authenticated and authorized via a RADIUS authentication domain. As with AD-based authentication domains, specific users are configured with the `addusertogroup` command. Domain groups are configured with `adddomaingroup`.

RADIUS-based authentication domains can support MFA as described in Definitions on page 21.

When accessing a RADIUS authentication domain, a Crestron device must authenticate itself using a shared secret, which is configured on the device and on the RADIUS server. This shared secret allows the device to join the RADIUS authentication domain.

Configuring a RADIUS authentication domain on Crestron devices requires the following information:

- The fully qualified name of the authentication domain (for example, "authdomain.com").
- The RADIUS server addresses, in name or numeric form.

  If a server's address is specified by name, only the first part of the FQDN is needed. The name of the authentication domain is appended to the controller's name to obtain the FQDN. Refer to the following example valid controller addresses:

  - 192.168.1.100
  - 2607:f8b0:4000:819::200e
  - radius-server1
- The shared secret used to authenticate the Crestron device to the RADIUS server.

Configuring a Crestron device on a RADIUS server may vary according to the server specifics, but in general, the following information is needed:

- The device's name or IP address.
- A shared secret known to the server and the device that is used by the device when accessing the server.

The following sections describe this configuration in more detail by showing how a device named "crestron-device-1" is added to a RADIUS authentication domain named "radius.crestron.com". The RADIUS server host address is 10.20.30.40.

Configure the RADIUS Server

Details concerning RADIUS server configuration are outside the scope of this document and will vary between different servers. In general, each Crestron device that uses RADIUS to authenticate and authorize users must be added to a device list on the server. A shared secret is also assigned to each device.

It may also be possible or necessary to configure rules on the server that control user-level access, such as which users are allowed, how and when access is allowed, specific per-user authorization levels, and so forth.

RADIUS based logins (with or without MFA) are supported using any A/A server that can both understand standard RADIUS messages sent to it and is capable to sending back a Crestron vendor-specific attribute to indicate the user's access level. The following Crestron vendor-specific attributes must be returned by the RADIUS server:

```
# Crestron vendor ID as assigned in IANA Enterprise Vendor Numbers:
#  https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers
VENDOR          Crestron                3212

BEGIN-VENDOR    Crestron

# Crestron-Access-Level can be used by a server to send a Crestron-specific
# access level to a device, typically in an Access-Accept message.
ATTRIBUTE       Crestron-Access-Level   64       string

# Crestron-Access-Group is used to send a single "domain group" name to a
# device, typically when a user authentication is successful.
# This attribute may appear multiple times in an Access-Accept message, if
# the user is a member of multiple user/domain groups.
ATTRIBUTE       Crestron-Access-Group   65       string

END-VENDOR      Crestron
```

Configure the Authentication Domain on the Crestron Device

The next step is to configure the authentication domain details on the device using the `addauthdomain` command. This can be done without an active network connection. By default, no interaction with the RADIUS server takes place during this operation.

For the example device and domain above, use the following process:

1. Open a console on the Crestron device and log in using a local administrator account.
2. Issue `addauthdomain` to add the "radius.crestron.com" authentication domain to the device:

   ```
   ADDAUTHDOMAIN -N:radius.crestron.com -H:10.20.30.40 -T:radius -P:password
   ```

   The `-T:` option identifies the type of the authentication domain. Because the default is "AD", it must be changed to a RADIUS-type authentication domain.

In this example, the shared secret used to identify the device to the RADIUS server is "password". As shown above, the shared secret can be configured using the `-P:` option. If the shared secret is not supplied as part of the `addauthdomain` command, the device will prompt for it.

3. Issue `listauthdomains` to show all configured authentication domains.

Validate the Authentication Domain

The `validateauthdomain` command can be used to verify that the configuration of a RADIUS-type authentication domain is correct. Validation of a domain is not required, but it provides a means to verify that the RADIUS server details as configured on the device are correct.

Validation can be performed at any point after the authentication domain is configured and may be repeated any number of times. A negative outcome does not prevent AD login attempts, but it is a good indication that any such attempts will fail. A positive result means that the device has successfully joined the domain.

Using the current example, the domain added to the device in the previous step is validated as follows:

```
VALIDATEAUTHDOMAIN radius.crestron.com
```

Network connectivity between the device and domain controller is required for a validation attempt. For a RADIUS-type authentication domain, validation consists of the device establishing a connection to the server and authenticating itself via the shared secret configured for the domain. The process uses a ping-like mechanism in which the device sends a RADIUS "Server-Status" message to the server and receives a response.

If the RADIUS server does not support the "Server-Status" message, then the validation attempt will fail (after a timeout/restransmit process takes place). Microsoft Network Policy Server is an example of a RADIUS server that does not support this RADIUS message and therefore cannot be validated using the `validateauthdomain` command. The `testlogin` command can be used for validation instead, but it requires actual user credentials.

Console Commands for Managing Authentication Domains

The following console commands can be used to manage authentication domains.

Add a New Authentication Domain

Use the `addauthdomain` command to add authentication domains to a device. Crestron devices support a fixed number of authentication domains. Once the maximum number of domains has been reached, no new domains can be added.

```
addauthdomain -n:domain_name -h:host [-t:{AD | RADIUS}] {-p:shared_secret} | {-
d:device_account [-k:keytab_filename] [-r:realm_name]} [-v]
```

- `-n:domain_name` - Specifies the name of the authentication domain to configure
- `-h:host` - Specifies either an IP address (v4 or v6) or server name that will authenticate/authorize users logging in to this device

- `-t:domain_type` - Indicates the authentication/authorization protocol to use for network logins with this domain (defaults to AD if not provided)

  For RADIUS domains:

  - `-p:shared_secret` -The password used to authenticate this device to a domain-side server

  For AD domains:

  - `-d:device_account` - The domain-side computer account associated with this device
  - `-k:keytab_filename` - The name of the keytab file to use when authenticating to the domain

    The keytab file must be uploaded to the **/SYS/** folder on the device, and will use a default name if not specified

  - `-r:realm_name` - The name of the authentication realm (defaults to <DOMAIN_NAME> if not specified)
  - `-V` - Attempts to validate the configuration with the domain

Example: `addauthdomain -n:radius.crestron.com -h:10.20.30.40 -t:radius -p:password`

The `-h:` option can be used multiple times to configure multiple hosts or multiple addresses for the same host. An authentication domain supports a fixed number of hosts. If too many hosts are provided, the command fails. The order in which the host addresses are used depends on several factors (OS settings, Kerberos, and so forth), but in general, they should be provided using `addauthdomain` in preference order from first to last. If an attempt to contact a host fails, the device will try the next configured host. Crestron devices do not support round-robin usage of domain hosts.

The `-v` option performs validation of the domain after configuration is saved. If this option is used and validation fails, the domain remains configured. If configuration is incorrect, the domain can be removed and added back.

Validate an Authentication Domain

Use the `validateauthdomain` to validate an authentication domain. Validation of an authentication domain ensures that configuration details are correct on both the domain side and on the device. The exact details of the validation process vary depending on the type of authentication domain that is being validated. There are two ways to validate authentication domains:

- Append the `-v` option to the `addauthdomain` command as described in Add a New Authentication Domain on page 28.
- Use the `validateauthdomain` command as described in this section.

`validateauthdomain domain_name`

- `domain_name` - Specifies the name of the authentication domain to validate. Depending on the domain type, validation may require network connectivity.

Example: `validateauthdomain radius.crestron.com`

It is not necessary to validate an authentication domain before use, but the outcome of a validation attempt is a strong indicator as to whether domain logins using the authentication domain will succeed or fail.

The `validateauthdomain` command can be executed any number of times and at any point in time to check the domain-joined status of the device.

List Configured Authentication Domains

Use the `listauthdomains` command to list all configured authentication domains. Refer to the following example:

```
Auth Domain Name | Type | Server
-----------------------+----------+--------------------
crestron-domain.org | AD | dc1
```

Delete an Authentication Domain

Use the `delethauthdomain` command to remove authentication domains from the device.

`deleteauthdomain domain_name`

- `domain_name` - Specifies the name of the authentication domain to delete. Only deletes domain configuration on the device (not on the domain side).

Example: `deleteauthdomain radius.crestron.com`

As noted above, domain-side configuration remains as is. If it must be removed, domain-specific commands must be used to remove it (for example, delete or disable the computer account, and so forth).

Once an authentication domain is removed from the device, users can no longer log in to that device using credentials from the deleted domain. Removal of an authentication domain does not affect any users currently logged in to the device with credentials from that domain.

Display or Change the Current Authentication Domain Mode

Use the `authdomainmode` command to do the following:

- Display the current authentication domain mode
- Enable domain-joined mode
- Switch back to legacy AD mode (if it is supported)

`authdomainmode [on|off]`

- `on` - Enables authentication domain mode and disables legacy AD login
- `off` - Disables authentication domain mode and enables legacy AD login
- No parameter - Displays the current setting

Example: `authdomainmode on`

If legacy AD is in use, this command can be used to enable authentication domains. When authentication domains are in use, this command can be used to switch back to legacy AD. The new mode will remain active until one of the following occurs:

- The next time the device is restored, at which point authentication domains will be enabled if the current mode is legacy AD
- The mode is changed again via the `authdomainmode` command.

Legacy AD cannot be enabled until all configured authentication domains are removed from the device via the `deleteauthdomain` command.

Add Users and Groups

It is likely that additional users—either local or via Active Directory® or RADIUS credential management—will need to be given access to the device. Refer to the Optional Configuration on page 18 section for instructions.

Enable Sending Audit Logs to Remote Syslog Server

Devices do not send audit logs to a remote Syslog server by default. To enable sending to a remote Syslog server, issue the following command:

```
remotesyslog [-s:] {-e:} {-a} [-i:address] [-p:port] {-t:protocol} {-v:on|off}
```

- `-s:on|off` enables or disables remote system error logging
- `-e:ok|info|notice|warning|error|fatal` decides which types of errors are logged. Selecting a tier results in logging errors of that level of importance and above in a hierarchy from `ok` to `fatal`.
  - `ok` - Logs all "OK" errors and above to Syslog
  - `info` - Logs all "info" errors and above to Syslog
  - `notice` - Logs all "notice" errors and above to Syslog (default)
  - `warning` - Logs all "warning" errors and above to Syslog
  - `error` - Logs all "error" errors and above to Syslog
  - `FATAL` - Logs all "fatal" errors and above to Syslog
- `-a log`
  - Accesses Syslog contents of the audit log if remote system error logging is enabled
- `-i:address`
  - Replaces `address` with the remote Syslog server IP address in dot decimal notation or an ASCII string containing the server host name (max 255 characters)
- `p:port`
  - Replaces `port` with the remote Syslog server port number in decimal notation
- `-t:tcp|udp|ssl`
- `-v:on|off`
  - If `ssl` is selected, select `on` to verify the server or `off` to not verify the server. Not entering a parameter displays the current setting.

To test the command, run the following script:

```
rsyslog -s:on -a -i:172.30.144.58 -p:23456 -t:SSL -v:off
```

As a security best practice, the options `-t:ssl` and `-v:on` should be used.

Secure Control System Connection

If this device is connected to another control system, set the user name and password for the control system CIP connection by issuing the following command:

```
setcsauthentication -n:username -p:password
```

- `n` - Specifies name of the user (domain users enter domain\username)
- `p` - Specifies password

Example: `setcsauthentication -n:remotecs -p:randompassword string`

Management Functions

The following sections provide information about device management functions.

Firmware Update

To perform a firmware update:

1. SFTP the .puf firmware file to the **/firmware** location on the device.
2. Enter the `puf <filename>` command in the console, where `<filename>` is the complete filename of the .puf file, including the filename extension.

User and Group Management

Local users and groups can be added to the device after an administrator account has been created. Additionally, the device can grant access levels to existing domain (Active Directory and RADIUS) users and groups.

The following sections describe how to manage users and groups on the device.

	1	2	3	4	5	6
Administrator	Yes	Yes	Yes	Yes	Yes	Yes
Programmer	Yes	Yes	Yes	No	Yes	Yes
Operator	Yes	Yes	Yes	No	No	Yes
User	No	Yes	No	No	No	No
Connection Only	No	Yes	Yes	No	No	No

Add Local User

To add a local user to the device, issue the following command:

```
adduser -n:username -p:password
```

- `username` - Specifies the name of the local user that is to be created
- `password` - Specifies a password for the local user

Example: `adduser -n:jsmith -p:user01`

A local user is created without access rights. To assign access rights to a local user, the user must be added to at least one local group. For more information, refer to the Add User to Group on page 36 section.

Delete Local User

To remove a local user from the device, issue the following command:

```
deleteuser username
```

- `username` - Specifies the name of the local user who is to be removed

When a local user is removed, the user is also removed from any local groups.

Add Local Group

To add a local group to the device, issue the following command:

```
addgroup -n:groupname -l:accesslevel
```

- `groupname` - Specifies the name of the local group that is to be created
- `accesslevel` - Specifies the access level for the local group:
  - a - Administrator
  - p - Programmer
  - o - Operator
  - u - User
  - c - Connection only

Example: `addgroup -n:cresprogs -l:p`

> **NOTE:** A predefined access level must be assigned to a group when it is created.

When a user is added to a group, the user inherits the access level set for the group. Certain device functions and console commands are accessible only to users with corresponding access levels.

If a user belongs to multiple groups, the user's access level is the combined access level of all groups that contain the user.

Delete Local Group

To remove a local group from the device, issue the following command:

```
deletegroup groupname
```

- `groupname` - Specifies the name of the local group

When a local user group is removed, users in the group are not removed from the device. However, the user will lose the access rights associated with the removed group.

List Local Groups

Users with administrator privileges can view all local groups added to the device. The device comes with the following built-in groups that cannot be deleted by any user: Administrators, Programmers, Operators, Users, and Connects.

To view a list of all local groups added to the device, issue the following command:

```
listgroups [a] [p] [o] [u] [c]
```

- `a` - Groups with administrator rights are listed
- `p` - Groups with programmer rights are listed
- `o` - Groups with operator rights are listed
- `u` - Groups with user rights are listed
- `c` - Groups with connect-only rights are listed

Example: `listgroups p`

Add Domain Group

To add an existing domain group to the device, issue the following command:

```
adddomaingroup -n:groupname -l:accesslevel
```

> **NOTE:** Use the `adlogin` command to log in to the Active Directory server.

- `groupname` - Specifies the name of the domain group to be added
- `accesslevel` - Specifies the access level for the domain group:
  - a - Administrator
  - p - Programmer
  - o - Operator
  - u - User
  - c - Connection only

Example: `adddomaingroup -n:adprogs -l:p`

> **NOTE:** The device cannot create or remove a group from the domain service, but it can grant an access level to an existing domain group.

All users of the domain group inherit the access level set for the group. Certain device functions and console commands are accessible only to users with corresponding access levels.

Remove Domain Group

To remove a domain group from the device, issue the following command:

```
deletedomaingroup groupname
```

- `groupname` - Specifies the name of the domain group

When a domain group is removed from the device, it is not deleted from the domain service. Once the group is removed from the device, all members of that group lose access to the device.

List Domain Groups

Users with administrator privileges can view all domain groups that were added to the device by issuing the following command:

`listdomaingroups [a] [p] [o] [u] [c]`

- `a` - Domain groups with administrator rights are listed
- `p` - Domain groups with programmer rights are listed
- `o` - Domain groups with operator rights are listed
- `u` - Domain groups with user rights are listed
- `c` - Domain groups with connect-only rights are listed

Example: `listdomaingroups p`

List Users

To view all users (local and domain) that have been added to local groups, issue the following command:

`listusers`

- No parameter - Lists all users that have been added to local groups

List Group Users

To view all users that have been added to a specific group, issue the following command:

`listgroupusers groupname`

- `groupname` - Specifies the group name that should be queried

Example: `listgroupusers cresprogs`

Show User Information

To view the access rights of a particular user, issue the following command:

`userinformation username`

- `username` - Specifies the user name that should be queried

Example: `userinformation jsmith1`

Add User to Group

To add a local or a domain user to a local group, issue the following command:

`addusertogroup -n:username -g:groupname`

- `username` - Specifies the name of the local or domain user
- `groupname` - Specifies the name of the local group

Example: `addusertogroup -n:jsmith1 -g:cresprogs`

Local users are created on the device without any access rights. Adding a user to a local group grants the user the access level assigned to the group.

> **NOTE:** The device cannot create or remove a user from the domain service, but it can grant an access level to an existing domain user. This may be accomplished either by adding the domain user to a local group on the device or by adding the domain group(s) of which the user is a member to the device.

Remove User from Group

To remove a local or a domain user from a local group, issue the following command:

`removeuserfromgroup -n:username -g:groupname`

- `username` - Specifies the name of the local or domain user
- `groupname` - Specifies the name of the local group

Example: `removeuserfromgroup -n:jsmith1 -g:cresprogs`

Update Local Password

To update the current user's password, issue the following command:

`updatepassword`

Users may update their password. The user is prompted to enter the current password once and the new password twice. If the old password does not match the current password, the operation fails and the password is not changed.

Reset User Password

To reset a user's password, issue the following command:

`resetpassword -n:username -p:defaultpassword`

- `username` - Specifies the user whose password will be reset
- `defaultpassword` - Specifies a default password that can be provided to the user following the reset

Example: `resetpassword -n:jsmith1 -p:Default321!`

User Login IP Blocking Management

When User Login IP Blocking is enabled and a user reaches the maximum number of login attempts over an Ethernet connection, the client's IP address is blocked. Administrators have access to commands that allow them to manage the blocked IP addresses, including manually blocking and unblocking IP addresses.

List Blocked IP Addresses

To view all blocked IP addresses, issue the following command:

`listblockedip`

- No parameter - Lists all blocked IP addresses

Add IP Address to Blocked List

To add an IP address to the blocked list manually, issue the following command:

`addblockedip [ipaddress]`

- `ipaddress` - Enter the IP address that is to be blocked
- No parameter - Lists all blocked IP addresses

Example: `addblockedip 255.255.255.255`

Remove IP Address from Blocked List

To remove an IP address from the blocked list manually, issue the following command:

`remblockedip [ALL|ipaddress]`

- `ipaddress` - Enter the IP address that will be removed from the blocked list
- `ALL` - Remove all blocked IP addresses
- No parameter - Lists all blocked IP addresses

Example: `remblockedip 255.255.255.255`

User Account Locking Management

When User Account Locking is enabled and a user reaches the maximum number of login attempts, the user account is locked. Administrators have access to commands that allow them to manage the user accounts, including manually locking and unlocking accounts.

Add User to Locked List

To add a user to the locked list, issue the following command:

`addlockeduser [name]`

- `name` - Specifies the user account that is to be locked.
- No parameter - Lists all locked user accounts

Example: `addlockeduser jsmith1`

Remove User from Locked List

To remove a user from the locked list, issue the following command:

`remlockeduser [name]`

- `name` - Specifies the user account that is to be removed from the locked list.
- No parameter - Lists all locked user accounts

Example: `remlockeduser jsmith1`

List Locked User

To view a list of locked user accounts, issue the following command:

`listlockeduser`

- No parameter - Lists all locked user accounts

Certificate Management

X.509 certificates are used for a number of purposes by the device, including authentication by various protocols. These certificates can be added, removed, and managed from the console. It is important to understand the different kinds of certificates, their purpose, and how to install and configure each of them.

The device supports three basic types of certificates:

- **Trust Certificates:** These certificates are used to determine whether certificates presented by other entities are trusted. There are two types of trust certificates: Root and Intermediate. Both types serve the same purpose.
- **Server Certificates:** A server certificate is a certificate presented by a protocol when acting as a server to prove its identity. Clients connecting to that server will verify that server certificate. Server certificates loaded onto the device must also load the associated private key for that certificate since the private key is required as part of the process of proving identity.
- **Client Certificates:** A client certificate is a certificate presented by a protocol when acting as a client to prove its identity. When a client connects to a server, that server will verify that client certificate. Client certificates loaded onto the device must also load the associated private key for that certificate since the private key is required as part of the process of proving identity.

> **NOTE:** There are some certificates that can be both a server and client certificate and, therefore, can be used for either purpose.

The device stores certificates by category based upon how they are used:

- **Root:** These are the default Trust Certificates to which the device will verify server certificates against when acting as a TLS client. Root certificates are the start of a certificate chain and can be identified by the Issuer and Subject fields of the certificate being the same. The device may use an alternate list of trusted certificates for certain protocols or use cases but, unless specifically indicated, this Root store will be used.
- **Intermediate:** This is identical to the Root category, except that this store contains only intermediate certificates, which are Trust Certificates that were signed by another certificate (the Issuer field will be different than the Subject field). The default list of trusted certificates is the combination of all the Root and Intermediate certificates.
- **Default Server:** This category contains a single server certificate and is the default server certificate. This must include a private key. If a server certificate is needed by the device, and none is specifically loaded for a particular purpose, then this one will be used. This certificate cannot be loaded by the standard certificate management commands, but is instead loaded by special commands and is required as part of activating full authentication on the device. Refer to the Default Server Certificate on page 43 section for more information.
- **Machine:** This category contains a single client certificate and is used only for 802.1X, and only when EAP-TLS authentication is chosen. This must include a private key.
- **Web Server:** This category contains a single server certificate and is the server certificate used by the web server. This must include a private key. If no web server certificate is loaded, the default server certificate will be used.

Certificate Requirements

The device supports standard X.509v3 certificates. The following algorithms are supported for the public key and signatures:

- **RSA**: Key lengths of 2048, 3072, or 4096 bits
- **ECC**: secp256r1, secp384r1, and secp521r1
- **Hash**: SHA-1, SHA-256, SHA-384, or SHA-512

Certificate Signing Request (CSR) generation for the default server certificate can only generate a 2048 bit RSA key and can only use a SHA-256 hash for its signature.

Certificate Commands

The following sections provide information about commands that allow the user to add, remove, and show certificates. These commands do not apply to the default server certificate.

Add a Certificate (Fixed File Name)

To add a certificate that has a predefined file name, load the certificate file into the **/cert** directory on the device using SFTP. The file must have the file name specified below, depending on the type of certificate.

```
certificate add <certificate store> [password]
```

- `certificate store` - Specifies the category name indicating the purpose of the certificate: `root`, `intermediate`, `machine`, or `webserver`.
- `password` - Specifies the password required to access a private key in the file. It is optional and only used when a password-protected private key is included in the file.

Example: `certificate add intermediate`

The file name to use along with the format and contents of the certificate file all depend on the category chosen:

- `root`: The file must be named **root_cert.cer** and must be in standard pem format. It should only contain a root certificate.
- `intermediate`: The file must be named **intermediate_cert.cer** and must be in standard pem format. It should only contain an intermediate certificate.
- `machine`: The file must be named **machine_cert.pfx** and must be in standard PKCS #12 format. It should only contain a client certificate and its associated private key. If a password is needed to access the file, it must be provided as part of the command.
- `webserver`: The file must be named **webserver_cert.pfx** and must be in standard PKCS #12 format. It should only contain a server certificate and its associated private key. If a password is needed to access the file, it must be provided as part of the command. Make sure to load the web server certificate's signing chain into the Root and Intermediate Trust stores before loading the web server certificate itself. If the signing chain is not present, loading of the web server certificate will fail. If that signing chain is not available, or loading it into the device is not desired, disable the verification check prior to loading the web server certificate by issuing the `sslverify -s:off` command.

Certificates are stored by category, which must be specified when using any of the standard certificate management commands.

Add a Certificate (Specified File Name)

To add a certificate that has a user-defined file name, the command is identical to the previous command for loading certificates with a fixed file name—the only difference is that the file name to be used is specified as part of the command. Load the certificate file into the **/cert** directory on the device using SFTP. The file must have the file name specified below, depending on the type of certificate.

```
certificate addf <certificate name> <certificate store> [password]
```

- `certificate name` - Specifies the file name containing the certificate
- `certificate store` - Specifies the category name indicating the purpose of the certificate: `root`, `intermediate`, `machine`, or `webserver`
- `password` - Specifies the password required to access a private key in the file. It is optional and only used when a password-protected private key is included in the file.

Example: `certificate addf device-server.pfx webserver secretpass`

The format and contents of the certificate file depend on the category chosen:

- `root`: The file must be in standard pem format. It should only contain a root certificate.
- `intermediate`: The file must be in standard pem format. It should only contain an intermediate certificate.
- `machine`: The file must be in standard PKCS #12 format. It should only contain a client certificate and its associated private key. If a password is needed to access the file, it must be provided as part of the command.
- `webserver`: The file must be in standard PKCS #12 format. It should only contain a server certificate and its associated private key. If a password is needed to access the file, it must be provided as part of the command. Make sure to load the web server certificate's signing chain into the Root and Intermediate Trust stores before loading the web server certificate itself. If the signing chain is not present, loading of the web server certificate will fail. If the signing chain is not available, or loading it into the device is not desired, disable the verification check prior to loading the web server certificate by issuing the `sslverify -s:off` command.

Remove a Certificate

To remove a certificate from the device, issue the following command:

```
certificate rem <certificate store> [certificate number] [certificate name]
[certificate uid]
```

- `certificate store` - Specifies the category name indicating the purpose of the certificate: `root`, `intermediate`, `machine`, or `webserver`
- `certificate number` - Specifies the number that identifies the specific certificate to remove
- `certificate name` - Specifies the name that identifies the specific certificate to remove
- `certificate uid` - Specifies the UID that identifies the specific certificate to remove

Example: `certificate rem intermediate 1`

Only one identifier (number, name, or UID) is needed. These identifiers can be determined by listing the certificates using the command described below.

View a Certificate

To view additional details about a certificate, issue the following command:

```
certificate view <certificate store> [certificate number] [certificate name]
[certificate uid]
```

- `certificate store` - Specifies the category name indicating the purpose of the certificate: `root`, `intermediate`, `machine`, or `webserver`
- `certificate number` - Specifies the number that identifies the specific certificate to view
- `certificate name` - Specifies the name that identifies the specific certificate to view
- `certificate uid` - Specifies the UID that identifies the specific certificate to view

Example: `certificate view intermediate 1`

Only one identifier (number, name, or UID) is needed. These identifiers can be determined by listing the certificates using the command described below.

List Certificates

To show the list of certificates loaded in the device for a specific category, issue the following command:

```
certificate listn <certificate store>
```

- `certificate store` - Specifies the category name indicating the purpose of the certificate: `root`, `intermediate`, `machine`, or `webserver`

Example: `certificate listn root`

The certificates will be listed with their name and identifiers, which can be used for the remove and view commands.

Default Server Certificate

The default server certificate must be loaded into the device in order for clients to properly authenticate TLS connections.

Make sure to load the default server certificate's signing chain into the Root and Intermediate Trust stores before loading the default server certificate itself. If the signing chain is not present, loading of the default server certificate will fail. If the signing chain is not available, or loading it into the device is not desired, disable the verification check prior to loading the default server certificate by issuing the `sslverify -s:off` command.

Prior to a default server certificate being loaded, a certificate that is self-signed and self-generated by the device will be used as the default server certificate.

Load Default Server Certificate and Enable Authentication

To load the default server certificate of the device, issue the following command:

```
ssl [off | self | ca [-p:privatekeypassword]]
```

- `off` - No effect, TLS cannot be turned off
- `self` - Reverts to using the self-signed and self-generated certificate
- `ca` - Loads the default server certificate and enables use of the certificate
- `p:privatekeypassword` - Indicates that the private key associated with the default server certificate is password protected and specifies the password that should be used to access it

Example: `ssl ca`

As a security best practice, a default server certificate should be loaded by issuing the `ssl ca` command.

After issuing the SSL command, the device must be rebooted in order for the changes to take effect.

If the private key is protected by a password and the `-p` option is not provided, the command will ask for the password interactively.

To replace the existing default server certificate with a new one, issue the `ssl ca` command again.

When the `ssl ca` command is executed, the default server certificate information must be in a specific location in specific file names. Some information may also need to be installed using the standard certificate management commands. The following requirements for this information must be met before executing the `ssl ca` command:

- All information related to the default server certificate must be broken up into separate files. This means one file for the server certificate, one file for the private key, one file for the root certificate, and one file for each intermediate certificate. If a CSR was generated on the device (see instructions below), no private key file will be needed because it is already on the device.
- Load the intermediate certificates into the intermediate store using the `certificate add` or `certificate addf` command as described above in the standard certificate management commands.
- Load the root certificate into a filed named **rootCA_cert.cer** in the **/sys** directory of the device using SFTP. The file must be in standard pem format. Because the **/sys** directory is not directly accessible via SFTP, transfer the file to the **/user** directory and use the `move` command to move the file to the **/sys** directory (for example. `move /user/rootCA_cert.cer /sys/rootCA_cert.cer`). It is recommended to use the `delete` command to delete any existing file with that name in the **/sys** directory (for example, `delete /sys/rootCA_cert.cer`).
- Load the server certificate into a filed named **srv_cert.cer** in the **/sys** directory of the device using SFTP. The file must be in standard pem format. Because the **/sys** directory is not directly accessible via SFTP, transfer the file to the **/user** directory and use the move command to move the file to the **/sys** directory (for example, `move/user/srv_cert.cer /sys/srv_cert.cer`). It is a recommended to use the `delete` command to delete any existing file with that name in the **/sys** directory (for example, `delete /sys/srv_cert.cer`).
- Load the private key for the server certificate into a filed named **srv_key.pem** in the /sys directory of the device using SFTP. The file must be in standard pem format. Because the **/sys** directory is not directly accessible via SFTP, transfer the file to the **/user** directory and use the `move` command to move the file to the **/sys** directory (for example, `move /user/srv_key.pem /sys/srv_key.pem`). It is recommended to use the `delete` command to delete any existing file with that name in the **/sys** directory (for example, `delete /sys/srv_key.pem`). As previously noted, if the device generated a Certificate Signing Request (CSR) for this certificate, no private key is needed because it is already on the device.

The `ssl CA` command can then be issued and the device can be rebooted.

Generate a Certificate Signing Request (CSR)

The device has the capability to generate a CSR for the default server certificate. This CSR is limited to using a 2048-bit RSA key pair and a SHA-256 hash for its signature. If any of the other algorithms supported by the device are required, do not generate the CSR with the device. Instead, generate the CSR externally and load the private key with the certificate.

To generate a CSR, issue the following command:

```
createcsr c:st:l:o:ou:cn:e [-i:<option>] [-s:<altname>[,<altname>],...]
```

- `c` - Two-letter country code
- `st` - State or province name
- `l` - Locality or city name
- `o` - Organization or company name

- `ou` - Organizational unit name or division
- `cn` - Site name or domain name
- `e` - Email address
- `-i` - Ignores blank parameters. <option> is `true` or `false`.
- `-s` - Subject Alternative Name parameter(s); <`altname`> is a type:value; valid types are `DNS`, `IP`, `email`, `URI`

> **NOTE:** Values that contain spaces must be enclosed in quotation marks.

Example: `createcsr us:nj:rockleigh:"Crestron Electronics"::device.crestron.com: -i:true -s:dns:altname.crestron.com,ip:192.168.0.1`

Be aware that generating a CSR will overwrite any previous CSR and private key, rendering that previous CSR useless. It will not affect any certificate and private key in use that may have been loaded.

Only the `ou` and `e` fields may be left blank and not included in the CSR by specifying the `-i:true` option. Other fields are not affected by the `-i` option and will always be included in the CSR. If the `-i:true` option is not specified, the `ou` and `e` fields will also always be included in the CSR, even if left blank. Fields that are left blank, but still in the CSR, will be set to default values. Because these default values are not likely to be accurate for most environments, it is recommended to always fill in all fields except `ou` and `e`, use the `-i:true` option, and fill in `ou` and `e` if needed.

Once generated, the CSR can be retrieved using SFTP. The CSR will be stored in a file named **request.csr** in the **/sys** directory of the device. Because the **/sys** directory is not directly accessible via SFTP, move the file to the **/user** directory and transfer the file from there (for example, `move /sys/request.csr /user/request.csr`). It is recommended to use the `delete` command to delete any existing file with that name in the **/user** directory (for example, `delete /user/request.csr`).

Backup and Restore Functionality

Crestron products provide users with the ability to back up and restore configuration information for a product.

- This functionality is currently supported only using the console interface.
- A configuration backed up on a certain device type can only be restored on the same device type.
- The configuration file generated is password protected. The following password rules apply:
  - The password must be less than 128 characters.
  - The string must be double quoted if it contains spaces.
  - Printable characters including spaces (with the exception of the single quote character) are permitted.
- If the directory option is given for `exportall`, then the same directory option must be given for `importall`.

> **NOTE:** A backup can be created only when there are no user programs running on the control system. To stop all programs, issue the `stopprog -p:all` command.

To use the backup and restore function, issue the following command:

`configutils [exportall|importall] [-p[:password]] [-d:directory]`

- `exportall` - Exports all data settings to the device firmware folder.
- `importall` - Imports all data settings from the device firmware folder.
- `-f` - Add to not prompt for an import.
- `-p[:password]` - Password to encrypt data. If a password is not provided, the console will prompt to enter it.
- `-d[:directory]` - An alternate directory to store the backup.

Example: `configutils importall -p:password`

> **CAUTION:** Issuing `configutils importall` will restart the device.

Additional Instructions

The instructions in this section are not specific to this device. However, they may be useful to an administrator when setting up and configuring the device.

Use OpenSSL to Create a Certificate Signing Request (CSR)

In most cases, a CSR must be provided to a certificate signing authority to receive a signed certificate. When requesting a signed certificate for this device, you may not want to or be able to generate the CSR on the device itself. In these cases, OpenSSL may be used to create the CSR.

This process can be accomplished by following these instructions on any Windows® or Linux® OS-based computer with OpenSSL version 1.0.2 or newer installed. As a security best practice, ensure that the version of OpenSSL installed is FIPS 140-2 certified.

> **NOTE:** In the following instructions, the example file names include a generic *name* descriptor. It is recommended to replace *name* with a string that identifies the device that will receive the requested certificate so you can more easily match the certificate files with the appropriate device.

Create a Configuration File

First, a configuration file that will be used to generate the CSR must be created. This file will contain information about the CSR and any information that should be included in the CSR.

Create a text file called *name*-csr-openssl.cnf with the following contents:

```
# OpenSSL configuration file for CSR generation

# CSR configuration - Change sha256 to alternate hash function if desired
[ req ]
default_md          = sha256
distinguished_name  = req_distinguished_name
string_mask         = utf8only
utf8                = yes
prompt              = no
req_extensions      = req_ext

# Extensions to be included - Currently SAN only
[req_ext]
subjectAltName = @alt_names

# Information to put in certificate Subject field - fill in desired values
# Comment out any items not desired (only commonName is required)
[ req_distinguished_name ]
commonName                      = Device.Fully.Qualified.Domain.Name
countryName                     = optional
stateOrProvinceName             = optional
localityName                    = optional
0.organizationName              = optional
organizationalUnitName          = optional
emailAddress                    = optional

# List of information to put in SAN extension - fill in desired values
# Additional names or IP addresses can be added if necessary
[ alt_names ]
DNS.1 = Device.Fully.Qualified.Domain.Name
```

Modify the text file to include the information specific to the device and the network site. This information will be put into the Subject field of the certificate and is specified in the `[ req_distinguished_name ]` section of the text file. The `commonName` entry must be filled in and should be the FQDN of the device.

All other fields are optional and should be filled in or commented out (if not commented out, the certificate will contain "optional" as the value of that field). Note that the `countryName` field is only allowed to be 2 characters.

The following example shows a sample of this section containing filled and empty fields:

```
[ req_distinguished_name ]
commonName                      = deviceName.crestron.com
countryName                     = US
stateOrProvinceName             = NJ
localityName                    = Rockleigh
0.organizationName              = Crestron Electronics
#organizationalUnitName          = optional
#emailAddress                    = optional
```

This CSR will also request the standard Subject Alternate Name (SAN) extension to be included in the certificate. The information to include in this extension is specified in the `[ alt_names ]` section of the text file. At least one entry is required, and that entry should match the FQDN specified in the `commonName` field above.

Add additional names that may be used when connecting to the device. Each additional name must use an incremented number in the suffix for the "DNS" identifier. IP addresses are also supported if needed.

The following example shows a sample of this section filled out for a device with three names and two IP addresses:

```
[ alt_names ]
DNS.1 = deviceName.crestron.com
DNS.2 = alternateName.crestron.com
DNS.3 = thirdname.crestron.com
IP.1 = 192.168.0.10
IP.2 = 10.0.0.5
```

Finally, if your certificate signing authority requires the CSR to be signed with a stronger hash than SHA256, the `default_md` field in the `[ req ]` section can be changed. Change `sha256` to `sha384` or `sha512` as needed.

Generate the Private Key

Generate a 2048 bit RSA key by issuing the following command:

```
openssl genrsa -out name.key.pem 2048
```

If desired, replace the `2048` parameter with `3092` or `4096` to generate a longer key of that length.

Create the CSR

Create the CSR using the key and information in the configuration file:

```
openssl req -config name-csr-openssl.cnf -key name.key.pem -new -out name.csr.pem
```

If you wish to view the CSR in text form to confirm it contained the expected information, use the following command:

```
openssl req -noout -text -in name.csr.pem
```

Create and Sign the Certificate

The certificate must be created and signed by the trusted signing authority for the network the device will be used on. Provide the CSR file (name.csr.pem) to your signing authority to create and sign the certificate. The signing authority should return the signed certificate along with the signing chain for that certificate.

Load the Certificate

To load the certificate as the Default Server Certificate, use the *name*.key.pem file that was created, along with the server certificate and signing chain from the signing authority, and follow the instructions provided in the Required Configuration on page 6 and Required Configuration on page 6 topics.

To load the certificate as the Web Server certificate, the certificate and key must be placed into a PKCS #12 file. Ensure that the certificate provided by the signing authority is in PEM format, and then issue the following command, where *name*.cert.pem is the file from the signing authority with the certificate in PEM format.:

```
openssl pkcs12 -export -out name.certandkey.pfx -inkey name.key.pem -in
name.cert.pem
```

OpenSSL will ask for an "Export Password". Enter a password which will be used to protect the PKCS #12 file. It will then ask you to confirm that password.

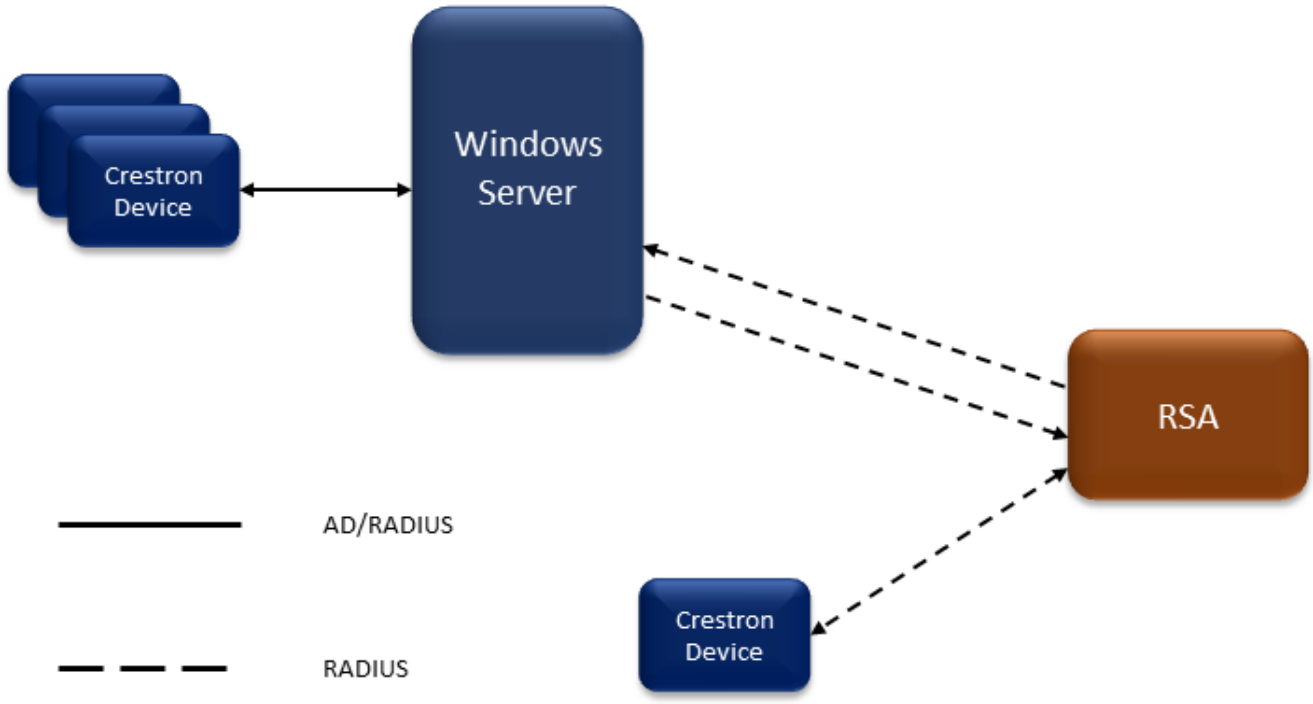
Next, follow the instructions in Required Configuration on page 6 for loading a Web Server certificate. Make sure to provide the Export Password that was entered above when loading the certificate file into the device.

Clean Up

Once successfully loaded onto the device, wipe the local copy of the private key (in the file *name*.key.pem) on the computer used to generate the CSR, as this contains the secret information specific to that certificate for that device.

Configure MFA Support with RSA Authentication Manager

An RSA authentication manager can be used with a Microsoft® Windows® server to provide Multi-Factor Authentication (MFA) capabilities for Crestron devices. The following network diagram provides an overview of this configuration.



The following sections provide information on how to configure RSA SecurID and Authentication Manager as a back-end authentication service. Instructions are provided to configure MFA support with and without a Windows integration. These instructions are not meant to be a complete reference but instead provide an introductory set of steps that show how to integrate a Crestron device with the RSA Authentication Manager.

> **NOTE:** More complex configurations are possible. In these scenarios, the RSA product documentation should be consulted.

In an ideal scenario, a Crestron device should support the following:

- Active Directory® service login using any server that uses Kerberos and LDAP.
- RADIUS-based logins, with or without MFA, using any server that can both understand standard RADIUS messages sent to it and is capable of sending back a Crestron vendor-specific attribute to indicate a user's access level.

Configure Crestron Vendor-Specific Settings for Authentication Manager

Vendor-specific RADIUS attributes (VSA) are used to send authorization information back to a Crestron device as part of a successful login. These VSAs must be configured on the Authentication Manager (AM) if the AM is configured to send authorization information (such as a Crestron access level or domain group name). It is possible to send authorization information from another component, such as Microsoft Network Policy Server (NPS).

The following example starts with RSA as a standalone component. More detailed information on Crestron vendor code is described in Authentication and Authorization on page 65.

To configure Crestron vendor-specific settings for the AM:

1. From the Operations Console (OC), navigate to **Deployment Configuration** > **RADIUS Servers**.
2. Select the server and then select **Manage Server Files**.
3. Obtain a copy of the Crestron RADIUS dictionary and rename the file "dictionary.crestron".
4. Select the **Dictionary Files** tab and then select **Add New**.
5. Select the Crestron dictionary file and then select **Submit**.
6. In the **Configuration Files** tab, edit the **vendor.ini** file. Add the following lines (keeping alphabetical order):

   ```
   vendor-product        = Crestron
   dictionary            = Crestron
   ```

7. Select **Save**.
8. Open the Crestron dictionary file and add the following line (keeping alphabetical order):

   ```
   $INCLUDE dictionary.crestron
   ```

9. Select **Save** and then restart the RADIUS server.

If the RADIUS server does not start (assuming that it was running before changes were made), ensure that the information entered in the procedure above has been entered correctly.

Add a RADIUS Client to an RSA AM

The RSA AM must know about every RADIUS client with which it communicates. To add a RADIUS client for a Crestron device:

1. In the Security Console (SC), navigate to **RADIUS Clients** > **Add New**.
2. Enter a descriptive name for the client (for example, "DM-NVX-363-001").
3. Enter the IP address of the Crestron device.
4. Set the **Make/Model** to "Crestron". This is required because the Crestron device and the AM are exchanging Crestron-specific RADIUS attributes (for authorization).
5. Configure a shared secret that will also be used on the Crestron device.

6. Select **Save and Create Associated Agent**.

   > **NOTE:** Agent-specific attributes can be set, but for this example, keep the default values as is.

7. Select **Save**.

RSA Authentication Use Cases

The following RSA authentication use cases are supported for Crestron devices.

Authenticate Users Directly to RSA With a Fixed Passcode

RSA AM can support both local users (provisioned on the AM itself) and users synced from a separate identity source, such as a Windows AD installation. Prior to Windows and SecurID integration, it is helpful to create a configuration for local users who use a fixed passcode to log in.

> **NOTE:** A PIN is never used in conjunction with a fixed passcode. When using a fixed passcode to authenticate, no other authentication method is required.

To authenticate users directly to RSA with a fixed passcode:

1. Use the `addauthdomain` command on the Crestron device to add a RADIUS authentication domain for RSA. Ensure that the shared secret matches the one configured in Add a RADIUS Client to an RSA AM on page 52.
2. Add user and authentication/authorization data on the AM. In this example, a new user attribute ("Identity Attribute" in RSA terms) will be added to the RSA AM internal database. This attribute is used to set the user's access level on a Crestron device. The attribute will be mapped to a RADIUS attribute that the AM sends back when user authentication is successful.
   a. Add a new identity attribute for **AccessLevel**. In the Security Console (SC), navigate to **Identity** > **Identity Attribute Definitions**, and then select **Add New**.
   b. Leave the category name as "Attributes". This new attribute can be mandatory or optional. Mandatory attributes must be set for any users added after the attribute has been defined.
   c. Set the data type format to "String".
   d. Add the values "administrator", "programmer", "operator", "user", and "connect" to the predefined list entries.
   e. The remaining options can retain their default values.
3. Add the RADIUS attribute:
   a. Navigate to **RADIUS** > **RADIUS User Attribute Definitions** > **Add New**.
   b. For the access level attribute, set the attribute number to 64.
   c. Set the attribute name to "Crestron-Access-Level". This name must match the attribute with the same number in the Crestron RADIUS dictionary.
   d. Select **Yes** for the **Map to an Identity Attribute** option, and select the new **AccessLevel** attribute from the drop-down menu.
   e. Select **Save**.

4. Add a local user:
    a. Navigate to **Identity** > **Users** > **Add New**.
    b. Set all required and optional fields in the **User Basics** section. The user ID is the one that will be used to log in to a Crestron device.
    c. Set a user password. This password is what the user provides when logging in to the RSA Self-Service Console (SSC).

        > **NOTE:** This password is not the same as the password used to authenticate the user when logging into a Crestron device.

    d. Set any other optional values for the user. These fields can retain their default values.
    e. Select the new **AccessLevel** attribute from the drop-down menu near the bottom of the field list. This is the access level that the user will have after logging in to a Crestron device.
    f. Select **Save**.
5. Configure authentication options for the user that was created in step 4. Authentication options define how a user can authenticate themselves when logging in to a Crestron device.
    a. Select the user added in step 4 and then select **Authentication Settings**. The user is assigned a fixed passcode. This is called a passcode and not a password to distinguish it from the password used for the SSC.
    b. Fill the **Allow authentication with a fixed passcode** check box and set the passcode. By default, the user will be required to change this passcode during their initial login.
6. Add the new RADIUS user attribute to this user so that RADIUS responses will include the user's access level.
    a. At the bottom of the same **Authentication Settings** page, select **64 - Crestron-Access-Level** from the drop-down menu of RADIUS user attributes.
    b. Select **Add**.
    c. Select **Save**.

The user is now able to log in to the Crestron device using the new username and authentication domain associated with the RSA AM.

Fixed Passcode Changes

Depending on the configured policies, the AM may require a user to change a fixed passcode at first login, at periodic intervals, or both. The messages displayed during the associated RADIUS challenge/response process can be confusing since, by default, a three word "PIN" is used instead of "passcode".

If challenged to change a PIN when logging in with a fixed passcode, the AM is actually requesting a change to the passcode. Upon completion of the passcode change, the AM will ask the user to wait for the token to change, and then to enter the new passcode (not tokencode). Here, the user must again enter the new passcode; there is no token code involved.

Once this process is complete, the user will have changed their fixed passcode successfully.

Using a RADIUS Profile

It is also possible to configure a RADIUS profile on the AM that uses RADIUS attributes. The profile can be assigned to a set of users. This way, if you need to make a change to a RADIUS attribute, you only need to change it in one place instead of for each user.

To use a RADIUS profile:

1. Navigate to **RADIUS** > **Profiles** > **Add New** to create the new RADIUS profile.
2. Name the profile "Crestron-Administrator".
3. In the **Return List Attributes**, select "Crestron-Access-Level" from the drop-down menu.
4. Enter "administrator" for the access level value. Unlike a RADIUS User Attribute, a RADIUS attribute added to a profile cannot be mapped to an Identity Attribute. The value of the attribute must be provided on this page.
5. Select **Add**, and then select **Save**.

Assuming the RADIUS User Attribute for the access level was added in Authenticate Users Directly to RSA With a Fixed Passcode on page 53, it should be removed at this point to avoid having two of the same attributes in the RADIUS reply (though the Crestron device should use only the first one).

To remove the RADIUS User Attribute for the access level:

1. Display the list of users by navigating to **Identity** > **Users** > **Manage Existing**.
2. Select the user to change and then select **Authentication Settings**.
3. In the **RADIUS** section, select the new RADIUS profile.
4. Select the existing RADIUS user attribute for **Crestron-Access-Level** and then select **Remove**.
5. Select **Save**.

The user should still be able to log in to a Crestron device with the same level of access, but the access is provided via the RADIUS profile and not a RADIUS user attribute.

A RADIUS profile can also be associated with a RADIUS client agent in a similar way, so that any client authenticating via that agent will have an access level assigned to them by the profile.

Authenticate Users Directly to RSA With a SecurID Token

The previous use case dealt with authenticating a user via a fixed passcode. Fixed passcodes are considered a weak form of authentication because they require only a single factor. A SecurID token (soft token or hardware token, SecurID 700) can be used to add a second factor to the user's authentication.

A SecurID token is a time-based numeric value provided either by a hardware device or an application (such as one on the user's smart device). Tokens provided via the SecurID application require a PIN to compute the correct value and are therefore a form of two-factor authentication. Tokens from a hardware device require the PIN to be entered as part of the user's token code.

Use SecurID Soft Tokens

A set of software tokens should have been configured on the AM as part of the installation process. In this step, one of those tokens will be assigned to a user.

Ensure the SecurID app is installed on the user's smart device. This process will vary according to the device type.

1. In the Security Console (SC), navigate to **Identity** > **Users** > **Manage Existing**.
2. Select the user, and then select **SecurID tokens**.
3. Select **assign tokens**.
4. From the list of tokens, select the one you want to assign to the user, and then select **Assign**.
5. On the next screen, tokens assigned to the user are listed. Select the token that was just assigned, and select **Distribute**.
6. This example will use the distribution method **MacOS_File_SDTID**. Select this from the token profile drop-down menu.
7. Highlight the **DeviceSerialNumber** attribute value and delete it to set the serial number to null/empty.
8. Assign an optional password to encrypt the file.
9. Select **Save & Distribute**, then select **Download Now**. This will download the token zip file to your local machine.
10. Unzip the zipped file and extract the .sdtid file. Assuming the user's SecurID application is on their phone, it will be easier to send the .sdtid file and not the .zip
11. Send the file to the user so they can access it via their smart device.

The next set of instructions occur on the user side for importing a token to the SecurID application and setting a PIN.

1. On the user's smart device, open the .sdtid file attachment in the SecurID application, which will import the token.
2. There is currently no PIN assigned to the token, so the user must log in to the RSA self-service console (for example, "https://<RSA_server_fqdn>/ssc"). If a password has been assigned to the user, it can be used to log in. If a fixed passcode has been assigned, it should also work. If neither has been assigned, the user can set a password after logging in.
3. After login, select **Set PIN** and then set a PIN. This PIN will be used in the SecurID application to obtain a correct token code.
4. After setting a PIN, use the link in the SSC to test the PIN/token code.
5. Log out of the SSC.

> **NOTE:** If the PIN is not set as described above, the user will be challenged to set a new PIN when they log in for the first time. Either method of setting the PIN will work.

After the procedures above have been completed, it is now possible to log in to a Crestron device using this SecurID token.

From the Crestron device, log in to the device with the username that has the authentication domain associated with RSA AM. When prompted for a password, use the SecurID app. Enter the PIN for the desired credential (there may be only one, but it is possible to have multiple tokens imported), then enter the resulting token code for the password. Log in to the Crestron device at this point in time should be successful.

The RSA AM may occasionally challenge a login and ask for the next token code. Most logins are not challenged, but some are. You can wait for the token code to change, or you can press the right-arrow button in the SecurID app to advance the token immediately. Waiting for the token should occur only if the token is going to change shortly. Otherwise, login timeouts may occur.

> **NOTE:** Entering the wrong PIN in the SecurID app will still generate a token, but it will be invalid and will not work for authentication.

Use SecurID 700 Hardware Tokens

A set of hardware tokens should have been configured on the AM as part of the installation process. In this step, one of those tokens will be assigned to a user.

1. In the SC, navigate to **Identity** > **Users** > **Manage Existing**.
2. Select the user, then select **SecurID tokens**.
3. Select **assign tokens**.
4. From the list of tokens, select the SecurID 700 token that you want to assign to the user, and then select **Assign**.

> **NOTE:** It may be possible to set an initial PIN during this process, but it is not necessary. A PIN may also be set by the user for this hardware token after assignment using the self-service console.

It is be possible to log in to a Crestron Device using this SecurID hardware token. From the Crestron device, log in to the device using the username that has the authentication domain associated with RSA AM. When prompted for a password, enter the password using the following format: <PIN><tokencode>. If no PIN has been assigned to the token, then use only the token code from the hardware device. AM will require the PIN to be set during the login process.

Refer to the following example scenario where there is a username of "user1", a PIN of "1234", and a hardware token currently displaying "567890". This user would log in to a Crestron device with a username of "user1" and a password of "1234567890".

RSA Integration with Windows Active Directory

Once device login with RSA AM local users is working, setup can be extended to work with users imported from Windows Active Directory (AD). The first step is to add Windows AD as an identity source in the RSA Operations Console. Adding an identity source to AM can be a complex process. This section assumes that the number of Windows users is relatively small and no Global Catalog resource is needed.

Add the Windows AD Identity Source

To add the Windows AD identity source:

1. Log in to the Operations Console (OC).
2. Select **Identity Sources** > **Add New**.
3. Give the identity source a meaningful name (such as "WindowsAD").
4. Enter the full directory pathname for the user. The Directory URL is usually formatted as follows: "ldap://<FQDN of Windows AD server>". The Directory User ID is the ID of a Windows user that RSA will use to access LDAP information. This can be an existing user, or you can create a new user dedicated to this purpose. For example, the user "rsauser" in the domain "mydomain.com" would be entered as "cn=rsauser,CN=Users,dc=mydomain,dc=com".
5. Enter the **Directory Password**. The Directory Password is the Windows password for the Directory User ID.
6. Once the information above has been set, use the **Test Connection** button to ensure RSA can log in to the Windows server with the supplied credentials.
7. Select **Next** to proceed to the identity source mapping screen.
8. Set the **User Base DN** to the point in the organizational hierarchy that represents the set of users you want to add to RSA AM. For example, to add all users in the domain "mydomain.com", set this value to "dc=mydomain,dc=com".
9. Set the **User Group Base DN** in a similar manner. In this example, it can be set to the same value as the **User Base DN** in order to import all user groups.
10. Ensure **Directory is an Active Directory Global Catalog** is not selected and **Authenticate users to this identity source** is selected.
11. Leave the remaining fields with their default values.
12. Select **Save and Finish**.

> **NOTE:** A user's Windows passwords are imported by RSA, but they are useful only for logging in to the RSA Self-Service Console. The windows password cannot be used for authenticating to a Crestron device.

Link the Identity Source to the RSA AM

Before linking the new identity source, log out of and then back into the SC. Then, complete the following:

1. Once you are logged back in to the SC, navigate to **Setup** > **Identity Sources** > **Link Identity Source to System**.
2. The WindowsAD source should appear in the **Available** box. Select it and click the arrow to move it to the **Linked** box.
3. Select **Save**.

To see the Windows users, navigate to **Identity** > **Users** > **Manage Existing**. Then, select the WindowsAD identity source from the drop-down menu on the left, and select **Search**.

Assign Authentication Methods to a Windows User

Although users from Windows have been added to AM, they cannot actually be authenticated at this point because they do not have any RSA authentication method available. As mentioned previously, Windows credentials cannot be used with RSA other than to log in to the self-service console. For RSA to authenticate a user, they must have one or both of a fixed passcode or SecurID token assigned.

To set a fixed passcode for the user:

1. Navigate to **Identity** > **Users** > **Manage Existing**.
2. Select the user and then select **Authentication Settings**.
3. Fill the **Allow Authentication with a Fixed Passcode** check box and set the passcode. By default, the user will be required to change the passcode when they first log in.
4. Select **Save**.

To assign a SecurID token to the user follow steps described in Authenticate Users Directly to RSA With a SecurID Token on page 55.

Assign an Access Level to Windows Users

As described in Authenticate Users Directly to RSA With a Fixed Passcode on page 53, a new Identity Attribute called "AccessLevel" was added to the AM with the values "Administrator", "Operator", and so forth. This attribute was mapped to the Crestron-Access-Level VSA that is sent back to the device in an Access-Accept RADIUS message. When a user is configured in the RSA AM local database, this attribute is set to assign an access level to the user.

To assign an access level to Windows users:

> **NOTE:** When users are added from an external identity source like Windows AD, this attribute must be set manually for each user. It is also possible to map the attribute to a Windows attribute.

1. Navigate to **Identity** > **Users** >**Manage Existing**.
2. Select the Windows identity source and then select **Search** to display the list of Windows users.
3. Select the desired user and then select **Edit**.
4. Scroll to the bottom of the page and select the appropriate access level from the drop-down menu. This is the access level that the user will have after logging in to a Crestron device.
5. Select **Save**.

After setting authentication methods and access levels, the user should be able to log in to the Crestron device with their Windows login ID and RSA fixed passcode or SecurID token.

Authenticate Users with RSA AM and Authorize with Windows NPS

Previous uses cases have described how to use RSA AM to perform both authentication and authorization of a user. It is also possible for RSA AM to perform user authentication and to have Windows Network Policy Server (NPS) perform authorization. This can be done whether users on AM are local or have been added from a Windows identity source. The approach should work both ways as long as usernames on both Windows and AM are the same.

The benefit of using this approach is that most administrators are used to working with Windows users and groups, and NPS can automatically handle hierarchical groups. This makes it easier to create network policies on NPS that return access level or group information. In particular, groups on the RSA AM do not seem to be very useful for returning authorization data to Crestron devices. If there are many Crestron devices, placing the AM behind NPS means that only one RADIUS client (NPS) must be configured on AM. This particular configuration is described in the following sections.

Add the RSA Authentication Manager to NPS

Windows NPS installation is discussed in Installation and Setup of Basic AD Login using MS Windows Server on page 63. RADIUS requests from Crestron devices are sent first to NPS, which then will forward them to RSA's Authentication Manager (AM). To get NPS/AM communication working, first add AM as a remote RADIUS server group to NPS and create a connection request policy.

1. In the NPS console, right click on **RADIUS Clients and Servers** > **Remote RADIUS Server Groups**, and then select **New**.
2. Give the group a name, enter the details for the RSA AM server, and then select **OK**. The shared secret must be the same on the NPS configuration and when NPS is configured later as a client on RSA AM.
3. Set up a connection request policy in NPS to forward RADIUS requests to RSA. In the NPS console, right click on **Policies** > **Connection Request Policies**, and then select **New**.
4. Give the policy a name (such as "Forward to RSA").
5. Within the **Conditions** tab, select **Add** to add any desired condition. A recommended condition is user match, so that RADIUS access requests from a specific user match the policy.

   > **NOTE:** The rest of this section assumes a user match policy exists. Use the same local user that was used in the previous sections since it has been verified to work. Note that this user does not need to be a Windows user (the name can be anything).

6. Select **Next** once all conditions have been added.
7. On the next window, select **Forward Requests...** and use the server group name configured above.
8. Select **Next**.
9. In the **Configure Settings** window, select **Vendor Specific** and then select **Add...**.
10. From the vendor drop-down menu, select **Custom**, select **Remote-RADIUS-To-Windows-User-Mapping**. and then select **Add...**.
11. In the window that pops up next, select **True** and then select **OK**.
12. Select **Close**, select **Next**, and then select **Finish**.

The custom RADIUS attribute set in this policy will cause NPS to use a Network Policy when a RADIUS access-accept comes back from RSA AM. The Network Policy (described in Add a Network Policy to NPS for User Authorization on page 61) is where NPS will return the user's access level.

> **NOTE:** If there already other connection request policies in NPS, the order in which they appear is important to consider. Connection request policies are listed in precedence order, and NPS uses the first policy that matches the incoming RADIUS access request.

Add a Network Policy to NPS for User Authorization

It is assumed that the connection request policy above matches a user or users who belong to an existing Windows group, either directly or via a group hierarchy. The previous section uses a group named "Crestron_Administrators", which will also be used in the following procedure.

1. Right-click **Policies** > **Network Policies**, and then select **New**.
2. Give the policy a name (such as "Connections from Crestron Administrators") and then select **Next**.
3. Within the **Conditions** window, select **Add**.
4. Select **User Groups**, select **Add**, then add each group that should have administrator access. In this example, only one group is needed ("Crestron_Administrators") since all administrators have been added to it. However, multiple groups could also be added using this process if desired.
5. Once all groups have been added, select **Next**.
6. Ensure **Access Granted** is selected, then select **Next**.
7. On the **Authentication Methods** screen, ensure only **MS-CHAP** and **Unencrypted** authentication are selected (others are possible but are not covered in this procedure).

   > **NOTE:** Windows may warn you about selecting an insecure method. The help text can be referenced for more information.

8. No constraints need to be configured, so select **Next**.
9. On the **Configure Settings** screen, select **Vendor Specific** under **RADIUS Attributes**, then select **Add**.
10. Select **Custom** from the **Vendor** drop-down menu, select **Vendor-Specific** from the attributes list, then select **Add**.
11. Select **Add** again within the **Attribute Information** window.
12. Select **Enter Vendor Code** , and then enter "3212" in the text field. 3212 is Crestron's IANA-assigned vendor ID.
13. Select **Yes, it conforms**, and then select **Configure Attribute...**.
14. Enter "64" in the vendor-assigned attribute number text field. This is the attribute number for Crestron-Access-Level.
15. Select **String** from the attribute format drop-down menu, then enter "administrator" in the **Attribute** value box.
16. Select **OK** three times, and then select **Close**. A Vendor-specific attribute should now be visible in the policy **Attributes** area.
17. Select **Next**, and then **Finish**.

This new policy will cause NPS to append a Crestron-specific access level to the RADIUS access-granted message. This level is assigned to the user on the device following a successful login.

> **NOTE:** More detailed notes on network policies and Crestron vendor code can be found in Authentication and Authorization on page 65.

As with connection request policies, network policies are arranged in priority order. NPS uses the first matching policy, so if you already have existing network policies, it is important to consider the order in which they appear.

Add NPS as a RADIUS Client in RSA AM

The NPS and AM must both have knowledge of a shared secret in order to communicate, so the next step is to configure NPS as a RADIUS client in the AM.

1. In the security console (SC), navigate to **RADIUS Clients** > **Add New**.
2. Give the client a meaningful name (such as "Windows NPS").
3. Enter the IP address of the Windows server.
4. Set the **Make/Model** to **Standard Radius** because NPS and AM are not exchanging Crestron-specific RADIUS attributes. NPS will append Crestron VSAs to RADIUS replies going back to the Crestron device.
5. Configure the same shared secret that was configured for the NPS Remote RADIUS Server Group in the previous section.
6. Select **Save and Create Associated Agent**.
7. Agent-specific attributes can be set, but for this example, leave the default values in place.
8. Select **Save**.

Remove RSA AM Authorization Processing

If Authentication Manager (AM) is configured to send authorization data back to the Crestron device in a Crestron VSA, it could conflict with a VSA added by NPS. The Crestron device will use the first Crestron-Access-Level VSA in a RADIUS Access-Accept message. This may cause a problem if AM and NPS differ on the authorization level, resulting in a user being granted an unexpected access level.

To remove a VSA added to a specific user:

1. Navigate to **Identity** > **Users** > **Manage Existing**.
2. Select the desired user and select **Authentication Settings**.
3. If the user has a Crestron-specific RADIUS profile assigned, change the profile to "None".
4. In the **RADIUS** section, select any Crestron-specific attributes shown in the **Attributes** list and then select **Remove**.
5. Select **Save**.

Add an NPS Authentication Domain to the Crestron Device

The final step is to add an NPS authentication domain to the device. Domain login requests using this configured domain name will be sent from the device to NPS. Use the `addauthdomain` command on the device to add a RADIUS-type authentication domain with the Windows NPS server as the host.

Now it should be possible to log in to the Crestron device with the username and authentication domain. The password will be either the RSA fixed passcode for the user or the user's SecurID token depending on how the user's authentication methods were configured.

Additional Field Notes

The following field notes are provided for more information on configuring MFA support with the RSA Authentication Manager.

Deployment and Network Design Choices for Customers

Two different models for integrating Crestron devices with RSA have been presented:

- Crestron devices send RADIUS requests directly to the AM, which performs both authentication and authorization. In this case, each device must be added to the AM along with an associated agent.
- Crestron devices send RADIUS requests to Windows NPS, which forwards them to RSA AM for authentication. Authorization may be performed by the AM or by NPS. In this case, only NPS needs to be added to the AM as a RADIUS client.

The first method is not a problem unless there are many Crestron devices that must be integrated with RSA. In this scenario, there are two ways that the RADIUS client configuration can be minimized:

- Have the Crestron send RADIUS to NPS instead of RSA AM, which is described in Authenticate Users with RSA AM and Authorize with Windows NPS on page 59.
- Use the <ANY>RADIUS client in the RSA AM as described below.

The <ANY> client in the AM is not specific to a device or IP address. It allows multiple different devices to send RADIUS requests to the AM while configuring only one RADIUS client.

To configure the <ANY> client in the AM:

1. From the Operations Console, navigate to **Deployment Configuration** > **RADIUS Servers**. Select the server and then select **Manage Server Files**.
2. Within the **Configuration Files** tab, edit the **securid.ini** file.
3. Locate the setting for "CheckUserAllowedByClient" and set it to "0".
4. Select **Save and Restart RADIUS Server**.
5. In the Security Console, navigate to **RADIUS** > **Clients** > **Add New**.
6. Give the client a meaningful name (such as "All Crestron Devices").
7. Check the **ANY Client** check box. Note that you can have only one <ANY> client configured.
8. Select **Crestron** from the **Make/Model** drop-down menu. This will be available only if the Crestron dictionary has been added as detailed in Configure Crestron Vendor-Specific Settings for Authentication Manager on page 52.
9. Enter the shared secret that all Crestron devices will use in their authentication domain configuration.
10. Select **Save**.

No further configuration is required. Use the `addauthdomain` command on the Crestron devices to add an authentication domain for the RSA AM as described previously.

Installation and Setup of Basic AD Login using MS Windows Server

Installation and setup of basic AD login using an MS Windows server is a complex topic that is beyond the scope of this documentation. Depending on the version of MS Windows Server, other documentation and training materials should be referenced.

The following training videos cover this process using Windows Server 2019.

- Microsoft Windows Server 2019 - Initial Setup & Configuration
- How to Configure a DNS Reverse Looukp Zone - Windows Server 2019

In order to authenticate and authorize domain users, a Crestron device must have the following Windows AD setup:

- A DNS server capable of resolving names on the local network (namely Kerberos and LDAP services) running on the server itself
- Kerberos and LDAP services
- Users and (possibly) user groups provisioned on the server

Crestron devices may perform a DNS search to locate a Kerberos service on the network. It is also possible to provision the Windows AD server details on a Crestron device to avoid a DNS search. In either case, the Kerberos service is used by the Crestron device to authenticate the user by asking the Windows server to validate the user's name and password. Once authenticated, the user's credentials may be used to perform an LDAP query to obtain the user's group memberships (if a domain group is used to authorize the user).

Once a basic Windows server installation is in place (and ideally is working for AD logins) it can be enhanced to support RADIUS-based authentication and authorization with or without MFA.

Windows Network Policy Server

Windows Network Policy Server is capable of performing many functions, such as sending proxy RADIUS messages between Crestron devices and a RADIUS backend server that performs user authentication and MFA. There are numerous reasons why this can be advantageous:

- A Windows Server is already configured and running for AD logins with all users and groups defined. It would be counterproductive to define all users and groups again on a separate RADIUS server.
- Pointing all of your Crestron devices to one single server for both AD and RADIUS/MFA is a cleaner solution. Windows can be configured to either forward an access request to another server or to handle it locally. Windows is also capable of load-balancing forwarded requests if needed.
- RADIUS is relatively simple to use for authentication, but authorization takes more effort. If you already have users and groups configured on a Windows server (in case you are already using AD logins on Crestron devices), it is easier to use Network Policy Server to handle authorizations for RADIUS logins.

To first step for using Windows Network Policy Server is to enable and configure it in your existing Windows AD setup. The Network Policy Server (NPS) on Windows is acting as a RADIUS proxy.

Windows Server 2019 already includes the NPS component. The referenced NPS installation procedure can be used to install the NPS component.

After NPS is installed, register NPS with a Windows Active directory domain, which will allow it to access domain resources like users and groups.

After the steps above have been completed, a RADIUS front-end setup will be enabled and configured. The back-end must now be configured to do the challenge/response/token handling.

Authentication and Authorization

Once a user has been authenticated using a password-only or password plus OTP, a device will need to determine the authorization/access level for the user.

For Crestron products, there are two sources that are used to determine a user's access level:

- The user list on the Crestron device (obtained by issuing `listusers`).
- The domain group list on the Crestron device (obtained by issuing `listdomaingroups`).

Both AD and RADIUS-based logins use the device's user or domain group list. If an authenticated user is in the user list (added to a device via `addusertogroup`), then the configured access level is assigned to the user. If the user has not been added to the device, the domain group list is consulted for access level. The device must determine all domain groups that include the user.

For Active directory logins, this is done by using LDAP queries to get a list of domain groups. RADIUS logins do not (by design) use LDAP queries from the device and there is no real way of using either Kerberos or RADIUS protocols to return what may be (in some cases) a very long list of groups. The solution Crestron uses is RADIUS's support for vendor-specfic attributes (VSA). A VSA is used to convey an authorization level back to the device.

In this solution, the desired outcome is to have RSA AM authentication (based on NPS policies) and Windows/NPS handle the authorization. This helps system administrators maintain the user and group information on the Windows side like the AD solution and to augment the system with RSA SecurID Authentication Manager for the token based second-factor authentication.

Vendor-Specific Attributes

Two vendor specific attributes have been defined by Crestron to support user authorization for RADIUS-based logins. They are defined in the client side RADIUS attribute dictionary (in this case, a Crestron device) located at `/etc/radiusclient/dictionary.crestron`. Details of this definition are provided in the following code sample.

```
# Crestron vendor ID as assigned in IANA Enterprise Vendor Numbers:
# https://www.iana.org/assignments/enterprise-numbers/enterprise-numbers
VENDOR            Crestron                3212

BEGIN-VENDOR    Crestron

# Crestron-Access-Level can be used by a server to send a Crestron-specific
# access level to a device, typically in an Access-Accept message.
ATTRIBUTE        Crestron-Access-Level    64        string

# Crestron-Access-Group is used to send a single "domain group" name to a
# device, typically when a user authentication is successful.
# This attribute may appear multiple times in an Access-Accept message, if
# the user is a member of multiple user/domain groups.
ATTRIBUTE        Crestron-Access-Group    65        string

END-VENDOR
```

The VSA definitions above imply that the RADIUS server can use either of these attributes (Crestron-Access-Level or Crestron-Access-Group) to communicate the access level back to the Crestron device.

A Crestron-Access-Level attribute is a field where the server specifically states the user's access level. This can be determined by a Network Policy. The attribute value can be one of the following strings (case insensitive):

- Administrator
- Operator
- Programmer
- Users
- Connection

> **NOTE:** Only one of these attributes per RADIUS reply is used. If more than one is present, only the first one will apply.

A Crestron-Access-Group attribute is used to indicate that the authenticated user is a member of a group. The attribute value is a string denoting the name of the group. Upon receiving this attribute, the Crestron device consults its configured `domaingroup` list to determine an access level. Similar to Crestron-Access-Level, multiple Crestron-Access-Group attributes can be returned to the device in a RADIUS reply. In such a case, the resulting access level is the highest access level found in the `domaingroup` list.

The creation of a Windows user group and a network policy that uses Crestron-Access-Level is described in Authenticate Users with RSA AM and Authorize with Windows NPS on page 59. The same approach may be taken for the Crestron-User-Group VSA as well.

Crestron Electronics, Inc.
15 Volvo Drive, Rockleigh, NJ 07647
Tel: 888.CRESTRON
Fax: 201.767.7656
www.crestron.com

Security Reference Guide — Doc. 9063D
07/18/23
Specifications subject to
change without notice.