3948/513x/5144/516x/5170/811x Routers and Platforms

Security

SAOS 10.7.1

323-1955-303 - Standard Revision A
May 2022

LEGAL NOTICES

Security

Ciena® cannot be responsible for unauthorized use of equipment and will not make allowance or credit for unauthorized use or access.

Contacting Ciena

Corporate Headquarters	410-694-5700 or 800-921-1144	www.ciena.com
Customer Technical Support/Warranty		www.ciena.com/support/
Sales and General Information	North America: 1-800-207-3714 International: +44 20 7012 5555	E-mail: sales@ciena.com
In North America	410-694-5700 or 800-207-3714	E-mail: sales@ciena.com
In Europe	+44-207-012-5500 (UK)	E-mail: sales@ciena.com
In Asia	+81-3-3248-4680 (Japan)	E-mail: sales@ciena.com
In India	+91-22-42419600	E-mail: sales@ciena.com
In Latin America	011-5255-1719-0220 (Mexico City)	E-mail: sales@ciena.com
Training		E-mail: learning@ciena.com

For additional office locations and phone numbers, please visit the Ciena web site at www.ciena.com.

READ THIS LICENSE AGREEMENT (“LICENSE”) CAREFULLY BEFORE INSTALLING OR USING CIENA SOFTWARE OR DOCUMENTATION. THIS LICENSE IS AN AGREEMENT BETWEEN YOU AND CIENA COMMUNICATIONS, INC. (OR, AS APPLICABLE, SUCH OTHER CIENA CORPORATION AFFILIATE LICENSOR) (“CIENA”) GOVERNING YOUR RIGHTS TO USE THE SOFTWARE. BY INSTALLING OR USING THE SOFTWARE, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS LICENSE AND AGREE TO BE BOUND BY IT.

1. License Grant. Ciena may provide “Software” to you either (1) embedded within or running on a hardware product (together with Software, “Product”) or (2) as a standalone application, and Software includes upgrades acquired by you from Ciena or a Ciena authorized reseller. The terms of this License apply to your use of the Software and associated documentation whether such has been provided by Ciena, an affiliate of Ciena, or by means of an authorized reseller or distributor. Subject to these terms, and payment of all applicable License fees including any usage-based fees, Ciena grants you, as end user, a non-exclusive, non-transferable, personal License to use the Software only in object code form, subject to any applicable authorized use, activation requirements, usage levels, scope of functionality and release level of the Software, as set forth in the applicable quote accepted by Buyer upon Buyer's issuance of an acceptable purchase order (“Order”), and in accordance with the detailed ordering information in the Ciena's generally available, applicable, Product documentation as of the date of such Order. Unless the context does not permit, Software also includes associated documentation. Where an Order is for a (a) perpetual license, you may use the Software and associated documentation for as long as you use the Product for internal business use, or a (b) subscription license, you may only use the Software and associated documentation during the subscription term. A subscription license includes Software upgrades and/or technical support Services during the subscription term (that are not included in a perpetual license), in accordance with the Order and as further described in the applicable Ciena's service description as of the date of the applicable Order. Prior to the expiration of each subscription term, Ciena will send you a quote for the annual renewal fee(s). To renew the subscription Software license(s) for additional subscription terms, you issue an Order in advance of the then-current expiration date of such subscription term.

2. Open Source and Third-Party Licenses. If any Software is subject to an open-source license that provides the end user with rights that are broader than this License, then such rights shall take precedence. Ciena warrants that using Software in accordance with its documentation will not subject you to any obligation to disclose, distribute or license your own software that interacts with Software.

3. Title. You are granted no title or ownership rights in or to the Software. Unless specifically authorized by Ciena in writing, you are not authorized to create any derivative works based upon the Software. Title to the Software, including any copies or derivative works based thereon, and to all copyrights, patents, trade secrets and other intellectual property rights in or to the Software, are and shall remain the property of Ciena and/or its licensors. Ciena's licensors are third party beneficiaries of this License. Ciena reserves to itself and its licensors all rights in the Software not expressly granted to you.

4. Confidentiality. The Software contains trade secrets of Ciena. Such trade secrets include, without limitation, the design, structure and logic of individual Software programs, their interactions with other portions of the Software, internal and external interfaces, and the programming techniques employed. The Software and related technical and commercial information, and other information received in connection with the purchase and use of the Software that a reasonable person would recognize as being confidential, are all confidential information of Ciena (“Confidential Information”).

5. Obligations. You shall:

i) Hold the Software and Confidential Information in strict confidence for the benefit of Ciena using your best efforts to protect the Software and Confidential Information from unauthorized disclosure or use, and treat the Software and Confidential Information with the same degree of care as you do your own similar information, but no less than reasonable care;
ii) Keep a current record of the location of each copy of the Software you make;
iii) Use the Software only in accordance with the authorized usage level;
iv) Preserve intact any copyright, trademark, logo, legend or other notice of ownership on any original or copies of the Software, and affix to each copy of the Software you make, in the same form and location, a reproduction of the copyright notices, trademarks, and all other proprietary legends and/or logos appearing on the original copy of the Software delivered to you; and
v) Issue instructions to your authorized personnel to whom Software is disclosed, advising them of the confidential nature of the Software and provide them with a summary of the requirements of this License.

6. Restrictions. You shall not:

i) Use the Software or Confidential Information a) for any purpose other than your own internal business purposes; and b) other than as expressly permitted by this License;
ii) Allow anyone other than your authorized personnel who need to use the Software in connection with your rights or obligations under this License to have access to the Software;
iii) Make any copies of the Software except such limited number of copies, in machine readable form only, as may be reasonably necessary for execution in accordance with the authorized usage level or for archival purposes only;
iv) Make any modifications, enhancements, adaptations, derivative works, or translations to or of the Software;
v) Reverse engineer, disassemble, reverse translate, decompile, or in any other manner decode the Software;
vi) Make full or partial copies of the associated documentation or other printed or machine-readable matter provided with the Software unless it was supplied by Ciena in a form intended for reproduction;
vii) Export or re-export the Software and/or the associated documentation from the country in which it was received from Ciena or its authorized reseller unless authorized by Ciena in writing; or
viii) Publish the results of any benchmark tests run on the Software.

7. Audit: Upon Ciena's reasonable request you shall permit Ciena to audit the use of the Software to ensure compliance with this License.

8. U.S. Government Use. The Software is provided to the Government only with restricted rights and limited rights. Use, duplication, or disclosure by the Government is subject to restrictions set forth in FAR Sections 52-227-14 and 52-227-19 or DFARS Section 52.227-7013(C)(1)(ii), as applicable. The Software and any accompanying technical data (collectively “Materials”) are commercial within the meaning of applicable Federal acquisition regulations. The Materials were developed fully at private expense. U.S. Government use of the Materials is restricted by this License, and all other U.S. Government use is prohibited. In accordance with FAR 12.212 and DFAR Supplement 227.7202, the Software is commercial computer software and the use of the Software is further restricted by this License.

9. Term of License. This License is effective until the applicable subscription term expires or the License is terminated. You may terminate this License by giving written notice to Ciena. This License will terminate immediately if (i) you breach any term or condition of this License or (ii) you become insolvent, cease to carry on business in the ordinary course, have a receiver appointed, enter into liquidation or bankruptcy, or any analogous process in your home country. Termination shall be without prejudice to any other rights or remedies Ciena may have. Upon any termination of this License, you shall destroy and erase all copies of the Software in your possession or control, and forward written certification to Ciena that all such copies of Software have been destroyed or erased. Your obligations to hold the Confidential Information in confidence, as provided in this License, shall survive the termination of this License.

10. Compliance with laws. You agree to comply with all laws related to your installation and use of the Software. Software is subject to U.S. export control laws and may be subject to export or import regulations in other countries. If Ciena authorizes you to import or export the Software in writing, you shall obtain all necessary licenses or permits and comply with all applicable laws.

11. Limitation of Liability. ANY LIABILITY OF CIENA SHALL BE LIMITED IN THE AGGREGATE TO THE AMOUNTS PAID BY YOU TO CIENA OR ITS AUTHORIZED RESELLER FOR THE SOFTWARE. THIS LIMITATION APPLIES TO ALL CAUSES OF ACTION, INCLUDING WITHOUT LIMITATION BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS. THE LIMITATIONS OF LIABILITY DESCRIBED IN THIS SECTION ALSO APPLY TO ANY LICENSOR OF CIENA. NEITHER CIENA NOR ANY OF ITS LICENSORS SHALL BE LIABLE FOR ANY INJURY, LOSS OR DAMAGE, WHETHER INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL INCLUDING WITHOUT LIMITATION ANY LOST PROFITS, CONTRACTS, DATA OR PROGRAMS, AND THE COST OF RECOVERING SUCH DATA OR PROGRAMS, EVEN IF INFORMED OF THE POSSIBILITY OF SUCH DAMAGES IN ADVANCE.

12. General. Ciena may assign this License to an affiliate or to a purchaser of the intellectual property rights in the Software. You shall not assign or transfer this License or any rights hereunder, and any attempt to do so will be void. This License shall be governed by the laws of the State of New York without regard to conflict of laws provisions. The U.N. Convention on Contracts for the International Sale of Goods shall not apply hereto. This License constitutes the complete and exclusive agreement between the parties relating to the license for the Software and supersedes all proposals, communications, purchase orders, and prior agreements, verbal or written, between the parties. If any portion hereof is found to be void or unenforceable, the remaining provisions shall remain in full force and effect.

Contents

CHAPTER 1
Overview 1

CHAPTER 2
User access security 3
TACACS+ 4
RADIUS 5
RADIUS Vendor-Specific Attributes (VSA) 6
RADIUS over TLS 7
Authentication configuration 7
List of procedures 7
Creating a server group using the CLI 10
Removing a server group using the CLI 11
Configuring the authentication method using the CLI 12
Configuring an accounting method using the CLI 13
Enabling TACACS or RADIUS authorization using the CLI 14
Adding a server to a server group using the CLI 15
Configuring a server using the CLI 16
Enabling a server using the CLI 18
Disabling a server using the CLI 19
Removing a server from a server group using the CLI 20
Configuring timeout from a RADSEC server using the CLI 21
Attaching a TLS service profile to a RADSEC server using the CLI 22
Configuring the search method using the CLI 23
Removing a reference to a server group from the authentication-method list using the CLI 24
Clearing AAA authentication provider statistics using the CLI 25
Clearing server statistics using the CLI 26
Configuring login accounting event support using CLI 27
Configuring command accounting event support using CLI 28
Removing the login accounting event type using CLI 29
Removing the command accounting event type using CLI 30
Displaying the server configuration using the CLI 31
Displaying server statistics using the CLI 32
Displaying the authentication configuration using the CLI 35
Displaying users using the CLI 36
Displaying accounting information using the CLI 37

Displaying your user account using CLI 38
Displaying user accounts using the CLI 39
Creating a server group using the YANG model 40
Removing a server group using the YANG model 43
Configuring the authentication method using the YANG model 44
Inserting a method as the first method in the authentication list using the YANG model 46
Inserting a method as the last method in the authentication list using the YANG model 48
Configuring an accounting method using the YANG model 50
Enabling TACACS or RADIUS authorization using the YANG model 52
Adding a server to a server group using the YANG model 53
Configuring a server using the YANG model 55
Enabling a server using the YANG model 57
Disabling a server using the YANG model 59
Removing a server from a server group using the YANG model 61
Configuring timeout from a RADSEC server using the YANG model 63
Attaching a TLS service profile to a RADSEC server using the YANG model 65
Configuring the search method using the YANG model 67
Removing a reference to a server group from the authentication-method list using the YANG model 69
Configuring command accounting event support using the YANG model 71

CHAPTER 3
Secure Shell public key authentication 73
OpenSSH compatibility 73
List of procedures 73
Installing an SSH user public key using the CLI 75
Deleting an SSH user public key using the CLI 77
Enabling SSH public key authentication using the CLI 78
Disabling SSH public key authentication using the CLI 79
Configuring encryption algorithms using the CLI 80
Configuring key exchange algorithms using the CLI 81
Configuring the rekey limit using the CLI 82
Configuring the rekey time using the CLI 83
Configuring a MAC algorithm using the CLI 84
Configuring a PKA algorithm using the CLI 85
Displaying SSH server information using the CLI 86
Installing an SSH user public key using the YANG model 87
Deleting an SSH user public key using the YANG model 89
Enabling SSH public key authentication using the YANG model 90
Disabling SSH public key authentication using the YANG model 91
Configuring encryption algorithms using the YANG model 92
Configuring key exchange algorithms using the YANG model 94
Configuring the rekey limit using the YANG model 96
Configuring the rekey time using the YANG model 98
Configuring a MAC algorithm using the YANG model 99
Configuring a PKA algorithm using the YANG model 101
Initiating an SSH connection using a host key algorithm 103
Initiating an SSH connection using a cipher algorithm 104

Initiating an SSH connection using a key exchange algorithm 105
Retrieving SSH server information using the YANG model 106

CHAPTER 4
X.509 certificates 107
Public key infrastructure 107
X.509 108
Example use cases for PKI and X.509 109
Certificate authority or certificate revocation lists installation 109
System certificate and private key installation 110
NTE authenticates another system 111
NTE authenticated by server 111
Check IP host 111
Online Certificate Status Protocol 112
OCSP server certificates 112
OCSP responder 113
OCSP configuration 113
Certificate fingerprint 113
List of procedures 114
Installing a CA certificate using the CLI 117
Installing a CA certificate locally using the CLI 118
Uninstalling a CA certificate using the CLI 119
Installing a CRL using the CLI 120
Installing a CRL from a local path using the CLI 121
Uninstalling a CRL using the CLI 122
Installing a device certificate and private key using the CLI 123
Installing a device certificate from a local path using the CLI 124
Uninstalling a device certificate and private key using the CLI 125
Generating a private key and certificate signing request using the CLI 126
Enabling check-fingerprint using the CLI 127
Disabling check-fingerprint using the CLI 128
Configuring a fingerprint list using the CLI 129
Deleting a fingerprint-list using the CLI 130
Modifying the OCSP default responder URL using the CLI 131
Removing the OCSP default responder URL using the CLI 132
Modifying the OCSP state using the CLI 133
Modifying the nonce state using the CLI 134
Displaying a CA certificate using the CLI 135
Displaying a CRL using the CLI 136
Displaying a device certificate and private key using the CLI 137
Displaying all installed certificates on the system using the CLI 138
Displaying check-fingerprint and fingerprint list using the CLI 139
Adding ip-host-list entries using the CLI 140
Deleting ip-host-list entries using the CLI 142
Displaying ip-host-list entries using the CLI 143
Enabling check IP host using the CLI 144
Installing a CA certificate using the YANG model 145
Installing a CA certificate locally using the YANG model 146
Uninstalling a CA certificate using the YANG model 147
Installing a CRL using the YANG model 148

Uninstalling a CRL using the YANG model 149
Installing a device certificate and private key using the YANG model 150
Installing a device certificate locally using the YANG model 151
Uninstalling a device certificate and private key using the YANG model 152
Generating a private key and certificate signing request using the YANG model 153
Enabling check-fingerprint using the YANG model 154
Disabling check fingerprint using the YANG model 155
Configuring a fingerprint-list using the YANG model 156
Deleting a fingerprint-list using the YANG model 158
Modifying the OCSP default responder URL using the YANG model 159
Removing the OCSP default responder URL using the YANG model 160
Modifying the OCSP state using the YANG model 161
Modifying the nonce state using the YANG model 162
Retrieving a CA certificate using the YANG model 163
Retrieving a CRL using the YANG model 164
Retrieving a device certificate and private key using the YANG model 165
Retrieving check-fingerprint and the finger-print list using the YANG model 166
Troubleshooting PKIX errors using the CLI 167
Troubleshooting TLS errors using the CLI 169
Troubleshooting TLS handshake errors using the CLI 171

CHAPTER 5
802.1x **173**
802.1x roles 174
Supplicant 174
Authenticator 175
Authentication server 175
Deployment example 176
802.1x on aggregation ports 177
Authentication verification 177
EAP-TLS 177
EAP-TLS supplicant authentication 178
EAP-TLS mutual authentication 179
List of procedures 179
Enabling 802.1x authentication using the CLI 182
Disabling 802.1x authentication using the CLI 183
Re-authenticating the supplicant using the CLI 184
Initializing a port using the CLI 185
Configuring a supplicant port in MD5 mode using the CLI 186
Configuring a supplicant port in TLS mode using the CLI 188
Configuring an authenticator port in MD5 mode using the CLI 191
Creating and attaching a server group for the RADIUS server to use in authentication using the CLI 195
Clearing all 802.1x port statistics using the CLI 197
Clearing specific port statistics using the CLI 198
Displaying 802.1x ports using the CLI 199
Displaying information about all authenticator ports using the CLI 201
Displaying information about a specific authenticator port using the CLI 202
Displaying information about all supplicant ports using the CLI 203

Displaying information about a specific supplicant port using the CLI 204
Displaying 802.1x global information using the CLI 205
Displaying 8021x authenticator statistics using the CLI 206
Displaying 802.1x authenticator statistics for a specific port using the CLI 207
Displaying 802.1x supplicant statistics using the CLI 208
Displaying 802.1x supplicant statistics for a specific port using the CLI 209
Enabling 802.1x using the YANG model 210
Disabling 802.1x using the YANG model 211
Re-authenticating the supplicant using the YANG model 212
Initializing a port using the YANG model 213
Configuring a port as a supplicant or an authenticator using the YANG model 214
Setting parameters on a supplicant port using the YANG model 216
Setting parameters on an authenticator port using the YANG model 219
Attaching a tls-service profile to the supplicant port using the YANG model 222
Creating and attaching a server group for RADIUS configuration on an authenticator DUT using the YANG model 224
Clearing all 802.1x port statistics using the YANG model 226
Clearing specific port statistics using the YANG model 227
Retrieving 802.1x port information using the YANG model 228
Retrieving information about all authenticator ports using the YANG model 229
Retrieving information about all supplicant ports using the YANG model 230
Retrieving 802.1x global information using the YANG model 231
Retrieving 802.1x authenticator statistics information using the YANG model 232
Retrieving 802.1x supplicant statistics using the YANG model 233
Retrieving 802.1x authenticator-session statistics using the YANG model 234

CHAPTER 6
Transport Layer Security 235
Profiles 236
Cipher suite 236
Elliptic curve 240
List of procedures 240
Creating a TLS profile using the CLI 242
Creating a peer authentication profile using the CLI 244
Creating a TLS service profile using the CLI 245
Displaying TLS profile information using the CLI 246
Displaying TLS server session information using the CLI 247
Creating a TLS profile using the YANG model 249
Creating a peer authentication profile using the YANG model 251
Creating a TLS service profile using the YANG model 252
Retrieving the TLS profile information using the YANG model 254

CHAPTER 1
Overview

This document explains how to manage and protect resources from unauthorized or detrimental access and use.

In order to configure security features, you need to install the Security license. Licenses can be purchased by contacting Ciena Customer Support.

The following table lists the chapters in this document.

Table 1 Chapters in Security

Chapter	Description
User access security	Describes how to manage remote servers for authentication, authorization and accounting (AAA) services on the client by means of server groups.
Secure Shell public key authentication	Describes how public and private key authentication works
X.509 certificates	Describes how to use X.509 functionality for authentication and encryption.
802.1x	Describes how the IEEE 802.1x-2010 standard is used to provide a method for authenticating customer premise equipment (CPE) and the systems used to provide the CPE network connection.
Transport Layer Security	Describes how to implement Transport Layer Security (TLS), which is a security protocol that creates secure network connections.

Audience

This document is intended for operations personnel such as Network Administrators, Network Planners and Security Administrators who perform tasks related to administering the system in a packet networking environment.

Trademark and copyright acknowledgments

- Ciena® is a trademark of Ciena Corporation.
- Portions of this document are copyright 2018 YumaWorks, Inc.

CHAPTER 2
User access security

Remote servers for authentication, authorization and accounting (AAA) services are managed on the client by means of server groups. Each server group can contain up to eight servers. Identical servers can be configured in multiple server groups.

Configurations for identical servers that appear in multiple server groups are preserved within each server group. This means that each server entry may have a unique name, timeout, administrative state, port and secret-key in its configuration, even if a server with the same address or hostname exists in another server group.

All servers in a server group must be able to provide the same authentication protocol, that is, TACACS, RADIUS, or RADSEC. For example, every server configured in a server group indicated by type TACACS is sent TACACS+ packets during an authentication session.

> ***Note:*** The add-on security license must be installed for TACACS, RADIUS, or RADSEC authentication.

The network operator can enable and disable the administrative state of each configured server list entry. The server can be removed from the connection queue without having to delete the configuration for that server.

The administrative state for a server entry can be enabled or disabled for each server group association. For example, if a server is configured in server group A and server group B, the administrative state of the server entry in server group A can be enabled or disabled without affecting the administrative state of the same server entry in server group B. The default state of a server entry is enabled.

The network operator can configure multiple authentication providers and specify the sequence that each provider tries. This sequence provides a backup if one provider fails to respond to an authentication request. Remote servers can use different authentication protocols, for example, TACACS+. The default authentication provider is local authentication.

The system supports:

- "TACACS+" on page 4
- "RADIUS" on page 5
- "RADIUS over TLS" on page 7

TACACS+

TACACS+ provides an industry-standard security protocol for controlling authentication, authorization and accounting functions. It also provides security by using a shared key to encrypt information between the Network Access Server (NAS) and the authentication server.

> ***Note:*** Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Authentication grants the user access when they first log in to the system or request a service. Traditional authentication uses a name and a fixed password. More modern authentication mechanisms use one-time passwords or a challenge-response query. TACACS+ supports all these types of authentication.

The TACACS+ protocol client is supported where the system operates as an NAS. There is no limit on the number of server-groups so with N server-groups you could have N* 8 authentication servers. If a TACACS+ server is not configured, a locally-configured password file is used for authentication. Local authentication is used only if the user authentication provider order is configured to allow it.

> ***Note:*** TACACS+ is not compatible with previous TACACS and XTACACS protocols.

TACACS+ provides a mechanism to request additional information from the user. The system works with any text-based multi-factor authentication provided by the TACACS+ server, such as RSA authenticators or the Google Authenticator application.

Multi-factor authentication allows the server to perform an arbitrary number of exchanges with the user which could include challenge/response, requesting a value from an authenticator dongle, requesting a new password, or asking the user what was your mother's maiden name.

A TACACS+ server can reply to access requests with a REPLY that contains a TAC_PLUS_AUTH_STATUS_GETDATA indicating that it needs data other than the username or password. This reply may include a server_msg which can be used as a prompt.

Configure the TACACS server to access the system using the SYSTEM_ROLE_DIAG local user role. For more information about local user roles, refer to Administration for the product release you using.

The following table describes the TACACS server configuration.

Table 2 TACACS server configuration

Server setting	Value
default service	permit
service	shell
priv-lvl	15
nacm-group	super or limited or admin

Note: If the TACACS user needs to have DIAG level permissions, priv-level 15 must be set. For the limited level, meaning non-diag level permissions, any value less than 15 must be set.

RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a client/server system used to secure networks against unauthorized remote access such as with Telnet.

RADIUS servers allow the network operator to configure and control user accounts in one central location instead of having to configure accounts on every system in the network. RADIUS enables access and authentication control to be very flexible in the way it regulates access.

When authenticating a Telnet user, the system sends authentication requests to one or more RADIUS servers. The RADIUS server keeps track of all user authentication and service access information. The RADIUS server returns authentication results to the system and the user is either allowed or denied access based on this information.

Configure the RADIUS server to access the system using the SYSTEM_ROLE_DIAG local user role. For more information about local user roles, refer to Administration for the product release you using.

The following table describes the RADIUS server configuration.

Table 3 RADIUS server configuration

Server setting	Value
ngsu Cleartext-Password	"ngsu"

Server setting	Value
Login-Service	Telnet
Ciena_CES_NACM_Groups	super or limited or admin
Ciena_CES_Priv_Level_New	4
Ciena_CES_Priv_Level	4

Note: If the RADIUS user needs to have DIAG level permissions, priv-level 4 must be set. For the limited level, meaning non-diag level permissions, any value less than 4 must be set.

RADIUS Vendor-Specific Attributes (VSA)

RFC 2865 specifies a method for communicating vendor-specific information between the network access system and the RADIUS server, by using the vendor-specific attribute, attribute 26. Attribute 26 allows vendors to support their own extended attributes otherwise not suitable for general use.

The Ciena RADIUS client supports vendor-specific options for user login which can be used to define the user privilege level assigned to the user being authenticated.

The following table describes the VSA attributes.

Table 4 VSA attributes

VSA attributes	Description
VSA for new installations and devices that interwork with the system.	• attribute type: 26 (VSA) vendor: 1271 (ciena) vsa-sub-type: 10 integer to indicate privilege level: — 1 (limited) 2 (admin) 3 (super-user) 4 (diag)
Ciena also supports a legacy VSA.	• attribute type: 26 (VSA) vendor: 1271 (ciena) vsa-sub-type: 1 and one of the following numbers to indicate user privilege level: — 1 (limited) 2 (admin) 3 (super-user) 4 (diag)

Note: Systems that run older software require the legacy VSA. You can configure servers that provide authentication to systems running older software with the legacy VSA or both VSAs.

RADIUS over TLS

RADIUS over TLS (RADSEC) allows the transmission of RADIUS messages through the transport layer using TLS. The system implements RADSEC for AAA.

RADSEC is modeled as a manager, client, and authentication source completely independent of UDP RADIUS.

RADSEC employs existing TLS/PKIX infrastructure for X.509 certificate and other TLS-specific parameters. You can configure a remote RADSEC server and a corresponding TLS service profile for sending radius authentication and accounting request securely using TLS.

The system initiates a TLS connection with the server. The two endpoints perform configured authentication and establish a TLS connection. RADIUS messages are then transmitted using the TLS connection.

Authentication configuration

To configure authentication by means of a remote server:

- ensure TACACS, RADIUS, or RADSEC authorization is enabled. For configuration using CLI, enabling TACACS, RADIUS, or RADSEC authorization is done on the TACACS, RADIUS, or RADSEC server respectively.
- create a server group and set the type to TACACS, RADIUS, or RADSEC.
- add one or more servers to the server group.
- configure the shared secret for each server in the server group.
- configure the authentication method to refer to the server group.

List of procedures

The following procedures are used to configure user access security:

- Procedure 1, “Creating a server group using the CLI”
- Procedure 2, “Removing a server group using the CLI”
- Procedure 3, “Configuring the authentication method using the CLI”
- Procedure 4, “Configuring an accounting method using the CLI”
- Procedure 5, “Enabling TACACS or RADIUS authorization using the CLI”
- Procedure 6, “Adding a server to a server group using the CLI”
- Procedure 7, “Configuring a server using the CLI”
- Procedure 8, “Enabling a server using the CLI”
- Procedure 9, “Disabling a server using the CLI”

- Procedure 10, “Removing a server from a server group using the CLI”
- Procedure 11, “Configuring timeout from a RADSEC server using the CLI”
- Procedure 12, “Attaching a TLS service profile to a RADSEC server using the CLI”
- Procedure 13, “Configuring the search method using the CLI”
- Procedure 14, “Removing a reference to a server group from the authentication-method list using the CLI”
- Procedure 15, “Clearing AAA authentication provider statistics using the CLI”
- Procedure 16, “Clearing server statistics using the CLI”
- Procedure 17, “Configuring login accounting event support using CLI”
- Procedure 18, “Configuring command accounting event support using CLI”
- Procedure 19, “Removing the login accounting event type using CLI”
- Procedure 20, “Removing the command accounting event type using CLI”
- Procedure 21, “Displaying the server configuration using the CLI”
- Procedure 22, “Displaying server statistics using the CLI”
- Procedure 23, “Displaying the authentication configuration using the CLI”
- Procedure 24, “Displaying users using the CLI”
- Procedure 25, “Displaying accounting information using the CLI”
- Procedure 26, “Displaying your user account using CLI”
- Procedure 27, “Displaying user accounts using the CLI”
- Procedure 28, “Creating a server group using the YANG model”
- Procedure 29, “Removing a server group using the YANG model”
- Procedure 30, “Configuring the authentication method using the YANG model”
- Procedure 31, “Inserting a method as the first method in the authentication list using the YANG model”
- Procedure 32, “Inserting a method as the last method in the authentication list using the YANG model”
- Procedure 33, “Configuring an accounting method using the YANG model”
- Procedure 34, “Enabling TACACS or RADIUS authorization using the YANG model”
- Procedure 35, “Adding a server to a server group using the YANG model”
- Procedure 36, “Configuring a server using the YANG model”

- Procedure 37, “Enabling a server using the YANG model”
- Procedure 38, “Disabling a server using the YANG model”
- Procedure 39, “Removing a server from a server group using the YANG model”
- Procedure 40, “Configuring timeout from a RADSEC server using the YANG model”
- Procedure 41, “Attaching a TLS service profile to a RADSEC server using the YANG model”
- Procedure 42, “Configuring the search method using the YANG model”
- Procedure 43, “Removing a reference to a server group from the authentication-method list using the YANG model”
- Procedure 44, “Configuring command accounting event support using the YANG model”

Procedure 1 Creating a server group using the CLI

Create a server group for authentication purposes.

Overview

The following table describes the parameters for creating a server group.

Table 5 Parameters for creating a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server group.
type	TACACS | RADIUS | RADSEC	Specifies the AAA protocol type for the server group. Setting the type when you create a server group enables defaults to be automatically populated.

Steps

1 Create a server group:

```
system aaa server-groups server-group <name> config name
<name> type <type>
```

Example

The following example creates a TACACS server group named tacacs group.

```
system aaa server-groups server-group tacacsgroup config name tacacsgroup type
TACACS
```

The following example creates a RADIUS server group named radiusgroup.

```
system aaa server-groups server-group radiusgroup config name radiusgroup type
RADIUS
```

The following example creates a RADSEC server group named radsecgroup.

```
system aaa server-groups server-group radsecgroup config name radsecgroup type
RADSEC
```

Procedure 2 Removing a server group using the CLI

Remove a server group configuration and all nested servers from the system as required.

Overview

The following table lists the parameter for removing a server group configuration.

Table 6 Parameter for removing a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server.

Steps

1 Remove a server group:

```
no system aaa server-groups server-group <name>
```

Example

The following example removes a server group named rwgroup.

```
no system aaa server-groups server-group rwgroup
```

Procedure 3 Configuring the authentication method using the CLI

Configure the authentication method used to authenticate users. Authentication can be local or remote. The default is local authentication.

Overview

Server groups are added in the order in which they are added into the authentication method.

The following table lists the parameter for configuring the authentication method.

Table 7 Parameter for configuring the authentication method

Parameter	Valid values	Description
authentication-method	string | AUTH_LOC	Identifies the server group to use for remote authentication. AUTH_LOC specifies local authentication.

CAUTION

Risk of loss of access to the system

Ensure that more than one provider is configured. If only one provider is configured and that provider fails to reply positively to an authentication request or fails to reply to the request the system may enter a lockout state.

Steps

1 Configure the authentication method:

```
system aaa authentication config authentication-method
<authentication-method>
```

Example

The following example configures the authentication method.

```
system aaa authentication config authentication-method rwgroup
```

Procedure 4 Configuring an accounting method using the CLI

Configure an accounting method to send messages to the server group to record user activity, for example, to account for services used, such as in a billing environment, or as an auditing tool for security services.

Overview

The following table describes the parameter for configuring the accounting method.

Table 8 Parameter for configuring an accounting method

Parameter	Valid values	Description
accounting-method	string	Specifies the server group to send accounting messages to.

Note: Session Accounting and Session Authorization by means of NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1 Configure an accounting method to send messages to the server group:

```
system aaa accounting config accounting-method
<accounting-method>
```

Procedure 5 Enabling TACACS or RADIUS authorization using the CLI

Enable TACACS or RADIUS authorization.

Overview

The following table describes the parameter for enabling TACACS or RADIUS authorization.

Table 9 Parameter for enabling TACACS or RADIUS authorization

Parameter	Valid values	Description
enable-external-groups	true | false	Enables or disables TACACS or RADIUS authorization.

Steps

1 Enable TACACS or RADIUS authorization.

```
nacm enable-external-groups <enable-external-groups>
```

Example

The following example enables authorization.

```
nacm enable-external-groups true
```

Procedure 6 Adding a server to a server group using the CLI

Add a server to a server group to establish the connection queue.

Requirements

Ensure that the server group exists.

Overview

Adding a server to server group creates a server in the system. A server group can contain up to eight servers.

The following table describes the parameters for creating a server.

Table 10 Parameters for creating a server

Parameter	Valid values	Description
name	string	Specifies the server group to associate to the new server.
address	IPv4 address or IPv6 address Format: x.x.x.x or x:x:x:x:x:x:x:x	Specifies the IP address or hostname of the server.

Steps

1 Add a server to the server group:

```
system aaa server-groups server-group <name> servers
server <address> config address <address>
```

Example

The following example adds a server identified by an IPv4 address to a server group.

```
system aaa server-groups server-group rwgroup servers server 192.0.2.2 config
address 192.0.2.2
```

The following example adds a server identified by an IPv6 address to a server group.

```
system aaa server-groups server-group rwgroup servers server
2620:11b:d03d:f108:216:3eff:fe50:600 config address
2620:11b:d03d:f108:216:3eff:fe50:600
```

Procedure 7 Configuring a server using the CLI

Configure a server to change the default settings.

Requirements

The server must be created in the system by adding it to a server group.

Overview

The following table describes the parameters for configuring a server.

Table 11 Parameters for configuring a server

Parameter	Valid values	Description
name	string	Specifies the server group in which the server is a member.
address	IPv4 address or IPv6 address Format: x.x.x.x or x:x:x:x:x:x:x:x	Specifies the IP address or hostname of the server.
type	TACACS | RADIUS | RADSEC	Identifies the AAA protocol type of the server group.
acct-port	0 - 65535 The default value for a tacacs server is 49. The default value for a radius or radsec server is 2083.	Specifies the port on which to contact the server for accounting.
auth-port	0 - 65535 The default value for a tacacs server is 49. The default value for a radius or radsec server is 2083.	Specifies the port on which to contact the server for authorization.
retransmit-attempts	0-3	Maximum number of retries before moving on to the next server.

Parameter	Valid values	Description
secret-key	string, maximum 128 characters	Specifies an unencrypted shared key used between the server and the system. Only an encrypted version of the key string will be preserved in the configuration. A secret-key string of zero length removes the secret-key without having to delete the secret-key leaf.

Steps

1 Configure the server:

```
system aaa server-groups server-group <name> servers
server <address> <type> config acct-port <acct-port>
auth-port<auth-port> retransmit-attempts <retransmit-
attempts> secret-key <secret-key>
```

Example

The following example configures a server.

```
system aaa server-groups server-group rwgroup servers server 192.0.2.2 radius
config acct-port 1813 auth-port 1812 retransmit-attempts 3 secret-key
theonekey
```

Procedure 8 Enabling a server using the CLI

Enable a server when you want to add the server to the connection queue.

Overview

The following table describes the parameters for enabling a server.

Table 12 Parameters for enabling a server

Parameter	Valid values	Description
name	string	Specifies the name of the server group that the server is associated with.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the hostname of the server.
admin-state	enabled | disabled The default value is enabled.	Specifies the administrative state of the AAA server.

Steps

1 Enable a server.

```
system aaa server-groups server-group <name> server
<address> config admin-state <admin-state>
```

Example

The following example enables a server.

```
system aaa server-groups server-group radserver server 192.0.2.2 config
admin-state enabled
```

Procedure 9 Disabling a server using the CLI

Disable a server when you want to remove the server from the connection queue but do not want to delete the configuration for that server.

Overview

The following table describes the parameters for disabling a server.

Table 13 Parameters for disabling a server

Parameter	Valid values	Description
name	string	Specifies the name of the server group that the server is associated with.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the hostname of the server.
admin-state	enabled | disabled The default value is enabled.	Specifies the administrative state of the AAA server.

Steps

1 Disable a server:

```
system aaa server-groups server-group <name> servers
server <address> config admin-state <admin-state>
```

Example

The following example disables a server identified by an IPv4 address.

```
system aaa server-groups server-group rwgroup servers server 10.33.80.91
config admin-state disabled
```

The following example disables a server identified by an IPv6 address.

```
system aaa server-groups server-group rwgroup servers server
2620:11b:d03d:f108:216:3eff:fe50:600 config admin-state disabled
```

Procedure 10 Removing a server from a server group using the CLI

Remove a server from a server group when you want to remove the configuration for the server.

Overview

The following table lists the parameters for removing a server from a server group.

Table 14 Parameters for removing a server from a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server group.
address	string	Specifies the IP address or hostname of the server.

Steps

1 Remove a server from a server group:

```
no system aaa server-groups server-group <name> servers
server <address>
```

Example

The following example removes a server with an IP v4 address from a server group.

```
no system aaa server-groups server-group rwgroup servers server 10.33.80.91
```

The following example removes a server with an IP v6 address from a server group.

```
no system aaa server-groups server-group rwgroup servers server
2620:11b:d03d:f108:216:3eff:fe50:600
```

Procedure 11 Configuring timeout from a RADSEC server using the CLI

Configure session timeout from a RADSEC server, as required by the network plan.

Overview

The following table describes the parameters for configuring timeout from a RADSEC server.

Table 15 Parameters for configuring timeout from a RADSEC server

Parameter	Valid values	Description
name	string	Specifies the server group name to associate with the new server.
address	IP address	Specifies the IP address or the hostname of the server.
timeout	1 - 30 The default value is 5.	Specifies the timeout in seconds on responses from the RADSEC server.

Steps

1 Configure the timeout from a RADSEC server.

```
system aaa server-groups server-group <name> servers
server <address> config timeout <timeout>
```

Example

The following example configures timeout from a RADSEC server.

```
system aaa server-groups server-group radserver servers server 192.0.2.0
config timeout 5
```

Procedure 12 Attaching a TLS service profile to a RADSEC server using the CLI

Attach a TLS service profile to a RADSEC server for sending radius authentication and accounting requests securely over TLS.

Requirements

Ensure that the TLS service profile is configured.

Overview

The following table describes the parameters for attaching a TLS service profile to a RADSEC server.

Table 16 Parameters for attaching a TLS service profile to a RADSEC server

Parameter	Valid values	Description
name	string	Specifies the server group name to associate with the new server.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the hostname of the server.
tls_service_profile	TLS service profile	Specifies the name of the TLS service profile.

Steps

1 Attach a TLS service profile to a RADSEC server.

```
system aaa server-groups server-group <name> servers
server <address> radsec config tls-service-profile
<tls_service_profile>
```

Example

The following example attaches a TLS service profile to a RADSEC server.

```
system aaa server-groups server-group radserver servers server 192.0.2.2
radsec config tls-service-profile tls_srv_profile1
```

Procedure 13 Configuring the search method using the CLI

Configure the search method to specify the order in which to contact servers.

Overview

The following table describes parameters for configuring a search method.

Table 17 Parameters for configuring a search method

Parameter	Valid	Description
name	string	Specifies the name of the server group.
search-method	priority | cached The default value is cached.	priority- retry all servers in priority order cached- uses the server that last returned a positive response

Steps

1. Configure the search method:

   ```
   system aaa server-groups server-group <name> config
   search-method <search-method>
   ```

2. Unset the priority search method to return to default (cached mode):

   ```
   no system aaa server-groups server-group rwgroup config
   search-method
   ```

Example

The following sample RPC sets the search method to priority.

```
system aaa server-groups server-group rwgroup config search-method priority
```

The following sample RPC sets the search method to cached.

```
system aaa server-groups server-group rwgroup config search-method cached
```

Procedure 14 Removing a reference to a server group from the authentication-method list using the CLI

Remove a reference to a server group from the authentication-method list when the reference to the server group is no longer required.

Overview

The following table lists the parameter for removing a reference to a server group from the authentication-method list.

Table 18 Parameter for removing a reference to a server group from the authentication-method list

Parameter	Valid values	Description
name	string	Specifies the name of the server.

Steps

1 Remove a reference to a server group from the authentication-method list

```
no system aaa authentication config authentication-method
<name>
```

Example

The following example removes a reference to a server group from the authentication-method list.

```
no system  aaa authentication config authentication-method rwgroup
```

Procedure 15 Clearing AAA authentication provider statistics using the CLI

Clear AAA authentication provider statistics when the statistics are no longer required.

Overview

The following table lists the parameter for clearing AAA authentication provider statistics.

Table 19 Parameter for clearing AAA authentication provider statistics

Parameter	Valid values	Description
name	string	Specifies the name of the server.

Steps

1 Clear AAA authentication provider statistics:

```
clear aaa stats authentication name <server-group>
```

Example

The following example clears AAA authentication provider statistics for a server group rwgroup.

```
clear aaa stats authentication name rwgroup
```

Procedure 16 Clearing server statistics using the CLI

Clear server statistics when the statistics are no longer required.

Overview

The following table describes parameters for clearing server statistics.

Table 20 Parameters for clearing server statistics

Parameter	Valid	Description
address	IP address	Specifies the IP address of the server.
name	string	Specifies the name of the server group.

Steps

1 Clear server statistics:

```
clear aaa stats server address <address> server-group
<name>
```

Example

The following example clears server statistics for a server group rwgroup.

```
clear aaa stats server address 192.0.2.0 server-group rwgroup
```

Procedure 17 Configuring login accounting event support using CLI

Configure login accounting event support when network requirements call for remote accounting to be enabled.

Overview

Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1 Configure login accounting event support:

```
system aaa accounting events event oc-aaa-
types:AAA_ACCOUNTING_EVENT_LOGIN config event-type oc-
aaa-types:AAA_ACCOUNTING_EVENT_LOGIN record START_STOP
```

Procedure 18 Configuring command accounting event support using CLI

Configure command accounting event support when network requirements call for remote accounting to be enabled.

Overview

Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1 Configure command accounting event support:

```
system aaa accounting events event oc-aaa-
types:AAA_ACCOUNTING_EVENT_COMMAND config event-type
AAA_ACCOUNTING_EVENT_COMMAND record START_STOP
```

Procedure 19 Removing the login accounting event type using CLI

Remove the login accounting event type when the event type is no longer required.

Overview

Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1 Remove the login accounting event type:

```
no system aaa accounting events event oc-aaa-
types:AAA_ACCOUNTING_EVENT_LOGIN
```

Procedure 20 Removing the command accounting event type using CLI

Remove the command accounting event type when the event type is no longer required.

Overview

Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1 Remove the command accounting event type:

```
no system aaa accounting events event oc-aaa-
types:AAA_ACCOUNTING_EVENT_COMMAND
```

Procedure 21 Displaying the server configuration using the CLI

Display the server configuration.

Steps

1. Display the server configuration:

   ```
   show aaa server config
   ```

Example

The following example shows the output from the show aaa server config command for the server group named rwgroup.

```
show aaa server config
+---------------------- SERVER GROUP ----------------------+
|Group Configuration                                       |
+------------------------------+---------------------------+
|Name                          | rwgroup                   |
|Type                          | TACACS                    |
|Search Method                 | cached                    |
+------------------------------+---------------------------+
|Per-Server Configuration                                  |
+----------------------------------------------------------+
|Ip Addr    |Name|Timeout|Admin-state|Port|Secret|Src addr|
+-----------+----+-------+-----------+----+------+--------+
|10.33.80.98|    |      5|enabled    |  49|******|        |
|10.33.80.91|    |      5|enabled    |  49|******|        |
+-----------+----+-------+-----------+----+------+--------+
|Cached Server Search Order                                |
+----------------------------------------------------------+
|10.33.80.91                                               |
|10.33.80.98                                               |
+----------------------------------------------------------+
```

Procedure 22 Displaying server statistics using the CLI

Display server statistics to learn more about the performance of the server.

Overview

The following table describes common server statistics.

Table 21 Common server statistics

Statistic	Description
Connection Opens	Number of new connection requests sent to this server.
Connection Closes	Number of connection close requests sent to this server.
Connection Aborts	Number of aborted connections to this server.
Connection Failures	Number of connection failures to this server.
Connection Timeouts	Number of timeouts that occurred while connecting to this server.
Messages Sent	Number of messages sent to this server.
Messages Received	Number of messages received from this server.
Errors Received	Number of error messages received from this server.
Access Requests	Number of access-request packets sent to this server.
Access Accepts	Number of access-accept packets received from this server.
Access Rejects	Number of access-reject packets received from this server.
Malformed Responses	Number of malformed response packets received from this server.
Bad Authenticators	Number of packets containing invalid authenticators or signature attributes.

The following table lists RADIUS-specific statistics.

Table 22 RADIUS-specific statistics

Statistic	Description
Retransmissions	number of retransmitted access-request messages.
Access Challenges	number of access-challenge packets received from this server.

Statistic	Description
Accounting Responses	number of accounting-response packets (valid or invalid) received from this server.
Unknown Types	number of RADIUS packets of unknown type received from this server on the authentication port.
Packets Dropped	number of RADIUS packets received from this server on the authentication port and dropped for some other reason.
Round Trip Time	the time interval (in hundredths of a second) between the most recent access-reply/access-challenge and the access-request that matched it from this RADIUS server.

The following table lists additional Radsec specific statistics.

Table 23 Radsec specific statistics

Statistic	Description
Failed-tcp-connections	Number of failure tcp connection with radius serve
Failed-tls-connections	Number of failure tls connection with radius serve

Steps

1 Display the server statistics:

```
show aaa server statistics
```

Example

The following example shows the output from the show aaa server statistics command for a TACACS+ server.

```
+------- admingroup (TACACS) -------+
| Ip address           | 10.33.80.91 |
+----------------------+-------------+
| Oper-state           |     enabled |
| Connection-opens     |           2 |
| Connection-closes    |           2 |
| Connection-aborts    |           0 |
| Connection-failures  |           0 |
| Connection-timeouts  |           0 |
| Messages-sent        |           3 |
| Messages-received    |           3 |
| Errors-received      |           0 |
| Access-requests      |           0 |
| Access-accepts       |           1 |
| Access-rejects       |           0 |
| Malformed-responses  |           0 |
| Bad-authenticators   |           0 |
+----------------------+-------------+
```

The following example shows the sample output for show aaa server statistics command for a RADSEC server.

```
+------------- SERVER GROUP -------------+
| Group Configuration                     |
+-------------------------+---------------+
| Name                    | radserver     |
| Type                    | RADSEC        |
| Search Method           | cached        |
+-------------------------+---------------+
| Per-Server Statistics                   |
+-------------------------+---------------+
| Ip address              | 192.0.2.0     |
+-------------------------+---------------+
| Oper-state              |       enabled |
| Connection-opens        |             1 |
| Connection-closes       |             1 |
| Connection-aborts       |             0 |
| Connection-failures     |             0 |
| Connection-timeouts     |             0 |
| Messages-sent           |             1 |
| Messages-received       |             1 |
| Errors-received         |             0 |
| Access-requests         |             1 |
| Access-accepts          |             1 |
| Access-rejects          |             0 |
| Malformed-responses     |             0 |
| Bad-authenticators      |             0 |
+-------------------------+---------------+
| Access-challenges       |             0 |
| Accounting-responses    |             0 |
| Unknown-types           |             0 |
| Packets-dropped         |             0 |
| Round-trip-time         |            36 |
| Failed-tcp-connections  |             0 |
| Failed-tls-Connections  |             0 |
+-------------------------+---------------+
```

Procedure 23 Displaying the authentication configuration using the CLI

Display the authentication configuration to verify the AAA configuration, that is, authentication, users, accounting, server-groups, and servers.

Steps

1 Display the authentication configuration:

```
show aaa authentication
```

Example

The following example shows sample output for the show aaa authentication command.

```
show aaa authentication
+------------------------ AUTHENTICATION PROVIDERS -----------------------+
| Name        | Oper-state | Type    | Called | Success | Failure | Skipped |
+-------------+------------+---------+--------+---------+---------+---------+
| AUTH_LOC    | enabled    | LOCAL   |      3 |       2 |       1 |       0 |
| admingroup  | enabled    | TACACS  |      1 |       1 |       0 |       0 |
+-------------+------------+---------+--------+---------+---------+---------+
```

Procedure 24 Displaying users using the CLI

Display users using the CLI.

Steps

1 Display users using the CLI.

```
show aaa users
```

Example

The following example displays the users using the CLI.

```
show aaa users
+--------------- USER ACCOUNT TABLE ---------------+
| Username | Role              | Sessions | Lockout |
+----------+-------------------+----------+---------+
| diag     | SYSTEM_ROLE_DIAG  |          |         |
| user     | SYSTEM_ROLE_USER  |          |         |
+----------+-------------------+----------+---------+
```

Procedure 25 Displaying accounting information using the CLI

Displaying accounting information.

Steps

1 Display accounting information.

```
show aaa accounting
```

Example

The following example displays accounting information.

```
show aaa accounting

+ ACCOUNTING METHODS +
| Name               |
+--------------------+
| admingroup         |
+--------------------+
+----------- ACCOUNTING EVENTS -----------+
| Event-type                  | Record     |
+-----------------------------+------------+
| AAA_ACCOUNTING_EVENT_LOGIN | START_STOP |
+-----------------------------+------------+
```

Procedure 26 Displaying your user account using CLI

Display your user account to view your account information.

Steps

1 Display your user account:

```
user whoami
```

Example

The following example displays your user account information.

```
user whoami

Username: diag
Privilege-level: SYSTEM_ROLE_DIAG
```

Procedure 27 Displaying user accounts using the CLI

Display the user accounts logged on to the system to learn more about user accounts and associated privilege levels.

Steps

1 Display user accounts:

```
user who
```

Example

The following example displays information about the user accounts that are logged on to the system.

```
user who

+------------ USERS SUMMARY ------------+
| Username | PID   | Terminal           |
+----------+-------+--------------------+
| diag     | 23347 | 10.33.80.209:49774 |
| root     | 27158 | Local:ttyS0        |
+----------+-------+--------------------+
```

Procedure 28 Creating a server group using the YANG model

Create a server group for authentication purposes.

Requirements

The YANG model ciena-openconfig-aaa.yang is used in this procedure.

Overview

The following table describes the parameters for creating a server group.

Table 24 Parameters for creating a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server group.
type	TACACS | RADIUS | RADSEC	Specifies the AAA protocol type for the server group. Setting the type when you create a server group enables defaults to be automatically populated.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to create a server group.

Example

The following sample RPC creates a TACACS server group named tacacsgroup.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>tacacsgroup</name>
              <config>
                <name>tacacsgroup</name>
                <type xmlns:oc-aaa="http://openconfig.net/yang/aaa">TACACS</
type>
              </config>
```

```
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

The following sample RPC creates a RADIUS server group named radiusgroup.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>radiusgroup</name>
              <config>
                <name>radiusgroup</name>
                <type xmlns:oc-aaa="http://openconfig.net/yang/aaa">RADIUS</
type>
              </config>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

The following sample RPC creates a RADSEC server group named radsecgroup.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="4" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
   <default-operation>merge</default-operation>
    <config>
     <system xmlns="http://openconfig.net/yang/system">
      <aaa>
       <server-groups>
        <server-group>
         <name>radsecgroup</name>
          <config>
           <name>radsecgroup</name>
            <type xmlns:ciena-oc-aaa="http://www.ciena.com/ns/yang/ciena-
openconfig-aaa">RADSEC</type>
         </config>
        </server-group>
```

```
        </server-groups>
       </aaa>
      </system>
     </config>
    </edit-config>
</rpc>
```

Procedure 29 Removing a server group using the YANG model

Remove a server group configuration and all nested servers from the system as required.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table lists the parameter for removing a server group configuration.

Table 25 Parameter for removing a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to remove the reference to the server group from the authentication-method list.
3. Send an RPC <get> to verify that the server group has been removed.

Example

The following sample RPC removes the server group named tac_authen.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group nc:operation="delete"
             xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
              <name>tac_authen</name>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 30 Configuring the authentication method using the YANG model

Configure the authentication method used to authenticate users. Authentication can be local or remote. The default is local authentication.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

Server groups are added in the order in which they are added into the authentication method.

The following table describes the parameter for configuring the authentication method.

Table 26 Parameter for configuring the authentication method

Parameter	Valid values	Description
authentication-method	string | AUTH_LOC	Identifies the server group to use for remote authentication. AUTH_LOC specifies local authentication.

CAUTION
Risk of loss of access to the system
Ensure that more than one provider is configured. If only one provider is configured and that provider fails to reply positively to an authentication request or fails to reply to the request the system may enter a lockout state.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to set the authentication method.

Example

The following sample RPC specifies the server group to use for remote authentication.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
```

```
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <authentication>
            <config>
              <authentication-method>tac_authen</authentication-method>
              <authentication-method>AUTH_LOC</authentication-method>
            </config>
          </authentication>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 31 Inserting a method as the first method in the authentication list using the YANG model

Insert a method as the first method in the authentication list.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table lists the parameter for inserting a method as the first method in the authentication list.

Table 27 Parameter for inserting a method as the first method in the authentication list

Parameter	Valid values	Description
authentication-method	<server group name>| AUTH_LOC	• server group name specifies the name of the server group used for remote authentication • AUTH_LOC specifies local authentication

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to insert the method.
3. Send an RPC <get> to verify the list.

Example

The following sample RPC inserts a method as the first method in the authentication list.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <authentication>
            <config>
              <authentication-method yg:insert="first" xmlns:yg="urn:ietf:
              params:xml:ns:yang:1">tac_authen</authentication-method>
              </config>
          </authentication>
        </aaa>
      </system>
```

```
    </config>
  </edit-config>
</rpc>
```

Procedure 32 Inserting a method as the last method in the authentication list using the YANG model

Insert a method as the last method in the authentication list.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table lists the parameter for inserting a method as the last method in the authentication list.

Table 28 Parameter for inserting a method as the last method in the authentication list

Parameter	Valid values	Description
authentication-method	<server group name>| AUTH_LOC	• server group name specifies the name of the server group used for remote authentication • AUTH_LOC specifies local authentication

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to insert the method.
3. Send an RPC <get> to verify the list.

Example

The following sample RPC inserts a method as the last method in the authentication list.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <authentication>
            <config>
              <authentication-method yg:insert="last" xmlns:yg="urn:ietf:
              params:xml:ns:yang:1">tac_authen</authentication-method>
              </config>
          </authentication>
        </aaa>
      </system>
```

```
    </config>
  </edit-config>
</rpc>
```

Procedure 33 Configuring an accounting method using the YANG model

Configure an accounting method to send messages to the server group to record user activity, for example, to account for services used, such as in a billing environment, or as an auditing tool for security services.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table describes the parameter for configuring the accounting method.

Table 29 Parameter for configuring an accounting method

Parameter	Valid values	Description
accounting-method	string	Specifies the server group to send accounting messages to.

Note: Session Accounting and Session Authorization by means of NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure the accounting method.
3. Send an RPC <get> to verify information for the accounting method.

Example

The following example configures the system to send accounting messages to the server group named admingroup.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
<system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <accounting>
            <config>
              <accounting-method>admingroup</accounting-method>
            </config>
```

```
            <events>
              <event>
            <event-type>oc-aaa-types:AAA_ACCOUNTING_EVENT_LOGIN</event-type>
                <config>
                 <event-type xmlns:oc-aaa-types="http://openconfig.net/yang/
aaa/types">oc-aaa-types:AAA_ACCOUNTING_EVENT_LOGIN</event-type>
                  <record>START_STOP</record>
                </config>
              </event>
            </events>
          </accounting>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 34 Enabling TACACS or RADIUS authorization using the YANG model

Enable TACACS or RADIUS authorization.

Requirements

The YANG model ietf-netconf-acm.yang is used in this procedure.

Overview

The following table describes the parameter for enabling TACACS or RADIUS authorization.

Table 30 Parameter for enabling TACACS or RADIUS authorization

Parameter	Valid values	Description
enable-external-groups	true | false	Enables or disables TACACS or RADIUS authorization.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable TACACS or RADIUS authorization.

Example

The following example enables TACACS or RADIUS authorization in the service.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
         <nacm xmlns="urn:ietf:params:xml:ns:yang:ietf-netconf-acm"
xmlns:ncx="http:/
        /netconfcentral.org/ns/yuma-ncx">
        <enable-external-groups>true</enable-external-groups>
      </nacm>
    </config>
  </edit-config>
</rpc>
```

Procedure 35 Adding a server to a server group using the YANG model

Add a server to a server group to establish the connection queue.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Ensure that the server group exists and has a type (TACACS, RADIUS, or RADSEC) configured.

Overview

A server group can contain up to eight servers. Server priority is defined by the order of the server groups in the authentication method list.

The following table describes the parameters for adding a server to a server group.

Table 31 Parameters for adding a server to a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server group.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or hostname of the server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to add servers to a server group.

Example

The following sample RPC adds a server to a server group.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
```

```
              <name>tac_authen</name>
              <servers>
                <server>
                  <address>192.0.2.2</address>
                  <config>
                    <address>192.0.2.2</address>
                  </config>
                </server>
              </servers>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 36 Configuring a server using the YANG model

Configure a server to change the default settings.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

The server must be created in the system by adding it to a server group.

Overview

The following table describes the parameters for configuring a server.

Table 32 Parameters for configuring a server

Parameter	Valid values	Description
name	string	Specifies the server group in which the server is a member.
address	IPv4 address or IPv6 address Format: x.x.x.x or x:x:x:x:x:x:x:x	Specifies the IP address or hostname of the server.
type	TACACS | RADIUS | RADSEC	Identifies the AAA protocol type of the server group.
acct-port	0 - 65535 The default value for a tacacs server is 49. The default value for a radius or radsec server is 2083.	Specifies the port on which to contact the server for accounting.
auth-port	0 - 65535 The default value for a tacacs server is 49. The default value for a radius or radsec server is 2083.	Specifies the port on which to contact the server for authorization.
retransmit-attempts	0-3	Maximum number of retries before moving on to the next server.

Parameter	Valid values	Description
secret-key	string, maximum 128 characters	Specifies an unencrypted shared key used between the server and the system. Only an encrypted version of the key string will be preserved in the configuration. A secret-key string of zero length removes the secret-key without having to delete the secret-key leaf.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure the server.

Example

The following sample RPC configures the server.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>tac_authen</name>
              <servers>
                <server>
                  <address>192.0.2.2</address>
                  <type>RADIUS</type>
                  <config>
                    <acct-port>192.0.2.2</acct-port>
                    <auth-port>192.0.2.2</auth-port>
                    <retransmit-attempts>3</retransmit-attempts>
                    <secret-key>theonekey</secret-key>
                  </config>
                </server>
              </servers>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 37 Enabling a server using the YANG model

Enable a server when you want to add the server to the connection queue.

Requirements

The YANG model ciena-openconfig-aaa.yang is used in this procedure.

Overview

The following table describes the parameters for enabling a server.

Table 33 Parameters for enabling a server

Parameter	Valid values	Description
name	string	Specifies the name of the server group that the server is associated with.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the hostname of the server.
admin-state	enabled | disabled The default value is enabled.	Specifies the administrative state of the AAA server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable a server.

Example

The following example enables a server.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="7" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
     </target>
      <default-operation>merge</default-operation>
       <config>
        <system xmlns="http://openconfig.net/yang/system">
         <aaa>
          <server-groups>
           <server-group>
            <name>radserver</name>
             <servers>
              <server>
               <address>192.0.2.2</address>
                <config>
```

```
          <admin-state xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
           aaa">enabled</admin-state>
          </config>
         </server>
        </servers>
       </server-group>
      </server-groups>
     </aaa>
    </system>
   </config>
  </edit-config>
</rpc>
```

Procedure 38 Disabling a server using the YANG model

Disable a server when you want to remove the server from the connection queue but do not want to delete the configuration for that server. The administrative state for a server can be set to enabled (default) or disabled. Enabled servers can be used for authentication.

Requirements

The YANG model ciena-openconfig-aaa.yang is used in this procedure.

Overview

The following table describes the parameters for disabling a server.

Table 34 Parameters for disabling a server

Parameter	Valid values	Description
name	string	Specifies the name of the server group that the server is associated to. This parameter is configured in the <server-group> node.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or hostname of the server. This parameter is configured in the <server> node.
admin-state	enabled | disabled The default value is enabled.	Specifies the administrative state of the AAA server. This parameter is configured in the <server> node.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to disable the server.

Example

The following sample RPC sets the administrative state of the server with IP address 10.10.20.20 to disabled.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
```

```
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>tac_authen</name>
              <servers>
                <server>
                  <address>192.0.2.0</address>
                  <config>
                    <admin-state xmlns="http://www.ciena.com/ns/yang/
                    ciena-openconfig-aaa">disabled</admin-state>
                  </config>
                </server>
              </servers>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 39 Removing a server from a server group using the YANG model

Remove a server from a server group when you want to remove the configuration for the server.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table lists the parameters for removing a server from a server group.

Table 35 Parameters for removing a server from a server group

Parameter	Valid values	Description
name	string	Specifies the name of the server group.
address	string	Specifies the IP address or hostname of the server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to remove a server from a server group.
3. Send an RPC <get> to verify information for the server group.

Example

The following example RCP removes a server with IP address 192.0.2.0 from a server group called tac_authen.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>tac_authen</name>
              <servers>
                <server nc:operation="delete" xmlns:nc="urn:ietf:
                params:xml:ns:netconf:base:1.0">
                  <address>192.0.2.0</address>
                </server>
              </servers>
```

```
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
```

Procedure 40 Configuring timeout from a RADSEC server using the YANG model

Configure session timeout from a RADSEC server, as required by the network plan.

Requirements

The YANG model ciena-openconfig-aaa.yang is used in this procedure.

Overview

The following table describes the parameters for configuring timeout from a RADSEC server.

Table 36 Parameters for configuring timeout from a RADSEC server

Parameter	Valid values	Description
name	string	Specifies the server group name to associate with the new server.
address	IP address	Specifies the IP address or the hostname of the server.
timeout	1 - 30 The default value is 5.	Specifies the timeout in seconds on responses from the RADSEC server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure timeout from a RADSEC server.

Example

The following example configures timeout from a RADSEC server.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="22" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
    <default-operation>merge</default-operation>
     <config>
      <system xmlns="http://openconfig.net/yang/system">
       <aaa>
        <server-groups>
         <server-group>
          <name>radserver</name>
          <servers>
          <server>
```

```
          <address>192.0.2.0</address>
           <config>
            <timeout>5</timeout>
           </config>
          </server>
         </servers>
        </server-group>
       </server-groups>
      </aaa>
     </system>
    </config>
   </edit-config>
</rpc>
```

Procedure 41 Attaching a TLS service profile to a RADSEC server using the YANG model

Attach a TLS service profile to a RADSEC server for sending radius authentication and accounting requests securely over TLS.

Requirements

The YANG model ciena-openconfig-aaa.yang is used in this procedure.

Ensure that the TLS service profile is configured.

Overview

The following table describes the parameters for attaching a TLS service profile to a RADSEC server.

Table 37 Parameters for attaching a TLS service profile to a RADSEC server

Parameter	Valid values	Description
name	string	Specifies the server group name to associate with the new server.
address	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the hostname of the server.
tls_service_profile	TLS service profile	Specifies the name of the TLS service profile.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to attach a TLS service profile to a RADSEC server.

Example

The following example attaches a TLS service profile to a RADSEC server.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="18" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
   <default-operation>merge</default-operation>
   <config>
    <system xmlns="http://openconfig.net/yang/system">
     <aaa>
```

```
      <server-groups>
       <server-group>
        <name>radserver</name>
         <servers>
          <server>
           <address>192.0.2.2</address>
              <radsec xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
aaa">
             <config>
              <tls-service-profile>tls_srv_profile1</tls-service-profile>
             </config>
         </radsec>
        </server>
       </servers>
      </server-group>
     </server-groups>
    </aaa>
   </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 42 Configuring the search method using the YANG model

Configure the search method to specify the order in which to contact servers.

Requirements

The YANG models ciena-openconfig-system.yang and ciena-openconfig-aaa.yang are used in this procedure.

Overview

The following table describes parameters for configuring the search method.

Table 38 Parameters for configuring the search method

Parameter	Valid	Description
name	string	Specifies the name of the server group.
search-method	priority | cached The default value is cached.	priority- retry all servers in priority order cached- uses the server that last returned a positive response

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure the search method.

Example

The following sample RPC sets the search method to priority for a TACACS server group named rwgroup.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="8" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <default-operation>merge</default-operation>
    <config>
      <system xmlns="http://openconfig.net/yang/system">
        <aaa>
          <server-groups>
            <server-group>
              <name>rwgroup</name>
              <config>
                <search-method xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-aaa">priority</search-method>
              </config>
            </server-group>
```

```
         </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

The following sample RPC unsets the priority search method to return to default (cached mode).

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="8" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <default-operation>merge</default-operation>
    <config>
      <system xmlns="http://openconfig.net/yang/system">
        <aaa>
          <server-groups>
            <server-group>
              <name>rwgroup</name>
              <config>
                <search-method xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-aaa">cached</search-method>
              </config>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 43 Removing a reference to a server group from the authentication-method list using the YANG model

Remove a reference to a server group from the authentication-method list when the reference to the server group is no longer required.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table lists the parameter for removing a reference to a server group from the authentication-method list.

Table 39 Parameter for removing a reference to a server group from the authentication-method list

Parameter	Valid values	Description
name	string	Specifies the name of the server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to remove the reference to the server group from the authentication-method list.
3. Send an RPC <get> to verify that the server group has been removed from the authentication-method list.

Example

The following sample RPC removes the reference to the server group named tac_authen from the authentication-method list.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <authentication>
            <config>
              <authentication-method nc:operation="delete"
              xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
              tac_authen</authentication-method>
            </config>
          </authentication>
        </aaa>
```

```
    </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 44 Configuring command accounting event support using the YANG model

Configure command accounting event support when network requirements call for remote accounting to be enabled.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

Session Accounting and Session Authorization via NACM groups are supported. Command Accounting is also supported. Command Authorization is not supported.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable command accounting.
3. Send an RPC <get> to verify information for command accounting.

Example

The following sample RPC enables remote command accounting.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
<system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <accounting>
            <config>
              <accounting-method>admingroup</accounting-method>
            </config>
            <events>
              <event>
          <event-type>oc-aaa-types:AAA_ACCOUNTING_EVENT_COMMAND</event-type>
                <config>
                 <event-type xmlns:oc-aaa-types="http://openconfig.net/yang/
aaa/types">oc-aaa-types:AAA_ACCOUNTING_EVENT_COMMAND</event-type>
                  <record>START_STOP</record>
                </config>
              </event>
            </events>
          </accounting>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

CHAPTER 3
Secure Shell public key authentication

Public key-based authentication is a mechanism that uses a key pair, which comprises a public key and a private key. The public key is known to everyone and is stored on the Secure Shell (SSH) server. The private key is known to the owner only and is stored on the SSH client.

When public key authentication is enabled, authentication to the server is done without typing a password. In this scenario, the private key acts as a complex password that is never transmitted. Instead, the server generates random data, which is encrypted with the public key and then sent to the client. The client then decrypts this message with the private key and returns the original data to the server. This validates the possession of the private key and allows authentication.

Optionally the private key can be encrypted with a passphrase while generating the key pair. In this scenario, when accessing the server using a passphrase protected private key, the client responds with a passphrase prompt.

OpenSSH compatibility

OpenSSH implements all cryptographic algorithms described in the Secure Shell (SSH) standard.

Ciena recommends configuring strong algorithms on the SSH server. In some scenarios it is not possible to modify the configuration on the SSH server, for example if a user has logged on to a device with a user name that does not have permission to modify the configuration. In such situations, the network operator can re-enable an algorithm to create the SSH connection.

List of procedures

The following procedures are available to configure public key based authentication:

- Procedure 45, “Installing an SSH user public key using the CLI”
- Procedure 46, “Deleting an SSH user public key using the CLI”

- Procedure 47, “Enabling SSH public key authentication using the CLI”
- Procedure 48, “Disabling SSH public key authentication using the CLI”
- Procedure 49, “Configuring encryption algorithms using the CLI”
- Procedure 50, “Configuring key exchange algorithms using the CLI”
- Procedure 51, “Configuring the rekey limit using the CLI”
- Procedure 52, “Configuring the rekey time using the CLI”
- Procedure 53, “Configuring a MAC algorithm using the CLI”
- Procedure 54, “Configuring a PKA algorithm using the CLI”
- Procedure 55, “Displaying SSH server information using the CLI”
- Procedure 56, “Installing an SSH user public key using the YANG model”
- Procedure 57, “Deleting an SSH user public key using the YANG model”
- Procedure 58, “Enabling SSH public key authentication using the YANG model”
- Procedure 59, “Disabling SSH public key authentication using the YANG model”
- Procedure 60, “Configuring encryption algorithms using the YANG model”
- Procedure 61, “Configuring key exchange algorithms using the YANG model”
- Procedure 62, “Configuring the rekey limit using the YANG model”
- Procedure 63, “Configuring the rekey time using the YANG model”
- Procedure 64, “Configuring a MAC algorithm using the YANG model”
- Procedure 65, “Configuring a PKA algorithm using the YANG model”
- Procedure 66, “Initiating an SSH connection using a host key algorithm”
- Procedure 67, “Initiating an SSH connection using a cipher algorithm”
- Procedure 68, “Initiating an SSH connection using a key exchange algorithm”
- Procedure 69, “Retrieving SSH server information using the YANG model”

Procedure 45 Installing an SSH user public key using the CLI

Install an SSH user public key on the SSH server to authenticate and initiate connection with the SSH client.

Overview

Public keys are stored on the system at /mnt/secure/ssh-server/users as <user>.pub. If the downloaded public key uses the SSH2 format, then it is converted to the openSSH format and stored on the system. After conversion the downloaded SSH2 format public key is deleted.

The following table describes the parameters for installing an SSH user public key.

Table 40 Parameters for installing an SSH user public key

Parameter	Valid values	Description
user	string	Specifies the shell user for whom the public key is being installed.
filename	filepath	Specifies the path to the public key file.
server-type	ftp-server | http-server	Specifies a list of supported servers to download the public key.
address	IPv4 address or IPv6 address Format: x.x.x.x or x:x:x:x:x:x:x:x	Specifies the host IP.
login-id	string	Specifies the login ID of the download server.
password	string	Specifies the password of the download server.
url	string	Specifies the transfer protocol, IP address/ hostname, port, path to the public key file, user name, and password.

Steps

1 Install an SSH user public key.

```
system ssh-server user-pubkey install user-name <user>
filename <filename> [server-type <server-type>] address
<address> [login <login-id> password <password>]
```

```
OR
```

```
system ssh-server user-pubkey install user-name <user>
url <url>
```

Example

The following examples installs an SSH user public key using the individual parameters of the command.

```
system ssh-server user-pubkey install user-name diag filename
/home/ubuntu/opensshKey.pub server-type ftp-server address 192.0.2.2
login diag password diag
```

The following examples installs an SSH user public key using the URL parameter.

```
system ssh-server user-pubkey install user-name diag url
http://192.0.2.2:8000/rsa_diag.pub
```

Procedure 46 Deleting an SSH user public key using the CLI

Delete an SSH user public key from the system when it is no longer needed.

Overview

The following table describes the parameter for deleting an SSH user public key.

Table 41 Parameter for deleting an SSH user public key

Parameter	Valid values	Description
user	string	Specifies the user name.

Steps

1 Delete an SSH user public key.

```
system ssh-server user-pubkey delete user-name <user>
```

Example

The following example deletes an SSH user public key.

```
system ssh-server user-pubkey delete user-name diag
```

Procedure 47 Enabling SSH public key authentication using the CLI

Enable SSH public key authentication to enable logging on to an SSH server using a public/private key pair.

Overview

The following table describes the parameter for enabling SSH public key authentication.

Table 42 Parameter for enabling SSH public key authentication

Parameter	Valid values	Description
public-key-authentication	enabled | disabled	Sets the state of public key authentication.

Steps

1 Enable SSH public key authentication.

```
system ssh-server config public-key-authentication
<public-key-authentication>
```

Example

The following example enables SSH public key authentication.

```
system ssh-server config public-key-authentication enabled
```

Procedure 48 Disabling SSH public key authentication using the CLI

Disable SSH public key authentication to revert to using password authentication to log on to the SSH server.

Overview

The following table describes the parameter for disabling SSH public key authentication.

Table 43 Parameter for disabling SSH public key authentication

Parameter	Valid values	Description
public-key-authentication	enabled | disabled	Sets the state of public key authentication.

Steps

1 Disable SSH public key authentication.

```
system ssh-server config public-key-authentication
<public-key-authentication>
```

Example

The following example disables SSH public key authentication.

```
system ssh-server config public-key-authentication disabled
```

Procedure 49 Configuring encryption algorithms using the CLI

Configure an encryption algorithm to allow the specified encryption algorithm to be used during an SSH handshake.

Overview

The following table describes the parameter for configuring encryption algorithms.

Table 44 Parameter for configuring encryption algorithms

Parameter	Valid values	Description
encryption-algorithm	3des-cbc | aes128-cbc | aes192-cbc | aes256-cbc | aes128-ctr | aes192-ctr | aes256-ctr | aes128-gcm@openssh.com | aes256-gcm@openssh.com | chacha20-poly1305@openssh.com | rijndael-cbc@lysator.liu.se	Specifies the encryption algorithms.

Steps

1 Configure an encryption algorithm.

```
system ssh-server config encryption-algorithm
<encryption-algorithm>
```

Example

The following example configures encryption algorithms.

```
system ssh-server config encryption-algorithm chacha20-poly1305-openssh.com
aes128-gcm-openssh.com
```

Procedure 50 Configuring key exchange algorithms using the CLI

Configure a key exchange algorithm to allow the specified key exchange algorithm to be used during an SSH handshake.

Overview

The following table describes the parameter for configuring key exchange algorithms.

Table 45 Parameter for configuring key exchange algorithms

Parameter	Valid values	Description
kex-algorithm	curve25519-sha256 | curve25519-sha256@libssh.org | ecdh-sha2-nistp384 | ecdh-sha2-nistp521 | diffie-hellman-group-exchange-sha1 | diffie-hellman-group-exchange-sha256 | diffie-hellman-group1-sha1 | diffie-hellman-group14-sha1 | diffie-hellman-group14-sha256 | diffie-hellman-group16-sha512 | diffie-hellman-group18-sha512	Specifies the key exchange algorithms.

Steps

1 Configure key exchange algorithms.

```
system ssh-server config kex-algorithm <kex-algorithm>
```

Example

The following example configures key exchange algorithms.

```
system ssh-server config kex-algorithm curve25519-sha256-libssh.org
ecdh-sha2-nistp521
```

Procedure 51 Configuring the rekey limit using the CLI

Configure the rekey limit to specify the maximum amount of data that can be transmitted before the session key is renegotiated.

Overview

The following table describes the parameter for configuring the rekey limit.

Table 46 Parameter for configuring the rekey-limit

Parameter	Valid values	Description
rekey-limit	1G | 500M | default	Specifies the amount of data between SSH session key renegotiations. When the rekey limit is set to default, it corresponds to rekeying performed after the encryption algorithm's default amount of data is sent or received.

Steps

1 Configure the rekey limit.

```
system ssh-server config rekey-limit <rekey-limit>
```

Example

The following example configures the rekey limit.

```
system ssh-server config rekey-limit 500M
```

Procedure 52 Configuring the rekey time using the CLI

Configure the rekey time to specify the time in seconds after which the session key can be renegotiated.

Overview

The following table describes the parameter for configuring the rekey time.

Table 47 Parameter for configuring the rekey time

Parameter	Valid values	Description
rekey-time	0 - 3600	Specifies the time between SSH session key renegotiations. Rekey time is measured in seconds. When set to 0, the rekey timeout is disabled.

Steps

1 Configure the rekey time.

```
system ssh-server config rekey-time <rekey-time>
```

Example

The following example configures the rekey time.

```
system ssh-server config rekey-time 50
```

Procedure 53 Configuring a MAC algorithm using the CLI

Configure the MAC algorithm to provide message authentication.

Overview

The following table describes the parameter for configuring a MAC algorithm.

Table 48 Parameter for configuring a MAC algorithms

Parameter	Valid values	Description
mac-algorithm	hmac-md5-etm | hmac-sha1-etm | umac-64-etm | unmac-128-etm | hmac-sha2-256-etm | hmac-sha2-512-etm | hmac-sha1-96-etm | hmac-md5-96-etm | hmac-md5 | hmac-sha1 | umac-64 | umac-128 | hmac-sha2-256 | hmac-sha2-512 | hmac-sha1-96 | hmac-md5-96	Specifies the MAC algorithms.

Steps

1 Configure a MAC algorithm.

```
system ssh-server config mac-algorithm <mac-algorithm>
```

Example

The following example configures a mac algorithm.

```
system ssh-server config mac-algorithm hmac-sha1-etm
```

Procedure 54 Configuring a PKA algorithm using the CLI

Configure a public key authentication (PKA) algorithm to send encrypted messages and enable the recipient to decrypt the message.

Overview

The following table describes the parameter for configuring a pka algorithm.

Table 49 Parameter for configuring a PKA algorithms

Parameter	Valid values	Description
pka-algorithm	ssh-rsa | ssh-dss | ssh-ed25519 | ecdsa-sha2-nistp256 | ecdsa-sha2-nistp384 | ecds-sha2-nistp521	Specifies pka algorithms.

Steps

1 Configure a PKA algorithm.

```
system ssh-server config pka-algorithm <pka-algorithm>
```

Example

The following example configures a pka algorithm.

```
system ssh-server config pka-algorithm ssh-rsa
```

Procedure 55 Displaying SSH server information using the CLI

View SSH server information to verify the configuration details.

Steps

1 View SSH server information.

```
show system ssh-server config
```

Example

The following example shows sample SSH server information.

```
+----------------- SSH SERVER CONFIG ----------------------------+
| Name                                  | Value                  |
+---------------------------------------+------------------------+
| Encryption Algorithm                  | Default Encryption Algo |
| Kex Algorithm                         | Default Kex Algo       |
| Public Key Authentication             | enabled                |
| Rekey Limit                           | Default                |
| Rekey Time                            | None                   |
+---------------------------------------+------------------------+
```

Procedure 56 Installing an SSH user public key using the YANG model

Install an SSH user public key on the SSH server to authenticate and initiate connection with the SSH client.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

Public keys are stored on the system at /mnt/secure/ssh-server/users as <user>.pub. If the downloaded public key uses the SSH2 format, then it is converted to the openSSH format and stored on the system. After conversion the downloaded SSH2 format public key is deleted.

The following table describes the parameters for installing an SSH user public key.

Table 50 Parameters for installing an SSH user public key

Parameter	Valid values	Description
user	string	Specifies the shell user for whom the public key is being installed.
filename	filepath	Specifies the path to the public key.
server-type	ftp-server | http-server	Specifies a list of supported servers to download the public key.
address	IPv4 address or IPv6 address Format: x.x.x.x or x:x:x:x:x:x:x:x	Specifies the host IP.
login-id	string	Specifies the login ID of the download server.
password	string	Specifies the password of the download server.

Parameter	Valid values	Description
url	string	Specifies the transfer protocol, IP address/ hostname, port, path to the public key file, user name, and password.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to install SSH user public key.

Example

The following sample RPC installs an SSH user public key using the individual parameters.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="14" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ssh-user-pubkey-install xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-system">
      <user>diag</user>
      <filename>filename.pub</filename>
      <server-type xmlns:ciena-ftt="http://www.ciena.com/ns/yang/ciena-file-
transfer-types">ciena-ftt:http-server</server-type>
      <address>192.0.2.2</address>
      <login-id>testLogin</login-id>
      <password>*</password>
   </ssh-user-pubkey-install>
</rpc>
```

The following examples installs an SSH user public key using the URL parameter.

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="11" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ssh-user-pubkey-install xmlns="http://www.ciena.com/ns/yang/ciena
   -openconfig-system">
      <user>diag</user>
      <url>http://192.0.2.2:8000/rsa_diag.pub</url>
   </ssh-user-pubkey-install>
</rpc>
```

Procedure 57 Deleting an SSH user public key using the YANG model

Delete an SSH public key from the system when it is no longer needed.

Requirements

The YANG model ciena-openconfig-system.yang is used in this procedure.

Overview

The following table describes the parameter for deleting an SSH user public key.

Table 51 Parameter for deleting an SSH user public key

Parameter	Valid values	Description
user	string	Specifies the user name.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to delete an SSH user public key.

Example

The following sample RPC deletes an SSH user public key.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="4" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <ssh-user-pubkey-delete xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-system">
      <user>user2</user>
   </ssh-user-pubkey-delete>
</rpc>
```

Procedure 58 Enabling SSH public key authentication using the YANG model

Enable SSH public key authentication to enable logging onto an SSH server using a public/private key pair.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for enabling SSH public key authentication.

Table 52 Parameter for enabling SSH public key authentication

Parameter	Valid values	Description
public-key-authentication	enabled | disabled	Sets the state of public key authentication.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable SSH public key authentication.

Example

The following sample RPC enables public key based authentication.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="5" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
          <running/>
      </target>
      <default-operation>merge</default-operation>
      <config>
          <system xmlns="http://openconfig.net/yang/system">
             <ssh-server>
                <config>
                      <public-key-authentication xmlns="http://
www.ciena.com/ns/yang/ciena-openconfig-system">enabled
                </public-key-authentication>
                </config>
             </ssh-server>
          </system>
      </config>
   </edit-config>
</rpc>
```

Procedure 59 Disabling SSH public key authentication using the YANG model

Disable SSH public key authentication to revert to using password authentication to log on to the SSH server.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for disabling SSH public key authentication.

Table 53 Parameter for disabling SSH public key authentication

Parameter	Valid values	Description
public-key-authentication	enabled | disabled	Sets the state of public key authentication.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to disable SSH public key authentication.

Example

The following sample RPC disables SSH public key authentication.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="5" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
   <edit-config>
      <target>
          <running/>
      </target>
      <default-operation>merge</default-operation>
      <config>
          <system xmlns="http://openconfig.net/yang/system">
             <ssh-server>
                <config>
                      <public-key-authentication xmlns="http://
www.ciena.com/ns/yang/ciena-openconfig-system">disabled
                </public-key-authentication>
                </config>
             </ssh-server>
          </system>
      </config>
   </edit-config>
</rpc>
```

Procedure 60 Configuring encryption algorithms using the YANG model

Configure an encryption algorithm to allow the specified encryption algorithm to be used during an SSH handshake.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for configuring encryption algorithms.

Table 54 Parameter for configuring an encryption algorithm

Parameter	Valid values	Description
encryption-algorithm	3des-cbc | aes128-cbc | aes192-cbc | aes256-cbc | aes128-ctr | aes192-ctr | aes256-ctr | aes128-gcm@openssh.com | aes256-gcm@openssh.com | chacha20-poly1305@openssh.com | rijndael-cbc@lysator.liu.se	Specifies the encryption algorithms.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure encryption algorithms.

Example

The following sample RPC configures encryption algorithms.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="6" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
   <default-operation>merge</default-operation>
    <config>
     <system xmlns="http://openconfig.net/yang/system">
      <ssh-server>
        <config>
          <encryption-algorithm xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-system" xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes128-ctr
          </encryption-algorithm>
          <encryption-algorithm
           xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-system"
```

```
           xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes192-ctr
          </encryption-algorithm>
        </config>
      </ssh-server>
    </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 61 Configuring key exchange algorithms using the YANG model

Configure a key exchange algorithm to allow the specified key exchange algorithm to be used during an SSH handshake.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for configuring key exchange algorithms.

Table 55 Parameter for configuring key exchange algorithms

Parameter	Valid values	Description
kex-algorithm	curve25519-sha256 | curve25519-sha256@libssh.org | ecdh-sha2-nistp384 | ecdh-sha2-nistp521 | diffie-hellman-group-exchange-sha1 | diffie-hellman-group-exchange-sha256 | diffie-hellman-group1-sha1 | diffie-hellman-group14-sha1 | diffie-hellman-group14-sha256 | diffie-hellman-group16-sha512 | diffie-hellman-group18-sha512	Specifies the key exchange algorithms.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure key exchange algorithms.

Example

The following sample RPC configures key exchange algorithms.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="7" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <edit-config>
  <target>
   <running/>
  </target>
   <default-operation>merge</default-operation>
    <config>
     <system xmlns="http://openconfig.net/yang/system">
      <ssh-server>
       <config>
        <kex-algorithm
          xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-system"
```

```
          xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:diffie-hellman-group-exchange-sha256
        </kex-algorithm>
        <kex-algorithm
        xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-system"
        xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:diffie-hellman-group16-sha512
        </kex-algorithm>
     </config>
    </ssh-server>
   </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 62 Configuring the rekey limit using the YANG model

Configure the rekey limit to specify the maximum amount of data that can be transmitted before the session key is renegotiated.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for the configuring rekey limit.

Table 56 Parameter for configuring the rekey limit

Parameter	Valid values	Description
rekey-limit	1G | 500M | default	Specifies the amount of data between SSH session key renegotiations. When it is set to default, it corresponds to rekeying performed after the encryption algorithm's default amount of data is sent or received.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure the rekey limit.

Example

The following sample RPC configure the rekey limit.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="8" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <edit-config>
  <target>
    <running/>
  </target>
  <default-operation>merge</default-operation>
  <config>
   <system xmlns="http://openconfig.net/yang/system">
    <ssh-server>
     <config>
      <rekey-limit xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
system">default</rekey-limit>
      </config>
    </ssh-server>
   </system>
  </config>
```

```
 </edit-config>
</rpc>
```

Procedure 63 Configuring the rekey time using the YANG model

Configure the rekey time to specify the time in seconds after which the session key can be renegotiated.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for configuring the rekey time.

Table 57 Parameter for configuring the rekey time

Parameter	Valid values	Description
rekey-time	0 - 3600	Specifies the time between SSH session key renegotiations. Rekey time is measured in seconds. When set to 0, the rekey timeout is disabled.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure the rekey time.

Example

The following sample RPC shows how to configure the rekey time.

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="8" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <edit-config>
  <target>
    <running/>
  </target>
  <default-operation>merge</default-operation>
  <config>
   <system xmlns="http://openconfig.net/yang/system">
    <ssh-server>
     <config>
      <rekey-time xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
system">3600</rekey-time>
      </config>
    </ssh-server>
   </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 64 Configuring a MAC algorithm using the YANG model

Configure a MAC algorithm to provide message authentication.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for configuring a MAC algorithm.

Table 58 Parameter for configuring a MAC algorithm

Parameter	Valid values	Description
mac-algorithm	hmac-md5-etm | hmac-sha1-etm | umac-64-etm | unmac-128-etm | hmac-sha2-256-etm | hmac-sha2-512-etm | hmac-sha1-96-etm | hmac-md5-96-etm | hmac-md5 | hmac-sha1 | umac-64 | umac-128 | hmac-sha2-256 | hmac-sha2-512 | hmac-sha1-96 | hmac-md5-96	Specifies the MAC algorithms.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure a MAC algorithm.

Example

The following sample RPC configures a MAC algorithm.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="6" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
   <default-operation>merge</default-operation>
    <config>
     <system xmlns="http://openconfig.net/yang/system">
      <ssh-server>
        <config>
         <mac-algorithm xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
system" xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes128-ctr
          </mac-algorithm>
          <mac-algorithm
           xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-system"
```

```
           xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes192-ctr
          </mac-algorithm>
        </config>
      </ssh-server>
    </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 65 Configuring a PKA algorithm using the YANG model

Configure a public key authentication (PKA) algorithm to send encrypted messages and enable the recipient to decrypt the message.

Requirements

The YANG models openconfig-system.yang and ciena-openconfig-system.yang are used in this procedure.

Overview

The following table describes the parameter for configuring a pka algorithm.

Table 59 Parameter for configuring a PKA algorithm

Parameter	Valid values	Description
pka-algorithm	ssh-rsa | ssh-dss | ssh-ed25519 | ecdsa-sha2-nistp256 | ecdsa-sha2-nistp384 | ecds-sha2-nistp521	Specifies pka algorithms.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure a MAC algorithm.

Example

The following sample RPC configures a MAC algorithm.

```
<?xml version="1.0" encoding="UTF-8"?>
 <rpc message-id="6" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
   <target>
    <running/>
   </target>
   <default-operation>merge</default-operation>
    <config>
     <system xmlns="http://openconfig.net/yang/system">
      <ssh-server>
        <config>
         <mac-algorithm xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-
system" xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes128-ctr
          </mac-algorithm>
          <mac-algorithm
           xmlns="http://www.ciena.com/ns/yang/ciena-openconfig-system"
           xmlns:sshtypes="http://ciena.com/ns/yang/ciena-ssh-
types">sshtypes:aes192-ctr
          </mac-algorithm>
        </config>
      </ssh-server>
```

```
    </system>
  </config>
 </edit-config>
</rpc>
```

Procedure 66 Initiating an SSH connection using a host key algorithm

Initiate an SSH connection using a host key algorithm to enable the host key algorithm for creating an SSH connection.

Overview

This procedure is performed by means of an SSH client running on the device used to access the system.

The following table describes the parameters for initiating an SSH connection using a host key algorithm.

Table 60 Parameters for initiating an SSH connection using a host key algorithm

Parameter	Valid values	Description
user	string	Specifies the user name on the remote SSH server.
hostname	IP address or hostname Format: x.x.x.x or x:x:x:x:x:x:x:x or example.com	Specifies the IP address or the domain name of the remote SSH server.

Steps

At the device used to access the system

1. Open an SSH client.
2. Initiate an SSH connection using a host key algorithm.

   ```
   ssh -oHostKeyAlgorithms=+ssh-dss <user>@<hostname>
   ```

Example

The following example initiates an SSH connection to 192.0.2.2 using a host key algorithm.

```
ssh -oHostKeyAlgorithms=+ssh-dss diag@192.0.2.2
```

Procedure 67 Initiating an SSH connection using a cipher algorithm

Initiate an SSH connection using a cipher algorithm to enable the cipher algorithm for creating an SSH connection.

Overview

This procedure is performed by means of an SSH client running on the device used to access the SAOS system.

The following table describes the parameters for initiating an SSH connection using a cipher algorithm.

Table 61 Parameters for initiating an SSH connection using a cipher algorithm

Parameter	Valid values	Description
cipher	aes128-cbc | aes192-cbc | aes256-cbc	Specifies a list of cipher algorithms.
user	string	Specifies the user name on the remote SSH server.
hostname	string	Specifies the IP address or the domain name of the remote SSH server.

Steps

At the device used to access the system

1. Open an SSH client.
2. Initiate an SSH connection using a cipher algorithm.

```
ssh -oCiphers=+<cipher> <user>@<hostname>
```

Example

The following example initiates an SSH connection to 192.0.2.2 using the aes256-cbc cipher algorithm.

```
ssh -oCiphers=+aes256-cbc diag@192.0.2.2
```

Procedure 68 Initiating an SSH connection using a key exchange algorithm

Initiate an SSH connection using a key exchange algorithm to enable the key exchange algorithm for creating an SSH connection.

Overview

This procedure is performed by means of an SSH client running on the device used to access the system.

The following table describes the parameters for initiating an SSH connection using a key exchange algorithm.

Table 62 Parameters for initiating an SSH connection using a key exchange algorithm

Parameter	Valid values	Description
weak kex algorithm	diffie-hellman-group-exchange-sha1 | diffie-hellman-group1-sha1	Specifies a list of weak key exchange algorithms.
user	string	Specifies the user name on the remote SSH server.
hostname	string	Specifies the IP address or the domain name of the remote SSH server.

Steps

At the device used to access the system

1. Open an SSH client.
2. Initiate an SSH connection using a key exchange algorithm.

```
ssh -oKexAlgorithms=+<weak kex algorithm>
<user>@<hostname>
```

Example

The following example initiates an SSH connection to 192.0.2.2 using the diffie-hellman-group1-sha1 key exchange algorithm.

```
ssh -oKexAlgorithms=+diffie-hellman-group1-sha1 diag@192.0.2.2
```

Procedure 69 Retrieving SSH server information using the YANG model

View SSH server information to verify the configuration details.

Steps

1. Establish a NETCONF connection to the server.
2. Send an RPC to retrieve SSH server information.

Example

The following sample RPC retrieve SSH server information.

CHAPTER 4
X.509 certificates

X.509 functionality is used by other functions/managers for authentication and encryption. There are two categories of X.509 certificates:

- System certificates are stored in a global directory that SAOS uses to identify itself.
- Trust store of Certificate Authority (CA) certificates and certificate revocation lists (CRLs) that are used to verify the identity of peers.

X.509 certificates contain owner information, a public key, and a signature which prove that the certificate is approved by a higher level certificate authority. They may also contain additional information which includes limitations on what the certificate can be used for and a valid date range.

End entities also require a private key to establish identity since it is paired with a public key in its certificate. Private keys are protected by encryption with a passphrase. This means that the key is useless without the passphrase, providing that the passphrase is protected. When a client wants to prove its identity to a server, the signed, public system certificate and private system key/passphrase must previously be installed on the client, and the root certificate that signed the client's certificate must be installed on the server. Intermediate certificates can be bundled with the root certificate on the server or with the system certificate on the client. During authentication, the client provides its system certificate to the server and uses its own private key to prove ownership of that certificate. The server verifies the certificate's signature against a chain of certificates back to the root certificate it holds and trusts. Authentication can be done in the opposite direction providing that the server has its own signed system certificate and private key and that the client has a copy of the root certificate that signed it. This can be the same root certificate that signed the client's system certificate or a different root certificate.

Public key infrastructure

Public Key Infrastructure (PKI) requires a public key and a private key (which must remain secret). The keys are linked so that messages encrypted with either of these keys can only be decrypted with the other.

PKI and X.509 are both are used for authentication and encryption.

The public key can be used to create messages that can only be decrypted by the owner of the private key. The owner of the private key is the only one who can create messages that can be decrypted with the public key. The public key can be used to verify the identify of the private key owner and to communicate securely with the private key owner. Private keys are encrypted with a pass-phrase that must be presented each time the key is used for any purpose.

SAOS provides the ability to install CA certificates, certificate revocation lists (CRLs), system certificates, and system private keys with passphrases. These are stored in non-volatile memory. Private keys and their passphrases are not readable.

During installation, SAOS performs basic checks on the certificates to verify that they are actual certificates in readable formats, and that the passphrase can correctly decrypt the private key. SAOS provides basic display functionality for certificates, and is able to verify the presence of the private key.

Trusted CA certificates are stored globally for use by any application or port since the infrastructure checks system certificates against any trusted certificate in a CA directory. System certificates are stored in a global directory and are referenced by the application/port that wanted to use the certificate.

Trusted CAs, CRLs, system certificates and private keys are pre-installed. Users can install or update these by commands that instruct SAOS to get them from an xFTP or HTTP server. Updated information can be reinstalled at any time.

X.509

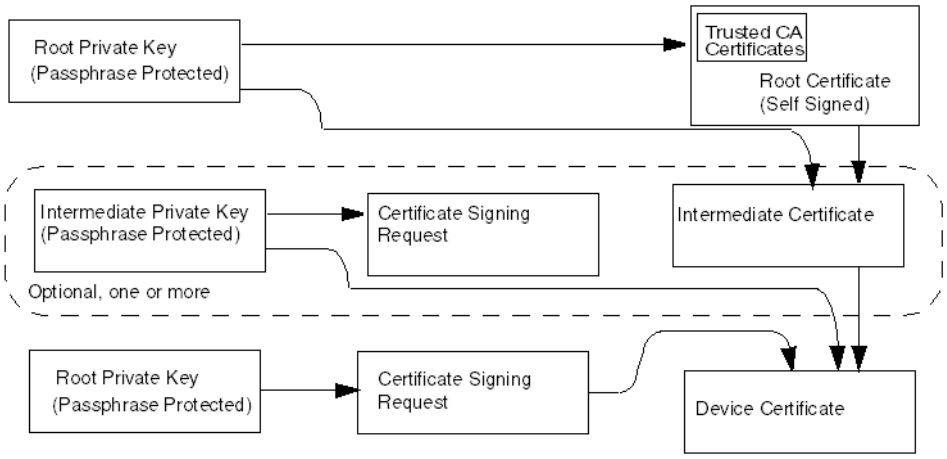
X.509 is an ITU-T standard for PKI. X.509 specifies standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm. PKI and X.509 are used for authentication and encryption.

X.509 provides standards for encapsulating public keys in certificates and certificate revocation lists, and the certification path validation algorithm. Certificates contain additional information to identify the owner of the associated private key, use restrictions, such as validity date range, and a digital signature.

Certificates are signed with a private key. They may be self-signed or signed by a more trusted private key owner using their private key.

The following figure shows an example of the relationship between private keys and system certificates, and certificate signing requests used to create certificates signed by a higher level certificate/owner.

Figure 1 Example relationship



Example use cases for PKI and X.509

This section outlines these use cases:

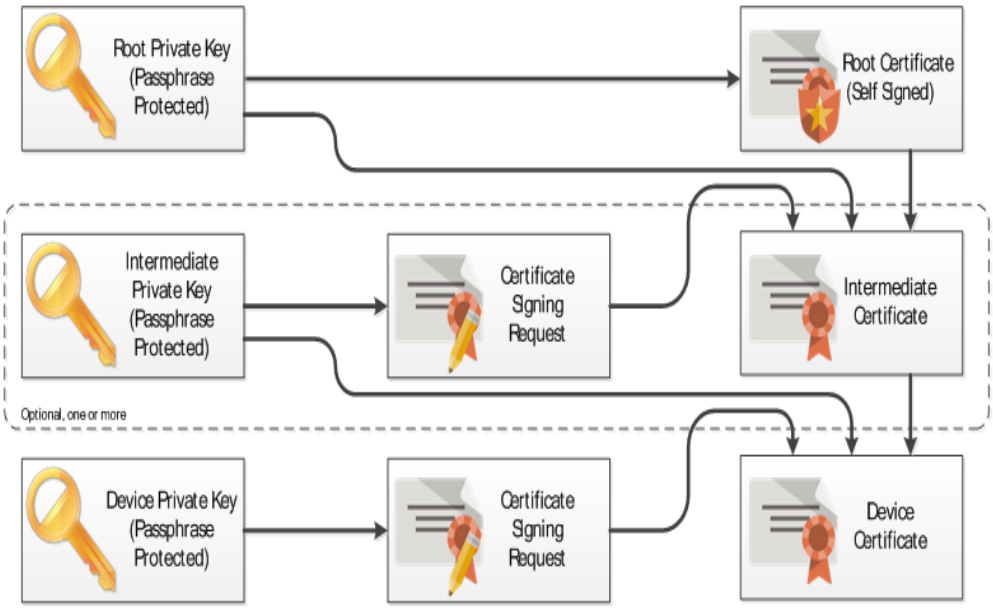
- “Certificate authority or certificate revocation lists installation” on page 109
- “System certificate and private key installation” on page 110
- “NTE authenticated by server” on page 111
- “NTE authenticates another system” on page 111

Certificate authority or certificate revocation lists installation

xFTP commands are used to get the certificates from an xFTP server and install them in a global CA directory. These installed certificates can be used to verify a system certificate. CAs can also be installed from a web server by executing a SAOS command that takes a URL and does an HTTP-Get to install them in the same global CA directory. CRL files are handled in the same manner and installed in the same directory. Applications requesting OpenSSL to validate a system certificate can specify this directory as the ca_path and OpenSSL does everything else.

The following figure shows the installation of a CA certificate or CRLs.

Figure 2 CA certificate or CRL installation



System certificate and private key installation

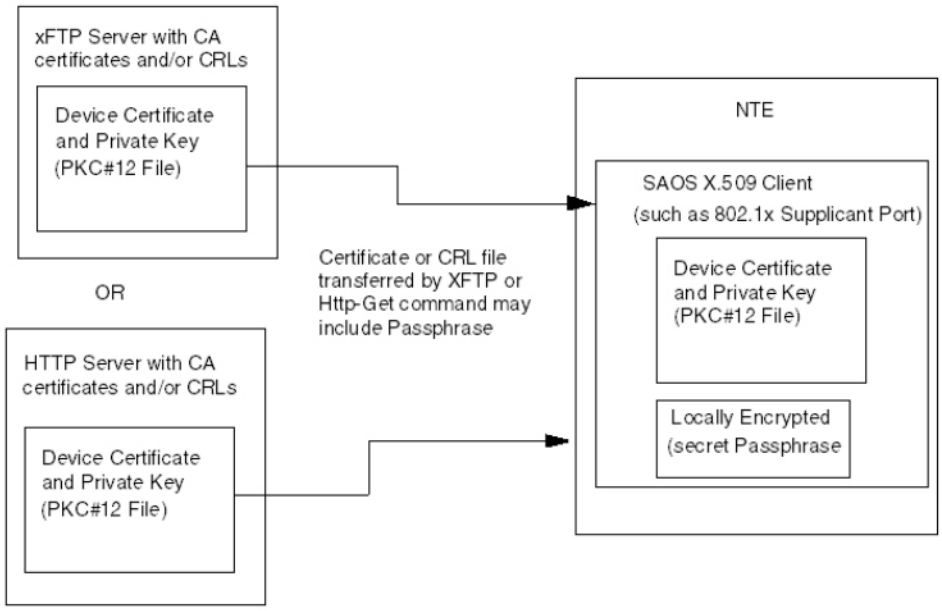
System certificates and their associated private keys must be created and signed outside SAOS. These are bundled into a PKCS#12 file, which is the standard format for combining private keys, system certificates and optional intermediate certificates. PKCS#12 provides FIPS 140-2 compliant passphrase protection on private keys. External tools, such as OpenSSL, provide a mechanism to convert other formats and bundle keys and certificates into PKCS#12 format before installing them in SAOS.

To install system certificates and private keys, xFTP commands can be used to get the certificate files from an xFTP server and install them for a specific application and/or port. System certificates can be installed from a web server by executing a SAOS command that takes a URL and does an HTTP-Get to install them in the same global CA directory.

If a file is passphrase protected (recommended), the passphrase can be specified as an optional part of the install command. The passphrase is locally encrypted and saved with the system certificate and private key file.

The following figure shows the installation of a system certificate and private key.

Figure 3 System certificate and private key installation



NTE authenticates another system

Authentication can be done in the opposite direction providing that the server has its own signed system certificate and private key. NTE must also have a copy of the root certificate that signed it. This may be the same root certificate that signed the client's system certificate or a different root certificate.

NTE authenticated by server

When the NTE wants to prove its identify to a server, the signed, public system certificate and the private system key/passphrase must previously be installed on the client. The root certificate that signed the client's certificate must also be installed on the server. Intermediate certificates can be bundled with the root certificate on the server or with the system certificate on the client.

During authentication, the client provides its system certificate to the server and uses its private key to prove ownership of the certificate. The server verifies the certificate's signature against the chain of certificates back to the root certificate that it holds and trusts.

Check IP host

The Check IP/host allows the user to specify which systems are allowed to connect. The user configures a list of acceptable IP addresses and DNSs then enables check IP host. The list is cross-referenced with either the Subject Alternate Name or the Common Name of the certificate. If it matches, then a TLS connection is established.

The list is called the ip-host-list. This ip-host-list must be configured by the user prior to enabling check IP host. The ip-host-list is part of peer-auth-profile allowing the customer to have different ip-host entries for different profiles.

Enabling check IP host without a configured ip-host-list and deleting the ip-host-list while check IP host is enabled returns error messages.

Online Certificate Status Protocol

Online Certificate Status Protocol (OCSP) is one of two methods used to maintain the security of a server and other network resources. The other method is known as Certificate Revocation List (CRL).

Unlike CRL, OCSP does not require updates to be frequently downloaded to keep the list current at the client end. When a user attempts to access a server, OCSP sends a request for certificate status information. The server sends back a response of current, expired or unknown. The protocol specifies the syntax for communication between the server (which contains the certificate status) and the client application (which is informed of that status).

OCSP is an Internet Protocol used to obtain the revocation status of a digital certificate. Messages that are communicated through OCSP are encoded in ASN.1 and are usually communicated over HTTP. The request/response nature of these messages results in OCSP servers being termed OCSP responders.

CRL is a list that contains all the serial numbers of certificates that have been revoked. These lists have to be updated frequently by the certificate issuer, unlike OCSP. When the list is outdated, it is no longer reliable to identify revoked certificates. Keeping the lists updated is tedious and the CRL process is often faulty due to the chance that revocation lists may not always be up to date.

When establishing an SSL/TLS session, clients can use OCSP to check the revocation status of the authentication certificate. The authenticating client sends a request containing the serial number of the certificate to the OCSP responder (server). The responder searches the database of the certificate authority (CA) that issued the certificate and returns response containing the status (good, revoked or unknown) to the client.

OCSP server certificates

The OCSP server must have the following certificates:

- Self-certificate (system cert) signed by the issuer (CA authority)
- Root certificate who signed the system certificate
- Root certificate of the client who is trying to initiate the connection

The OCSP protocol is used in the SSL/TLS connection between the client and the server. Valid certificates of the issuer should be present at the server and client locations.

The client certificate is signed by the CA authority. Then the OCSP responder, running on the server, responds to OCSP requests and sends them to the OCSP responder.

OCSP responder

The OCSP responder is a server that is usually run by the certificate issuer). It is a process that runs on the OCSP server which entertains OCSP requests. Every certificate has an AIA field which has the responder address URI in it. If an AIA field is empty or the responder provided is unreachable, OCSP requests are directed to the default OCSP responder configured on the system under test.

An OCSP responder may return a signed response signifying that the certificate specified in the request is good, revoked or unknown. If it cannot process the request, it may return an error code.

OCSP configuration

OCSP is a process that runs while establishing a TLS connection. When the client/server sends its certificate to a peer, the peer directs the OCSP request to the OCSP server to validate the sender's certificate. The OCSP server responds with 'valid', 'revoked', or 'unknown' to identify the certificate's condition.

For a SSL/TLS session, a TLS profile must be present on the server. The TLS profile uses the following parameters:

- peer-auth-profiles
- cipher-suite
- elliptic-curves
- server certificate or system certificate
- root or CA certificate of the server
- root of CA certificate of the client

Certificate fingerprint

A certificate fingerprint is a hash of a certificate that has been computed from the certificate data and its signature.

Fingerprints are unique identifiers for certificates and are used in applications that require trust decisions, in configurations, and are displayed in interfaces.

A fingerprint in hash cryptography is a short key that helps to identify a longer public key. Fingerprints are used for key authentication and other elements of cryptography security, which provides efficiency with smaller data sets.

Fingerprints allow systems to check these smaller datasets for key authentication more easily ensuring that they are accessing the correct public key. Fingerprints can also be more efficiently stored.

Security certificate systems manually perform key authentication to promote best security practices. In modern systems, fingerprints help refine the way cryptography works and streamlines datasets. This is essential in public-key cryptography and in other modern types of digital security.

Two hashing algorithms are supported for certificate fingerprints:

- SHA-256: Cryptographic Hash Algorithm. A cryptographic hash (sometimes called 'digest') is a signature for a text or a data file. SHA-256 generates an almost-unique 256-bit (32-byte) signature for a text.
- SHA-1: Secure Hash Algorithm. A cryptographic hash function that takes an input and produces a 160-bit (20-byte) hash value known as a message digest. This is typically rendered as a hexadecimal number, 40 digits long

List of procedures

The following procedures are available for X.509 certificates.

- Procedure 70, “Installing a CA certificate using the CLI”
- Procedure 71, “Installing a CA certificate locally using the CLI”
- Procedure 72, “Uninstalling a CA certificate using the CLI”
- Procedure 73, “Installing a CRL using the CLI”
- Procedure 74, “Installing a CRL from a local path using the CLI”
- Procedure 75, “Uninstalling a CRL using the CLI”
- Procedure 76, “Installing a device certificate and private key using the CLI”
- Procedure 77, “Installing a device certificate from a local path using the CLI”
- Procedure 78, “Uninstalling a device certificate and private key using the CLI”
- Procedure 79, “Generating a private key and certificate signing request using the CLI”
- Procedure 80, “Enabling check-fingerprint using the CLI”
- Procedure 81, “Disabling check-fingerprint using the CLI”
- Procedure 82, “Configuring a fingerprint list using the CLI”

- Procedure 83, “Deleting a fingerprint-list using the CLI”
- Procedure 84, “Modifying the OCSP default responder URL using the CLI”
- Procedure 85, “Removing the OCSP default responder URL using the CLI”
- Procedure 86, “Modifying the OCSP state using the CLI”
- Procedure 87, “Modifying the nonce state using the CLI”
- Procedure 88, “Displaying a CA certificate using the CLI”
- Procedure 89, “Displaying a CRL using the CLI”
- Procedure 90, “Displaying a device certificate and private key using the CLI”
- Procedure 91, “Displaying all installed certificates on the system using the CLI”
- Procedure 92, “Displaying check-fingerprint and fingerprint list using the CLI”
- Procedure 93, “Adding ip-host-list entries using the CLI”
- Procedure 94, “Deleting ip-host-list entries using the CLI”
- Procedure 95, “Displaying ip-host-list entries using the CLI”
- Procedure 96, “Enabling check IP host using the CLI”
- Procedure 97, “Installing a CA certificate using the YANG model”
- Procedure 98, “Installing a CA certificate locally using the YANG model”
- Procedure 99, “Uninstalling a CA certificate using the YANG model”
- Procedure 100, “Installing a CRL using the YANG model”
- Procedure 101, “Uninstalling a CRL using the YANG model”
- Procedure 102, “Installing a device certificate and private key using the YANG model”
- Procedure 103, “Installing a device certificate locally using the YANG model”
- Procedure 104, “Uninstalling a device certificate and private key using the YANG model”
- Procedure 105, “Generating a private key and certificate signing request using the YANG model”
- Procedure 106, “Enabling check-fingerprint using the YANG model”
- Procedure 107, “Disabling check fingerprint using the YANG model”
- Procedure 108, “Configuring a fingerprint-list using the YANG model”
- Procedure 109, “Deleting a fingerprint-list using the YANG model”

- Procedure 110, “Modifying the OCSP default responder URL using the YANG model”
- Procedure 111, “Removing the OCSP default responder URL using the YANG model”
- Procedure 112, “Modifying the OCSP state using the YANG model”
- Procedure 113, “Modifying the nonce state using the YANG model”
- Procedure 114, “Retrieving a CA certificate using the YANG model”
- Procedure 115, “Retrieving a CRL using the YANG model”
- Procedure 116, “Retrieving a device certificate and private key using the YANG model”
- Procedure 117, “Retrieving check-fingerprint and the finger-print list using the YANG model”
- Procedure 118, “Troubleshooting PKIX errors using the CLI”
- Procedure 119, “Troubleshooting TLS errors using the CLI”
- Procedure 120, “Troubleshooting TLS handshake errors using the CLI”

Procedure 70 Installing a CA certificate using the CLI

Install a CA certificate to validate peer certificates.

Steps

1. Install a CA certificate:

```
pkix-ca install ca-cert-name <ca_cert_name> remote-file-
uri scp://<server_ip>/<cert_path>/<ca.cert> login-id
<login_id> password <login_password>
```

Example

The following example installs a CA certificate named test.

```
pkix-ca install ca-cert-name test remote-file-uri scp://192.0.2.0/certs/
SaosCertificate.pem login-id User1 password abc
```

Procedure 71 Installing a CA certificate locally using the CLI

Install a CA certificate from a local path at an authenticator device under test (DUT).

Requirements

The CA certificate must be downloaded to a local directory on DUT.

Overview

Certificates are downloaded at a local path on DUT:

```
file:///mnt/config/pkix/newcerts/
```

Locally downloaded certificates are intended for installation only once. After the certificates are download and installed, they are no longer present at the local path.

If certificates are downloaded, but not installed on the system, they are retained even after system reboots, upgrades, or container restarts, except for RTFD or Open Network Install Environment (ONIE) installations.

Steps

1. Install a CA certificate from a local path on DUT:

   ```
   pkix-ca install ca-cert-name <cert_name> remote-file-uri
   file:///mnt/config/pkix/newcerts/<ca.cert.pem>
   ```

   Example

   The following example installs a CA certificate named Testca from a local path on DUT.

```
pkix-ca install ca-cert-name Testca remote-file-uri file:///mnt/config/pkix/
newcerts/<ca.cert.pem>
```

Procedure 72 Uninstalling a CA certificate using the CLI

Uninstall a CA certificate when it is no longer required.

Steps

1 Uninstall a CA certificate:

```
pkix-ca uninstall ca-cert-name <ca.cert>
```

Example

The following example uninstalls a CA certificate named testCa.

```
pkix-ca uninstall ca-cert-name testCa
```

Procedure 73 Installing a CRL using the CLI

Install a CRL to remove digital certificates that are nearing expiration and should no longer be used. A CRL contains a list of digital certificates have been revoked by the Certificate Authority.

Steps

1 Install a CRL:

```
pkix-crl-install crl-name <crl_name>
```

Example

The following example installs a CRL named TestCrl.

```
pkix-crl-install crl-name TestCrl
```

Procedure 74 Installing a CRL from a local path using the CLI

Install a certificate revocation list (CRL) from a local path on DUT.

Requirements

The CRL certificate must be downloaded to a local directory on DUT.

Overview

Certificates are downloaded at a local path on DUT:

```
file:///mnt/config/pkix/newcerts/
```

Locally downloaded certificates are intended for installation only once. After the certificates are download and installed, they are no longer present at the local path.

If certificates are downloaded, but not installed on the system, they are retained even after system reboots, upgrades, or container restarts, except for RTFD or Onie installations.

Steps

1 Install a CRL from a local path on DUT:

```
pkix-crl install crl-cert-name <crl-cert-name> remote-
file-uri file:///mnt/config/pkix/newcerts/<crl.pem>
```

Example

The following example installs a CRL named Testcrl from a local path on DUT.

```
pkix-crl install crl-cert-name Testcrl remote-file-uri file:///mnt/config/
pkix/newcerts/<crl.pem>
```

Procedure 75 Uninstalling a CRL using the CLI

Uninstall a CRL when all certificates in the list are expired.

Steps

1 Uninstall a CRL:

```
pkix-crl-uninstall crl-name <crl_name>
```

Example

The following example uninstalls a CRL named TestCrl.

```
pkix-crl-uninstall crl-name TestCrl
```

Procedure 76 Installing a device certificate and private key using the CLI

Install a device certificate and private key as required by the network plan.

Steps

1 Install a device certificate and private key:

```
pkix-certificates install <cert_name> remote-file-uri
scp://<server_ip>/<cert_path>/<device.p12> login-id
<login_id> password <login_password> cert-passphrase
<cert_pass_phrase>
```

Example

The following example installs a device certificate and private key.

```
pkix-certificates install TestCa remote-file-uri scp://192.0.2.0/certs/
TestClient.p12 login-id User1 password abc cert-passphrase test
```

Procedure 77 Installing a device certificate from a local path using the CLI

Install a device certificate from a local path on DUT.

Requirements

The device certificate must be downloaded to a local directory on DUT.

Overview

Certificates can be downloaded to a local directory on DUT and installed later.

Certificates are downloaded to a local path on DUT:

```
file:///mnt/config/pkix/newcerts/
```

Locally downloaded certificates are intended for installation only once. After the certificates are download and installed, they are no longer present at the local path.

If certificates are downloaded, but not installed on the DUT, they are retained even after the system reboots, upgrades, or container restarts, except for RTFD or Open Network Install Environment (ONIE) installations.

Steps

1 Install a device certificate from a local path on DUT:

```
pkix-certificates install cert-name <certName> cert-
passphrase <cert-passphrase> cert-only <true/false>
remote-file-uri file:///mnt/config/pkix/newcerts/
<device.cert.pem>
```

Example

The following example installs a device certificate named Testca from a local path on DUT.

```
pkix-ca install cert-name Testca cert-passphrase abc cert-only <true/false>
remote-file-uri file:///mnt/config/pkix/newcerts/<ca.cert.pem>
```

Procedure 78 Uninstalling a device certificate and private key using the CLI

Uninstall a device certificate and private key if all certificates and private keys are expired.

Steps

1 Uninstall a device certificate and private key:

```
pkix-certificates uninstall cert-name <cert_name>
```

Example

The following example uninstalls a device certificate and private key named TestCa.

```
pkix-certificates uninstall cert-name TestCa
```

Procedure 79 Generating a private key and certificate signing request using the CLI

Generate a private key and certificate signing request on the system, sign the certificate externally, and install the certificate as required.

Steps

1 Generate a private key and certificate signing request on the system, sign the certificate externally, and install the certificate:

```
pkix-certificates-csr-generate cert-name <cert_name>
algorithm-identifier <algorithm-identifier> remote-file-
uri ftp://server_ip/<path>/<cert.cnf> cert-passphrase
<cert_passPhrase>
```

Example

The following example generates a private key and certificate signing request.

```
pkix-certificates-csr-generate cert-name testCsrGen algorithm-identifier
pkix-types:rsa1024 remote-file-uri ftp://1.2.3.4/certs/ClientCert.pem cert-
passphrase test
```

Procedure 80 Enabling check-fingerprint using the CLI

Enable check-fingerprint for a peer authentication profile.

Overview

The following table lists parameters for enabling check-fingerprint for a peer authentication profile.

Table 63 Parameters for enabling check-fingerprint for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile	string	Specifies the profile name.
check-fingerprint	• true • false	Enables or disables check-fingerprint.

Steps

1 Enable check-fingerprint:

```
pkix peer-auth-profiles peer-auth-profile <peer-auth-profile> check-fingerprint <true|false>
```

Example

The following example enables check-fingerprint for a peer authentication profile named baseConf.

```
pkix peer-auth-profiles peer-auth-profile baseConf check-fingerprint true
```

Procedure 81 Disabling check-fingerprint using the CLI

Disable check-fingerprint for a peer authentication profile.

Overview

The following table lists parameters for disabling check-fingerprint for a peer authentication profile.

Table 64 Parameters for disabling check-fingerprint for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile	string	Specifies the profile name.
check-fingerprint	• true • false	Enables or disables check-fingerprint.

Steps

1 Disable check-fingerprint:

```
pkix peer-auth-profiles peer-auth-profile <peer-auth-profile> check-fingerprint <true|false>
```

Example

The following example disables check-fingerprint for a peer authentication profile named baseConf.

```
pkix peer-auth-profiles peer-auth-profile baseConf check-fingerprint false
```

Procedure 82 Configuring a fingerprint list using the CLI

Configure a fingerprint-list for a peer authentication profile.

Requirements

The fingerprint-list has been created using an open SSL command:

```
SHA-1 > openssl x509 -noout -fingerprint -sha1 -inform pem
<certificate>

SHA-256 > openssl x509 -noout -fingerprint -sha256 -inform pem
<certificate>
```

Overview

If check-fingerprint is enabled and a fingerprint-list entry is not configured, then check-fingerprint is considered disabled and fingerprint validation of the peer certificate doesn't occur. This behavior supports backwards compatibility.

The following table lists parameters for configuring a fingerprint list for a peer authentication profile.

Table 65 Parameters for creating a fingerprint list for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile	string	Specifies the profile name.
fingerprint-list	• SHA-1 • SHA-256	Specifies the hashing algorithm.

Steps

1 Configure a fingerprint-list:

```
pkix peer-auth-profiles peer-auth-profile <peer-auth-
profile> fingerprint-list <SHA-1|SHA-256>
```

Example

The following example creates a fingerprint-list using the algorithm SHA-1 for a peer authentication profile named baseConf.

```
pkix peer-auth-profiles peer-auth-profile baseConf fingerprint-list
sha-1:E1:11:69:1B:92:39:62:7C:7C:E9:10:10:E8:47:48:B8:F5:B9:23:16
```

Procedure 83 Deleting a fingerprint-list using the CLI

Delete a fingerprint-list for a peer authentication profile.

Overview

The following table lists parameters for deleting a fingerprint-list for a peer authentication profile.

Table 66 Parameters for deleting a fingerprint list for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
fingerprint-list	• SHA-1 • SHA-256	Specifies the hashing algorithm.

Steps

1 Delete a fingerprint-list:

```
no pkix peer-auth-profiles peer-auth-profile <peer-auth-
profile> fingerprint-list <SHA-1|SHA-256>
```

Example

The following example deletes a fingerprint-list using the algorithm SHA-1 for a peer authentication profile named baseConf.

```
no pkix peer-auth-profiles peer-auth-profile baseConf fingerprint-list
sha-1:E7:21:4B:5F:48:70:AC:90:B5:4D:91:CA:AB:2C:55:58:A4:7F:DA:47
```

Procedure 84 Modifying the OCSP default responder URL using the CLI

Modify the OCSP default responder URL.

Overview

The following table describes the parameters for modifying the OCSP default responder URL.

Table 67 Parameters for modifying the OCSP default responder URL

Parameter	Valid values	Description
profile-name	string	Identifies the TLS profile. For more information, refer to "Creating a TLS profile using the CLI" on page 242.
URL	http://IP:port	Default OCSP responder URL. Only HTTP is supported.

Steps

1 Modify the OCSP default responder URL.

```
hello-params <profile-name> default-ocsp-responder-url
<URL>
```

Example

The following example modifies the OCSP default responder URL for the TLS profile named baseConf.

```
hello-params baseConf default-ocsp-responder-url http://203.0.113.4:80
```

Procedure 85 Removing the OCSP default responder URL using the CLI

Remove the OCSP default responder URL when it is no longer required.

Overview

The following table describes the parameter for removing the OCSP default responder URL.

Table 68 Parameter for removing the OCSP default responder URL

Parameter	Valid values	Description
profile-name	string	Identifies the profile. For more information, refer to "Creating a TLS profile using the CLI" on page 242.

Steps

1 Remove the OCSP default responder URL.

```
no hello-params <profile-name> default-ocsp-responder-
url
```

Example

The following example removes the OCSP default responder URL for the TLS profile named baseConf.

```
no hello-params baseConf default-ocsp-responder-url
```

Procedure 86 Modifying the OCSP state using the CLI

Modify the OCSP state.

Overview

The following table describes the parameters for modifying the OCSP state.

Table 69 Parameters for modifying the OCSP state

Parameter	Valid values	Description
profile-name	string	Identifies the profile. For more information, refer to "Creating a TLS profile using the CLI" on page 242.
state	enabled | disabled	Enables or disables the OCSP state.

Steps

1 Modify the OCSP state.

```
hello-params <profile-name> ocsp-state <state>
```

Example

The following example enables the OCSP state for the TLS profile named baseConf.

```
hello-params baseConf ocsp-state enabled
```

Procedure 87 Modifying the nonce state using the CLI

Modify the nonce state.

Overview

The following table describes the parameters for modifying the nonce state.

Table 70 Parameters for modifying the nonce state

Parameter	Valid values	Description
profile-name	string	Identifies the profile.
state	enabled | disabled	Enables or disables the nonce state.

Steps

1 Modify the nonce state.

```
hello-params <profile-name> nonce-state <state>
```

Example

The following example modifies the nonce state for the TLS profile named baseConf.

```
hello-params baseConfig nonce-state enabled
```

Procedure 88 Displaying a CA certificate using the CLI

Display a CA certificate to view the installed CA certificates.

Steps

1 Display a CA certificate:

```
show pkix
```

Procedure 89 Displaying a CRL using the CLI

Display a CRL to view all CRLs which are nearing expiration and should not be trusted.

Steps

1 Display a CRL:

```
show pkix
```

Procedure 90 Displaying a device certificate and private key using the CLI

Display a device certificate and private key to view device certificates and private keys. Private keys cannot be seen in the CLI: the device certificate and private key are concatenated.

Steps

1 Display a device certificate and private key:

```
show pkix
```

Procedure 91 Displaying all installed certificates on the system using the CLI

Display all installed certificates on the system.

Steps

1 Display all certificates on the system:

```
show pkix
```

Example

The following example displays all certificates installed on the system.

```
> show pkix

+---------------------- CA CERTIFICATES ----------------------+
| Name                  | Value                               |
+-----------------------+-------------------------------------+
| CA Name               | rootCert                            |
| Subject Common Name   | test2CA                             |
| Issuer Common Name    | test2CA                             |
| Valid Until           | Aug 22 07:22:29 2039 UTC (19 years) |
+-----------------------+-------------------------------------+

+---- CERTIFICATE REVOCATION LISTS ---+
| Name              | Value           |
+-------------------+-----------------+
| No Entries                          |
+-------------------+-----------------+

+-------------------- DEVICE CERTIFICATES --------------------+
| Name                  | Value                               |
+-----------------------+-------------------------------------+
| Certificate Name      | server_cert                         |
| Algorithm ID          | rsa1024                             |
| Private Key           | present                             |
| Subject Common Name   | server                              |
| Issuer Common Name    | test2CA                             |
| Valid Until           | Sep  5 07:30:35 2020 UTC (2 months) |
+-----------------------+-------------------------------------+
```

Procedure 92 Displaying check-fingerprint and fingerprint list using the CLI

Display the status of check-fingerprint and the fingerprint-list for a peer authentication profile.

Steps

1 Display the status of check-fingerprint and the fingerprint-list:

```
show tls
```

Example

The following example displays the status of check-fingerprint and the fingerprint-list for peer authentication profiles.

```
> show tls
+----------- TLS SERVICE PROFILES -----------+
| Name                    | Value            |
+-------------------------+------------------+
| Service Profile Name    | test             |
| TLS Profile Name        | tls-profile      |
| Peer Auth Profile Name  | peer-auth-profile|
| Certificate Name        | server_cert      |
+-------------------------+------------------+
+---------------------------------- PEER AUTH PROFILES ----------------------------------+
| Name               | Value                                                            |
+--------------------+------------------------------------------------------------------+
| Profile Name       | peer-auth-profile                                                |
| Check Expiry       | False                                                            |
| Check IP/Host      | False                                                            |
| Check Fingerprint  | True                                                             |
| IP/Host List       | TLS_Client_NoAia                                                 |
| Fingerprint List   | sha-1:E1:11:69:1B:92:39:62:7C:7C:E9:10:10:E8:47:48:B8:F5:B9:23:16|
+--------------------+------------------------------------------------------------------+
+-------------------------- HELLO PARAMS --------------------------+
| Name                          | Value                           |
+-------------------------------+---------------------------------+
| Profile Name                  | tls-profile                     |
| Protocol Versions             | tls-1.2                         |
| Cipher Suites                 | ecdhe-rsa-with-aes-256-gcm-sha384|
| Elliptic Curves               | secp384r1                       |
| Sess. Resumption Timeout (s)  | 3600                            |
| OCSP State                    | disabled                        |
| NONCE State                   | enabled                         |
| Default OCSP Responder URL    | -                               |
+-------------------------------+---------------------------------+
```

Procedure 93 Adding ip-host-list entries using the CLI

Add IP addresses and host names to the ip-host-list to allow the systems to connect.

Overview

Wildcards used in accordance with https://tools.ietf.org/html/rfc6125#section-6.4.3 are supported except for:

- The wildcard has to be in the left most label. For example, example.*.com is not allowed.
- The wildcard can not be in the two right most labels. For example, *.example.com is allowed but *.com is not allowed.
- Only one wildcard is allowed per entry. For example, *.*.com is not allowed.

Multiple entries can be added in the same command.

Steps

1 Add entries to ip-host-list:

```
pkix peer-auth-profiles peer-auth-profile https-peer-
auth-profile ip-host-list <ip-address|hostname> <ip-
address|hostname> <ip-address|hostname>
```

Example

The following example adds 10.33.80.81 and three host entries to the ip-host-list.

```
pkix peer-auth-profiles peer-auth-profile https-peer-auth-profile ip-host-
list 10.33.80.81 eit-21.ca.stalab.ciena.com entry1 entry2 entry3
show tls
+---------- TLS SERVICE PROFILES ----------+
| Name                    | Value          |
+-------------------------+----------------+
| Service Profile Name    | baseConf       |
| TLS Profile Name        | baseConf       |
| Peer Auth Profile Name  | baseConf       |
| Certificate Name        | testCert       |
+-------------------------+----------------+
+------------ PEER AUTH PROFILES ------------+
| Name           | Value                     |
+----------------+---------------------------+
| Profile Name   | https-peer-auth-profile   |
| Check Expiry   | True                      |
| Check IP/Host  | True                      |
| IP/Host List   | 10.33.80.81               |
|                | eit-21.ca.stalab.ciena.com|
|                | entry1                    |
|                | entry2                    |
|                | entry3                    |
+----------------+---------------------------+
```

```
+---------------------------- HELLO PARAMS ----------------------------+
| Name                          | Value                                |
+-------------------------------+--------------------------------------+
| Profile Name                  | baseConf                             |
| Protocol Versions             | tls-1.2                              |
| Cipher Suites                 | ecdhe-ecdsa-with-aes-256-gcm-sha384, |
|                               | ecdhe-ecdsa-with-aes-256-cbc-sha384, |
|                               | ecdhe-ecdsa-with-aes-128-cbc-sha256, |
|                               | rsa-with-aes-256-gcm-sha384,         |
|                               | rsa-with-aes-256-cbc-sha256,         |
|                               | rsa-with-aes-256-cbc-sha,            |
|                               | ecdhe-rsa-with-aes-128-gcm-sha256,   |
|                               | ecdhe-rsa-with-aes-128-cbc-sha256,   |
|                               | rsa-with-aes-128-gcm-sha256,         |
|                               | rsa-with-aes-128-cbc-sha,            |
|                               | rsa-with-3des-ede-cbc-sha            |
| Elliptic Curves               | secp521r1, secp384r1, secp256r1      |
| Sess. Resumption Timeout (s)  | 3600                                 |
+-------------------------------+--------------------------------------+
```

Procedure 94 Deleting ip-host-list entries using the CLI

Delete IP addresses and host names from the ip-host-list to deny the systems from connecting.

Steps

1 Delete entries from ip-host-list:

```
no pkix peer-auth-profiles peer-auth-profile peer-
profile-name ip-host-list <ip-address|hostname> <ip-
address|hostname> <ip-address|hostname>
```

Example

The following example deletes entry1 from ip-host-list.

```
no pkix peer-auth-profiles peer-auth-profile peer-profile-name ip-host-list
entry1
show tls
+----------- TLS SERVICE PROFILES ----------+
| Name                    | Value           |
+-------------------------+-----------------+
| Service Profile Name    | baseConf        |
| TLS Profile Name        | baseConf        |
| Peer Auth Profile Name  | baseConf        |
| Certificate Name        | testCert        |
+-------------------------+-----------------+
+------------ PEER AUTH PROFILES ------------+
| Name           | Value                     |
+----------------+---------------------------+
| Profile Name   | https-peer-auth-profile   |
| Check Expiry   | True                      |
| Check IP/Host  | True                      |
| IP/Host List   | 10.33.80.81               |
|                | eit-21.ca.stalab.ciena.com|
|                | entry2                    |
|                | entry3                    |
+----------------+---------------------------+
+---------------------------- HELLO PARAMS ----------------------------+
| Name                          | Value                                |
+-------------------------------+--------------------------------------+
| Profile Name                  | baseConf                             |
| Protocol Versions             | tls-1.2                              |
| Cipher Suites                 | ecdhe-ecdsa-with-aes-256-gcm-sha384, |
|                               | ecdhe-ecdsa-with-aes-256-cbc-sha384, |
|                               | ecdhe-ecdsa-with-aes-128-cbc-sha256, |
|                               | rsa-with-aes-256-gcm-sha384,         |
|                               | rsa-with-aes-256-cbc-sha256,         |
|                               | rsa-with-aes-256-cbc-sha,            |
|                               | ecdhe-rsa-with-aes-128-gcm-sha256,   |
|                               | ecdhe-rsa-with-aes-128-cbc-sha256,   |
|                               | rsa-with-aes-128-gcm-sha256,         |
|                               | rsa-with-aes-128-cbc-sha,            |
|                               | rsa-with-3des-ede-cbc-sha            |
| Elliptic Curves               | secp521r1, secp384r1, secp256r1      |
| Sess. Resumption Timeout (s)  | 3600                                 |
+-------------------------------+--------------------------------------+
```

Procedure 95 Displaying ip-host-list entries using the CLI

Display the ip-host-list to see the IP addresses and host names of systems allowed to connect.

Steps

1 Display ip-host-list entries:

```
show tls
```

Example

The following example shows the systems allowed to connect in the PEER AUTH PROFILES output.

```
show tls
+---------- TLS SERVICE PROFILES ----------+
| Name                    | Value          |
+-------------------------+----------------+
| Service Profile Name    | baseConf       |
| TLS Profile Name        | baseConf       |
| Peer Auth Profile Name  | baseConf       |
| Certificate Name        | testCert       |
+-------------------------+----------------+
+------------ PEER AUTH PROFILES ------------+
| Name           | Value                    |
+----------------+--------------------------+
| Profile Name   | https-peer-auth-profile  |
| Check Expiry   | True                     |
| Check IP/Host  | True                     |
| IP/Host List   | 10.33.80.81              |
|                | eit-21.ca.stalab.ciena.com |
|                | entry1                   |
|                | entry2                   |
|                | entry3                   |
+----------------+--------------------------+
+---------------------------- HELLO PARAMS ----------------------------+
| Name                          | Value                                |
+-------------------------------+--------------------------------------+
| Profile Name                  | baseConf                             |
| Protocol Versions             | tls-1.2                              |
| Cipher Suites                 | ecdhe-ecdsa-with-aes-256-gcm-sha384, |
|                               | ecdhe-ecdsa-with-aes-256-cbc-sha384, |
|                               | ecdhe-ecdsa-with-aes-128-cbc-sha256, |
|                               | rsa-with-aes-256-gcm-sha384,         |
|                               | rsa-with-aes-256-cbc-sha256,         |
|                               | rsa-with-aes-256-cbc-sha,            |
|                               | ecdhe-rsa-with-aes-128-gcm-sha256,   |
|                               | ecdhe-rsa-with-aes-128-cbc-sha256,   |
|                               | rsa-with-aes-128-gcm-sha256,         |
|                               | rsa-with-aes-128-cbc-sha,            |
|                               | rsa-with-3des-ede-cbc-sha            |
| Elliptic Curves               | secp521r1, secp384r1, secp256r1      |
| Sess. Resumption Timeout (s)  | 3600                                 |
+-------------------------------+--------------------------------------+
```

Procedure 96 Enabling check IP host using the CLI

Enable check IP/host after generating an ip-host-list which specifies which systems are allowed to connect.

Steps

1 Enable check IP/host:

```
pkix peer-auth-profiles peer-auth-profile peer-profile-
name check-ip-host true
```

Procedure 97 Installing a CA certificate using the YANG model

Install a CA certificate to validate peer certificates.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to install a CA certificate.

Example

The following sample RPC installs a CA certificate.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="133">
  <pkix-transfer:pkix-ca-install xmlns:pkix-transfer="http://www.ciena.com/
ns/yang/ciena-pkix-transfer">
    <pkix-transfer:ca-cert-name>testCa</pkix-transfer:ca-cert-name>
    <pkix-transfer:remote-file-uri>ftp://1.2.3.4/certs/SaosCertificate.pem</
pkix-transfer:remote-file-uri>
    <pkix-transfer:force>true</pkix-transfer:force>
  </pkix-transfer:pkix-ca-install>
</rpc>
```

Procedure 98 Installing a CA certificate locally using the YANG model

Install a CA certificate from a local path at an authenticator device under test (DUT).

Requirements

The ciena-pkix-transfer.yang model is used in this procedure.

The CA certificate must be downloaded to a local directory on DUT.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-ca-install> to install a CA certificate.

Example

The following sample RPC installs a CA certificate.

```
<rpc message-id="9" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
<pkix-ca-install xmlns="http://www.ciena.com/ns/yang/ciena-pkix-transfer">
      <ca-cert-name>CA_certificateName</ca-cert-name>
     <remote-file-uri>scp://certificatePath/CA_certificate.pem</remote-file-
uri>
      <login-id>remoteServerUserName</login-id>
      <password>remoteServerPassword</password>
</pkix-ca-install>
</rpc>
```

Procedure 99 Uninstalling a CA certificate using the YANG model

Uninstall a CA certificate if that CA certificate is no longer required.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to uninstall a CA certificate.

Example

The following RPC example uninstalls a CA certificate.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-ca-uninstall xmlns:pkix-transfer="http://
www.ciena.com/ns/yang/ciena-pkix-transfer">
    <pkix-transfer:ca-cert-name>testCa</pkix-transfer:ca-cert-name>
  </pkix-transfer:pkix-ca-uninstall>
</rpc>
```

Procedure 100 Installing a CRL using the YANG model

A CRL contains a list of digital certificates have been revoked by the Certificate Authority. These digital certificates are nearing expiration and should no longer be used.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to install a CRL.

Example

The following sample RPC installs a CRL.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-crl-install xmlns:pkix-transfer="http://www.ciena.com/
ns/yang/ciena-pkix-transfer">
    <pkix-transfer:crl-cert-name>TestCrl</pkix-transfer:crl-cert-name>
    <pkix-transfer:remote-file-uri>FTP://1.2.3.4/certs/crl.pem</pkix-
transfer:remote-file-uri>
    <pkix-transfer:force>true</pkix-transfer:force>
  </pkix-transfer:pkix-crl-install>
</rpc>
```

Procedure 101 Uninstalling a CRL using the YANG model

Uninstall a CRL if all certificates are expired.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to uninstall a CRL.

Example

The following sample RPC uninstalls a CRL.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-crl-uninstall xmlns:pkix-transfer="http://
www.ciena.com/ns/yang/ciena-pkix-transfer">
    <pkix-transfer:crl-name>TestCrl</pkix-transfer:crl-name>
  </pkix-transfer:pkix-crl-uninstall>
</rpc>
```

Procedure 102 Installing a device certificate and private key using the YANG model

Install a device certificate and private key as required by the network plan.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to install a device certificate.

Example

The following sample RPC installs a device certificate and private key.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-certificates-install xmlns:pkix-
transfer="http://www.ciena.com/ns/yang/ciena-pkix-transfer">
    <pkix-transfer:cert-name>testCert</pkix-transfer:cert-name>
    <pkix-transfer:cert-passphrase>test</pkix-transfer:cert-passphrase>
    <pkix-transfer:cert-only>false</pkix-transfer:cert-only>
    <pkix-transfer:remote-file-uri>ftp://192.0.2.0/certs/TestClient.p12</
pkix-transfer:remote-file-uri>
    <pkix-transfer:force>true</pkix-transfer:force>
  </pkix-transfer:pkix-certificates-install>
</rpc>
```

Procedure 103 Installing a device certificate locally using the YANG model

Install a device certificate from a local path on DUT.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

The device certificate must be downloaded to a local directory on DUT.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-certificates-install> to install a device certificate.

Example

The following sample RPC installs a device certificate from a local path.

```
<?xml version="1.0" encoding="UTF-8"?>
   <rpc message-id="10" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
     <pkix-certificates-install xmlns="http://www.ciena.com/ns/yang/ciena-
pkix-transfer">
       <cert-name>Device_certificateName</cert-name>
      <remote-file-uri>scp://certificatePath/Device_certificate.pem</remote-
file-uri>
       <cert-passphrase>passPhrase</cert-passphrase>
       <cert-only>false</cert-only>
       <login-id> remoteServerUserName </login-id>
       <password> remoteServerPassword </password>
     </pkix-certificates-install>
   </rpc>
```

Procedure 104 Uninstalling a device certificate and private key using the YANG model

Uninstall a device certificate and private key if all certificates and private keys are expired.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to uninstall a device certificate and private key.

Example

The following sample RPC uninstalls a device certificate and private key.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-crl-uninstall xmlns:pkix-transfer="http://
www.ciena.com/ns/yang/ciena-pkix-transfer">
    <pkix-transfer:crl-name>cienaDevCrl</pkix-transfer:crl-name>
  </pkix-transfer:pkix-crl-uninstall>
</rpc>
```

Procedure 105 Generating a private key and certificate signing request using the YANG model

Generate a private key and certificate signing request on the system, sign the certificate externally, and install the certificate as required.

Requirements

The YANG model ciena-pkix-transfer.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <pkix-transfer> to generate a private key and certificate signing request.
3. Sign the certificate externally.
4. Send an RPC <pkix-transfer to install the signed certificate.

Example

The following sample RPC generates a certificate signing request.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-certificates-csr-generate xmlns:pkix-transfer="http://
www.ciena.com/ns/yang/ciena-pkix-transfer" xmlns:pkix-types="http://
www.ciena.com/ns/yang/ciena-pkix-types">
    <pkix-transfer:cert-name>testCsrGen</pkix-transfer:cert-name>
    <pkix-transfer:algorithm-identifier>pkix-types:rsa1024</pkix-
transfer:algorithm-identifier>
    <pkix-transfer:cert-passphrase>test</pkix-transfer:cert-passphrase>
    <pkix-transfer:tls-service-profile>test</pkix-transfer:tls-service-
profile>
   <pkix-transfer:remote-file-uri>https://10.32.8.252/certs/ClientCert.cnf</
pkix-transfer:remote-file-uri>
    <pkix-transfer:force>true</pkix-transfer:force>
  </pkix-transfer:pkix-certificates-csr-generate>
</rpc>
```

The following sample RPC installs the externally signed certificate and associates it with the previously generated private key.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <pkix-transfer:pkix-certificates-install xmlns:pkix-
transfer="http://www.ciena.com/ns/yang/ciena-pkix-transfer">
    <pkix-transfer:cert-name>testCsrGen</pkix-transfer:cert-name>
    <pkix-transfer:cert-passphrase>test</pkix-transfer:cert-passphrase>
    <pkix-transfer:cert-only>true</pkix-transfer:cert-only>
    <pkix-transfer:remote-file-uri>ftp://1.2.3.4/certs/ClientCert.pem</
pkix-transfer:remote-file-uri>
    <pkix-transfer:force>true</pkix-transfer:force>
</pkix -transfer:pkix-certificates-install>
</rpc>
```

Procedure 106 Enabling check-fingerprint using the YANG model

Enable check-fingerprint for a peer authentication profile.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Overview

The following table describes the parameters for enabling a check-fingerprint for a peer authentication profile.

Table 71 Parameters for enabling a check-fingerprint for peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
check-fingerprint	true | false	Enables or disables check-fingerprint.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable check-fingerprint for a peer authentication profile.

Example

The following sample RPC enables check-fingerprint for a peer authentication profile.

```
  <?xml version="1.0" encoding="UTF-8"?>
    <rpc message-id="18" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <edit-config>
        <target>
        <running/>
        </target>
        <config>
          <pkix xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
            <peer-auth-profiles>
              <peer-auth-profile
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create">
          <peer-auth-profile-name>peer-auth-profile</peer-auth-profile-name>
                <check-fingerprint>true</check-fingerprint>
             <fingerprint-list>sha-
1:E7:21:4B:5F:48:70:AC:90:B5:4D:91:CA:AB:2C:55:58:A4:7F:DA:47</fingerprint-
list>
              </peer-auth-profile>
            </peer-auth-profiles>
          </pkix>
        </config>
      </edit-config>
    </rpc>
```

Procedure 107 Disabling check fingerprint using the YANG model

Disable check-fingerprint for a peer authentication profile.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Overview

The following table describes the parameters for disabling a check-fingerprint for a peer authentication profile.

Table 72 Parameters for disabling a check-fingerprint for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
check-fingerprint	true | false	Enables or disables check-fingerprint.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to disable check-fingerprint for the peer authentication profile.

Example

The following sample RPC disables check-fingerprint for a peer authentication profile.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pkix:pkix xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
        <pkix:peer-auth-profiles>
          <pkix:peer-auth-profile>
            <pkix:peer-auth-profile-name>baseConf</pkix:peer-auth-profile-
name>           <pkix:check-cert-expiry>true</pkix:check-cert-expiry>
            <pkix:check-ip-host>true</pkix:check-ip-host>
            <pkix:check-fingerprint>false</pkix:check-fingerprint>
           <pkix:periodic-reauthorization-max-interval>14400</pkix:periodic-
reauthorization-max-interval>
          </pkix:peer-auth-profile>
        </pkix:peer-auth-profiles>
      </pkix:pkix>
    </config>
  </edit-config>
</rpc>
```

Procedure 108 Configuring a fingerprint-list using the YANG model

Configure a fingerprint-list for a peer authentication profile.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

The fingerprint-list has been created.

Overview

The following table describes the parameters for configuring a fingerprint-list for a peer authentication profile.

Table 73 Parameters for configuring a fingerprint-list for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
fingerprint-list	SHA-1 | SHA-256	Specifies the hashing algorithm.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure a fingerprint-list for a peer authentication profile.

Example

The following sample RPC configures a fingerprint-list for a peer authentication profile.

```
<?xml version="1.0" encoding="UTF-8"?>
  <rpc message-id="18" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
    <edit-config>
      <target>
      <running/>
      </target>
      <config>
        <pkix xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
          <peer-auth-profiles>
            <peer-auth-profile
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create">
        <peer-auth-profile-name>peer-auth-profile</peer-auth-profile-name>
              <check-fingerprint>true</check-fingerprint>
            <fingerprint-list>sha-
1:E7:21:4B:5F:48:70:AC:90:B5:4D:91:CA:AB:2C:55:58:A4:7F:DA:47</fingerprint-
list>
            </peer-auth-profile>
          </peer-auth-profiles>
        </pkix>
```

```
      </config>
    </edit-config>
  </rpc>
```

Procedure 109 Deleting a fingerprint-list using the YANG model

Delete a fingerprint-list for a peer authentication profile.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Overview

The following table lists parameters for deleting a fingerprint-list for a peer authentication profile.

Table 74 Parameters for deleting a fingerprint-list for a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
fingerprint-list	SHA-1 | SHA-256	Specifies the hashing algorithm.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to delete a fingerprint-list for a peer authentication profile.

Example

The following sample RPC disables check-fingerprint for a peer authentication profile.

```
  <?xml version="1.0" encoding="UTF-8"?>
    <rpc message-id="18" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
      <edit-config>
        <target>
        <running/>
        </target>
        <config>
          <pkix xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
            <peer-auth-profiles>
              <peer-auth-profile
 xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" nc:operation="create">
          <peer-auth-profile-name>peer-auth-profile</peer-auth-profile-name>
                <check-fingerprint>false</check-fingerprint>
             <fingerprint-list>sha-
1:E7:21:4B:5F:48:70:AC:90:B5:4D:91:CA:AB:2C:55:58:A4:7F:DA:47</fingerprint-
list>
              </peer-auth-profile>
            </peer-auth-profiles>
          </pkix>
        </config>
      </edit-config>
    </rpc>
```

Procedure 110 Modifying the OCSP default responder URL using the YANG model

Modify the OCSP default responder URL.

Requirements

The YANG model ciena-tls.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to modify the OCSP default responder URL.

Example

The following sample RPC modifies the OCSP default responder URL for the TLS profile named baseConf.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <hello-params xmlns="http://www.ciena.com/tls/yang/ciena-tls"
xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <profile-name>baseConf</profile-name>
        <default-ocsp-responder-url>http://203.0.113.4:8090</default-ocsp-
responder-url>
      </hello-params>
    </config>
  </edit-config>
</rpc>
```

Procedure 111 Removing the OCSP default responder URL using the YANG model

Remove the OCSP default responder URL when it is no longer required.

Requirements

The YANG model ciena-tls.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to remove the OCSP default responder URL.

Example

The following sample RPC removes the OCSP default responder URL for the TLS profile named baseConf.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <hello-params xmlns="http://www.ciena.com/tls/yang/ciena-tls"
xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <profile-name>baseConf</profile-name>
        <default-ocsp-responder-url nc:operation="delete"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">http://10.121.230.37:80</
default-ocsp-responder-url>
      </hello-params>
    </config>
  </edit-config>
</rpc>
```

Procedure 112 Modifying the OCSP state using the YANG model

Modify the OCSP state.

Requirements

The YANG model ciena-tls.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to modify the OCSP state.

Example

The following sample PRC modifies the OCSP state for the TLS profile named baseConf.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <hello-params xmlns="http://www.ciena.com/tls/yang/ciena-tls"
                    xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <profile-name>baseConf</profile-name>
        <ocsp-state>enabled</ocsp-state>
      </hello-params>
    </config>
  </edit-config>
</rpc>
```

Procedure 113 Modifying the nonce state using the YANG model

Modifying the nonce state.

Requirements

The YANG model ciena-tls.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to modify the nonce state.

Example

The following sample RPC modifies the nonce state for the TLS profile named baseConf.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <hello-params xmlns="http://www.ciena.com/tls/yang/ciena-tls"
xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <profile-name>baseConf</profile-name>
        <nonce-state>enabled</nonce-state>
      </hello-params>
    </config>
  </edit-config>
</rpc>
```

Procedure 114 Retrieving a CA certificate using the YANG model

Retrieve a CA certificate to view the installed CA certificates.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve a CA certificate.

Example

The following sample RPC retrieves a CA certificate.

```
<get>
  <filter type="subtree">
    <pkix:pkix-state xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
      <pkix:ca-certificates/>
    </pkix:pkix-state>
  </filter>
</get>
```

The following example shows the output from retrieving the CA certificate.

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
      xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
  <pkix-state xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
    <ca-certificates>
      <ca-certificate>
        <ca-name>testCa</ca-name>
        <subject-common-name>SaosCertificate</subject-common-name></subject-
common-name>
        <issuer-common-name>MyCA</issuer-common-name>
       <valid-not-after>Feb 20 20:10:25 2028 GMT (9 years)</valid-not-after>
      </ca-certificate>
    </ca-certificates>
  </pkix-state>
</data>
```

Procedure 115 Retrieving a CRL using the YANG model

Retrieve a CRL to view all CRLs which are nearing expiration and should not be trusted.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve a CRL.

Example

The following sample RPC retrieves a CRL.

```
<get>
  <filter type="subtree">
    <pkix:pkix-state xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
      <pkix:crls>
        <pkix:crl/>
      </pkix:crls>
    </pkix:pkix-state>
  </filter>
 </get>
```

The following example shows the output from retrieving a CRL.

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
      xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
  <pkix-state xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
    <crls>
      <crl>
        <crl-name>cienaDevCrl</crl-name>
        <issuer-common-name>Sample Signer Cert</issuer-common-name>
        <last-update>Feb 18 10:32:00 2013 GMT</last-update>
        <revoked-cert-count>5</revoked-cert-count>
      </crl>
    </crls>
  </pkix-state>
</data>
```

Procedure 116 Retrieving a device certificate and private key using the YANG model

Retrieve a device certificate and private key to view device certificates and private keys.

Requirements

The YANG model ciena-pkix.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve a device certificate and private key.

Example

The following example retrieves device certificates and private keys.

```
<get>
  <filter type="subtree">
    <pkix:pkix-state xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
      <pkix:device-certificates/>
    </pkix:pkix-state>
  </filter>
</get>
```

The following example shows the output from retrieving a device certificate.

```
<?xml version="1.0" encoding="utf-8"?>
<data xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
      xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
  <pkix-state xmlns="http://www.ciena.com/ns/yang/ciena-pkix">
    <device-certificates>
      <device-certificate>
     <device-certificate>      <certificate-name>testCert</certificate-name>
       <algorithm-identifier xmlns:pkix-types="http://www.ciena.com/ns/yang/
ciena-pkix-types">pkix-types:rsa2048</algorithm-identifier>
        <private-key>present</private-key>
        <subject-common-name>ValimarDevClient</subject-common-name>
        <issuer-common-name>ValimarDevCA</issuer-common-name>
       <valid-not-after>Feb 20 20:41:53 2028 GMT (9 years)</valid-not-after>
      </device-certificate>
    </device-certificates>
  </pkix-state>
</data>
```

Procedure 117 Retrieving check-fingerprint and the finger-print list using the YANG model

Retrieve the status of check-fingerprint and the fingerprint-list for a peer authentication profile.

Requirements

The ciena-pkix.yang model is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve the status of check-fingerprint and the fingerprint-list.

Example

The following example shows the output from retrieving the status of check-fingerprint and the fingerprint-list.

```
"yuma-netconf:rpc-reply": {
    "yuma-netconf:data": {
      "ciena-pkix:pkix": {
        "peer-auth-profiles": {
          "peer-auth-profile": [
            {
              "peer-auth-profile-name":"peer-auth-profile",
              "check-cert-expiry":false,
              "check-ip-host":false,
              "check-fingerprint":true,
              "periodic-reauthorization-max-interval":16000,
              "ip-host-list": [
                "TLS_Client_NoAia"],
              "fingerprint-list": [
"sha1:E1:11:69:1B:92:39:62:7C:7C:E9:10:10:E8:47:48:B8:F5:B9:23:16"
```

Procedure 118 Troubleshooting PKIX errors using the CLI

Troubleshoot PKIX failures by viewing the PKIX and certificate errors as required.

Overview

A number of errors can display while downloading and installing certificates and establishing a TLS connection.

Debug errors provide pertinent information to troubleshoot and debug the root cause of TLS connection failures.

Steps

1 Display PKIX errors:

```
log view troubleshoot
```

Example

The following example illustrates a scenario in which the user is attempting to download a file named https-device-cert-dummy from the HTTPS server using a TLS profile.

```
>pkix-certificates install cert-name https-device-cert-dummy cert-only false
remote-file-uri https://10.33.80.81/source/staRadSecClientCert.p12 cert-
passphrase test tls-service-profile https-tls-service-profile

Error:Certificate verification error: The certificate chain could be built up
using the untrusted certificates but the root could not be found locally. Error
#19 Certificate: client
```

The configuration is not correct as the CA certificate has not been downloaded on DUT.

Expected Error: To establish a TLS connection with a successful HTTPS operation, a CA certificate of the received/sender's certificate is required. A PKIX error would occur.

```
>show pkix
+---------- CA CERTIFICATES ----------+
| Name             | Value            |
+------------------+------------------+
| No Entries                          | {A CA certificate is not available.}
+------------------+------------------+

+---- CERTIFICATE REVOCATION LISTS ---+
| Name             | Value            |
+------------------+------------------+
| No Entries                          |
+------------------+------------------+

+------------------- DEVICE CERTIFICATES -------------------+
| Name                | Value                                |
```

```
+---------------------+-------------------------------------+
| Certificate Name    | https-device-cert                   |
| Algorithm ID        | rsa1024                             |
| Private Key         | present                             |
| Subject Common Name | client                              |
| Issuer Common Name  | test2CA                             |
| Valid Until         | Sep  5 07:28:30 2020 UTC (2 months) |
+---------------------+-------------------------------------+

>log view troubleshoot
ERROR 2020-06-25 08:44:31.495660 cn-node-pkix Certificate verification error:
The certificate chain could be built up using the untrusted certificates but
the root could not be found locally. Error #19 Certificate: client
```

Procedure 119 Troubleshooting TLS errors using the CLI

Troubleshoot TLS errors by viewing the TLS errors as required.

Overview

A number of errors can display while downloading and installing certificates and establishing a TLS connection.

Debug errors provide pertinent information to troubleshoot and debug the root cause of TLS connection failures.

Steps

1 Display TLS errors:

```
log view troubleshoot
```

Examples

The following example illustrates a TLS error.

This example illustrates a scenario in which the user is trying to download a file named https-device-cert-dummy from the HTTPs server using a TLS profile.

```
> pkix-certificates install cert-name https-device-cert-dummy cert-only false
remote-file-uri https://10.33.80.81/source/staRadSecClientCert.p12 cert-
passphrase test tls-service-profile https-tls-service-profile

Error:IP or Hostname Mismatch Certificate: server client
```

The domain/IP value on DUT under check-ip-host 100.100.100.100 is configured incorrectly.

Expected Error: For a TLS connection to be established and the HTTPS operation to be successful, the ip-host list should have a value configured equal to the value present inside the sender's certificate under the SAN field. A TLS error would occur.

```
> show tls
+--------------- TLS SERVICE PROFILES ---------------+
| Name                    | Value                    |
+-------------------------+--------------------------+
| Service Profile Name    | https-tls-service-profile |
| TLS Profile Name        | https-tls-profile         |
| Peer Auth Profile Name  | https-peer-auth-profile   |
| Certificate Name        | https-device-cert         |
+-------------------------+--------------------------+

+------------- PEER AUTH PROFILES ------------+
| Name               | Value                  |
+--------------------+------------------------+
| Profile Name       | https-peer-auth-profile |
| Check Expiry       | False                   |
| Check IP/Host      | True                    |
| Check Fingerprint  | -                       |
```

```
| IP/Host List       | 100.100.100.100          |
| Fingerprint List   | -                        |
+--------------------+--------------------------+

+------------------------- HELLO PARAMS -------------------------+
| Name                           | Value                         |
+--------------------------------+-------------------------------+
| Profile Name                   | https-tls-profile             |
| Protocol Versions              | tls-1.2                       |
| Cipher Suites                  | ecdhe-rsa-with-aes-128-cbc-sha |
| Elliptic Curves                | secp256r1                     |
| Sess. Resumption Timeout (s)   | 3600                          |
| OCSP State                     | disabled                      |
| NONCE State                    | enabled                       |
| Default OCSP Responder URL     | -                             |
+--------------------------------+-------------------------------+

> log view troubleshoot
ERROR 2020-06-25 08:53:13.574737 cn-node-pkix IP or Hostname Mismatch
Certificate: server client
```

Procedure 120 Troubleshooting TLS handshake errors using the CLI

Troubleshoot TLS handshake errors by viewing the TLS handshake errors as required.

Overview

A number of errors can display while downloading and installing certificates and establishing a TLS connection.

Debug errors provide pertinent information to troubleshoot and debug the root cause of TLS handshake errors.

Steps

1 Display TLS handshake errors:

```
log view troubleshoot
```

Example

The following example illustrates an OCSP Error.

This example illustrates a scenario in which the user is trying to download a file named https-device-cert-dummy from the HTTPs server using a TLS profile.

```
> pkix-certificates install cert-name https-device-cert-dummy cert-only false
remote-file-uri https://10.33.80.81/source/staRadSecClientCert.p12 cert-
passphrase test tls-service-profile https-tls-service-profile

Error:Certificate verification error: OCSP Response Failed Verification:
certificate verify error Error #501 Certificate: client
```

The OCSP is configured, but the OCSP responder is not configured. When the OCSP is configured, a request is sent to the OCSP responder to query if the certificate received from the client is valid.

Expected error: For a TLS connection to be established and the HTTPS operation to be successful, the OCSP responder needs to respond with the message: Valid client certificate.

```
> show tls
+--------------- TLS SERVICE PROFILES ---------------+
| Name                    | Value                    |
+-------------------------+--------------------------+
| Service Profile Name    | https-tls-service-profile |
| TLS Profile Name        | https-tls-profile         |
| Peer Auth Profile Name  | https-peer-auth-profile   |
| Certificate Name        | https-device-cert         |
+-------------------------+--------------------------+

+-------------- PEER AUTH PROFILES ------------+
| Name               | Value                  |
+--------------------+------------------------+
```

```
| Profile Name       | https-peer-auth-profile |
| Check Expiry       | False                   |
| Check IP/Host      | True                    |
| Check Fingerprint  | -                       |
| Fingerprint List   | -                       |
+--------------------+-------------------------+

+------------------------- HELLO PARAMS -------------------------+
| Name                           | Value                         |
+--------------------------------+-------------------------------+
| Profile Name                   | https-tls-profile             |
| Protocol Versions              | tls-1.2                       |
| Cipher Suites                  | ecdhe-rsa-with-aes-128-cbc-sha |
| Elliptic Curves                | secp256r1                     |
| Sess. Resumption Timeout (s)   | 3600                          |
| OCSP State                     | enabled                       |
| NONCE State                    | enabled                       |
| Default OCSP Responder URL     | -                             |
+--------------------------------+-------------------------------+


> log view troubleshoot
ERROR 2020-06-25 09:01:38.035489 cn-node-pkix Ocsp Error: OCSP Response Failed
Verification
ERROR 2020-06-25 09:01:38.035674 cn-node-pkix Ocsp Error: OCSP Response Failed
Verification
ERROR 2020-06-25 09:01:38.037215 cn-node-pkix Certificate verification error:
OCSP Response Failed Verification: certificate verify error Error #501
Certificate: client
```

CHAPTER 5
802.1x

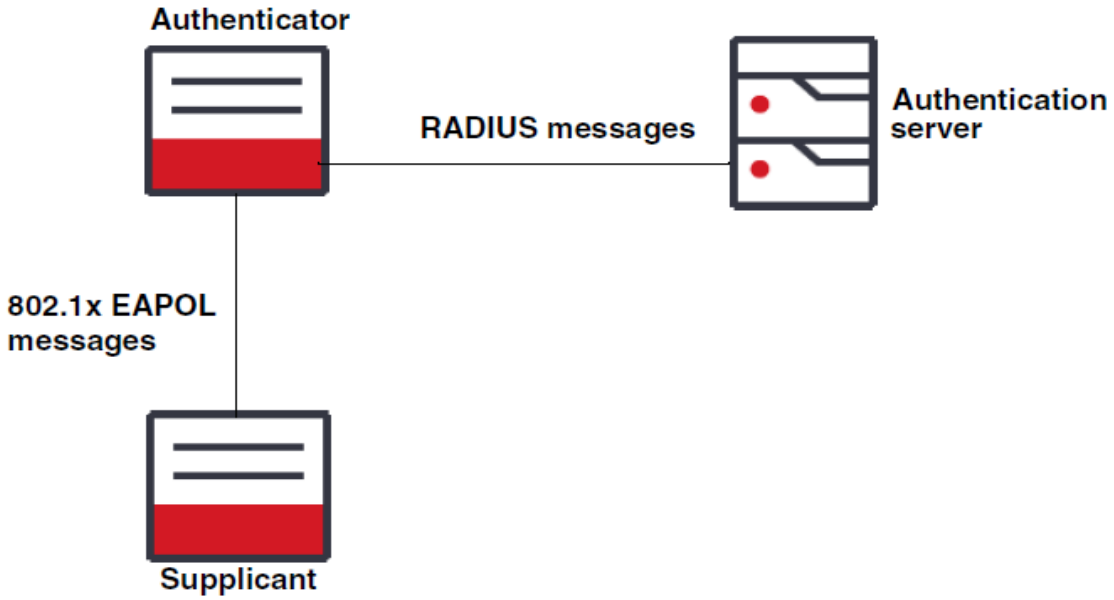
The IEEE 802.1x-2010 standard defines an authentication protocol that uses a centralized authentication server (typically a RADIUS server) to provide port-based and user-based network access control. This provides a method for authenticating customer premise equipment (CPE) and the system is used to provide the CPE network connection.

When a system configured for 802.1x authentication is connected to the network, it passes an authentication request to the system provides its uplink. That system then passes the request through the network to the authentication server, which compares the user's credentials to a pre-entered subscriber database entry and determines whether to allow the system full access to the network.

The use of 802.1x with RADIUS authentication differs from standard RADIUS management authentication. 802.1x uses port-based authentication to authenticate users who are attempting to access a system to change or monitor its configuration. The same RADIUS server can be used for authentication in both instances.

The following figure shows an example of 802.1x authentication.

Figure 4 IEEE 802.1x authentication example



802.1x roles

The 802.1x standard specifies these roles in the authentication process:

- “Supplicant” on page 174
- “Authenticator” on page 175
- “Authentication server” on page 175

Supplicant

The supplicant is the system that requests access to the network. The supplicant can be a subscriber system, such as a PC, or a port on a system that is connected to another system that provides its uplink. In the case of a PC, the PC’s network interface card (NIC) is configured for 802.1x authentication using EAP-MD5 or EAP-TLS. When the NIC is enabled, it issues an 802.1x authentication request to a port on the system. As part of the authentication request, it provides its uplink which is configured as an 802.1x authenticator. Until the supplicant is successfully authenticated, only extensible application protocol over LAN (EAPOL) messages from the supplicant are accepted by the authenticator port on the uplink system, while other data packets are dropped.

Note: If the supplicant is configured to use DHCP to obtain an IP address, the DHCP request is not passed to the DHCP server until after the supplicant has successfully authenticated. If the supplicant does not authenticate, it does not receive an IP address, which prevents it from

being reached by using an uplink or a subscriber connection. A direct serial port connection to the supplicant system is then required to correct the problem.

Authenticator

The authenticator acts as an intermediary between the 802.1x supplicant and the RADIUS authentication server. It receives the EAPOL formatted authentication request from the supplicant, encapsulates the authentication request into a RADIUS message, and passes the authentication request to the RADIUS authentication server. The response from the authentication server is sent back to the authenticator, which forwards the response to the supplicant. The authenticator also uses the RADIUS response to determine whether to begin passing regular data traffic to and from the supplicant.

The following table describes that the port, when it's acting as the authenticator, can be configured to respond to 802.1x frames.

Table 75 Authenticator port configurations

Configuration	Description
Auto	Provides 802.1x operation on a port, allowing only EAPOL frames to be sent and received until the client successfully authenticates. Once authenticated, regular traffic is allowed.
Force Authorized	Disables 802.1x and puts the port is an authorized state. The port transmits and receives normal traffic without 802.1x-based authentication of the client.
Force Unauthorized	Causes all communications from an 802.1x client to be blocked, preventing the client from authenticating through this port.

Authentication server

The authentication server receives RADIUS authentication requests sent from the authenticator, determines whether the user's credentials are in its subscriber data base, and responds with a message to allow or deny network access to the supplicant. For the authentication to succeed, the authentication server must be configured to accept the same encryption type as the supplicant.

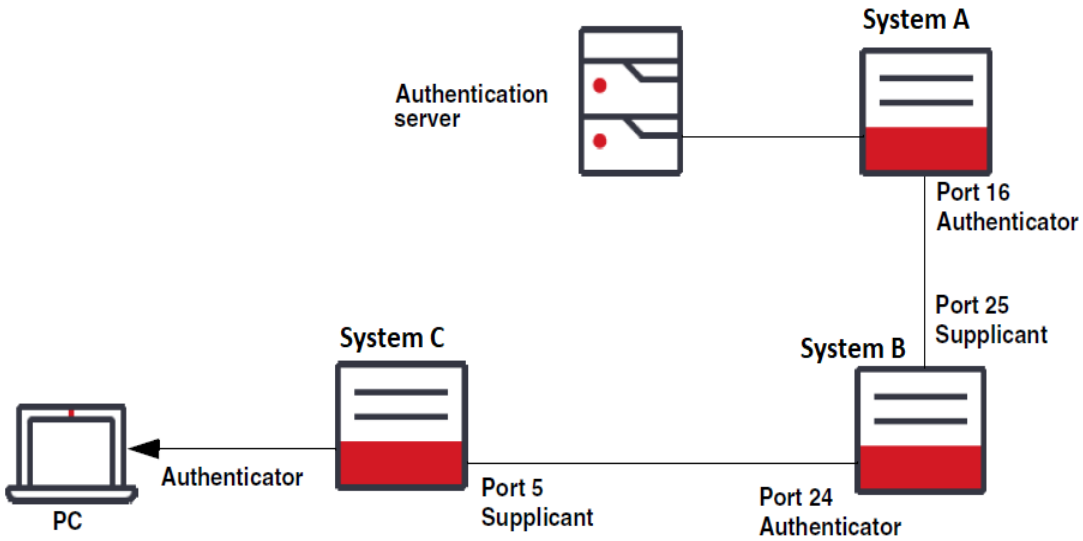
The authentication server is a third-party system that is separately controlled by the network administrator, but must be accessible by the authenticator to authenticate the supplicant. The user names and credentials used by supplicants must be configured on the server. The server must also be configured to use EAP-MD5 or EAP-TLS authentication.

Most authentication servers can be configured to allow multiple supplicants to use the same user credentials to authenticate, but can also require each supplicant to have a unique user name or password.

Deployment example

The following figure illustrates a sample network topology where 802.1x is used. System A has a connection to an authentication server that is configured with a list of user names and passwords. Port 16 on System A is configured as an authenticator, and is connected to port 25 on System B. System B is configured as a supplicant. When System B is connected to System A and is powered on, System B sends out EAPOL messages to System A to begin the authentication process. System A forms the EAPOL messages into a RADIUS message and forwards the request to the authentication server. Once authenticated, port 16 on System A allows regular traffic to ingress from port 25 on System B. If port 25 on System B fails to authenticate, regular traffic from that port is blocked: traffic from downstream systems is blocked from reaching the network.

Figure 5 IEEE 802.1x deployment example



After port 25 on System B has successfully authenticated, it can pass data from downstream systems that receive their uplink from that port, such as System C connected to port 24. When System C connected to System B is powered on, it sends EAPOL messages out port 5 to port 24 on System B, which in turn forms the message into a RADIUS message and forwards it upstream to the authentication server. Once System C has successfully authenticated, port 24 on System B allows regular traffic from System C to ingress that port.

If a PC is connected to a subscriber port on System C, the same 802.1x process can be used to authenticate the PC and unblock the port on System C to provide network access.

802.1x on aggregation ports

802.1x operation controls the state of physical ports: it allows or denies a port access to the system based on its authentication state. Ports that are configured as 802.1x supplicants which are also members of a link aggregation group must be authenticated to pass traffic as part of that link aggregation group. If a port that is a member of a link aggregation group becomes unauthenticated during operation, it is removed from the distribution list.

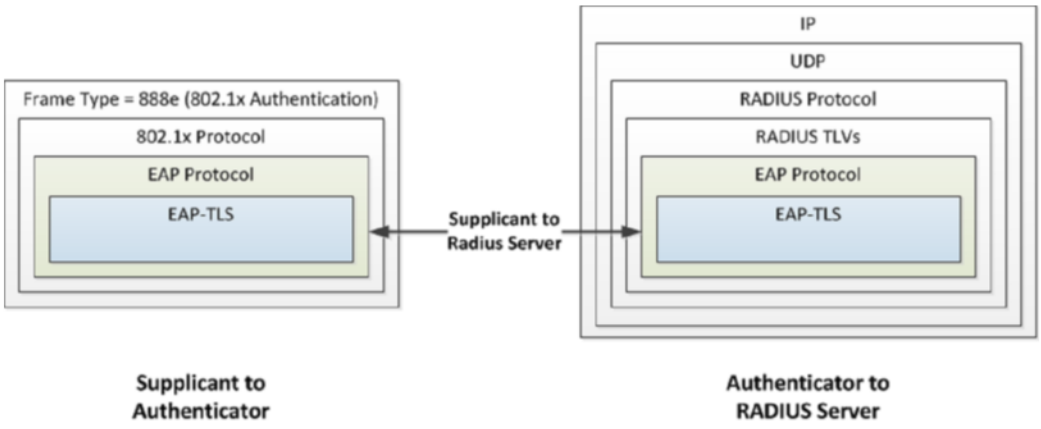
Authentication verification

The Port Access Entity (PAE) state for the supplicant and the authenticator monitor the current state of authentication. When the supplicant has authenticated, both systems indicate that state.

EAP-TLS

Extensible Authentication Protocol Transport Layer Security (EAP-TLS) is an Internet Engineering Task Force (IETP) open standard that uses the TLS protocol for authentication. EAP-TLS messages are carried inside EAP messages carried in 802.1x frames or RADIUS UDP packets as shown in the following figure.

Figure 6 EAP-TLS messages



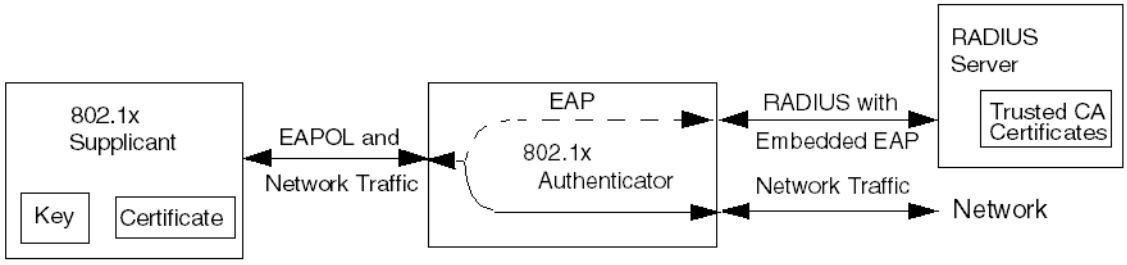
EAP-TLS provides a significantly higher level of security than EAP-MD5. EAP-MD5 uses a weak password hash and only provides authentication of the EAP peer to the EAP server, but no mutual authentication. As a result, EAP-MD5 is vulnerable to attacks. EAP-TLS uses X.609 certificates to authenticate the supplicant. The supplicant can optionally verify the identify of the RADIUS authentication server.

EAP-TLS supplicant authentication

In 802.1x, EAP-TLS is the supplicant and must have a system certificate and private key and passphrase installed. The server is the RADIUS server. It must have a copy of the root certificate. Intermediate certificates are committed for simplicity.

The following figure describes the authentication of the EAP-TLS supplicant. During authentication, the supplicant presents its system certificate to the server which validates it against its store of trusted certificate authority (CA) certificates. The server can optionally be configured to validate system certificates against certificate revocation lists (CRLs).

Figure 7 Certificate configuration for supplicant authentication

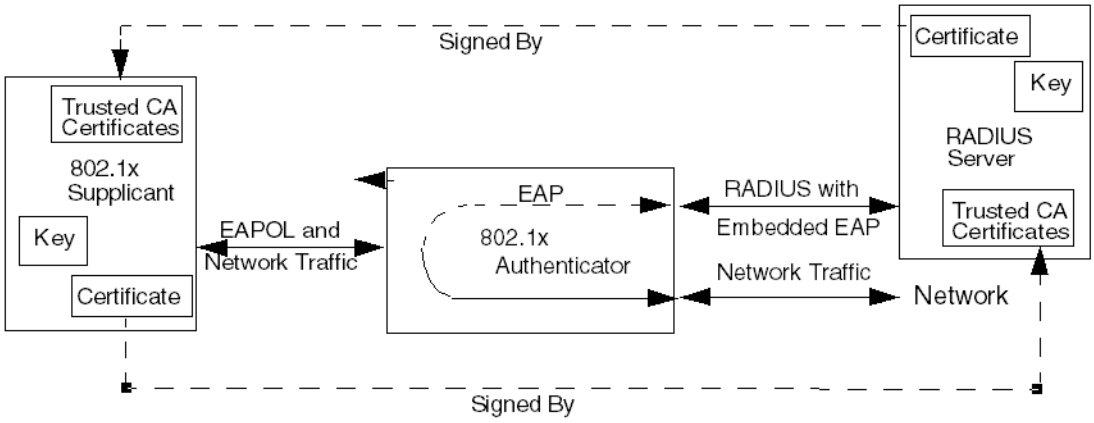


EAP-TLS mutual authentication

If the supplicant is required to authenticate the server, the server must have a private key and passphrase and signed certificate. The supplication must also have a copy of the root certificate that signed it. During mutual authentication, the supplicant and server exchange their system certificates and each validates the other's against its trusted CA certificates and optionally a CRL.

The following figure describes how NTE authenticates the server.

Figure 8 Supplicant EAP-TLS mutual authentication



In this example, the RADIUS server provides its private key to provide ownership of the certificate. The supplicant verifies the certificate's signature against the chain of certificates back to the root certificate it holds and trusts.

List of procedures

The following procedures are available for 802.1x.

- Procedure 121, "Enabling 802.1x authentication using the CLI"
- Procedure 122, "Disabling 802.1x authentication using the CLI"
- Procedure 123, "Re-authenticating the supplicant using the CLI"
- Procedure 124, "Initializing a port using the CLI"
- Procedure 125, "Configuring a supplicant port in MD5 mode using the CLI"
- Procedure 126, "Configuring a supplicant port in TLS mode using the CLI"
- Procedure 127, "Configuring an authenticator port in MD5 mode using the CLI"
- Procedure 128, "Creating and attaching a server group for the RADIUS server to use in authentication using the CLI"
- Procedure 129, "Clearing all 802.1x port statistics using the CLI"
- Procedure 130, "Clearing specific port statistics using the CLI"

- Procedure 131, “Displaying 802.1x ports using the CLI”
- Procedure 132, “Displaying information about all authenticator ports using the CLI”
- Procedure 133, “Displaying information about a specific authenticator port using the CLI”
- Procedure 134, “Displaying information about all supplicant ports using the CLI”
- Procedure 135, “Displaying information about a specific supplicant port using the CLI”
- Procedure 136, “Displaying 802.1x global information using the CLI”
- Procedure 137, “Displaying 8021x authenticator statistics using the CLI”
- Procedure 138, “Displaying 802.1x authenticator statistics for a specific port using the CLI”
- Procedure 139, “Displaying 802.1x supplicant statistics using the CLI”
- Procedure 140, “Displaying 802.1x supplicant statistics for a specific port using the CLI”
- Procedure 141, “Enabling 802.1x using the YANG model”
- Procedure 142, “Disabling 802.1x using the YANG model”
- Procedure 143, “Re-authenticating the supplicant using the YANG model”
- Procedure 144, “Initializing a port using the YANG model”
- Procedure 145, “Configuring a port as a supplicant or an authenticator using the YANG model”
- Procedure 146, “Setting parameters on a supplicant port using the YANG model”
- Procedure 147, “Setting parameters on an authenticator port using the YANG model”
- Procedure 148, “Attaching a tls-service profile to the supplicant port using the YANG model”
- Procedure 149, “Creating and attaching a server group for RADIUS configuration on an authenticator DUT using the YANG model”
- Procedure 150, “Clearing all 802.1x port statistics using the YANG model”
- Procedure 151, “Clearing specific port statistics using the YANG model”
- Procedure 152, “Retrieving 802.1x port information using the YANG model”
- Procedure 153, “Retrieving information about all authenticator ports using the YANG model”

- Procedure 154, “Retrieving information about all supplicant ports using the YANG model”
- Procedure 155, “Retrieving 802.1x global information using the YANG model”
- Procedure 156, “Retrieving 802.1x authenticator statistics information using the YANG model”
- Procedure 157, “Retrieving 802.1x supplicant statistics using the YANG model”
- Procedure 158, “Retrieving 802.1x authenticator-session statistics using the YANG model”

Procedure 121 Enabling 802.1x authentication using the CLI

Enable 802.1x authentication as required by the network plan.

Steps

1 Enable 802.1x:

```
system dot1x config system-auth-control enabled
```

Example

The following example enables 802.1x.

```
system dot1x config system-auth-control enabled
```

Procedure 122 Disabling 802.1x authentication using the CLI

Disable 802.1x when 802.1x authentication is no longer required on the system.

Steps

1 Disable 802.1x:

```
system dot1x config system-auth-control disabled
```

Example

The following example disables 802.1x.

```
system dot1x config system-auth-control disabled
```

Procedure 123 Re-authenticating the supplicant using the CLI

Re-authenticate the supplicant to re-establish the connection between the supplicant and the authenticator.

Overview

The following table describes the parameter for re-authenticating the supplicant.

Table 76 Parameter for re-authenticating a port

Parameter	Valid values	Description
port name	name-string	Specifies the name of the port.

Steps

1 Re-authenticate a supplicant.

```
dot1x reauthenticate port <port name>
```

Example

The following example re-authenticates port 8 on the supplicant.

```
dot1x reauthenticate port 8
```

Procedure 124 Initializing a port using the CLI

Initialize a port to establish a connection to the authentication server.

Overview

The following table describes the parameter for initializing a port.

Table 77 Parameter for initializing a port

Parameter	Valid values	Description
port name	name-string	Specifies the name of the port.

Steps

1 Initialize a port.

```
dot1x initialize port <port name>
```

Example

The following example initializes port 8.

```
> dot1x initialize port 8
```

Procedure 125 Configuring a supplicant port in MD5 mode using the CLI

Configure a supplicant port in MD5 mode as required by the network plan.

Overview

The following table describes the parameters for configuring a supplicant port in MD5 mode.

Table 78 Parameters for configuring a supplicant port in MD5 mode

Parameter	Valid values	Description
interface	integer	Identifies the port to be configured.
admin-status	enabled disabled	Enables or disables the administrative status.
identity	name-type	Specifies the username that the supplicant will use as the identity in the authentication process.
password	password-type	Specifies the password that the supplicant will use for EAP-MD5 in the authentication process. For security, the password is displayed as xxxx.

Steps

1 Configure a port as a supplicant.

```
oc-if:interfaces interface <interface> config dot1x port-capabilities supplicant true
```

2 Enable the interface.

```
oc-if:interfaces interface <interface> config dot1x supplicant admin-status enabled
```

3 Specify the username that the supplicant will use as the identity in the authentication process.

```
oc-if:interfaces interface <interface> config dot1x supplicant identity <identity>
```

4 Specify the password that the supplicant will use for EAP-MD5 in the authentication process.

```
oc-if:interfaces interface <interface> config dot1x
supplicant password <password>
```

5 Enable system authentication control.

```
system dot1x config system-auth-control enable
```

Example

The following example configures port 27 as a supplicant port in MD5 mode.

```
oc-if:interfaces interface 27 config dot1x port-capabilities supplicant true
oc-if:interfaces interface 27 config dot1x supplicant admin-status enabled
oc-if:interfaces interface 27 config dot1x supplicant identity mysecurityid
oc-if:interfaces interface 27 config dot1x supplicant password mypassword
system dot1x config system-auth-control enabled
```

Procedure 126 Configuring a supplicant port in TLS mode using the CLI

Configure a supplicant port in TLS mode as required by the network plan.

Requirements

This procedure assumes that the CA certificate is installed and the peer-auth-profile exists. For more information, refer to "X.509 certificates" on page 107.

Overview

The following table describes the parameters for configuring a supplicant port in TLS mode.

Table 79 Parameters for configuring a supplicant port in TLS mode

Parameter	Valid values	Description
interface	integer	Identifies the port to be configured.
admin-status	enabled disabled	Enables or disables the administrative status.
identity	string	Specifies the username that the supplicant will use as the identity in the authentication process.
password	string	Specifies the password that the supplicant will use for EAP-MD5 in the authentication process. For security, the password is displayed as xxxx.
eap-method	TLS	Selects the EAP-method.
tls-service-profile	TLS service profiles created using "Creating a TLS profile using the CLI" on page 242	Specifies the TLS service profile to be used when the eap-method is set to eap-TLS.

Steps

1 Install a CA certificate.

```
pkix-ca install ca-cert-name <ca_cert_name> remote-file-
uri ftp://<server_ip>/<cert_path>/<ca.cert> login-id
<login_id> password <login_password>
```

2 Install a system certificate.

```
pkix-certificates install <cert_name> https-device-cert
remote-file-uri ftp://<server_ip>/<cert_path>/
<device.p12> cert-only <true|false> cert-passphrase
<cert_pass_phrase> login-id <login_id> password
<login_password>
```

3 Create a peer authentication profile.

```
pkix peer-auth-profiles peer-auth-profile https-peer-
auth-profile check-ip-host <true|false> check-cert-
expiry <true|false> check-fingerprint <true|false>
periodic-reauthorization-max-interval <periodic-
reauthorization-max-interval_value>
```

4 Set the TLS version.

```
hello-params https-tls-profile tls-versions tls-version
<tls-version>
```

5 Identify the TLS profile.

```
tls-service-profiles https-tls-service-profile tls-
profile-name <tls-profile-name>
```

6 Identify the peer authentication profile.

```
tls-service-profiles https-tls-service-profile tls-peer-
auth-profile-name <peer-auth-profile>
```

7 Identify the TLS certificate.

```
tls-service-profiles https-tls-service-profile tls-
certificate-name <certificate>
```

8 Configure a port as a supplicant.

```
oc-if:interfaces interface <interface> config dot1x port-
capabilities supplicant true
```

9 Enable the interface.

```
oc-if:interfaces interface <interface> config dot1x
supplicant admin-status enabled
```

10 Specify the username that the supplicant will use as the identity in the authentication process.

```
oc-if:interfaces interface <interface> config dot1x
supplicant identity <identity>
```

11 Specify the password that the supplicant will use for EAP-MD5 in the authentication process.

```
oc-if:interfaces interface <interface> config dot1x
supplicant password <password>
```

12 Set the EAP method to TLS.

```
oc-if:interfaces interface <interface> config dot1x
supplicant eap-method tls <eap-method tls>
```

13 Attach the TLS service profile.

```
oc-if:interfaces interface 27 config dot1x supplicant
tls-service-profile <TLS service profile>
```

14 Enable system authentication control.

```
system dot1x config system-auth-control enabled
```

Example

The following example configures port 27 as a supplicant port in TLS mode. This example shows the commands for installing the CA certificate, creating the peer authentication profile and creating the TLS service profile.

```
pkix-ca install ca-cert-name ca-cert-tls remote-file-uri ftp://10.33.85.130/
/home/User1/radius_adhoc/ca2.pem login-id User1 password abc
pkix-certificates install https-device-cert remote-file-uri ftp://
10.33.85.130//home/User1/radius_adhoc/client.p12 cert-only false cert-
passphrase test login-id User1 password abc
pkix peer-auth-profiles peer-auth-profile https-peer-auth-profile check-ip-
host false check-cert-expiry false check-fingerprint false periodic-
reauthorization-max-interval 600
hello-params https-tls-profile tls-versions tls-version tls-1.2
tls-service-profiles https-tls-service-profile tls-profile-name https-tls-
profile
tls-service-profiles https-tls-service-profile tls-peer-auth-profile-name
https-peer-auth-profile
tls-service-profiles https-tls-service-profile tls-certificate-name https-
device-cert
oc-if:interfaces interface 27 config dot1x port-capabilities supplicant true
oc-if:interfaces interface 27 config dot1x supplicant admin-status enabled
oc-if:interfaces interface 27 config dot1x supplicant identity mysecurityid
oc-if:interfaces interface 27 config dot1x supplicant password mypassword
oc-if:interfaces interface 27 config dot1x supplicant eap-method tls
oc-if:interfaces interface 27 config dot1x supplicant tls-service-profile
https-tls-service-profile
system dot1x config system-auth-control enabled
```

Procedure 127 Configuring an authenticator port in MD5 mode using the CLI

Configure an authentication port in MD5 mode as required by the network plan.

Overview

The following table describes the parameters for configuring an authenticator port.

Table 80 Parameters for configuring an authenticator port

Parameter	Valid values	Description
admin-status	enabled disabled	Enables or disables the administrative status.
eapol-protocol-version	1|2 The default value is 2.	Specifies the EAPOL version. Set this parameter to 1 to work around issues with client implementations that drop EAPOL frames that use version 2.
control-direction	both	Specifies the current value of the administrative controlled directions parameter for the port.

Parameter	Valid values	Description
port-control	auto force-authorized force-unauthorized	Sets the port authorization mode. auto: Provides 802.1x operation on a port, allowing only EAPOL frames to be sent and received until the client successfully authenticates. Once authenticated, regular traffic is allowed. force-authorized: Disables 802.1x and puts the port in an authorized state. The port transmits and receives normal traffic without 802.1x-based authentication of the client. force-unauthorized: Causes all communications from an 802.1x client to be blocked, preventing the client from authenticating through this port.
reauth-enabled	true false	Determines whether re-authentication is initiated. The re-authentication interval is specified with the reauth-period.
quiet-period	number <1..65535>	Specifies the number of seconds that the system remains in the quiet state following a failed authentication exchange with the client.
reauth-max	number <1..10>	Specifies the maximum number of re-authentication attempts on an authenticator port before the port is unauthorized.

Parameter	Valid values	Description
reauth-period	number <1..65535>	This object indicates the time period of the re-authentication to the supplicant.
server-timeout	number <1..180>	Specifies the timeout value for an authenticator on a port connecting to a backend authentication server.

Steps

1. Enable the interface.

   ```
   config dot1x authenticator admin-status enabled
   ```

2. Specify the eapol-version.

   ```
   config dot1x authenticator eapol-protocol-version 1|2
   ```

3. Specify the value of the controlled directions parameters.

   ```
   config dot1x authenticator control-director both
   ```

4. Set the port authorization mode.

   ```
   config dot1x authenticator port-control <auto|force-
   authorized|force-unauthorized>
   ```

5. If re-authentication is enabled, specify the re-authentication level.

   ```
   config dot1x authenticator reauth-period <1..65535>
   ```

6. Determines whether re-authentication is required.

   ```
   config dot1x authenticator reauth-enabled <true|false>
   ```

7. Specify the quiet-period.

   ```
   config dot1x authenticator quiet-period < 1..65535>
   ```

8. Specify the re-authentication attempt value.

   ```
   config dot1x authenticator reauth-max <1..10>
   ```

9. Specify the server-timeout value.

   ```
   config dot1x authenticator server-timeout <1..180>
   ```

Example

The following example configures an autheticator.

```
config dot1x authenticator admin-status enabled
config dot1x authenticator eapol-protocol-version 2
config dot1x authenticator control-director both
config dot1x authenticator port-control auto
config dot1x authenticator reauth-enabled false
config dot1x authenticator quiet-period 60
```

```
config dot1x authenticator reauth-max 2
config dot1x authenticator reauth-period 360
config dot1x authenticator server-timeout 30
```

Procedure 128 Creating and attaching a server group for the RADIUS server to use in authentication using the CLI

Create and attach a server group to configure a RADIUS server on an authenticator device under test (DUT).

Overview

The following table describes the parameters to create and attach a server group.

Table 81 Server group parameters

Parameter	Valid values	Description
name	string	Specifies the name of the server.
server-group-name	string	Specifies the name of the server group.
config-type	RADIUS	Specifies the AAA server type. All servers in the group must be of this type.
server IP address	IP address	Sets the IP address of the authentication server.
config IP address	IP address	Is the IP address of the server group.
timeout	<1... 30> seconds	Sets the timeout in seconds on responses from the AAA server.
auth-port	oc-inet:port-number	Specifies the port number for authentication requests.
secret-key	name	Specifies the secret key for RADIUS configuration.

Steps

1 Create and attach a server group for RADIUS authentication on an authenticator:

```
system aaa server-groups server-group <server-group-name>
config name <name>
```

2 Configure the server group configuration type as RADIUS:

```
system aaa server-groups-server-group <server-group-
name> config type <config-type>
```

3 Configure the IP addresses for the server group:

```
system aaa server-groups server-group authserver servers
server <server IP address> config address <config IP
address>
```

4 Enable the server group:

```
system aaa server-groups server-group authserver servers
server <IP address> config admin-state enabled
```

5 Configure the server group timeout:

```
system aaa server-groups server-group <server-group>
servers server <server IP address> config timeout
<timeout>
```

6 Configure the authentication port:

```
system aaa server-groups server-group authserver servers
server <IP address> radius config auth-port <oc-
inet:port-number>
```

7 Configure the RADIUS configuration secret key:

```
system aaa server-groups server-group authserver servers
server <server IP address> radius config secret-key
<name>
```

8 Configure the RADIUS configuration secret key:

```
system system dot1x config authentication-method
authserver
```

9 Enable system authentication control.

```
system dot1x config system-auth-control enabled
```

Example

The following example creates a server group for the RADIUS server to be used in authentication.

```
system aaa server-groups server-group authserver config name authserver
system aaa server-groups server-group authserver config type RADIUS
system aaa server-group server-group authserver servers server 10.33.85.130
config address 10.33.85.130
system aaa server-groups server-group authserver servers server 10.33.85.130
config admin-state enabled
system aaa server-groups server-group authserver servers server 10.33.85.130
config timeout 8
system aaa server-groups server-group authserver servers server 10.33.85.130
radius config auth-port 1812
system aaa server-groups server-group authserver servers server 10.33.85.130
radius config secret-key testing123
system dot1x config authentication-method authserver
system dot1x config system-auth-control enabled
```

Procedure 129 Clearing all 802.1x port statistics using the CLI

Clear all 802.1x port statistics to reset statistics collection.

Steps

1 Clear all 802.1x port statistics:

```
clear dot1x statistics
```

Procedure 130 Clearing specific port statistics using the CLI

Clear specific port statistics to reset statistics collection.

Overview

The following table describes the parameter for clearing specific port statistics.

Table 82 Parameter for clearing port statistics

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1 Clear statistics for a specific port:

```
clear dot1x statistics port <port name>
```

Example

The following example shows output for the clear dot1x statistics port command for port 8.

```
clear dot1x statistics port 8
```

Procedure 131 Displaying 802.1x ports using the CLI

Display 802.1x ports to verify the configuration.

Steps

1 Display 802.1x ports:

```
show dot1x ports
```

Example

The following example shows sample output from the show dot1x ports command.

```
> show dot1x ports
+-------------------------- DOT1X PORT SUMMARY ---------------------------+
|      | Port Role      | Admin     | Oper      | PAE          | Controlled   |
| Port | (Auth/Supp)    | State     | State     | State        | Port Status  |
+------+----------------+-----------+-----------+--------------+--------------+
| 1    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 2    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 3    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 4    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 5    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 6    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 7    | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 8    | Authenticator  | Enabled   | Disabled  | Disconnected | Unauthorized |
| 9    | Authenticator  | Disabled  | Disabled  | Disconnected | Authorized   |
| 10   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 11   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 12   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 13   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 14   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 15   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 16   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 17   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 18   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 19   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 20   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 21   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 22   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 23   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 24   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 25   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 26   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 27   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 28   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 29   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 30   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 31   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 32   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 33   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 34   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 35   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 36   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 37   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
| 38   | None           | Disabled  | Disabled  | Disconnected | Authorized   |
```

```
| 39    | None           | Disabled | Disabled | Disconnected | Authorized   |
| 40    | Supplicant     | Disabled | Disabled | Disconnected | Authorized   |
| 41    | Supplicant     | Disabled | Disabled | Disconnected | Authorized   |
| 42    | None           | Disabled | Disabled | Disconnected | Authorized   |
| 43    | None           | Disabled | Disabled | Disconnected | Authorized   |
| 44    | None           | Disabled | Disabled | Disconnected | Authorized   |
+-------+----------------+----------+----------+--------------+--------------+
```

Procedure 132 Displaying information about all authenticator ports using the CLI

Display information about all authenticator ports to verify the configuration.

Steps

1 Display information about all authenticator ports:

```
show dot1x ports auth
```

Example

The following example shows output from the show dot1x ports auth command.

```
> show dot1x ports auth
+------------- DOT1X PORT SUMMARY -------------+
| Name                       | Value           |
+----------------------------+-----------------+
| Port Name                  | 8               |
| Admin State                | Enabled         |
| Port Control               | Auto            |
| ReAuth Enabled             | False           |
| Quiet Period (sec)         | 60              |
| Server Timeout (sec)       | 30              |
| ReAuth Period (sec)        | 3,600           |
| Max Retries                | 2               |
| EAP Version                | 2               |
| Control Direction          | Both            |
| Operational State          | Disabled        |
| Controlled Port Status     | Unauthorized    |
| Last EAPOL Frame Version   | 2               |
| Last EAPOL Frame Source    | 54:c3:3e:4b:da:92 |
+----------------------------+-----------------+
| Port Name                  | 9               |
| Admin State                | Disabled        |
| Port Control               | Auto            |
| ReAuth Enabled             | False           |
| Quiet Period (sec)         | 60              |
| Server Timeout (sec)       | 30              |
| ReAuth Period (sec)        | 3,600           |
| Max Retries                | 2               |
| EAP Version                | 2               |
| Control Direction          | Both            |
| Operational State          | Disabled        |
| Controlled Port Status     | Authorized      |
| Last EAPOL Frame Version   | 0               |
| Last EAPOL Frame Source    | 00:00:00:00:00:00 |
+----------------------------+-----------------+
```

Procedure 133 Displaying information about a specific authenticator port using the CLI

Display information about a specific authenticator port to verify the configuration.

Overview

The following table describes the parameter to display information about a specific authenticator port

Table 83 Parameter for a specific authenticator port

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1 Display information about a specific authenticator port:

```
show dot1x ports auth port <port name>
```

Example

The following example shows sample output for the show dot1x ports auth port command for port 2.

```
> show dot1x ports auth port 2
+------------- DOT1X PORT SUMMARY -------------+
| Name                       | Value           |
+----------------------------+-----------------+
| Port Name                  | 2               |
| Admin State                | Enabled         |
| Port Control               | Auto            |
| ReAuth Enabled             | False           |
| Quiet Period (sec)         | 60              |
| Server Timeout (sec)       | 30              |
| ReAuth Period (sec)        | 3,600           |
| Max Retries                | 2               |
| EAP Version                | 2               |
| Control Direction          | Both            |
| Operational State          | Enabled         |
| PAE State                  | Authenticating  |
| Last EAPOL Frame Version   | 2               |
| Last EAPOL Frame Source    | 74:87:bb:c9:c0:0b |
+----------------------------+-----------------+
```

```
+------------------------------------------------------------ DOT1X PORT STATISTICS ------------------------------------------------------------+
|      | Eapol     | Eapol     | Eapol Start | Eapol Logoff | Eapol Resp | Eapol Resp | Eapol Req | Invalid Eapol | Eapol Req |
| Port | Frame Tx  | Frame Rx  | Frame Rx    | Frame Rx     | Id Rx      |  Rx        |  Tx       | Frame Rx      | Id Tx     |
+------+-----------+-----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
| 2    | 3838      | 5756      | 1919        | 0            | 1919       | 3837       | 0         | 0             | 3838      |
+------+-----------+-----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
```

Procedure 134 Displaying information about all supplicant ports using the CLI

Display information about all supplicant ports to verify the configuration.

Steps

1 Display information about all supplicant ports:

```
show dot1x ports supp
```

Example

The following example shows sample output from the show dot1x ports supp command:

```
> show dot1x ports supp
+------------- DOT1X PORT SUMMARY -------------+
| Name                       | Value           |
+----------------------------+-----------------+
| Port Name                  | 40              |
| Admin State                | Disabled        |
| Start Period (sec)         | 30              |
| Held Period (sec)          | 60              |
| Auth Period (sec)          | 30              |
| Max Start                  | 3               |
| EAP Version                | 2               |
| EAP Method                 | md5             |
| Operational State          | Disabled        |
| Controlled Port Status     | Authorized      |
| Last EAPOL Frame Version   | 0               |
| Last EAPOL Frame Source    | 00:00:00:00:00:00 |
+----------------------------+-----------------+
| Port Name                  | 41              |
| Admin State                | Disabled        |
| Start Period (sec)         | 30              |
| Held Period (sec)          | 60              |
| Auth Period (sec)          | 30              |
| Max Start                  | 3               |
| EAP Version                | 2               |
| EAP Method                 | md5             |
| Operational State          | Disabled        |
| Controlled Port Status     | Authorized      |
| Last EAPOL Frame Version   | 0               |
| Last EAPOL Frame Source    | 00:00:00:00:00:00 |
+----------------------------+-----------------+
```

Procedure 135 Displaying information about a specific supplicant port using the CLI

Display information about a specific supplicant port to verify the configuration.

Overview

The following table describes the parameter for displaying information about a specific supplicant port.

Table 84 Parameter for displaying information about a supplicant

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1 Display information about a specific supplicant port:

```
show dot1x ports supp port <port name>
```

Example

The following example shows sample output for the show dot1x ports supp port command for port 1.

```
> show dot1x ports supp port 1

+------------- DOT1X PORT SUMMARY --------------+
| Name                        | Value           |
+-----------------------------+-----------------+
| Port Name                   | 1               |
| Admin State                 | Enabled         |
| Start Period (sec)          | 30              |
| Held Period (sec)           | 60              |
| Auth Period (sec)           | 30              |
| Max Start                   | 3               |
| Username                    | ngsu            |
| Password                    | Set             |
| EAP Version                 | 2               |
| EAP Method                  | md5             |
| Operational State           | Enabled         |
| Controlled Port Status      | Unauthorized    |
| PAE State                   | Authenticating  |
| Last EAPOL Frame Version    | 2               |
| Last EAPOL Frame Source     | 74:87:bb:c9:c0:0c |
+-------------------------------------+-------------------------+

+------------------------------- DOT1X PORT STATISTICS --------------------------------+
|      | Eapol     | Eapol     | Eapol Start | Eapol Logoff | Invalid Eapol | Eapol Length |
| Port | Frame Tx  | Frame Rx  | Frame Tx    | Frame Tx     | Frame Rx      | Error Frames |
+------+-----------+-----------+-------------+--------------+---------------+--------------+
| 1    | 5756      | 3837      | 1919        | 0            | 0             | 0            |
+------+-----------+-----------+-------------+--------------+---------------+--------------+
```

Procedure 136 Displaying 802.1x global information using the CLI

Display 802.1x global information to verify global settings.

Steps

1 Display 802.1x global information:

```
show dot1x
```

Example

The following example shows sample output from the show dot1x command.

```
> show dot1x
+-------- DOT1X GLOBAL INFO ---------+
| Name                   | Value      |
+------------------------+------------+
| Admin Status           | Enabled    |
| Authentication Method  | authserver |
| Operation State        | Enabled    |
+------------------------+------------+
```

Procedure 137 Displaying 8021x authenticator statistics using the CLI

Display 802.1x authenticator statistics to view performance information.

Steps

1 Display 802.1x authenticator statistics.

```
show dot1x ports auth statistics
```

Example

The following example shows the output from the show dot1x ports auth statistics command.

```
> show dot1x ports auth statistics
+------------------------------------------------ DOT1X PORT STATISTICS -------------------------------------------------+
|      | Eapol     | Eapol     | Eapol Start | Eapol Logoff | Eapol Resp | Eapol Resp | Eapol Req | Invalid Eapol | Eapol Req |
| Port | Frame Tx  | Frame Rx  | Frame Rx    | Frame Rx     | Id Rx      | Rx         | Tx        | Frame Rx      | Id Tx     |
+------+-----------+-----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
| 2    | 3844      | 5765      | 1922        | 0            | 1922       | 3843       | 0         | 0             | 3844      |
| 5    | 15        | 0         | 0           | 0            | 0          | 0          | 0         | 0             | 15        |
+------+-----------+-----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
```

Procedure 138 Displaying 802.1x authenticator statistics for a specific port using the CLI

Display 802.1x authenticator statistics for a specific port to view performance information for the port.

Overview

The following table describes the parameter for displaying statistics about a specific authenticator port.

Table 85 Parameter for displaying statistics about an authenticator

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1 Display 802.1x authenticator statistics for a specific port:

```
show dot1x ports auth port <port name> statistics
```

Example

The following example shows the statistics for port 8.

```
> show dot1x ports auth port 8 statistics
+------------- DOT1X PORT SUMMARY -------------+
| Name                       | Value           |
+----------------------------+-----------------+
| Port Name                  | 8               |
| Admin State                | Enabled         |
| Port Control               | Auto            |
| ReAuth Enabled             | False           |
| Quiet Period (sec)         | 60              |
| Server Timeout (sec)       | 30              |
| ReAuth Period (sec)        | 3,600           |
| Max Retries                | 2               |
| EAP Version                | 2               |
| Control Direction          | Both            |
| Operational State          | Enabled         |
| Controlled Port Status     | Unauthorized    |
| PAE State                  | Authenticating  |
| ReAuth Count               | 2               |
| Last EAPOL Frame Version   | 2               |
| Last EAPOL Frame Source    | 54:c3:3e:4b:da:92 |
+----------------------------+-----------------+
```

```
+------------------------------------------------ DOT1X PORT STATISTICS ------------------------------------------------+
|      | Eapol    | Eapol    | Eapol Start | Eapol Logoff | Eapol Resp | Eapol Resp | Eapol Req | Invalid Eapol | Eapol Req |
| Port | Frame Tx | Frame Rx | Frame Rx    | Frame Rx     | Id Rx      |  Rx        |  Tx       | Frame Rx      | Id Tx     |
+------+----------+----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
| 2    | 3838     | 5756     | 1919        | 0            | 1919       | 3837       | 0         | 0             | 3838      |
+------+----------+----------+-------------+--------------+------------+------------+-----------+---------------+-----------+
```

Procedure 139 Displaying 802.1x supplicant statistics using the CLI

Display 802.1x supplicant statistics to view performance information.

Steps

1 Display 802.1x supplicant statistics.

```
show dot1x ports supp statistics
```

Example

The following example shows sample output from the show dot1x ports supp statistics command.

```
> show dot1x ports supp statistics
+------------------------------ DOT1X PORT STATISTICS ---------------------------------+
|      | Eapol      | Eapol      | Eapol Start | Eapol Logoff | Invalid Eapol | Eapol Length |
| Port | Frame Tx   | Frame Rx   | Frame Tx    | Frame Tx     | Frame Rx      | Error Frames |
+------+------------+------------+-------------+--------------+---------------+--------------+
| 1    | 5765       | 3843       | 1922        | 0            | 0             | 0            |
| 6    | 0          | 0          | 0           | 0            | 0             | 0            |
+------+------------+------------+-------------+--------------+---------------+--------------+
```

Procedure 140 Displaying 802.1x supplicant statistics for a specific port using the CLI

Display 802.1x supplicant statistics for a specific port to view performance information.

Overview

The following table describes the parameter for displaying statistics about a specific supplicant port.

Table 86 Parameter for displaying statistics about a supplicant

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1 Display 802.1x supplicant statistics for a specific port:

```
show dot1x ports supp port <port name> statistics
```

Example

The following example shows sample statistics for port 40.

```
> show dot1x ports supp port 40 statistics
+------------- DOT1X PORT SUMMARY -------------+
| Name                       | Value           |
+----------------------------+-----------------+
| Port Name                  | 40              |
| Admin State                | Disabled        |
| Start Period (sec)         | 30              |
| Held Period (sec)          | 60              |
| Auth Period (sec)          | 30              |
| Max Start                  | 3               |
| EAP Version                | 2               |
| EAP Method                 | md5             |
| Operational State          | Disabled        |
| Controlled Port Status     | Authorized      |
| Last EAPOL Frame Version   | 0               |
| Last EAPOL Frame Source    | 00:00:00:00:00:00 |
+----------------------------+-----------------+

+------------------------------ DOT1X PORT STATISTICS -------------------------------+
|      | Eapol     | Eapol     | Eapol Start | Eapol Logoff | Invalid Eapol | Eapol Length |
| Port | Frame Tx  | Frame Rx  | Frame Tx    | Frame Tx     | Frame Rx      | Error Frames |
+------+-----------+-----------+-------------+--------------+---------------+--------------+
| 40   | 0         | 0         | 0           | 0            | 0             | 0            |
+------+-----------+-----------+-------------+--------------+---------------+--------------+
```

Procedure 141 Enabling 802.1x using the YANG model

Enable 802.1x as required by the network plan.

Requirements

The YANG model ciena-dot1x is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to enable 802.1x.
3. Send an RPC <get> to verify that 802.1x is enabled.

Example

The following sample RPC enables 802.1x.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
          <config>
            <system-auth-control>enabled</system-auth-control>
          </config>
        </dot1x>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 142 Disabling 802.1x using the YANG model

Disable 802.1x when 802.1x authentication is no longer required on the system.

Requirements

The YANG model ciena-dot1x is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to disable 802.1x.
3. Send an RPC <get> to verify that 802.1x is disabled.

Example

The following sample RPC disables 802.1x.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
          <config>
            <system-auth-control>disabled</system-auth-control>
          </config>
        </dot1x>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 143 Re-authenticating the supplicant using the YANG model

Re-authenticate the supplicant to re-establish the connection between the supplicant and the authenticator.

Requirements

The YANG model ciena-dot1x is used in this procedure.

Overview

The following table describes the parameter for re-authenticating the supplicant.

Table 87 Parameter for re-authenticating a port

Parameter	Valid values	Description
port name	string	Specifies the name of the port.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to re-authenticate the supplicant.
3. Send an RPC <get> to verify information after the supplicant is re-authenticated.

Example

The following sample RPC re-authenticates the supplicant.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <cn-dot1x:dot1x-port-reauthenticate xmlns:cn-dot1x="http://www.ciena.com/
ns/yang/ciena-dot1x">
    <cn-dot1x:port-name>8</cn-dot1x:port-name>
  </cn-dot1x:dot1x-port-reauthenticate>
</rpc>
```

Procedure 144 Initializing a port using the YANG model

Initialize a port to establish a connection to the authentication server.

Requirements

This procedure uses the YANG model ciena-dot1x.

Overview

The following table describes the parameter for initializing a port.

Table 88 Parameter for initializing a port

Parameter	Valid values	Description
port name	name-string	Specifies the name of the port.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to initializing a port.
3. Send an RPC <get> to initialize a port.

Example

The following sample RPC initializes a port.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <cn-dot1x:dot1x-port-initialize xmlns:cn-dot1x="http://www.ciena.com/ns/
yang/ciena-dot1x">
    <cn-dot1x:port-name>8</cn-dot1x:port-name>
  </cn-dot1x:dot1x-port-initialize>
</rpc>
```

Procedure 145 Configuring a port as a supplicant or an authenticator using the YANG model

Configure a port as a supplicant or authenticator as required by the network plan.

Requirements

This procedure uses the YANG model ciena-dot1x.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to configure a port as a supplicant or an authenticator.
3. Send an RPC <get> to verify the configuration information.

Example

The following sample RPC configures port 8 as a supplicant and port 9 as an authenticator.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces xmlns="http://openconfig.net/yang/interfaces"
                  xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <interface>
          <name>8</name>
          <config>
            <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
              <port-capabilities>
                <supplicant>true</supplicant>
                <authenticator>false</authenticator>
              </port-capabilities>
            </dot1x>
          </config>
        </interface>
        <interface>
          <name>9</name>
          <config>
            <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
              <port-capabilities>
                <supplicant>false</supplicant>
                <authenticator>true</authenticator>
              </port-capabilities>
            </dot1x>
          </config>
        </interface>
      </interfaces>
    </config>
  </edit-config>
```

```
</rpc>
```

Procedure 146 Setting parameters on a supplicant port using the YANG model

Set the parameters on a supplicant port as required by the network plan.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Overview

The following table describes the parameters that can be set on a supplicant port.

Table 89 Parameters on a supplicant port

Parameter	Valid values	Description
admin-status	enabled disabled	Enables or disables the administrative status.
identity	string	Specifies the username that the supplicant will use as the identity in the authentication process.
password	string	Specifies the password that the supplicant will use for EAP-MD5 in the authentication process. For security, the password is displayed as xxxx.
auth-period	number <1..65535>	Specifies the amount of time a supplicant waits for a (EAP) request from an authenticator before timing out.
held-period	number <1..65535>	Specifies the held period for a port in the supplicant role. The held period is the time period for which the supplicant stops trying to authenticate itself to the authenticator after an authentication failure.

Parameter	Valid values	Description
start-period	number <1..65535)	Specifies he amount of time a supplicant waits for a response from an authenticator on a connection request. If the authenticator has not responded after this time interval, the supplicant assumes that there is no valid authenticator on the other side.
max-start	number <1..65535>	Specifies the maximum number of times a supplicant tries to connect to an authenticator before concluding that there is no authenticator present.
eapol-protocol-version	number <1..2>	Specifies the eapol-version: 1 or 2 (2 is the default). This value is set to 1 to work around issues with client implementations that drop EAPOL frames that use version 2.
eap-method	TLS	Selects the EAP-method.
tls-service-profile	TLS service profiles created using "Creating a TLS profile using the CLI" on page 242	Specifies the TLS service profile to be used when the eap-method is set to eap-TLS.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to set parameters on a supplicant port.
3. Send an RPC <get> to verify information about the parameters.

Example

The following sample RPC sets parameters on a supplicant port.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
```

```
    </target>
    <config>
      <interfaces xmlns="http://openconfig.net/yang/interfaces"
                  xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <interface>
          <name>8</name>
          <config>
            <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
              <supplicant>
                <auth-period>30</auth-period>
                <held-period>60</held-period>
                <start-period>30</start-period>
                <max-start>3</max-start>
                <admin-status>enabled</admin-status>
                <identity>pvadmin</identity>
                <password>pvadmin</password>
                <eapol-protocol-version>2</eapol-protocol-version>
                <eap-method>tls</eap-method>
                <tls-service-profile>tls_srv_profile1</tls-service-profile>
              </supplicant>
            </dot1x>
          </config>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

Procedure 147 Setting parameters on an authenticator port using the YANG model

Set parameters on an authenticator port as required by the network plan.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Overview

The following table describes the parameters that can be set on an authenticator port.

Table 90 Parameters on an authenticator port

Parameter	Valid values	Description
admin-status	enabled disabled	Enables or disables the administrative status.
eapol-protocol-version	1|2 The default value is 2.	Specifies the EAPOL version. Set this parameter to 1 to work around issues with client implementations that drop EAPOL frames that use version 2.
control-direction	both	Specifies the current value of the administrative controlled directions parameter for the port.

Parameter	Valid values	Description
port-control	auto force-authorized force-unauthorized	Sets the port authorization mode. auto: Provides 802.1x operation on a port, allowing only EAPOL frames to be sent and received until the client successfully authenticates. Once authenticated, regular traffic is allowed. force-authorized: Disables 802.1x and puts the port in an authorized state. The port transmits and receives normal traffic without 802.1x-based authentication of the client. force-unauthorized: Causes all communications from an 802.1x client to be blocked, preventing the client from authenticating through this port.
reauth-enabled	true false	Determines whether re-authentication is initiated. The re-authentication interval is specified with the reauth-period.
quiet-period	number <1..65535>	Specifies the number of seconds that the system remains in the quiet state following a failed authentication exchange with the client.
reauth-max	number <1..10>	Specifies the maximum number of re-authentication attempts on an authenticator port before the port is unauthorized.

Parameter	Valid values	Description
reauth-period	number <1..65535>	This object indicates the time period of the re-authentication to the supplicant.
server-timeout	number <1..180>	Specifies the timeout value for an authenticator on a port connecting to a backend authentication server.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to set parameters on an authenticator port.
3. Send an RPC <get> to verify information about the parameters.

Example

The following sample RPC sets parameters on an authenticator port.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces xmlns="http://openconfig.net/yang/interfaces"
                  xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <interface>
          <name>9</name>
          <config>
            <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
              <authenticator>
                <eapol-protocol-version>2</eapol-protocol-version>
                <admin-status>disabled</admin-status>
                <control-direction>both</control-direction>
                <port-control>auto</port-control>
                <reauth-enabled>false</reauth-enabled>
                <quiet-period>60</quiet-period>
                <reauth-max>2</reauth-max>
                <reauth-period>3600</reauth-period>
                <server-timeout>30</server-timeout>
              </authenticator>
            </dot1x>
          </config>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

Procedure 148 Attaching a tls-service profile to the supplicant port using the YANG model

Attach a tls-service-profile to the supplicant port in order to use the eap-method as TLS for authentication.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

The tls-service-profile must be created with all the necessary parameters and certificates. For more information, refer to “Creating a TLS service profile using the CLI” on page 245.

Overview

The following table describes the parameters for the tls-service-profile.

Table 91 Tls-service-profile parameters

Parameter	Valid values	Description
port number	<string>	Is the number of the port.
tls-service-profiles-name	<string>	Is the name of the service profile.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to attached a TLS service profile.
3. Send an RPC <get> to verify that the TLS service profile was attached to the supplicant port.

Example

The following sample RPC attaches a TLS service profile to the supplicant port.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <interfaces xmlns="http://openconfig.net/yang/interfaces"
                  xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <interface>
          <name>9</name>
          <config>
            <dot1x xmlns="http://www.ciena.com/ns/yang/ciena-dot1x">
              <supplicant>
                <tls-service-profile>tls_srv_profile1</tls-service-profile>
```

```
              </supplicant>
            </dot1x>
          </config>
        </interface>
      </interfaces>
    </config>
  </edit-config>
</rpc>
```

Procedure 149 Creating and attaching a server group for RADIUS configuration on an authenticator DUT using the YANG model

Create and attach a server group to configure a RADIUS server on an authenticator DUT.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Overview

The following table describes the parameters to create and attach a server group.

Table 92 Server group parameters

Parameter	Valid values	Description
server-group	string	Specifies the configured name of the server group.
config-type	RADIUS	Specifies the AAA server type. All servers in the group must be of this type.
server	IP address	Sets the IP address of the authentication server.
config address	IP address	Is the IP address of the server group.
config timeout	<1... 30> seconds	Sets the timeout in seconds on responses from the AAA server.
auth-port	oc-inet:port-number	Specifies the port number for authentication requests.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to create and attach a server group for RADIUS configuration.
3. Send an RPC <get> to verify the configuration.

Example

The following sample RPC creates a server group for RADIUS configuration.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <system xmlns="http://openconfig.net/yang/system"
              xmlns:ncx="http://netconfcentral.org/ns/yuma-ncx">
        <aaa>
          <server-groups>
            <server-group>
              <name>authserver</name>
              <config>
                <name>authserver</name>
                <type xmlns:oc-aaa="http://openconfig.net/yang/aaa">oc-
aaa:RADIUS</type>
              </config>
              <servers>
                <server>
                  <address>10.33.85.130</address>
                  <config>
                    <address>10.33.85.130</address>
                    <timeout>7</timeout>
                    <admin-state xmlns="http://www.ciena.com/ns/yang/ciena-
openconfig-aaa">enabled</admin-state>
                  </config>
                  <radius>
                    <config>
                      <auth-port>1812</auth-port>
                      <acct-port>1813</acct-port>
                      <secret-key>testing123</secret-key>
                      <retransmit-attempts>3</retransmit-attempts>
                    </config>
                  </radius>
                </server>
              </servers>
            </server-group>
          </server-groups>
        </aaa>
      </system>
    </config>
  </edit-config>
</rpc>
```

Procedure 150 Clearing all 802.1x port statistics using the YANG model

Clear all 802.1x port statistics.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to clear all 802.1x statistics.
3. Send an RPC <get> to verify that all 802.1x statistics were cleared.

Example

The following sample RPC clears all 802.1x port statistics.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <cn-dot1x:clear-dot1x-statistics xmlns:cn-dot1x="http://www.ciena.com/ns/
yang/ciena-dot1x">
    <cn-dot1x:all/>
  </cn-dot1x:clear-dot1x-statistics>
</rpc>
```

Procedure 151 Clearing specific port statistics using the YANG model

Clear specific port statistics to reset statistics collection.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Overview

The following table describes the parameter for clearing specific port statistics.

Table 93 Parameter for clearing port statistics

Parameter	Valid values	Description
port name	name-string	Specifies the name of the port.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to clear specific port statistics.
3. Send an RPC <get> to verify that the specific port statistics were cleared.

Example

The following sample RPC clears specific port statistics.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <cn-dot1x:clear-dot1x-statistics xmlns:cn-dot1x="http://www.ciena.com/ns/
yang/ciena-dot1x">
    <cn-dot1x:port-name>1</cn-dot1x:port-name>
  </cn-dot1x:clear-dot1x-statistics>
</rpc>
```

Procedure 152 Retrieving 802.1x port information using the YANG model

Retrieve 802.1x port information to verify the configuration.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve port information.

Example

The following sample RPC retrieves 802.1x port information.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                  xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:state>
          <cn-dot1x:dot1x/>
        </oc-if:state>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

Procedure 153 Retrieving information about all authenticator ports using the YANG model

Retrieve information about all authenticator ports to verify the configuration.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve information about all authenticator ports.

Example

The following sample RPC retrieves information about all authenticator ports.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                      xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:config>
          <cn-dot1x:dot1x>
            <cn-dot1x:authenticator/>
          </cn-dot1x:dot1x>
        </oc-if:config>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

Procedure 154 Retrieving information about all supplicant ports using the YANG model

Retrieve information about all supplicant port to verify the configuration.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve information about all supplicant ports.

Example

The following sample RPC retrieves information about all supplicant ports.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                  xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:config>
          <cn-dot1x:dot1x>
            <cn-dot1x:supplicant/>
          </cn-dot1x:dot1x>
        </oc-if:config>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

Procedure 155 Retrieving 802.1x global information using the YANG model

Retrieve 802.1x global information to verify the configuration.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve the global information.

Example

The following sample RPC retrieves 802.1x global information in the administrative state.

```
<get>
  <filter type="subtree">
    <oc-sys:system xmlns:oc-sys="http://openconfig.net/yang/system"
                   xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <cn-dot1x:dot1x>
        <cn-dot1x:config>
          <cn-dot1x:system-auth-control/>
        </cn-dot1x:config>
      </cn-dot1x:dot1x>
    </oc-sys:system>
  </filter>
</get>
```

The following sample RPC retrieves 802.1x global information in the operational state.

```
<get>
  <filter type="subtree">
    <oc-sys:system xmlns:oc-sys="http://openconfig.net/yang/system"
                   xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <cn-dot1x:dot1x>
        <cn-dot1x:state>
          <cn-dot1x:operational-status/>
        </cn-dot1x:state>
      </cn-dot1x:dot1x>
    </oc-sys:system>
  </filter>
</get>
```

Procedure 156 Retrieving 802.1x authenticator statistics information using the YANG model

Retrieve 802.1x authenticator statistics information to learn more about 802.1x authenticator statistics.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve 802.1x authenticator statistics information.

Example

The following sample RPC retrieves 802.1x authenticator statistics.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                  xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:state>
          <cn-dot1x:dot1x>
            <cn-dot1x:authenticator/>
          </cn-dot1x:dot1x>
        </oc-if:state>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

Procedure 157 Retrieving 802.1x supplicant statistics using the YANG model

Retrieve 802.1x supplicant statistics information to learn more about 802.1x supplicant statistics.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve 802.1x supplicant statistics information.

Example

The following sample RPC retrieves 802.1x supplicant statistics information.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                      xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:state>
          <cn-dot1x:dot1x>
            <cn-dot1x:supplicant/>
          </cn-dot1x:dot1x>
        </oc-if:state>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

Procedure 158 Retrieving 802.1x authenticator-session statistics using the YANG model

Retrieve 802.1x authenticator-session statistics information to learn more about the 802.1x authenticator session.

Requirements

The YANG model ciena-dot1x.yang is used in this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to retrieve the authenticator-session statistics information.

Example

The following sample RPC retrieves 802.1x authenticator-session statistics information.

```
<get>
  <filter type="subtree">
    <oc-if:interfaces xmlns:oc-if="http://openconfig.net/yang/interfaces"
                  xmlns:cn-dot1x="http://www.ciena.com/ns/yang/ciena-dot1x">
      <oc-if:interface>
        <oc-if:state>
          <cn-dot1x:dot1x>
            <cn-dot1x:authenticator-session/>
          </cn-dot1x:dot1x>
        </oc-if:state>
      </oc-if:interface>
    </oc-if:interfaces>
  </filter>
</get>
```

CHAPTER 6
Transport Layer Security

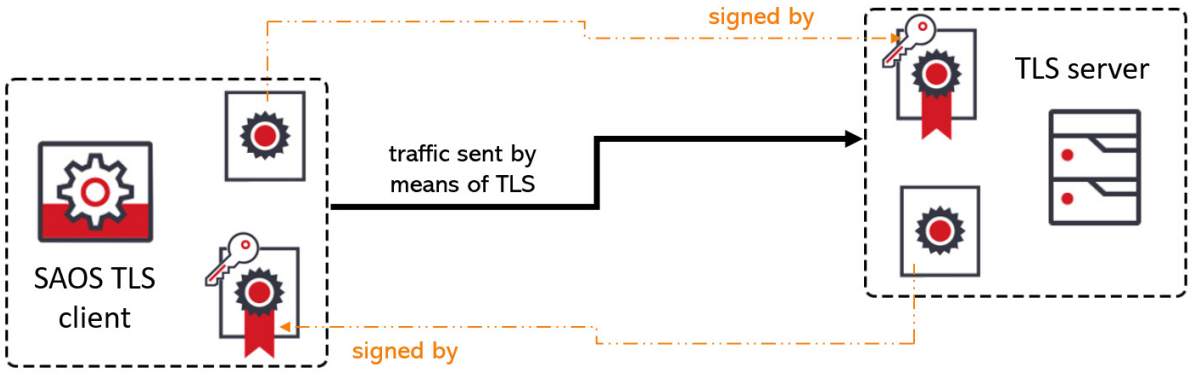
Transport Layer Security (TLS) is a security protocol that creates secure network connections. TLS operates on top of a TCP connection and provides a secure connection between peers. The connection is encrypted and either endpoint can authenticate its peer by various methods. TLS infrastructure is used by other functions or services for creating secure TLS connections to external systems.

The client and external server are pre-configured with signed system certificates and private keys in order to use TLS. The client and server are also pre-configured with the CA certificate that signed each other's system certificates. The server hostname or IP address is used to add the server to a list in a service on the system.

The system invokes the TLS client to create a connection, as needed. The client connects to the servers using TCP, and then attempts to create a TLS connection over that stream. The two endpoints perform configured authentications and establish a TLS connection. Messages can then be transmitted securely using the TLS connection.

The following figure shows a sample use case.

Figure 9 Sample TLS use case



TLS is used by:

- sZTP
- telemetry. For more information about configuring telemetry, refer to *Base, Advanced Ethernet and OAM Configuration* for the product release you are using.
- RADIUS over TLS (RADSEC)

Profiles

TLS services are configured through shared profiles. Profiles are named collections of configurations that can be referenced by one or more TLS services. This approach allows for maximum reuse without having to fully configure every TLS-based service.

TLS service profiles are the top level of TLS configuration. A TLS service profile is referenced by a service.

The following table lists the components of a TLS service profile.

Table 94 Components of a TLS service profile

Component	Description
TLS certificate name	Provides the certificate and key that establishes the identity of the system. The TLS certificate name is a reference to a certificate and private key stored in the system PKIX. It is used to identify the system to its TLS peer.
peer authentication profile	Determines how the TLS peer is authenticated.
TLS profile name	Configures the TLS connection parameters. TLS profiles reference cipher suites and elliptic curves.

TLS servers use the same TLS service profile as TLS client services. Any unused TLS options are disregarded.

Cipher suite

In TLS, cipher suites are combinations of authentication, encryption, message authentication code (MAC) and key exchange algorithms used to negotiate the security settings for a TLS.

The following table lists the cipher suites supported by the system.

Table 95 Cipher suites supported by the system

Default priority	Standard name	OpenSSL name	Cipher suite
1	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	ECDHE-RSA-AES256-GCM-SHA384	0xC030
2	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	ECDHE-ECDSA-AES256-GCM-SHA384	0xC02C
3	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	ECDHE-RSA-AES256-SHA384	0xC028
4	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	ECDHE-ECDSA-AES256-SHA384	0xC024
5	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	ECDHE-RSA-AES256-SHA	0xC014
6	TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	ECDHE-ECDSA-AES256-SHA	0xC00A
7	TLS_DHE_DSS_WITH_AES_256_GCM_SHA384	DHE-DSS-AES256-GCM-SHA384	0xA3
8	TLS_DHE_RSA_WITH_AES_256_GCM_SHA384	DHE-RSA-AES256-GCM-SHA384	0x9F
9	TLS_DHE_RSA_WITH_AES_256_CBC_SHA256	DHE-RSA-AES256-SHA256	0x6B
10	TLS_DHE_DSS_WITH_AES_256_CBC_SHA256	DHE-DSS-AES256-SHA256	0x6A
11	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	DHE-RSA-AES256-SHA	0x39
12	TLS_DHE_DSS_WITH_AES_256_CBC_SHA	DHE-DSS-AES256-SHA	0x38
13	TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384	ECDH-RSA-AES256-GCM-SHA384	0xC032
14	TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384x	ECDH-ECDSA-AES256-GCM-SHA384	0xC02E
15	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	ECDH-RSA-AES256-SHA384	0xC02A
16	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	ECDH-ECDSA-AES256-SHA384	0xC026
17	TLS_ECDH_RSA_WITH_AES_256_CBC_SHA	ECDH-RSA-AES256-SHA	0xC00F

Default priority	Standard name	OpenSSL name	Cipher suite
18	TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA	ECDH-ECDSA-AES256-SHA	0xC005
19	TLS_RSA_WITH_AES_256_GCM_SHA384	AES256-GCM-SHA384	0x9D
20	TLS_RSA_WITH_AES_256_CBC_SHA256	AES256-SHA256	0x3D
21	TLS_RSA_WITH_AES_256_CBC_SHA	AES256-SHA	0x35
22	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	ECDHE-RSA-AES128-GCM-SHA256	0xC02F
23	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	ECDHE-ECDSA-AES128-GCM-SHA256	0xC02B
24	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	ECDHE-RSA-AES128-SHA256	0xC027
25	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	ECDHE-ECDSA-AES128-SHA256	0xC023
26	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	ECDHE-RSA-AES128-SHA	0xC013
27	TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	ECDHE-ECDSA-AES128-SHA	0xC009
28	TLS_DHE_DSS_WITH_AES_128_GCM_SHA256	DHE-DSS-AES128-GCM-SHA256	0xA2
29	TLS_DHE_RSA_WITH_AES_128_GCM_SHA256	DHE-RSA-AES128-GCM-SHA256	0x9E
30	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256	DHE-RSA-AES128-SHA256	0x67
31	TLS_DHE_DSS_WITH_AES_128_CBC_SHA256	DHE-DSS-AES128-SHA256x	0x40
32	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	DHE-DSS-AES128-SHA	0x33
33	TLS_DHE_DSS_WITH_AES_128_CBC_SHA	DHE-DSS-AES128-SHA	0x32
34	TLS_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	ECDHE-RSA-DES-CBC3-SHA	0xC012
35	TLS_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	ECDHE-ECDSA-DES-CBC3-SHA	0xC008

Default priority	Standard name	OpenSSL name	Cipher suite
36	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA	EDH-RSA-DES-CBC3-SHA	0x16
37	TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA	EDH-DSS-DES-CBC3-SHA	0x13
38	TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256	ECDH-RSA-AES128-GCM-SHA256	0xC031
39	TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	ECDH-ECDSA-AES128-GCM-SHA256	0xC02D
40	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256	ECDH-RSA-AES128-SHA256	0xC029
41	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	ECDH-ECDSA-AES128-SHA256	0xC025x
42	TLS_ECDH_RSA_WITH_AES_128_CBC_SHA	ECDH-RSA-AES128-SHA	0xC004x
43	TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA	ECDH-ECDSA-AES128-SHA	0xC004x
44	TLS_ECDH_RSA_WITH_3DES_EDE_CBC_SHA	ECDH-RSA-DES-CBC3-SHA	0xC003
45	TLS_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHAx	ECDH-ECDSA-DES-CBC3-SHAx	0xC003
46	TLS_RSA_WITH_AES_128_CBC_SHA256	AES128-SHA256x	0x9C
47	TLS_RSA_WITH_AES_128_CBC_SHA256	AES128-SHA256x	0x3C
48	TLS_RSA_WITH_AES_128_CBC_SHA	DES-CBC3-SHA	0x2F
49	TLS_RSA_WITH_3DES_EDE_CBC_SHA	DES-CBC3-SHA	0x0A
50	TLS_RSA_WITH_RC4_128_SHA ***Note:*** non-fips compliant (RC4) but is recommended as part of RFC6614 (RadSec). It will be admin disabled by default and operationally disabled by FIPS mode	RC4-SHA	0x05

Default priority	Standard name	OpenSSL name	Cipher suite
51	TLS_RSA_WITH_RC4_128_MD5 ***Note:*** is non-fips compliant (RC4 & MD5) but is recommended as part of RFC5215 (EAP-TLS). It will be admin disabled by default and operationally disabled by FIPS mode.	RC4-MD5	0x04

Elliptic curve

Cipher suites use elliptic curves for key exchange and authentication.

The following table lists the elliptic curves supported by the system.

Table 96 Elliptic curves supported by the system

Default priority	Standard name	ID	FIPS module support
1	secp256r1	0x0017	yes
2	secp521r1	0x0019	yes
3	brainpoolP512r1	0x001C	no
4	brainpoolP384r1	0x001B	no
5	secp384r1	0x0018	yes
6	brainpoolP256r1	0x001A	no
7	secp256k1	0x0016	yes
8	sect571r1	0x000E	yes
9	sect571k1	0x000D	yes
10	sect409k1	0x000B	yes
11	sect409r1	0x000C	yes
12	sect283k1	0x0009	yes
13	sect283r1	0x000A	yes

List of procedures

Procedures for configuring TLS are:

- Procedure 159, “Creating a TLS profile using the CLI”
- Procedure 160, “Creating a peer authentication profile using the CLI”

- Procedure 161, “Creating a TLS service profile using the CLI”
- Procedure 162, “Displaying TLS profile information using the CLI”
- Procedure 163, “Displaying TLS server session information using the CLI”
- Procedure 164, “Creating a TLS profile using the YANG model”
- Procedure 165, “Creating a peer authentication profile using the YANG model”
- Procedure 166, “Creating a TLS service profile using the YANG model”
- Procedure 167, “Retrieving the TLS profile information using the YANG model”

Procedure 159 Creating a TLS profile using the CLI

Create a TLS profile according to the network plan.

Overview

The following table lists parameters for creating a TLS profile.

Table 97 Parameters for creating a TLS profile

Parameters	Valid values	Description
profile-name	string	Identifies the profile.
tls-version	• tls-1.0 • tls-1.1 • tls-1.2 The default value is tls-1.1.	Sets the minimum TLS version.
cipher-suite	Refer to "Cipher suite" on page 236.	Identifies the cipher suites to use for the profile. Cipher suites are listed in order of priority.
elliptic-curve	Refer to "Elliptic curve" on page 240.	Identifies the elliptic curves to use for the cipher suites. Elliptic curves are listed in order of priority.
session-resumption-timeout	60-86400 The default value is 3600.	Sets the timeout for the session. The timeout is set to avoid reusing stale sessions without a fresh authentication.

Steps

1 Specify the name of the TLS profile:

```
hello-params tls-profiles tls-profile <name>
```

2 Set the minimum TLS version:

```
hello-params tls-profile tls-versions tls-version <tls-version>
```

3 Specify the cipher suite:

```
hello-params https-tls-profile cipher-suites cipher-suite <cipher suite>
```

4 Specify the elliptic curve:

```
hello-params https-tls-profile elliptic-curves elliptic-curve <elliptic curve>
```

5 Specify the session-resumption-timeout:

```
tls-profile session-resumption-timeout <timeout>
```

Example

The following example creates a TLS profile named baseConf.

```
hello-params tls-profiles tls-profile baseConf
hello-params https-tls-profile cipher-suites cipher-suite ecdhe-rsa-with-aes-
128-cbc-sha
hello-params https-tls-profile elliptic-curves elliptic-curve ciena-tls-
types:secp256r1
hello-params tls-profile session-resumption-timeout 3600
```

Procedure 160 Creating a peer authentication profile using the CLI

Create a peer authentication profile to determine how the TLS peer is authenticated.

Overview

The following table lists parameters for creating a peer authentication profile.

Table 98 Parameters for creating a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Specifies the profile name.
check-cert-expiry	• true • false	Determines whether the certificate is checked for expiry.

Steps

1 Specify a name for the peer authentication profile:

```
pkix peer-auth-profiles peer-auth-profile <peer-auth-profile-name>
```

2 Determine whether the certificate is checked for expiry:

```
pkix peer-auth-profiles peer-auth-profile <peer-auth-profile-name> check-cert-expiry <true|false>
```

Example

The following example creates a peer authentication profile named baseConf.

```
pkix peer-auth-profiles peer-auth-profile baseConf
pkix peer-auth-profiles peer-auth-profile baseConf check-cert-expiry
true
```

Procedure 161 Creating a TLS service profile using the CLI

Create a TLS service profile so that TLS services can use the same TLS information.

Overview

The following table lists parameters for creating a TLS service profile.

Table 99 Parameters for creating a TLS service profile

Parameters	Valid values	Description
tls-profile-name	string	Identifies the TLS profile.
tls-peer-auth-profile-name	string	Identifies the peer authentication profile. If you omit the peer authentication profile, the peer is authenticated: this can be helpful for initial setup or debugging, but it is significantly less secure.
tls-certificate-name	string	Identifies the TLS certificate.

Steps

1 Identify the TLS profile:

```
tls-service-profiles test tls-profile-name <tls-profile>
```

2 Identify the peer authentication profile:

```
tls-service-profiles test tls-peer-auth-profile-name
<peer-auth-profile>
```

3 Identify the TLS certificate:

```
tls-service-profiles test tls-certificate-name
<certificate>
```

Example

The following example assigns the profile components to the TLS service profile names test.

```
tls-service-profiles test tls-profile-name tls-profile
tls-service-profiles test tls-peer-auth-profile-name peer-auth-profile
tls-service-profiles test tls-certificate-name testCert
```

Procedure 162 Displaying TLS profile information using the CLI

View TLS profile information to verify TLS configuration.

Steps

1 View TLS profile information:

```
show tls
```

Example

The following example shows sample output for the show tls command.

```
+----------- TLS SERVICE PROFILES ----------+
| Name                    | Value           |
+-------------------------+-----------------+
| Service Profile Name    | baseConf        |
| TLS Profile Name        | baseConf        |
| Peer Auth Profile Name  | baseConf        |
| Certificate Name        | testCert        |
+-------------------------+-----------------+
+------------- PEER AUTH PROFILES ------------+
| Name                               | Value    |
+------------------------------------+----------+
| Profile Name                       | baseConf |
| Check Expiry                       | True     |
| Check IP/Host                      | True     |
| IP/Host List                       | abc      |
+------------------------------------+----------+
+----------------------------- HELLO PARAMS -----------------------------+
| Name                          | Value                                  |
+-------------------------------+----------------------------------------+
| Profile Name                  | baseConf                               |
| Protocol Versions             | tls-1.2                                |
| Cipher Suites                 | ecdhe-ecdsa-with-aes-256-gcm-sha384,   |
|                               | ecdhe-ecdsa-with-aes-256-cbc-sha384,   |
|                               | ecdhe-ecdsa-with-aes-128-cbc-sha256,   |
|                               | rsa-with-aes-256-gcm-sha384,           |
|                               | rsa-with-aes-256-cbc-sha256,           |
|                               | rsa-with-aes-256-cbc-sha,              |
|                               | ecdhe-rsa-with-aes-128-gcm-sha256,     |
|                               | ecdhe-rsa-with-aes-128-cbc-sha256,     |
|                               | rsa-with-aes-128-gcm-sha256,           |
|                               | rsa-with-aes-128-cbc-sha,              |
|                               | rsa-with-3des-ede-cbc-sha              |
| Elliptic Curves               | secp521r1, secp384r1, secp256r1        |
| Sess. Resumption Timeout (s)  | 3600                                   |
+-------------------------------+----------------------------------------+
```

Procedure 163 Displaying TLS server session information using the CLI

View TLS server session information to see information about clients connected to a system in real-time.

Overview

Only active real-time session can be viewed. When the connection is terminated, the command does not return session information.

The following table describes the TLS server information.

Table 100 TLS server session information

Statistic	Description
Unique ID	A unique arbitrary number assigned to active TLS connections.
Client IP	IP of the client machine.
Client Application Name	Name of the client application.
Session Cipher Suite	Cipher-suite selected between server and client during TLS handshake.
Elliptic Curve	Elliptical curve selected between server and client during TLS handshake.
TLS Version	TLS version decided between selected between server and client during TLS handshake.
TLS Profile Name	TLS Service Profile used by server/DUT.
OCSP Operation State	TLS connection created using OCSP. Enabled: Yes, Disabled: No.
OCSP Responder URL	OCSP Responder URL if OCSP is used in the TLS connection.
Client Certificate Serial Number	Serial number of the client's certificate.
Client Certificate Expiry	Expiry date information for the client's certificate.
Client Subject Common Name	The common name of the client's certificate.
Server Subject Common Name	The common name of the server's certificate.

Steps

1 Display the tls server sessions information:

```
show tls-server-sessions
```

Example

The following example shows the output from the show tls-server-sessions command for TLS service profile named test.

```
show tls-server-sessions
+--------------------- TLS SERVER SESSIONS ----------------------+
| Name                                | Value                      |
+-------------------------------------+----------------------------+
| Unique ID                           | 6                          |
| Client IP                           | ::ffff:10.33.80.209        |
| Client Application Name             | GNMI                       |
| Session Cipher Suite                | ECDHE-RSA-AES256-GCM-SHA384 |
| Elliptic Curve                      | P-384                      |
| TLS Version                         | TLSv1.2                    |
| TLS Service Profile Name            | test                       |
| OCSP Operation State                | enabled                    |
| OCSP Responder URL                  | http://10.121.230.37:8090  |
| Client Certificate Serial Number    | 12293                      |
| Client Certificate Expiry           | Jul 15 16:53:50 2020 GM    |
| Client Subject Common Name          | TLS_Client_NoAia           |
| Server Subject Common Name          |                            |
+-------------------------------------+----------------------------+
```

Procedure 164 Creating a TLS profile using the YANG model

Create a TLS profile as required by the network plan.

Requirements

The YANG model ciena-tls.yang is used in this procedure.

Overview

The following table lists parameters for creating a TLS profile.

Table 101 Parameters for creating a TLS profile

Parameters	Valid values	Description
profile-name	string	Identifies the profile.
tls-version	• tls-1.0 • tls-1.1 • tls-1.2 The default value is tls-1.1.	Sets the minimum TLS version.
cipher-suite	Refer to “Cipher suite” on page 236.	Identifies the cipher suites to use for the profile. Cipher suites are listed in order of priority.
elliptic-curve	Refer to “Elliptic curve” on page 240.	Identifies the elliptic curves to use for the cipher suites. Elliptic curves are listed in order of priority.
session-resumption-timeout	60-86400 The default value is 3600.	Sets the timeout for the session. The timeout is set to avoid reusing stale sessions without a fresh authentication.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to create a TLS profile.
3. Send an RPC <get> to verify TLS profile information.

Example

The following sample RPC creates a TLS profile.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
```

```
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
     <ciena-tls:hello-params xmlns:ciena-tls="http://www.ciena.com/tls/yang/
ciena-tls">
        <ciena-tls:profile-name>baseConf</ciena-tls:profile-name>
        <ciena-tls:tls-versions>
          <ciena-tls:tls-version
xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-common">tlscmn:tls-1.2</
ciena-tls:tls-version>
        </ciena-tls:tls-versions>
        <ciena-tls:cipher-suites>
 <ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-ecdsa-with-aes-256-gcm-sha384</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:ecdh-ecdsa-with-aes-256-gcm-sha384</
ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-ecdsa-with-aes-256-cbc-sha384</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:rsa-with-aes-256-gcm-sha384</ciena-
tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:rsa-with-aes-256-cbc-sha256</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:rsa-with-aes-256-cbc-sha</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-rsa-with-aes-128-gcm-sha256</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-ecdsa-with-aes-128-cbc-sha256</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-rsa-with-aes-128-cbc-sha256</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:ecdhe-ecdsa-with-aes-128-cbc-sha256</ciena-tls:cipher-suite>
 <ciena-tls:cipher-suite xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:rsa-with-aes-128-gcm-sha256</ciena-
tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:rsa-with-aes-256-cbc-sha256</ciena-tls:cipher-suite>
 <ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:rsa-with-aes-128-cbc-sha</ciena-tls:cipher-suite>
<ciena-tls:cipher-suite xmlns:tlscmn="urn:ietf:params:xml:ns:yang:ietf-tls-
common">tlscmn:rsa-with-3des-ede-cbc-sha</ciena-tls:cipher-suite>
 </ciena-tls:cipher-suites>
 <ciena-tls:elliptic-curves>
<ciena-tls:elliptic-curve xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:secp521r1</ciena-tls:elliptic-curve>
<ciena-tls:elliptic-curve xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:secp384r1</ciena-tls:elliptic-curve>
<ciena-tls:elliptic-curve xmlns:ciena-tls-types="http://www.ciena.com/tls/
yang/ciena-tls-types">ciena-tls-types:secp256r1</ciena-tls:elliptic-curve>
 </ciena-tls:elliptic-curves>
 <ciena-tls:session-resumption-timeout>3600</ciena-tls:session-resumption-
timeout>
 </ciena-tls:hello-params>
   </config>
 </edit-config>
</rpc>
```

Procedure 165 Creating a peer authentication profile using the YANG model

Create a peer authentication profile to determine how the TLS peer is authenticated.

Requirements

The YANG model ciena-pkix.yang is used for this procedure.

Overview

The following table lists parameters for creating a peer authentication profile. profile.

Table 102 Parameters for creating a peer authentication profile

Parameters	Valid values	Description
peer-auth-profile-name	string	Identifies the profile.
check-cert-expiry	• true • false	Determines whether the certificate is checked for expiry.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to create a peer authentication profile.
3. Send an RPC <get> to verify peer authentication profile information.

Example

The following RPC creates a peer authentication profile named baseConf.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
    <config>
      <pkix:pkix xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
        <pkix:peer-auth-profiles>
          <pkix:peer-auth-profile>
         <pkix:peer-auth-profile-name>baseConf</pkix:peer-auth-profile-name>
            <pkix:check-cert-expiry>true</pkix:check-cert-expiry>
          </pkix:peer-auth-profile>
        </pkix:peer-auth-profiles>
      </pkix:pkix>
    </config>
  </edit-config>
</rpc>
```

Procedure 166 Creating a TLS service profile using the YANG model

Create a TLS service profile to enable TLS services to use the same TLS information.

Requirements

The YANG model ciena-tls-service-profile.yang is used for this procedure.

Overview

The following table lists parameters for creating a TLS service profile.

Table 103 Parameters for creating a TLS service profile

Parameters	Valid values	Description
tls-profile-name	string	Identifies the TLS profile.
tls-peer-auth-profile-name	string	Identifies the peer authentication profile.
tls-certificate-name	string	Identifies the TLS certificate.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <edit-config> to create a TLS service profile.
3. Send an RPC <get> to verify the TLS service profile information.

Example

The following RPC creates a TLS service profile.

```
<?xml version="1.0" encoding="utf-8"?>
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="">
  <edit-config>
    <target>
      <running/>
    </target>
<config>
      <ciena-tls-service-profile:tls-service-profiles xmlns:ciena-tls-
service-profile="http://www.ciena.com/tls/yang/ciena-tls-service-profile">
        <ciena-tls-service-profile:tls-service-profile-name>baseConf</ciena-
tls-service-profile:tls-service-profile-name>
        <ciena-tls-service-profile:tls-profile-name>baseConf</ciena-tls-
service-profile:tls-profile-name>
       <ciena-tls-service-profile:c>baseConf</ciena-tls-service-profile:tls-
peer-auth-profile-name>
        <ciena-tls-service-profile:tls-certificate-name>testCert</ciena-tls-
service-profile:tls-certificate-name>
      </ciena-tls-service-profile:tls-service-profiles>
    </config>
```

```
  </edit-config>
</rpc>
```

Procedure 167 Retrieving the TLS profile information using the YANG model

View TLS profile information to verify the TLS configuration.

Requirements

The YANG model ciena-tls-service-profile.yang is used for this procedure.

Steps

1. Establish a NETCONF connection to the system.
2. Send an RPC <get> to view TLS profile information.

Example

The following sample RPC retrieves the TLS profile information.

```
<get>
  <filter type="subtree">
    <ciena-tls-service-profile:tls-service-profiles xmlns:ciena-tls-service-
profile="http://www.ciena.com/tls/yang/ciena-tls-service-profile"/>
  </filter>
</get>

<get>
  <filter type="subtree">
    <ciena-tls:hello-params xmlns:ciena-tls="http://www.ciena.com/tls/yang/
ciena-tls"/>
  </filter>
</get>

<get>
  <filter type="subtree">
    <pkix:pkix xmlns:pkix="http://www.ciena.com/ns/yang/ciena-pkix">
      <pkix:peer-auth-profiles/>
    </pkix:pkix>
  </filter>
</get>
```

3948/513x/5144/516x/5170/811x Routers and Platforms

Security

SAOS 10.7.1
Publication: 323-1955-303
Document status: Standard
Revision A
Document release date: May 2022

CONTACT CIENA
For additional information, office locations, and phone numbers, please visit the Ciena web site at **www.ciena.com**