Firewall Support of Skinny Client Control Protocol

Last Updated: September 24, 2012

The Firewall Support of Skinny Client Control Protocol (SCCP) feature enables Context-Based Access Control (CBAC) inspection to support the Voice over IP (VoIP) protocol, Skinny Client Control Protocol (SCCP). That is, CBAC inspects Skinny control packets that are exchanged between a Skinny client and the Call Manager (CM); CBAC then configures the router (also known as the Cisco IOS Firewall) to enable the Skinny data channels to traverse through the router. In addition,the Firewall Support of Skinny Client Control Protocol (SCCP) feature extends the support of SCCP to accomodate video channels.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Firewall Support of Skinny Client Control Protocol

The Skinny inspection module is part of the inspection subsystem; thus, your router must be running an image that has firewall support.

Restrictions for Firewall Support of Skinny Client Control Protocol

This feature has the following restrictions:

- Skinny inspection will inspect only the SCCP sessions that have been established after the firewall is configured with Skinny inspection. That is, any SCCP sessions that were established through the firewall before the Skinny inspection was configured will not be inspected.
- This feature does not support Music on Hold (MOH) when a device other than the CM is the music server. (This feature does support MOH when the CM is the music server.)
- This feature does not address either the multicast functionality of SCCP or the functionality of multiple active calls on a single Skinny client.

This feature does not support the following Skinny and firewall configurations:

- The firewall and CM cannot be in the same router. Skinny inspection does not support this configuration because the current firewall implementation does not inspect sessions that start or terminate at the router. Thus, Skinny inspection will work only with an external CM.
- The CM and the Skinny client cannot be on three different networks that are separated at the firewall. The current firewall implementation does not inspect sessions that have devices residing on more than two distinct networks that are segregated at the firewall. That is, if there are more than two interfaces at the firewall, session inspection is not supported.

Information About Firewall Support of Skinny Client Control Protocol

Context-Based Access Control Overview

CBAC extends the concept of static access control lists (ACLs) by introducing dynamic ACL entries that open the necessary application ports on the basis of a specific application and close these ports at the end of the application session. CBAC achieves this functionality by inspecting the application data, checking for conformance of the application protocol, extracting the relevant port information to create the dynamic ACL entries, and closing these ports at the end of the session. CBAC is designed to easily allow a new application inspection whenever support is needed.

Skinny Overview

Skinny enables voice communication between two Skinny clients through the use of a CM. Typically, the CM provides service to the Skinny clients on TCP Port 2000. Initially, a Skinny client connects to the CM by establishing a TCP connection; the client will also establish a TCP connection with a secondary CM, if available. After the TCP connection is established, the client will register with the primary CM, which will be used as the controlling CM until it reboots or there is a keepalive failure. Thus, the Skinny TCP connection between the client and the CM exists forever and is used to establish calls coming to or from the client. If a TCP connection failure is detected, the secondary CM is used. All data channels established with the previous CM remain active and will be closed after the end parties hang up the call.

The table below lists the set of messages that are necessary for the data sessions to open and close. Skinny inspection will examine the data sessions that are deemed for opening and closing the access list pin holes.

Table 1 Skinny Data Session Messages

Skinny Inspection Message	Description
StationOpenReceiveChannelAckMessage	Contains the IP address and port information of the Skinny client sending this message. This message also contains the status of whether or not the client is willing to receive the voice traffic.
StationStartMediaTransmissionMessage	Contains the IP address and port information of the remote Skinny client.
StationCloseReceiveChannelMessage	CM instructs the Skinny client (on the basis of the information in this message) to close the receiving channel.
StationStopMediaTransmissionMessage	CM instructs the Skinny client (on the basis of the information in this message) to stop transmitting voice traffic.
StationStopSessionTransmissionMessage	CM instructs the Skinny client (on the basis of the information in this message) to end an indicated session.
StationOpenMultiMediaReceiveChannelAckMessage	Contains the IP address and port information of the skinny Client sending this message. It also contains the status of whether client is willing to receive the video and data channels.
StationCloseMultiMediaReceiveChannel	This message is sent from the Cisco Unified Communications Manager to the Skinny endpoint to request closing the receiving video or data channel.
StationStartMultiMediaTransmitMessage	This message is sent from Cisco Unified Communications Manager to the Skinny endpoint whenever Cisco Unified Communications Manager gets back OpenLogicalChannelAck for video or data channel.

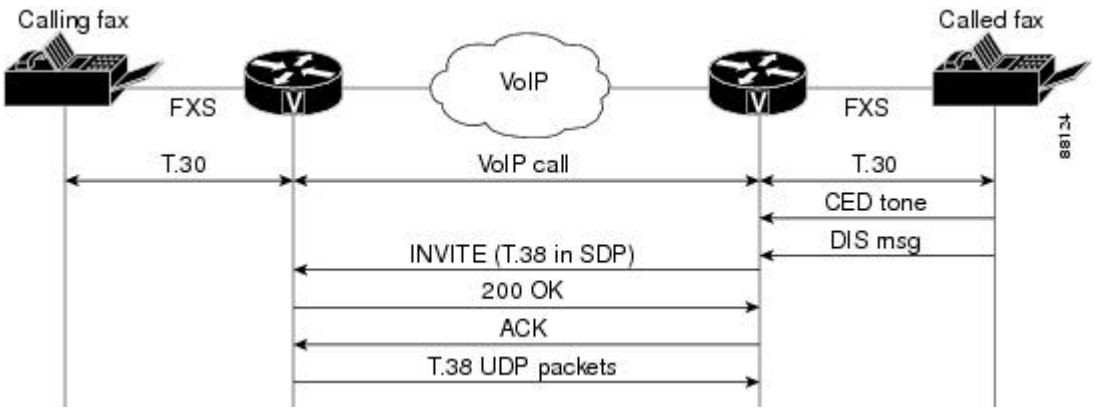
Skinny Inspection Message	Description
StationStopMultiMediaTransmission	This message is sent to Skinny endpoints to request transmission of video\data channel to stop.

CBAC and Skinny Functionality Overview

The figure below depicts typical deployment solutions that are supported by CBAC inspection for Skinny. According to Figure 1, a firewall with Skinny inspection can be configured on Cisco IOS Router A, Cisco IOS Router B, or both routers, thereby addressing the following three scenarios:

- A Cisco IOS router with a firewall on the customer premises equipment (CPE) side, supporting Skinny VoIP phone
- A Cisco IOS router with a firewall on the CM side
- A Cisco IOS router with a firewall at both ends of the connection

Figure 1 CBAC Inspection for Skinny Sample Topology



SCCP Video Call Flow

The figure below illustrates the communication paths between the clients and the Call Manager (CM). The firewall resides either a) in the path from Client A to CM and from Client A to Client B as indicated by Firewall-1 or b) in the path from Client B to CM and from Client B to Client A as indicated by Firewall-2.

Figure 2 Skinny Client to Skinny Client Communication

Setting Skinny CBAC Session Timeouts

Session timeouts are triggered when traffic is not seen on a particular session for a configured amount of time. (This value is configured via the **ip inspect name** command.) After the inactivity timeout is triggered, the firewall will clean up the session and deallocate all of the session data.

You must set the inactivity timeout value for Skinny to a greater value than the keepalive timeout value that is configured between the CM and Skinny clients. Otherwise, the Skinny connection may become inaccessible for inspection because the firewall might delete the session-related information due to inactivity.

After the inactivity timeout is triggered, the inspection module will send reset (RST packets) to both ends of the connection. Any data channels that are associated with the control channel will not be closed. After both end parties hang up, there will not be any traffic on the data channels and the connection will eventually timeout.

Note If the inactivity timeout of the control channel that is connected to the primary CM is less than the keepalive timeout that is sent by the CM to the Skinny client, the firewall will set the inactivity timeout to three times the keepalive timeout. If a timeout is not configured, the default value of 3600 seconds will be used.

How to Configure Your Firewall for Skinny Support

Configuring Basic Skinny CBAC Inspection

Perform the following required steps to configure a basic Skinny CBAC configuration:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip inspect name** *inspection-name protocol* **alert on off** }] [**audit-trail on**| **off**}] [**timeout** *seconds*
4. ip inspect name *inspection-name protocol* **alert on**| **off**}] [**audit-trail on**| **off**}] [**timeout** *seconds*
5. **interface** *type number*
6. **ip access-group** {*access-list-number*} {**in** | **out**}
7. **ip inspect** *inspection-name* **in** | **out**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip inspect name** *inspection-name protocol* **alert on off** }] [**audit-trail on**| **off**}] [**timeout** *seconds* **Example:** `Router(config)# ip inspect name firewall skinny`	Enables CBAC Skinny inspections.
Step 4	ip inspect name *inspection-name protocol* **alert on**| **off**}] [**audit-trail on**| **off**}] [**timeout** *seconds* **Example:** `Router(config)# ip inspect name firewall tftp`	(Optional. Required if the TFTP server is outside the firewall.) Defines a set of inspection rules.
Step 5	**interface** *type number* **Example:** `Router(config)# interface FastEthernet 0/0`	Configures an interface type and enters interface configuration mode.
Step 6	**ip access-group** {*access-list-number*} {**in** | **out**} **Example:** `Router(config-if)# ip access-group 100 in`	Control access to an interface. Number of the access list that is blocking incoming traffic.
Step 7	**ip inspect** *inspection-name* **in** | **out** **Example:** `Router(config-if)# ip inspect firewall out`	Applies a set of inspection rules to an interface.

Configuring Port to Application Mapping

By default, the Skinny inspection will inspect SCCP messages to or from the CM on TCP port 2000. If you prefer to configure the CM to use a different port, the port to application mapping (PAM) feature should be used to specify the desired port to the Cisco IOS firewall. Thus, the firewall will inspect the SCCP messages in the desired port and in port 2000. To configure the CM to use a different port via PAM, use the ip port-map command.

Before you can configure PAM, you must first configure the steps in the section, "Configuring Basic Skinny CBAC Inspection."

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip port map** *appl_name* **port** *port_num* [**list** *acl_num*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**ip port map** *appl_name* **port** *port_num* [**list** *acl_num*] **Example:** `Router(config)# ip port map skinny port 2100`	(Optional) Creates a port to address mapping for SCCP. This command allows you to indicate additional ports that need to be monitored for SCCP.

Verifying Cisco IOS Firewall for Skinny Support

To display active Skinny session information, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. **show ip inspect** {**name** *inspection-name* | **config** | **interfaces** | **session** [**detail**] | **all**}
3. **show ip access-list**
4. show ip port-map [*appl_name* | **port** *port_num*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.
Step 2	**show ip inspect** {**name** *inspection-name* | **config** | **interfaces** | **session** [**detail**] | **all**} **Example:** `Router# show ip inspect session detail`	(Optional) Displays existing sessions that are currently being tracked and inspected by CBAC. The optional **detail** keyword causes additional details about these sessions to be shown.
Step 3	**show ip access-list** **Example:** `Router# show ip access-list`	(Optional) Displays the contents of all current IP access lists, which includes the dynamic access lists created by Skinny inspection.

Step 4	show ip port-map [*appl_name* | **port** *port_num*
	
	Example:
	`Router#`
	`show ip port-map skinny`
		(Optional) Displays information about the active port to application mappings on the router.
		Use this command to view Skinny port map information.
		• *appl_name* --Displays Skinny-specific PAM information. (You must specify the *skinny* argument.)

Monitoring Cisco IOS Firewall for Skinny Support

Note Effective with Cisco IOS Release 12.4(20)T, the **debug ip inspect** command is replaced by the **debug policy-firewall** command. See the *Cisco IOS Debug Command Reference* for more information.

To monitor debugging messages related to Skinny inspection, perform the following optional steps:

SUMMARY STEPS

1. **enable**
2. **debug ip inspect** {*sccp* | **detailed**

DETAILED STEPS

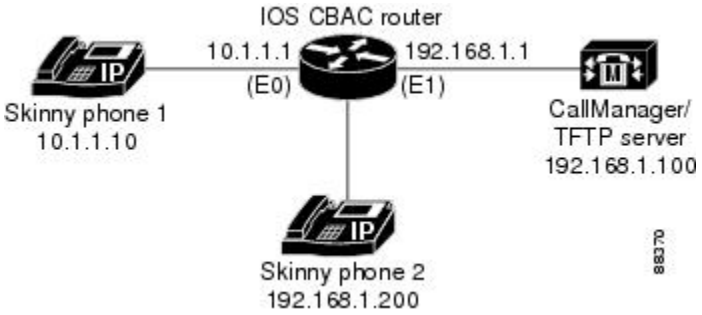
	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables higher privilege levels, such as privileged EXEC mode. • Enter your password if prompted.

Step 2	**debug ip inspect** {*sccp* | **detailed**
	
	Example:
	
	`Router#`
	`debug ip inspect sccp`
		(Optional) Displays and logs the debugging messages related to SCCP inspection.

Configuration Examples for Firewall Skinny Support

Example Firewall and Skinny Configuration

Figure 3 Skinny and CBAC Configuration



The following is an example of how to configure a Cisco IOS firewall for Skinny support (see the figure above):

```
!  Define the name of the router as “CBAC-Firewall.”
!
host CBAC-Firewall
!
!  Create a DHCP server process to offer out 10.1.1.x addresses on the
!  inside network. Option 150 is used by Cisco IP phones as where to
!  look for their configuration file. A default router is required so that all
!  the IP phones can talk to networks other than just to the local 10.1.1.x.
!
ip dhcp pool localnetwork
   network 10.1.1.0 255.255.255.0
```

```
   option 150 ip 192.168.1.100
   default-router 10.1.1.1
!
!  Prevent the DHCP server process from assigning 10.1.1.1 -.9 as an IP
!  address on the local network. This is done to hold the addresses .2 - .9 as static-
!  defined addresses.
!
ip dhcp excluded-address 10.1.1.1 10.1.1.9
!
!  Define firewall rules to all Skinny traffic in/out along with TFTP
!  services.
!
ip inspect name fwout tftp
ip inspect name fwout skinny
!
!  Prevent any traffic from coming in.
!
access-list 100 deny ip any any
!
interface ethernet 1
  ip access-group 100 in
  ip inspect firewall out
If the CallManager is requiring Skinny registration to happen on port tcp/2100, you will
still need the above configuration plus the following additional step.
ip port map skinny port 2100
```

Additional References

Related Documents

Related Topic	Document Title
Additional CBAC information and configuration tasks	*Configuring Context-Based Access Control*
Security commands	• *Cisco IOS Security Command Reference: Commands A to C* • *Cisco IOS Security Command Reference: Commands D to L* • *Cisco IOS Security Command Reference: Commands M to R* • *Cisco IOS Security Command Reference: Commands S to Z*
PAM information and configuration tasks	*Configuring Port to Application Mapping*

Standards

Standards	Title
None	—

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	--

Technical Assistance

Description	Link
Technical Assistance Center (TAC) home page, containing 30,000 pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/public/support/tac/home.shtml

Firewall Support of Skinny Client Control Protocol

The following table lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn . An account on Cisco.com is not required.

Note The following table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 2 Feature Information for Zone-Based Policy Firewall

Feature Name	Releases	Feature Information
Firewall Support of Skinny Client Control Protocol	12.3(1)	The Firewall Support of Skinny Client Control Protocol (SCCP) feature enables Context-Based Access Control (CBAC) inspection to support the Voice over IP (VoIP) protocol, Skinny Client Control Protocol (SCCP).