



Cisco IOS Scripting with TCL Command Reference

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Cisco and the Cisco Logo are trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and other countries. A listing of Cisco's trademarks can be found at www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1005R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

© 2011 Cisco Systems, Inc. All rights reserved.



CONTENTS

[cli through telsh](#) 5

[cli](#) 6

[policy-list](#) 8

[scripting tcl encdir](#) 10

[scripting tcl init](#) 12

[scripting tcl low-memory](#) 14

[scripting tcl secure-mode](#) 16

[scripting tcl trustpoint name](#) 18

[scripting tcl trustpoint untrusted](#) 20

[telquit](#) 22

[telsafe](#) 24

[telsh](#) 26



cli through telsh

cli

To specify EXEC command-line interface (CLI) commands within a Command Scheduler policy list, use the **cli** command in kron-policy configuration mode. To delete a CLI command from the current policy list, use the **no** form of this command.

cli *command*

no cli *command*

Syntax Description

command

EXEC-mode CLI command that must not generate a prompt or allow interruption by a keystroke.

Command Default

No CLI commands are specified.

Command Modes

Kron-policy configuration (config-kron-policy)

Command History

Release

Modification

12.3(1)

This command was introduced.

12.2(33)SRA

This command was integrated into Cisco IOS Release 12.2(33)SRA.

12.2(33)SB

This command was integrated into Cisco IOS Release 12.2(33)SB.

12.2(33)SXI

This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

Use the **cli** command in conjunction with the **kron policy-list** command to create a policy list containing EXEC CLI commands to be scheduled to run on the router at a specified time. Use the **kron occurrence** and **policy-list** commands to schedule one or more policy lists to run at the same time or interval.

The Command Scheduler process is useful to automate the running of EXEC commands at recurring intervals, and it can be used in remote routers to minimize manual intervention.

Examples

The following example shows how to configure the EXEC command **cns image retrieve** within the policy list named three-day-list:

```
Router(config)# kron policy-list three-day-list
```

```
Router(config-kron-policy)# cli cns image retrieve server https://10.19.2.3/cns/image/  
status https://10.19.2.3/cnsstatus/imageinfo/
```

Related Commands

Command	Description
kron occurrence	Specifies schedule parameters for a Command Scheduler occurrence and enters kron-occurrence configuration mode.
kron policy-list	Specifies a name for a Command Scheduler policy and enters kron-policy configuration mode.
policy-list	Specifies the policy list associated with a Command Scheduler occurrence.

policy-list

To associate a policy list with a Command Scheduler occurrence, use the **policy-list** command in kron-occurrence configuration mode. To delete a policy list from the Command Scheduler occurrence, use the **no** form of this command.

policy-list *list-name*

no policy-list *list-name*

Syntax Description

<i>list-name</i>	Name of the policy list.
------------------	--------------------------

Command Default

No policy list is associated.

Command Modes

Kron-occurrence configuration (kron-config-occurrence)

Command History

Release	Modification
12.3(1)	This command was introduced.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.

Usage Guidelines

Use the **policy-list** command with the **kron occurrence** command to schedule one or more policy lists to run at the same time or interval. Use the **kron policy-list** command in conjunction with the **cli** command to create a Command Scheduler policy list containing EXEC command line interface (CLI) commands to be scheduled to run on the router at a specified time.

When the *list-name* is new, a policy list structure is created. When the *list-name* is not new, the existing policy list is edited.

The Command Scheduler process is useful to automate the running of EXEC commands at recurring intervals, and can be used in remote routers to minimize manual intervention.

Examples

The following example shows how to create a Command Scheduler occurrence named may and associate a policy list named sales-may with the occurrence:

```
Router(config)# kron occurrence may at 6:30 may 20 oneshot  
Router(config-kron-occurrence)# policy-list sales-may
```

Related Commands

Command	Description
cli	Specifies EXEC CLI commands within a Command Scheduler policy list.
kron occurrence	Specifies schedule parameters for a Command Scheduler occurrence and enters kron-occurrence configuration mode.
kron policy-list	Specifies a name for a Command Scheduler policy and enters kron-policy configuration mode.

scripting tcl encdir

To specify the default location of external encoding files used by the Tool Command Language (Tcl) shell, use the **scripting tcl encdir** command in global configuration mode. To remove the default location, use the **no** form of this command.

scripting tcl encdir *location-url*

no scripting tcl encdir

Syntax Description

location-url

The URL used to access external encoding files used by Tcl.

Command Default

Tcl does not use external encoding files.

Command Modes

Global configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.
12.2(31)SB	This command was integrated into Cisco IOS Release 12.2(31)SB.
12.2(33)SB	This command's behavior was modified and implemented on the Cisco 10000 series router for the PRE3 and PRE4.

Usage Guidelines

Character strings in Tcl are encoded using 16-bit Unicode characters. Different operating system interfaces or applications can generate character strings using other encoding methods. Use the **scripting tcl**

encdir command to configure a location URL for the external Tcl character encoding files to support the Tcl **encoding** command.

Tcl contains only a few character sets within the Tcl shell. Additional characters sets are loaded, as needed, from external files.

Cisco 10000 Series Router Usage Guidelines

In Cisco IOS Release 12.2(33)SB, the router removes the `no scripting tcl encdir` command from the default configuration.

Examples

The following example shows how to specify a default location for external encoding files to be used by Tcl:

```
Router# configure terminal
Router(config)# scripting tcl encdir tftp://10.18.117.23/file2/
```

Related Commands

Command	Description
scripting tcl init	Specifies an initialization script for the Tcl shell.
tclsh	Enables the Tcl shell and enters Tcl configuration mode.

scripting tcl init

To specify an initialization script for the Tool Command Language (Tcl) shell, use the **scripting tcl init** command in global configuration mode. To remove the initialization script, use the **no** form of this command.

scripting tcl init *init-url*

no scripting tcl init

Syntax Description

init-url

The URL used to access the initialization script to be used by Tcl.

Command Default

Tcl does not run an initialization script.

Command Modes

Global configuration

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.
12.2(31)SB	This command was integrated into Cisco IOS Release 12.2(31)SB.
12.2(33)SB	This command's behavior was modified and implemented on the Cisco 10000 series router for the PRE3 and PRE4.

Usage Guidelines

Use the **scripting tcl init** command when you want to predefine Tcl procedures to run in an initialization script. The initialization script runs when the Tcl shell is entered and saves manual sourcing of the individual scripts.

Cisco 10000 Series Router Usage Guidelines

In Cisco IOS Release 12.2(33)SB, the router removes the no scripting tcl init command from the default configuration.

Examples

The following example shows how to specify an initialization script to run when the Tcl shell is enabled:

```
Router# configure terminal
Router(config)# scripting tcl init ftp://user:password@172.17.40.3/tclscript/initfile3.tcl
```

Related Commands

Command	Description
scripting tcl encdir	Specifies the default location of external encoding files used by the Tcl shell.
tclsh	Enables the Tcl shell and enters Tcl configuration mode.

scripting tcl low-memory

To set a low memory threshold for free memory for Tool Command Language (Tcl)-based applications, use the **scripting tcl low-memory** command in global configuration mode. To remove the specific low memory threshold and return to using the default value, use the **no** form of this command.

scripting tcl low-memory *bytes*

no scripting tcl low-memory

Syntax Description

bytes

Specifies the low memory threshold. The memory threshold can be set from 0 to 4294967295 bytes.

Command Default

The default value is 25 percent of the available free memory at start up when Tcl initializes.



Note

The default is platform-specific. (It depends on how much memory is installed, and how much memory is free when Tcl initializes).

Command Modes

Global configuration (config)

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SRC	This command was integrated into Cisco IOS Release 12.2(33)SRC.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

Use the **scripting tcl low-memory** command to set the threshold for free memory. If minimum free RAM drops below this threshold, Tcl aborts the current script. This prevents the Tcl interpreter from allocating too much RAM and crashing the router.

Examples

The following example shows how to set the threshold for free memory when the Tcl shell is initialized:

```
Router# configure terminal
Router(config)# scripting tcl low-memory 33117513
```

Related Commands

Command	Description
scripting tcl encdir	Specifies the default location of external encoding files used by the Tcl shell.
scripting tcl init	Specifies an initialization script for the Tcl shell.
tclsh	Enables the Tcl shell and enters Tcl configuration mode.

scripting tcl secure-mode

To enable signature verification of the interactive Tool Command Language (Tcl) scripts, use the **scripting tcl secure-mode** command in global configuration mode. To disable signature verification of the interactive Tcl scripts, use the **no** form of this command.

scripting tcl secure-mode
no scripting tcl secure-mode

Syntax Description

This command has no arguments or keywords.

Command Default

The signature verification of the interactive Tcl scripts is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use the **scripting tcl secure-mode** command to enable signature verification of all Tcl scripts run on the router. By default, the signature verification of the interactive Tcl scripts is disabled. You must enable the signature verification in order to verify whether the Tcl scripts match their digital signature. That would indicate they have not been altered since the digital signature was generated. If the script does not contain the digital signature, the script may run in a limited mode for untrusted script (that is, a script that has failed signature verification) or may not run at all. After receiving the results from the signature verification, the scripts are executed.

A Cisco IOS Crypto image software is required to enable this command and configure the Signed Tcl Scripts feature. The Crypto configuration commands enable the Cisco x.509 certificate storage. The **scripting tcl secure-mode** command can be enabled after the Crypto configuration trustpoint commands are enabled.

The **scripting tcl trustpoint name** command must be configured with the **scripting tcl secure-mode** command to verify the integrity of Tcl script signatures run on the router. Both commands must be configured to fully operate the feature; otherwise, a syslog message is generated:

```
*Jun 13 17:35:14.219: %SYS-6-SCRIPTING_TCL_INVALID_OR_MISSING_SIGNATURE: tcl signing
validation failed on script signed with trustpoint name mytrust, cannot run the signed
TCL script.
```

In addition, the **crypto pki trustpoint name** command provided should contain a certificate that matches the certificate that was originally used to generate the digital signature on the Tcl script.

Examples

The following example shows how to enable signature verification of the interactive Tcl scripts:

```

Router(config)# crypto pki trustpoint mytrust
Router(ca-trustpoint)# enrollment terminal
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate mytrust
Enter the base 64 encoded CA certificate.

End with a blank line or the word "quit" on a line by itself
MIEuDCCA6GcAwIBAgIBADANBgkqhkiG9w0BAQQFADCBnjELMAkGALUEBhMCVVMx
EzARBgNVBAGTCKNhbgG1mb3JuaWEwETAPBgNVBACTCFNhbiBKB3N1MRwwGgYDVQK
ExNDAXNjbYBTEtXN0ZWZlZLCBjBmMuMQ4wDAYDVQQLEWV0U1NURzEWEWBBQGA1UEAxMN
Sm90bDIuY28yOGUwBWFubjEhMB8GCSqGSIb3DQEJARYSAmxhbdXRtYW5AY2IzY28uY29t
MB4XDTA2MTEtXNzE3NTgwMV0eXDTA5MTEtXNjE3NTgwMV0wZ4xZCzAJBgNVBAYTAlVT
MRMtMGEyQDVQIIEwDPWxpZm9ybmlhMREwDwYDVQQHEWhlYWw4Sm9zZTEtCmB0GA1UE
CRMTQ2l2Y28yOGUwZGVtcyYwZS5lJlEOMAwGA1UECjMFTlNTVEcxZjAUBGNVBAMT
DUpvaG4gTGFlbG1hbm4xIETAFBgkqhkiG9w0BCQEWEmpsYXV0bWFWUz02NvLmNmV
bTCCASiIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALxtqTMCirMb+CdyWLuH
oWAM8CEJdWQggL7MVBh0i3TSMd/wW2XBB9biBtdlH6jHsJiCiOwAR5OorakwPyf7
mvrJ2PqJALs+Vn93VBKIG6rzU14+wdOx686BvdIdZvEjQpBR0iYtZfawVW70aLMV
bd7/B7vF1SG1YK9y1tX9p9nZyZ0x470AXetwOaGinvlG7VNuTXaASBLUjCRZsIlz
SBrXXedBz26+BuWm1FK45EYSlag5Rt9CGXWMBqzx91iyhrJ3zDmDKExa45yKJET
mAgDVmcpeteJtufI47UDZJK30g4M0Myx/8RWghmJ54qRGLF2BEPmMfQKnpL108MAL
Q8sCAwEAAaO0B/jCB+ZadBgNVHQ4EFgQU9/ToDvbMR3JfJ4xExa4X470nFq5khgcsG
AlUdIwSBwzCBwIAU9/ToDvbMR3JfJ4xExa4X470nFq5mhgaSkgaEwgZ4xCzAJBgNV
BAYTAlVTMRMtMGEyQDVQIIEwDPWxpZm9ybmlhMREwDwYDVQQHEWhlYWw4Sm9zZTEt
CmB0GA1UECRMTQ2l2Y28yOGUwZGVtcyYwZS5lJlEOMAwGA1UECjMFTlNTVEcxZjAUBG
NVBAMTDTUpvaG4gTGFlbG1hbm4xIETAFBgkqhkiG9w0BCQEWEmpsYXV0bWFWUz02NvLmNmV
bYBIBADAMBGNVRHREMBETAQAQH/MA0GCSqGSIb3DQEBAUAA4IBAQBtEs/4
MQEn9pT+XPCPg20bQU8y2AadI+I34YK+fDHsFOh68zhpsZ2T2VpNEvkFXpAdhgr
7dKngTwtCl481v70iNFViQVL+inNrZwWmXoTnUNCK7Hc5kHkXt6cj0mvsefVUzx
Xl70mauTHESRVlrmY0RjXsSrE1LerZYsuV5hBfndand+/rErmp2HvYfdntLnK8SzmXJ
5lwe/ET2QtYNGor00B1Lesowfslr3LhHi4w+5is7rLEqGnw/NFQ1UirlzLh80eB4M
wcpBJtsLauJ6ZUJQ17IqdsWsa3fhD5qq0/k8P9Z0YAYrf3+MFQr4ibvsYvHl0087
o2JsIgw4qz34pqNh
Certificate has the following attributes:
    Fingerprint MD5: 1E327DDB 330936EB 4FD1133E
    Fingerprint SHA1: EE7FF9F4 05148842 B9D50FAC D76FDC9C E0703246
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
Router(config)# scripting tcl secure-mode

Router(config)# scripting tcl trustpoint name mytrust

```

Related Commands

Command	Description
<code>scripting tcl trustpoint name</code>	Associates an existing configured trustpoint name with a certificate to verify Tcl scripts.

scripting tcl trustpoint name

To associate an existing configured trustpoint name with a certificate to verify Tool Command Language (Tcl) scripts, use the **scripting tcl trustpoint name** command in global configuration mode. To remove an existing configured trustpoint name, use the **no** form of this command.

scripting tcl trustpoint name *name*

no scripting tcl trustpoint name *name*

Syntax Description

name

Name of the configured trustpoint name associated with a certificate. Only one name can be associated with one certificate.

Command Default

A trustpoint name is not associated with a certificate to verify the Tcl scripts.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use the **scripting tcl trustpoint name** command to associate an existing configured trustpoint name with a certificate to verify Tcl scripts. This way, Tcl identifies which certificate is used for verifying the Tcl scripts. The name must match an existing configured trustpoint name, otherwise, the command is rejected with an error message on the console. You can enter the command multiple times and configure multiple trustpoint names. Once you enter the command, you cannot modify the trustpoint name. However, you can remove the trustpoint name using the **no** form of the command. You must individually remove each name. When the last name is removed, no signature checking is performed, and the untrusted script (that is, a script that has failed signature verification) action configured by the **scripting tcl trustpoint untrusted** command is also removed.

A Cisco IOS Crypto image software is required to enable this command and configure the Signed Tcl Scripts feature. The Crypto configuration commands enable the Cisco x.509 certificate storage. The **scripting tcl trustpoint name** command can be enabled after the Crypto configuration trustpoint commands are enabled.

The **scripting tcl secure-mode** command must be configured with the **scripting tcl trustpoint name** command to verify the integrity of Tcl script signatures run on the router. Both commands must be configured to fully operate this feature; otherwise, a syslog message is generated:

```
*Jun 13 17:53:31.659: %SYS-6-SCRIPTING_TCL_SECURE_TRUSTPOINT: scripting tcl secure-mode
```

is enabled, however no scripting tcl trustpoint names configured, cannot verify signed TCL script.

In addition, the **crypto pki trustpoint** *name* command provided should contain a certificate that matches the certificate that was originally used to generate the digital signature on the Tcl script.

Examples

The following example shows how the **scripting tel trustpoint name** command is used to associate existing trustpoint names. Different names can be used for different departments with certificates:

```

Router(config)# crypto pki trustpoint mytrust
Router(ca-trustpoint)# enrolment terminal
Router(ca-trustpoint)# exit
Router(config)# crypto pki authenticate mytrust
Enter the base 64 encoded CA certificate.

End with a blank line or the word "quit" on a line by itself
MIEuDCCA6CgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBnJELMAKGA1UEBhMCVVMx
EzARBgNVBAgTCKNhbgG1mb3JuaWEwExTAPBgNVBAcTCFNhbiBKB3N1MRwwGgYDVQVK
ExNDAXNjYyBTEuXN0ZW1zLCBjbmMuMQ4wDAYDVQQLEWV0U1NURzEWMWQGA1UEAxMN
Sm90b1BMYX0w0bWfuYjEhMB8GCSqGSIb3DQEJARYSAWxhbmRlXHRtYW5AY2l2Y28uY29t
MB4XDTA2MTEuXN0Z3NTgwMV0uXDTA5MTEuXN0Z3NTgwMV0wZ4xZCzAJBgNVBAYTAlVT
MRMwEQYDVQVKIEdpYWwZm9ybmlhMREwDwYDVQKEHwTYWwZSm9zZTECMBoGA1UE
CHM0Q2l2Y28uY312dGVtcywgSW5jLjEOMAwGA1UECjMFTlNTVEcxFjAUBgNVBAMT
DUpvaG4gTGFG1dG1hbm4xITAFBgkqhkiG9w0BCQEWEmPsYXV0bWfuYQGNpC2NvLmNv
bTCCAS1dWQYJKoZIhvcNAQEBBQADggEPADCCAQcCggEBAALxtqTMCirMb+CdyLUhL
oWAM8CEJDuQggL7MWBhoi3TSMd/wW2XBB9giBtdH6jHsJc1oWAr50orakwPyf7
mvrRJ2PqJALs+Vn93VBKIG6rZU14+wd0x686BvdIdZvEJQPbRoIyTzfzWV70aLmV
bd7/B7vF1SG1YK9yltX9p9nzyZ0x470AXetwoAGinv1G7VNUtXaASBLUjCRZsIlz
SBrXXedBz6+BuOwmlF424EYSlag5Rt9RGXhXMBqzX91iyhrJ3zDDmkExa45yKJET
mAGDVMcpeteJtiff4UDZJK30g4MbMyr/+c8WGhmJ54qrL9BZEPmZMQKNP1018MAl
Q8sCAwEAAQOB/jCB+ZadbGnVHQ4EFgQUU9/ToDvbmMR3JfJ4xEa4X47oNFq5mhwgS
A1UdIwSwBwCBWIAU9/ToDvbmMR3JfJ4xEa4X47oNFq5mhwgSkgAEwgZ4xCzAJBgNV
BAYTAlVTMRMEIEdpYWQYDVQVKIEdpYWwZm9ybmlhMREwDwYDVQKEHwTYWwZSm9zZTEc
MBoGA1UEChMTQ2l2Y28uY312dGVtcywgSW5jLjEOMAwGA1UECjMFTlNTVEcxFjAUB
BgNVBAMTDUpvaG4gTGFG1dG1hbm4xITAFBgkqhkiG9w0BCQEWEmPsYXV0bWfuYQGN
pC2NvLmNvbYyIBADANBgNVMEBETAADAg/MA0GCSqGSIb3DQEBAUAA4IBAQQBES/4
MQEn9pT+XPCPg2ObQu8y2AadI+I34YK+fDHsFOh68hZhszTN2VpNEvKfFXpAdhgr
7DKngTwTC1a481v70iNFViQVL+inNrZwMxcoTnUNCK7Hc5kHkXt6cj0mvsefVUuz
L170mauE2SRVl0rYbJXssRfELerZysuvfHbFdand+/rErmP2HvYfntLhKDSzmXJ
51wE/bt2QYNGor00W1LesowfslR3LhH14wn+5is7fHkLgNw/NuT1Ur1zH10eB4m
wcpB1JsLaJu6ZUJQ17IqdsWsa3fhD5qq0/k8P9z0YAYrf3+MFqr4ibvsYvH10087
02JslgW4qz34pqNh

Certificate was the following attributes:
    Fingerprint MD5: 1E327DBB 330936EB 2FB8EACB 4FD1133E
    Fingerprint SHA1: EE7FF9F4 05148842 B9D50FAC D76DFC9C E0703246

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
Router(config)# scripting tcl secure-mode

Router(config)# scripting tcl trustpoint name mytrust
Router(config)# scripting tcl trustpoint name dept_accr
Router(config)# scripting tcl trustpoint name dept_hr

```

Related Commands

Command	Description
scripting tcl secure-mode	Enables signature verification of the interactive Tcl scripts.

scripting tcl trustpoint untrusted

To allow the interactive Tool Command Language (Tcl) scripts to run regardless of the scripts failing the signature check, use the **scripting tcl trustpoint untrusted** command in global configuration mode. To disallow the interactive Tcl scripts to run regardless of the scripts failing the signature check, use the **no** form of this command.

scripting tcl trustpoint untrusted {execute | safe-execute | terminate}

no scripting tcl trustpoint untrusted

Syntax Description

execute	Executes Tcl scripts even if the signature verification fails. Caution Use of this keyword is usually not recommended because the signature verification is not performed if the execute keyword is configured.
safe-execute	Executes the Tcl script in safe mode if the signature verification fails.
terminate	Does not run the Tcl script if the signature verification fails. The default keyword is terminate .

Command Default

No script that fails signature verification can run; the script immediately stops.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use the **scripting tcl trustpoint untrusted** command to allow the interactive Tcl scripts to run regardless of the scripts failing the signature check or in untrusted mode. The untrusted script (that is, a script that has failed signature verification) is not safe to use.

**Caution**

Use of the **execute** keyword is usually not recommended because the signature verification is not performed.

The **execute** keyword is provided for internal testing purposes and to provide flexibility. For example in a situation where a certificate has expired but the other configurations are valid and you want to work with the existing configuration, then you can use the **execute** keyword to work around the expired certificate.

The **safe-execute** keyword allows the script to run in safe mode. You can use the **tclsafe** command and also enter the interactive Tcl shell safe mode to explore the safe mode Tcl commands that are available. In order to get a better understanding of what is available in this limited safe mode, use the **tclsafe** Exec command to explore the options.

The **terminate** keyword stops any script from running and reverts to default behavior. The default policy is to terminate. When the last trustpoint name is removed, the untrusted action is also removed. The untrusted action cannot be entered until at least one trustpoint name is configured for Tcl.

**Note**

This command only applies to the Tcl shell; it does not impact other components that make use of Tcl. For example, Embedded Event Manager (EEM) cannot perform any signature checking.

Examples

The following example shows how to execute the Tcl script in safe mode if the signature verification fails:

```
Router(config)# scripting tcl trustpoint untrusted safe-execute
```

Related Commands

Command	Description
scripting tcl trustpoint name	Associates an existing configured trustpoint name with a certificate to verify Tcl scripts.
tclsafe	Enables the interactive Tcl shell untrusted safe mode.

tclquit

To quit the interactive Tool Command Language (Tcl) shell, use the **tclquit** command in privileged EXEC mode.

tclquit

Syntax Description

This command has no arguments or keywords.

Command Default

The Tcl shell is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Examples

The following example shows how to disable the interactive Tcl shell:

```
Router# tclsh
Router(tcl)#
Router(tcl)# tclquit
Router#
```

Related Commands

Command	Description
tclsh	Enables the interactive Tcl shell.
tclsafe	Enables the interactive Tcl shell untrusted safe mode.

tclsafe

To enable the interactive Tool Command Language (Tcl) shell untrusted safe mode, use the **tclsafe** command in privileged EXEC mode. To exit from the safe mode, use the **exit** or the **tclquit** command.

tclsafe

Syntax Description

This command has no arguments or keywords.

Command Default

The Tcl shell untrusted safe mode is disabled.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(15)T	This command was introduced.

Usage Guidelines

Use the **tclsafe** command when you want to manually run Tcl commands from the Cisco IOS command-line interface (CLI) in untrusted safe mode. When you use the **tclsafe** command and enter the interactive Tcl shell safe mode, you can explore the safe mode Tcl commands that are available. When a script fails the signature check for a configured trustpoint name, it is determined to be untrusted. Untrusted Tcl scripts execute in limited safe mode, if **scripting tcl trustpoint untrusted safe-execute** command is configured. In order to get a better understanding of what is available in this limited safe mode, use the **tclsafe** Exec command to explore the options.

After Tcl commands are entered they are sent to a Tcl interpreter. If the commands are recognized as valid Tcl commands, the command is executed and the result is sent to the tty. If a command is not a recognized Tcl command, it is sent to the Cisco IOS CLI parser. If the command is not a Tcl or Cisco IOS command, two error messages are displayed.

A predefined Tcl script can be created outside of Cisco IOS software, transferred to flash or disk memory, and run within Cisco IOS software. It is also possible to create a Tcl script and precompile the code before running it under Cisco IOS software. To exit from this mode, use the **exit** or the **tclquit** command to disable the use of the Tcl shell and return to privileged EXEC mode.

You can also use the **tclsafe** command with a script name such as **tclsafe disk0:hello.tcl**. The script **hello.tcl** executes immediately and allows you to exit from the untrusted safe mode and return to privileged EXEC mode.

Examples

The following example shows how to enable the Tcl shell untrusted safe mode and run **info commands**:

```
Router# tclsafe
Router(safe)(tcl)# info commands
info commands
tell socket subst open eof glob list pid time eval lrange tcl_trace fblocked lsearch gets
case lappend proc break variable llength return linsert error catch clock info split
array if fconfigure concat join lreplace source fcopy global switch update close cd for
file append format read package set binary namespace scan seek while flush after vwait
uplevel continue hostname foreach rename fileevent regexp upvar unset encoding expr load
regsub interp history puts incr lindex lsort string
```

The following example shows how to execute the script **hello.tcl** to exit from the untrusted safe mode and return to privileged EXEC mode.

```
Router# tclsafe disk0:hello.tcl
```

Related Commands

Command	Description
scripting tcl trustpoint untrusted	Allows the interactive Tcl scripts to run regardless of the scripts failing the signature check.
tclquit	Quits Tcl shell.
tclsh	Enables the interactive Tcl shell and enters Tcl configuration mode.

tclsh

To enable the interactive Tool Command Language (Tcl) shell, use the **tclsh** command in privileged EXEC mode.

tclsh

Syntax Description

This command has no arguments or keywords.

Command Default

The Tcl shell is disabled.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.3(2)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2SX	This command is supported in the Cisco IOS Release 12.2SX train. Support in a specific 12.2SX release of this train depends on your feature set, platform, and platform hardware.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.

Usage Guidelines

Use the **tclsh** command when you want to run Tcl commands from the Cisco IOS command-line interface (CLI). When the interactive Tcl shell is enabled and Tcl configuration mode is entered, Tcl commands can be entered line by line or a predefined Tcl script can be run. After Tcl commands are entered they are sent to a Tcl interpreter. If the commands are recognized as valid Tcl commands, the command is executed and the result is sent to the tty. If a command is not a recognized Tcl command, it is sent to the Cisco IOS CLI parser. If the command is not a Tcl or Cisco IOS command, two error messages will be displayed.

A predefined Tcl script can be created outside of Cisco IOS software, transferred to Flash or disk memory, and run within Cisco IOS software. It is also possible to create a Tcl script and precompile the code before running it under Cisco IOS.

Use the **exit** or the **tclquit** command to disable the use of the Tcl shell and return to privileged EXEC mode.

Examples

The following example shows how to enable the Tcl interactive shell:

```
Router# tclsh  
Router(tcl)#
```

Related Commands

Command	Description
scripting tcl encdir	Specifies the default location of external encoding files used by the Tcl shell.
scripting tcl init	Specifies an initialization script for the Tcl shell.

