DDoS Mitigation Support on CGSE

Distributed denial-of-service (DDoS) attacks target network infrastructures or computer services resources. The primary goal of DDoS attacks is to deny legitimate users access to a particular computer or network resources, which results in service degradation, loss of reputation, and irretrievable data loss.

DDoS Defense is based on mitigating the attack traffic at entry point into the network.

DDoS Mitigation is the process of detecting increasingly complex and deceptive assaults and mitigating the effects of the attack to ensure business continuity and resource availability.

Threat Management System software is ported to the Cisco Carrier Grade Services Engine (CGSE) to mitigate the attacks and send clean traffic back to the targeted host or network. Cisco CGSE is an integrated multi-CPU service module offering carrier-class performance and scale in support of various applications and services.

For detailed information about DDoS mitigation support concepts, configuration tasks, and examples, see the *Implementing DDoS Mitigation Support on CGSE* module of the *System Security Configuration Guide for Cisco CRS Routers*.

Implementing DDoS Mitigation Support on CGSE

What is Distributed Denial-of-service (DDoS)?

Distributed denial-of-service (DDoS) is one in which numerous compromised systems attack a single target system, thereby causing denial of service for users of the targeted system. DDoS attacks target network infrastructures or computer services resources. The primary goal of DDoS attacks is to deny legitimate users access to a particular computer or network resources, which results in service degradation, loss of reputation, and irretrievable data loss.

What is DDoS Mitigation?

DDoS Mitigation is the process of detecting increasingly-complex and deceptive assaults, and mitigating the effects of such attacks, to ensure business operations continuity and resource availability. DDoS mitigation is based on mitigating the attack traffic at entry point into the network.

Complete DDoS protection provides these benefits:

- Detection and Mitigation of DDoS attacks
- Distinguish good traffic from bad traffic to preserve business continuity
- Include performance and architecture to deploy upstream to protect all points of vulnerability
- Maintain reliable and cost-efficient scalability

Implementing DDoS Mitigation Support on CGSE

Threat Management System (TMS) software is ported to the Cisco Carrier Grade Services Engine (CGSE) to mitigate the attacks, and to send clean traffic back to the targeted host or network. The Cisco CGSE is a single-slot module supported on all models of Cisco's proven high-end carrier-class routing system: CRS-1 and CRS-3. CGSE offers carrier-class performance and scale in support of various applications and services.

For more information on Implementing DDoS Mitigation Support on CGSE, refer to the *Implementing DDoS Mitigation Support on CGSE* chapter in the *System Security Configuration Guide for Cisco CRS Routers*.

For a complete description of the DDoS Mitigation Support commands, refer to the *DDoS Mitigation Support on CGSE Commands* module of the *System Security Command Reference for Cisco CRS Routers*.

Feature History for Implementing DDoS Mitigation Support on CGSE

Release	Modification
Release 4.2.3	This feature was introduced.

Restrictions for Implementing DDoS Mitigation

This solution does not provide support for these features on Cisco CRS Router:

- TACACS
- CPU performance evaluation for the TMS–CGSE application
- Latency or Jitter on box performance analysis
- H/W Time stamping of packets
- Co-existence of other services (e.g. CCN, CGN) in the same CGSE blade with TMS–CGSE scrubber
- Incremental routing requirements

Prerequisites for Implementing DDoS Mitigation

These prerequisites are required to implement DDoS Mitigation support on CGSE:

- Install the CGSE in your Cisco CRS chassis. For information about how to install a CGSE, refer to the *Cisco CRS 20 Gbps Carrier Grade Services Engine Physical Layer Interface Module Installation Note*, made available at URL: http://www.cisco.com/en/US/partner/docs/routers/crs/crs1/plim/installation/guide/20gbpscrscgseplim.html

Installing and Activating the PIE

The Package Installation Envelope (PIE) files, are installable software files with the .pie extension. PIE files are used to copy one or more software components onto the router. A PIE may contain a single component, a group of components (called a package), or a set of packages (called a composite package).

Use the **show install committed** command in EXEC mode to verify the committed software packages.

You must install and activate the services PIE before you install and use the TMS–CGSE software. Download the *hfr-services-px.pie* to a TFTP server.

For more information about installing PIEs, refer to *Upgrading and Managing Cisco IOS XR Software section* of the *System Management Configuration Guide for Cisco CRS Routers*.

Note The TMS–CGSE software is part of a separate image that you download from Cisco.com. For information about the specific images, refer to the *Release Notes for Cisco CRS-1 and Cisco CRS-3 for Cisco IOS XR Software Release 4.2.3.*

SUMMARY STEPS

1. **admin**
2. **install add** *tftp://<IP address of tftp server>/<location of pie on server>*
3. **install activate** *device:package*
4. **install commit**
5. **exit**
6. **show install committed**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**admin** **Example:** `RP/0/RP0/CPU0:router# admin`	Enters administration EXEC mode.

	Command or Action	Purpose
Step 2	**install add** *tftp://<IP address of tftp server>/<location of pie on server>* **Example:** `RP/0/RP0/CPU0:router(admin)# install add tftp://172.201.11.140/auto/tftp-users1/pie/`	Copies the contents of a package installation envelope (PIE) file to a storage device.
Step 3	**install activate** *device:package* **Example:** `RP/0/RP0/CPU0:router(admin)# install activate disk0:hfr-services-px.pie`	Activates the respective package and adds more functionality to the existing software.
Step 4	**install commit** **Example:** `RP/0/RP0/CPU0:router(admin)# install commit`	Saves the active software set to be persistent across designated system controller (DSC) reloads.
Step 5	**exit** **Example:** `RP/0/RP0/CPU0:router(admin)# exit`	Exits from the admin mode.
Step 6	**show install committed** **Example:** `RP/0/RP0/CPU0:router# show install committed`	Shows the list of the committed software packages.

Copying TMS-CGSE RPM Package Manager to Route Processor

Perform this task to copy RPM Package Manager (RPM) of TMS–CGSE to Route Processor (RP) disk and to a standby RP.

Note RPM Package Manager is a package management system. The name RPM refers to two things: software packaged in the .rpm file format, and the package manager itself. RPM was intended primarily for GNU/Linux distributions; the file format is the baseline package format of the Linux Standard Base.

Copy the TMS-CGSE RPM to a primary RP. You should also copy the RPM to a standby RP to enable TMS to operate in case of a route processor switchover or failover.

We recommend to store the **.rpm** image on a flash card.

Before you begin

Download the TMS–CGSE RPM image using TFTP, and store it in the "tftp root" directory.

SUMMARY STEPS

1. **configure**
2. **tftp ipv4 server homedir** *tftp-home-directory*

3. **commit**
4. **copy tftp:**//*<IP address of tftp server> <location of TMS–CGSE RPM image on tftp server>/<TMS–CGSE RPM image filename >* **disk0:***<destination filename>*
5. **copy disk0:***<TMS–CGSE RPM image name>* **location***<R/S/I of Active RP>* **disk0:location***<R/S/I of Standby RP>*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**tftp ipv4 server homedir** *tftp-home-directory* **Example:** `RP/0/RP0/CPU0:router(config)# tftp ipv4 server homedir disk0`	Enables the TFTP server or a feature running on the TFTP server.
Step 3	**commit**	
Step 4	**copy tftp:**//*<IP address of tftp server> <location of TMS–CGSE RPM image on tftp server>/<TMS–CGSE RPM image filename >* **disk0:***<destination filename>* **Example:** `RP/0/RP0/CPU0:router# copy tftp://198.51.100.1/tftp_directory/tms-cgse.rpm disk0:tms-cgse.rpm`	Copies the TMS–CGSE RPM image to disk0:
Step 5	**copy disk0:***<TMS–CGSE RPM image name>* **location***<R/S/I of Active RP>* **disk0:location***<R/S/I of Standby RP>* **Example:** `RP/0/RP0/CPU0:router# copy disk0:tms-cgse.rpm location 0/RP0/CPU0 disk0: location 0/RP1/CPU0`	Copies the TMS–CGSE RPM image to the standby RP disk0: **Note** Use **show hfr** command to identify the active RP and standby RP

How to Implement DDoS Mitigation Support on CGSE

To implement DDoS Mitigation Support, perform the tasks described in this section. The TMS application hosted on CGSE implements DDoS Mitigation Support on CGSE. Perform these procedures in the order presented to host the TMS application on CGSE.

Configuring the CGSE Service Role as Service Engine Service Hosting (SESH)

Configure the CGSE service role as Service Engine Service Hosting (SESH). This configuration is done to allow CGSE to start the TMS–CGSE service.

Important The removal of the service role is strictly not recommended while the card is active. This puts the card into a FAILED state, and impacts service.

SUMMARY STEPS

1. **configure**
2. **hw-module service sesh location**<*R/S/I*>
3. **commit**
4. **show running-config service sesh**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**hw-module service sesh location**<*R/S/I*> **Example:** `RP/0/RP0/CPU0:router(config)# hw-module service sesh location 0/1/CPU0`	Configures the service role as SESH for the specified CGSE location in rack/slot/interface format.
Step 3	**commit**	

Step 4	**show running-config service sesh**
	Example:
	`RP/0/RP0/CPU0:router# show running-config service sesh`
	`Wed Jul 11 14:24:31.560 PST`
	`service sesh sesh1`
	`service-location preferred-active 0/1/CPU0`
		Shows the location of the SESH.

Configuring the Service Infrastructure Interface

Configure the service infrastructure (ServiceInfra) interface and associate it with a CGSE module. This configuration needs to be done to send the infrastructure traffic to CGSE and to download the TMS–CGSE. Reboot the CGSE module after ServiceInfra interface configuration and association. Each CGSE should have one ServiceInfra interface.

Note The ServiceInfra interface IP address should be configured with a network mask. The maximum limit for this mask is /29. The network should be configured with a minimum of 5 hosts. You must assign a x.x.x.1 IP address. Other ServiceInfra IP addresses do not work for SESH.

SUMMARY STEPS

1. **configure**
2. **interface ServiceInfra** <*id*>
3. **ipv4 address** <*A.B.C.D*>/<*prefix*>
4. **service-location** <*R/S/I*>
5. **commit**
6. **hw-module location** <*R/S/I*> **reload**
7. **show services role**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**interface ServiceInfra** *<id>* **Example:** `RP/0/RP0/CPU0:router(config)# interface ServiceInfra 1`	Enters interface configuration mode for the service infrastructure.
Step 3	**ipv4 address** *<A.B.C.D>/<prefix>* **Example:** `RP/0/RP0/CPU0:router(config-if)# ipv4 address 100.1.1.1/29`	Sets the IP address for this interface.
Step 4	**service-location** *<R/S/I>* **Example:** `RP/0/RP0/CPU0:router(config-if)# service-location 0/1/CPU0`	Location of the CGSE you set in **Configuring the Service Role** section in rack/slot/interface format. **Note** To determine where the CGSE modules are installed in the chassis, use the **show platform** command in the EXEC mode. The **show platform** command displays the list of cards that includes CGSE modules with their service location.
Step 5	**commit**	
Step 6	**hw-module location** *<R/S/I>* **reload** **Example:** `RP/0/RP0/CPU0:router# hw-module location 0/1/CPU0 reload`	Reloads CGSE. Use the **show platform** command to monitor the CGSE boot state. The card is fully booted when it switches from the initially BOOTING state to the OK state.

Step 7	**show services role**
	Example:
	`RP/0/RP0/CPU0:router# show services role`
	`Node  Configured Role Enacted Role Enabled Services`
	`--------------------------------------------------`
	`0/1/CPU0 SESH           SESH        ServiceInfra`
		Displays information about the configured service role.

Configuring ServiceEngine–ServiceHost Instance

Configure the SESH instance to run on the CGSE node. The service location specifies the CGSE card location. One active card is supported with no failover, so only the preferred-active argument is supported.

Note Before configuring the SESH instance and reloading it, wait approximately 15 minutes for the CGSE to come up in the OK state.

SUMMARY STEPS

1. **configure**
2. **service sesh** *<name of the sesh instance>*
3. **service-location preferred-active** *<R/S/I>*
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**service sesh** *<name of the sesh instance>* **Example:** `RP/0/RP0/CPU0:router(config)# service sesh sesh1`	Configures service hosting instance.
Step 3	**service-location preferred-active** *<R/S/I>* **Example:** `RP/0/RP0/CPU0:router(config-sesh)# service-location preferred-active 0/1/CPU0`	Specifies the CGSE card location in rack/slot/interface format for the SESH instance. Only one active card is supported with no failover.
Step 4	**commit**	

Configuring Service Application Interfaces

Before configuring the TMS-CGSE software on a CGSE module, configure three Service Application (ServiceApp) interfaces and bind the interfaces with the created SESH instance. Configure one ServiceApp interface for the management path to CGSE. Configure the other two ServiceApp interfaces for the outgoing (offramp) traffic to the TMS-CGSE and for the incoming (onramp) traffic from the TMS-CGSE.

SUMMARY STEPS

1. **configure**
2. **vrf** *<vrf name>*
3. **commit**
4. **interface ServiceApp** *<ID>*
5. **description** *string*
6. **ipv4 address** *<A.B.C.D>/<prefix>*
7. **service sesh** *<name of the sesh instance>*
8. **interface ServiceApp** *<ID>*
9. **description** *string*
10. **ipv4 address** *<A.B.C.D>/<prefix>*
11. **service sesh** *<name of the sesh instance>*
12. **interface ServiceApp** *<ID>*
13. **description** *string*
14. **vrf** *<vrf name>*
15. **ipv4 address** *<A.B.C.D>/<prefix>*
16. **service sesh** *<name of the sesh instance>*

17. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**vrf** *<vrf name>* **Example:** `RP/0/RP0/CPU0:router(config)# vrf arbor-tms`	Configures the VRF reference.
Step 3	**commit** **Example:** `RP/0/RP0/CPU0:router(config)# commit`	Use the **commit** command to save the configuration changes to the running configuration file, and remain within the configuration session.
Step 4	**interface ServiceApp** *<ID>* **Example:** `RP/0/RP0/CPU0:router(config)# interface ServiceApp 11`	Enters the interface configuration mode for the service application.
Step 5	**description** *string* **Example:** `RP/0/RP0/CPU0:router(config-if)# description tms1 mgmt interface`	Creates a description for the Service Application Interface.
Step 6	**ipv4 address** *<A.B.C.D>/<prefix>* **Example:** `RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.10.76.1/29`	Sets the IP address for the management interface.
Step 7	**service sesh** *<name of the sesh instance>* **Example:** `RP/0/RP0/CPU0:router(config-if)# service sesh sesh1`	Associates the interface with the SESH service instance.
Step 8	**interface ServiceApp** *<ID>* **Example:** `RP/0/RP0/CPU0:router(config)# interface ServiceApp 21`	Enters the interface configuration mode for the service application.
Step 9	**description** *string* **Example:** `RP/0/RP0/CPU0:router(config-if)# description tms1 scrb ingress interface`	Creates a description for the Service Application Interface.
Step 10	**ipv4 address** *<A.B.C.D>/<prefix>* **Example:**	Sets the IP address for the scrubber ingress interface.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-if)# ipv4 address 204.0.0.1/24`	
Step 11	**service sesh** <*name of the sesh instance*> **Example:** `RP/0/RP0/CPU0:router(config-if)# service sesh sesh1`	Associates the interface with the SESH service instance.
Step 12	**interface ServiceApp** <*ID*> **Example:** `RP/0/RP0/CPU0:router(config)# interface ServiceApp 22`	Enters the interface configuration mode for the service application.
Step 13	**description** *string* **Example:** `RP/0/RP0/CPU0:router(config-if)# description tms1 scrb egress interface`	Creates a description for the ServiceApp interface.
Step 14	**vrf** <*vrf name*> **Example:** `RP/0/RP0/CPU0:router(config-if)# vrf arbor-tms`	Places the service interface in VRF. **Note** One ServiceApp interface (either onramp or offramp) must be in VRF to avoid loops.
Step 15	**ipv4 address** <*A.B.C.D*>/<*prefix*> **Example:** `RP/0/RP0/CPU0:router(config-if)# ipv4 address 205.0.0.1/24`	Sets the IP address for the scrubber egress interface.
Step 16	**service sesh** <*name of the sesh instance*> **Example:** `RP/0/RP0/CPU0:router(config-if)# service sesh sesh1`	Associates the interface with the SESH service instance.
Step 17	**commit**	

Configuring TMS–CGSE Service and Applications

To enable the TMS–CGSE Service and Applications, configure them first.

Create a Service Engine Service Hosting (SESH) instance and bind the ServiceApp interfaces to CGSE module while configuring TMS-CGSE.

SUMMARY STEPS

1. **configure**
2. **service sesh** <*name of the sesh instance*>
3. **service-location preferred-active** <*R/S/I*>
4. **service-type** <*service type name*> <*service instance name*>
5. **description** *string*

6. **package** *<name of the TMS–CGSE RPM image >*
7. **application tms-mgmt**
8. **interface ServiceApp** *<ID>*
9. **remote ipv4 address** *<A.B.C.D>/<prefix>*
10. exit
11. exit
12. **application tms-scrb**
13. **map ingress-interface ServiceApp***<ID>* **egress-interface ServiceApp** *<ID>*
14. **commit**
15. **show run service sesh**
16. **show service sesh instance**<name of instance>

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**service sesh** *<name of the sesh instance>* **Example:** `RP/0/RP0/CPU0:router(config)# service sesh sesh1`	Configures service hosting instance.
Step 3	**service-location preferred-active** *<R/S/I>* **Example:** `RP/0/RP0/CPU0:router(config-sesh)# service-location preferred-active 0/1/CPU0`	Specifies the CGSE card location in rack/slot/interface format for the SESH instance. Only one active card is supported with no failover.
Step 4	**service-type** *<service type name> <service instance name>* **Example:** `RP/0/RP0/CPU0:router(config-sesh)# service-type ddos-tms tms1`	Sets the service type.
Step 5	**description** *string* **Example:** `RP/0/RP0/CPU0:router(config-tms1)# description ddos TMS instance 1`	Creates a description for the service.
Step 6	**package** *<name of the TMS–CGSE RPM image >* **Example:** `RP/0/RP0/CPU0:router(config-tms1)# package tms-cgse.rpm`	Adds the TMS–CGSE image that is part of the instance. **Note** The TMS–CGSE RPM image should be in the tftp_root directory. It takes the TMS–CGSE application approximately 10 minutes to start executing, after committing the configuration.
Step 7	**application tms-mgmt** **Example:**	Specifies the TMS management application.

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router(config-tms1)# application tms-mgmt`	
Step 8	**interface ServiceApp** *<ID>* **Example:** `RP/0/RP0/CPU0:router(config-tms-mgmt)# interface ServiceApp 11`	Enters the interface mode of the service application.
Step 9	**remote ipv4 address** *<A.B.C.D>/<prefix>* **Example:** `RP/0/RP0/CPU0:router(config-if)# remote ipv4 address 10.10.76.2/29`	Specifies the remote IPv4 address of the service application. **Note** Remote management IP requires a minimum /29 mask.
Step 10	exit **Example:** `RP/0/RP0/CPU0:router(config-if)# exit`	Exits the Interface configuration mode.
Step 11	exit **Example:** `RP/0/RP0/CPU0:router(config-tms-mgmt)# exit`	Exits the TMS Management configuration mode.
Step 12	**application tms-scrb** **Example:** `RP/0/RP0/CPU0:router(config-tms1)# application tms-scrb`	Specifies the TMS scrubber application.
Step 13	**map ingress-interface ServiceApp***<ID>* **egress-interface ServiceApp** *<ID>* **Example:** `RP/0/RP0/CPU0:router(config-tms-scrb)# map ingress-interface ServiceApp 21 egress-interface ServiceApp 22`	Maps the incoming interface and outgoing interface.
Step 14	**commit**	

Step 15	**show run service sesh**
	Example:
	`RP/0/RP0/CPU0:router# sh running-config service sesh`
	`service sesh sesh1`
	`service-location preferred-active 0/1/CPU0`
	`service-type ddos-tms tms1`
	` description 'ddos TMS instance 1'`
	` package arbor-cgse.rpm`
	` application tms-mgmt`
	`  interface ServiceApp11`
	`   remote ipv4 address 10.10.76.2/29`
	`  !`
	` !`
	` application tms-scrb`
		Shows the configured parameters.

	Command or Action	Purpose

	`map ingress-interface ServiceApp21`
	`egress-interface ServiceApp22`	

Step 16	**show service sesh instance**<name of instance>
	Example:
	`RP/0/RP0/CPU0:router# show services sesh instance`
	`all`
	
	`Service Infra instance sesh1`
	`Application tms1 hosted on Location 0/1/CPU0`
	`Octeon 0`
	`State - UP - Application Spawned and`
	`Service App Interfaces Ready`
	`Error Messages - None`
		Displays the state of the application. Values are:
		• INIT—Application configuration download is initiated.
		• WAITING—Application download is complete, but the service application interface is not ready.
		• UP—Application download is complete, and the service application interface is ready.
		An error message is displayed when the service application is missing or not configured.

Configuring the Zone Secret

Zone Secret is the phrase used by all appliances in the system for internal communication. Zone Secret phrase is required to configure Peakflow SP Leader as Manager of CGSE.

Access TMS–CGSE to configure the Zone Secret.

Refer to the Accessing TMS–CGSE, on page 15 section for the steps for accessing TMS–CGSE.

SUMMARY STEPS

1. **services tms stop**
2. **services tms secret set** *<zone secret phrase>*
3. **services tms start**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**services tms stop** **Example:** `admin@arbos:/# services tms stop`	Stops the TMS service in CGSE.
Step 2	**services tms secret set** *<zone secret phrase>* **Example:** `admin@arbos:/# services tms secret set arbor`	Sets the Zone Secret phrase.
Step 3	**services tms start** **Example:** `admin@arbos:/# services tms start`	Starts the TMS service in CGSE.

Configuring Peakflow SP Leader as Manager of CGSE

Peakflow SP leader controls the TMS for all the mitigations. Mitigation is defined in Peakflow SP by the user and Peakflow SP installs the mitigation in TMS.

Configure TMS-CGSE to make Peakflow SP leader the manager of CGSE.

- To enable communication between TMS_CGSE with the Peakflow SP leader
- To enable Peakflow SP leader to control TMS

Access the services line card to configure TMS-CGSE to make Peakflow SP leader the manager of CGSE.

Refer to the Accessing TMS–CGSE, on page 15 section for the steps for accessing TMS–CGSE.

SUMMARY STEPS

1. **services tms bootstrap** *<Peakflow SP leader IP Address> < Zone secret password>*
2. **config write**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**services tms bootstrap** *<Peakflow SP leader IP Address> < Zone secret password>* **Example:** `admin@arbos:/# services tms bootstrap 121.10.23.1 arbor`	Sets the Peakflow SP Leader the manager of the CGSE module.
Step 2	**config write** **Example:** `admin@arbos:/# config write`	Saves the configuration changes.

Configuring TMS-CGSE in the Peakflow SP Web UI

Peakflow SP Web UI provides interface to configure and manage TMS-CGSE.

Configure TMS-CGSE in the Peakflow SP Web UI.

- To configure TMS-CGSE, select **Administration** > **Peakflow Appliances** from the Peakflow SP Web UI.
- To create a cluster that contains one or more TMS-CGSEs, select **Administration** > **Mitigation** > **TMS-CGSE Clusters** from the Peakflow SP Web UI.

For more information about configuring a TMS-CGSE, see *About Configuring Peakflow SP Appliances* module in the *Peakflow SP User Guide Version 5.7*.

For information about creating a cluster of TMS-CGSEs, see *Configuring TMS-CGSE Clusters* module in the *Peakflow SP User Guide Version 5.7*.

For information about TMS-CGSE deployment scenarios, see *TMS-CGSE Deployment Scenarios* module in the *Peakflow SP User Guide Version 5.7*.

Accessing TMS–CGSE

To access TMS–CGSE, connect to TMS–CGSE from a Linux server through SSH.

Note SSH requires the k9crypto.pie to be installed.

Note TMS–CGSE must be reachable from the Linux server. This can be achieved by configuring appropriate routes between the Linux server and the CRS.

SUMMARY STEPS

1. **ssh tms@** *<TMS Management IP Address>*
2. *Username*
3. *Password*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ssh tms@** *<TMS Management IP Address>* **Example:** `eng-1032:~ lnx_server$ ssh tms@10.10.76.2`	Connects to the TMS–CGSE.
Step 2	*Username* **Example:** `arbos login: admin`	Enter the login username.

Step 3	*Password*
	Example:
	`Password:`
	`admin@arbos:/#`
		Enter the Password.
		Note The password will not be visible when entered.

What to do next

It is recommended to change the default password after logging into TMS–CGSE for the first time.

Refer to the Changing TMS–CGSE Login Password, on page 15 section for the steps for changing the username and password.

Changing TMS–CGSE Login Password

It is recommended to change the default password after logging into TMS–CGSE for the first time.

SUMMARY STEPS

1. **services tms stop**

2. **services aaa local password** *<old password string>* **interactive**
3. *new password*
4. *new password*
5. **services tms start**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**services tms stop** **Example:** `admin@arbos:/# services tms stop`	Stops the TMS service in CGSE.
Step 2	**services aaa local password** *<old password string>* **interactive** **Example:** `admin@arbos:/# services aaa local password admin interactive`	Prompts for password change for the user "admin".
Step 3	*new password* **Example:** `Password:`	Enter the new password. **Note** The password will not be visible when entered.

Step 4	*new password*
	Example:
	`Password:`
	`admin@arbos:/#`
		Re-enter the new password.
		Note The password will not be visible when re-entered.

Step 5	**services tms start** **Example:** `admin@arbos:/# services tms start`	Starts the TMS service in CGSE.

Configuring TMS-CGSE Time Zone and Clock

To configure TMS-CGSE time zone and clock follow these steps.

SUMMARY STEPS

1. **system timezone set**
2. *<name of the timezone>*
3. *<name of the sub-timezone>*
4. **clock set** *[MMDDhhmm]*
5. **clock**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**system timezone set** **Example:** `admin@arbos:/# system timezone set`	Enters the time zone configuration.

Step 2	*<name of the timezone>*
	Example:
	`admin@arbos:/# system timezone set`
	``What timezone are you in? [`?' for list]  Asia``
		Sets the time zone.

Step 3	*<name of the sub-timezone>*
	Example:
	`admin@arbos:/# system timezone set`
	``What timezone are you in? [`?' for list]  Asia``
	``Select a sub-timezone [`?' for list]: Calcutta``
		Sets the sub time zone.

Step 4	**clock set** *[MMDDhhmm]*
	Example:
	`admin@arbos:/# clock set 01131401`
	`Fri Jan 13 14:01:00 EST 2012`
		Sets the clock.

Step 5	**clock**
	Example:
	`admin@arbos:/# clock`
	`Fri Jan 13 14:22:10 IST 2012`
		Displays the clock.

Configuration Examples for Implementing DDoS Mitigation Support on CGSE

This section contains the configuration examples for Implementing DDoS Mitigation Support on CGSE.

Configuring the CGSE Service Role as Service Engine Service Hosting: Example

This example shows how to configure CGSE Service Role as Service Engine Service Hosting (SESH):

```
configure
 hw-module service sesh location 0/1/CPU0
 end

Uncommitted changes found, commit them? [yes]: yes
```

```
show running-config service sesh

Wed Jul 11 14:24:31.560 PST
service sesh sesh1
service-location preferred-active 0/1/CPU0
```

Configuring the Service Infrastructure Interface: Example

This example shows the Service Infrastructure Interface configuration:

```
configure
 interface ServiceInfra 1
 ipv4 address 100.1.1.1/29
 service-location 0/1/CPU0
 end

Uncommitted changes found, commit them? [yes]: yes

hw-module location 0/1/CPU0 reload

show services role

Node  Configured Role Enacted Role Enabled Services
--------------------------------------------------
0/1/CPU0 SESH           SESH        ServiceInfra
```

Configuring ServiceEngine–ServiceHost Instance: Example

The following example shows how to configure ServiceEngine–ServiceHost Instance:

```
configure
 service sesh sesh1
  service-location preferred-active 0/1/CPU0
  end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes
```

Configuring Service Application Interfaces: Example

This example shows how to configure service application interfaces:

```
configure
 vrf arbor-tms
 commit
 interface ServiceApp 11
  description tms1 mgmt interface
  ipv4 address 10.10.76.1/29
  service sesh sesh1
 interface ServiceApp 21
  description tms1 scrb ingress interface
  ipv4 address 204.0.0.1/24
  service sesh sesh1
 interface ServiceApp 22
```

```
  description tms1 scrb egress interface
  vrf arbor-tms
  ipv4 address 205.0.0.1/24
  service sesh sesh1
  end

Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes
```

Configuring TMS–CGSE Service and Applications: Example

This example shows how to configure TMS–CGSE Service and Applications:

```
configure
 service sesh sesh1
  service-location preferred-active 0/1/CPU0
  service-type ddos-tms tms1
   description ddos TMS instance 1
   package arbor.rpm
   application tms-mgmt
   interface ServiceApp 11
   remote ipv4 address 10.10.76.2/29
   application tms-scrb
   map ingress-interface ServiceApp 21 egress-interface ServiceApp 22
   end
Uncommitted changes found, commit them before exiting(yes/no/cancel)? [cancel]:yes
```

This example shows the output of **show run service sesh** command:

```
show running-config service sesh

service sesh sesh1
service-location preferred-active 0/1/CPU0
service-type ddos-tms tms1
 description 'ddos TMS instance 1'
 package arbor-cgse.rpm
 application tms-mgmt
  interface ServiceApp11
   remote ipv4 address 10.10.76.2/29
  !
 !
 application tms-scrb
  map ingress-interface ServiceApp21 egress-interface ServiceApp22
 !
```

This example shows the output of **show service sesh instance** command:

```
show services sesh instance all

Service Infra instance sesh1
Application tms1 hosted on Location 0/1/CPU0
       Octeon 0
       State - UP - Application Spawned and Service App Interfaces Ready
       Error Messages - None
```

Configuring ACL to Limit Access to CGSE

Access Control Lists(ACLs) are used to restrict access to CGSE modules. You can configure ACLs so that the access to these modules are limited to components of the PeakFlow SP deployment and valid SSH users. To apply this restriction, you have to perform the following tasks:

1. Configure an IPv4 ACL for the Management ServiceApp interface
2. Apply the configured IPv4 ACL to the Management ServiceApp interface.
3. Configure an IPv6 ACL for the Management ServiceApp interface
4. Apply the configured IPv6 ACL to the Management ServiceApp interface.
5. Configure an IPv4 ACL for the ServiceInfra interface.
6. Apply the configured IPv4 ACL to the ServiceInfra interface.

Configuring an IPv4 ACL for the Management ServiceApp Interface

To configure an IPv4 ACL for the management ServiceApp interface of the CGSE module, follow these steps:

SUMMARY STEPS

1. **configure**
2. **ipv4 access-list** *access_list_name*
3. **10 permit tcp any host** *tms-cgse_mgmt_IP_address* **eq ssh**
4. **20 permit tcp any eq ident host** *tms-cgse_mgmt_IP_address*
5. **30 permit tcp host** *SP_leader_IP_address* **eq 443 host** *tms-cgse_mgmt_IP address*
6. **40 permit tcp host** *SP_leader_IP_address* *tms-cgse_mgmt_IP address* **eq 443**
7. **50 permit icmp any host** *tms-cgse_mgmt_IP address* **echo**
8. **60 permit icmp any host** *tms-cgse_mgmt_IP address* **echo-reply**
9. **35 permit tcp host** *SP_leader_IP_address* **eq 443 host** *tms-cgse_mgmt_IP address*
10. **45 permit tcp host** *SP_leader_IP_address* *tms-cgse_mgmt_IP address* **eq 443**
11. **30 deny IPv4 any any**
12. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ipv4 access-list** *access_list_name* **Example:** `RP/0/RP0/CPU0:router (config) # ipv4 access-list acl`	Creates an IPv4 ACL.
Step 3	**10 permit tcp any host** *tms-cgse_mgmt_IP_address* **eq ssh** **Example:**	Adds an entry into the IPv4 ACL. This entry permits the SSH access to the TMS management interface CGSE modules . The *tms-cgse_mgmt_IP_address* parameter

	Command or Action	Purpose
	`RP/0/RP0/CPU0:router (config) # 10 permit tcp any host 10.10.76.2 eq ssh`	specifies the IPv4 address chosen for the TMS management interface.
Step 4	**20 permit tcp any eq ident host** *tms-cgse_mgmt_IP_address* **Example:** `RP/0/RP0/CPU0:router (config) # 20 permit tcp any eq ident host 10.10.76.2`	Adds an entry into the IPv4 ACL. This entry permits the TCP traffic from the ident port(Port number 113) on any host to the TMS management interface on the CGSE modules. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 5	**30 permit tcp host** *SP_leader_IP_address* **eq 443 host** *tms-cgse_mgmt_IP address* **Example:** `RP/0/RP0/CPU0:router (config) # 30 permit tcp host 121.10.23.1 eq 443 host 10.10.76.2`	Adds an entry into the IPv4 ACL. This entry permits the TCP traffic from the port 443 on the PeakFlow SP Leader host to the TMS management interface on the CGSE modules. The *SP_leader_IP_address* parameter specifies the IP address assigned to the PeakFlow SP Leader appliance. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 6	**40 permit tcp host** *SP_leader_IP_address* *tms-cgse_mgmt_IP address* **eq 443** **Example:** `RP/0/RP0/CPU0:router (config) # 40 permit tcp host 121.10.23.1 10.10.76.2 eq 443`	Adds an entry into the IPv4 ACL. This entry permits the TCP traffic from the PeakFlow SP Leader host to the TMS management interface on port 443 on the CGSE modules. The *SP_leader_IP_address* parameter specifies the IP address assigned to the PeakFlow SP Leader appliance. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface..
Step 7	**50 permit icmp any host** *tms-cgse_mgmt_IP address* **echo** **Example:** `RP/0/RP0/CPU0:router (config) # 50 permit icmp any host 10.10.76.2 echo`	Adds an entry into the IPv4 ACL. This entry permits the ICMP ping requests to the TMS management interface on the CGSE modules. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 8	**60 permit icmp any host** *tms-cgse_mgmt_IP address* **echo-reply** **Example:** `RP/0/RP0/CPU0:router (config) # 60 permit icmp any host 10.10.76.2 echo-reply`	Adds an entry into the IPv4 ACL. This entry permits the ICMP echo replies to the TMS management interface on the CGSE module. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 9	**35 permit tcp host** *SP_leader_IP_address* **eq 443 host** *tms-cgse_mgmt_IP address* **Example:** `RP/0/RP0/CPU0:router (config) # 35 permit tcp host 121.10.23.1 eq 443 host 10.10.76.2`	Adds an entry into the IPv4 ACL. This entry permits the traffic from port 443 on the Peakflow SP Leader appliance to the CGSE modules if the Peakflow SP Leader is not the manager appliance. The *SP_leader_IP_address* parameter specifies the IP address assigned to the PeakFlow SP Leader appliance. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 10	**45 permit tcp host** *SP_leader_IP_address* *tms-cgse_mgmt_IP address* **eq 443**	Adds an entry into the IPv4 ACL. This entry permits the traffic the Peakflow SP Leader appliance to port 443 on

	Command or Action	Purpose
	Example: `RP/0/RP0/CPU0:router (config) # 45 permit tcp host 121.10.23.1 10.10.76.17/29 eq 443`	the CGSE modules if the Peakflow SP Leader is not the manager appliance. The *SP_leader_IP_address* parameter specifies the IP address assigned to the PeakFlow SP Leader appliance. The *tms-cgse_mgmt_IP_address* parameter specifies the IPv4 address chosen for the TMS management interface.
Step 11	**30 deny IPv4 any any** **Example:** `RP/0/RP0/CPU0:router (config) # 30 deny IPv4 any any`	Adds an entry into the IPv4 ACL. This entry denies all other access to the CGSE modules except the access given in the previous steps.
Step 12	**commit**	

Applying the IPv4 ACL to the Management Service Application Interface

Perform the following steps to apply the IPv4 Access Control List (ACL) to the Management Service Application (ServiceApp) interface.

SUMMARY STEPS

1. **configure**
2. **interface ServiceApp** *id*
3. **ipv4 access-group** *access_group_name* **egress**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**interface ServiceApp** *id* **Example:** `RP/0/RP0/CPU0:router (config) # interface ServiceApp 11`	Enters the interface mode of the ServiceApp interface. The *id* parameter specifies the number that was assigned to the ServiceApp interface.
Step 3	**ipv4 access-group** *access_group_name* **egress** **Example:** `RP/0/RP0/CPU0:router (config) # ipv4 access-group acl egress`	Controls access to the ServiceApp interface towards the outbound direction.
Step 4	**commit**	

Configuring an IPv6 ACL for the Management ServiceApp Interface

To configure an IPv6 ACL for the management Service Application (ServiceApp) interface, follow these steps:

SUMMARY STEPS

1. **configure**
2. **ipv6 access-list** *access_list_name*
3. **10 deny ipv6 any any log**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ipv6 access-list** *access_list_name* **Example:** `RP/0/RP0/CPU0:router (config) # ipv6 access-list acl1`	Creates an IPv6 ACL.
Step 3	**10 deny ipv6 any any log** **Example:** `RP/0/RP0/CPU0:router (config) # 10 deny ipv6 any any log`	Adds an entry into the IPv6 ACL. This entry denies all access to the management ServiceApp interface of the CGSE module except the access given in the previous steps.
Step 4	**commit**	

Applying the IPv6 ACL to the Management ServiceApp Interface

Perform the following steps to apply the IPv6 Access Control List (ACL) to the Management ServiceApp interface.

SUMMARY STEPS

1. **configure**
2. **interface ServiceApp** *id*
3. **ipv6 access-group** *access_group_name* **egress**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**interface ServiceApp** *id* **Example:** `RP/0/RP0/CPU0:router (config) # interface ServiceApp 11`	Enters the interface mode of the ServiceApp interface. The *id* parameter specifies the number that was assigned to the ServiceApp interface.
Step 3	**ipv6 access-group** *access_group_name* **egress** **Example:** `RP/0/RP0/CPU0:router (config) # ipv6 access-group acl1 egress`	Controls access to the ServiceApp interface towards the outbound direction.

	Command or Action	Purpose
Step 4	**commit**	

Configuring an IPv4 ACL for the ServiceInfra Interface

To configure an IPv4 ACL for the Service Infrastructure (ServiceInfra) interface, follow these steps:

SUMMARY STEPS

1. **configure**
2. **ipv4 access-list** *access_list_name*
3. **10 permit tcp host** *serviceinfra_IP_address* **range 6005 6008** *serviceinfra_subnet/mask*
4. **20 permit tcp host** *serviceinfra_IP_address* **range 16000 16003** *serviceinfra_subnet/mask*
5. **30 permit tcp host** *serviceinfra_IP_address* **range 4000 4003** *serviceinfra_subnet/mask*
6. **40 permit udp host** *serviceinfra_IP_address* **eq 5567** *serviceinfra_subnet/mask*
7. **50 permit icmp any** *serviceinfra_subnet/mask* **echo**
8. **60 permit icmp any** *serviceinfra_subnet/mask* **echo-reply**
9. **70 deny IPv4 any any log**
10. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**ipv4 access-list** *access_list_name* **Example:** `RP/0/RP0/CPU0:router (config) # ipv4 access-list acl`	Creates an IPv4 ACL.
Step 3	**10 permit tcp host** *serviceinfra_IP_address* **range 6005 6008** *serviceinfra_subnet/mask* **Example:** `RP/0/RP0/CPU0:router (config) # 10 permit tcp host 100.1.1.1 range 6005 6008 100.1.1.0/29`	Adds an entry into the IPv4 ACL. This entry permits TCP traffic from the ServiceInfra interface on ports (6005 to 6008) to a host configured on the ServiceInfra subnet on the CGSE modules. The *serviceinfra_IP_address* parameter specifies the IP address assigned to the ServiceInfra interface. The *serviceinfra_subnet/mask* specifies the destination subnet for which traffic will be permitted.
Step 4	**20 permit tcp host** *serviceinfra_IP_address* **range 16000 16003** *serviceinfra_subnet/mask* **Example:** `RP/0/RP0/CPU0:router (config) # 20 permit tcp host 100.1.1.1 range 16000 16003 100.1.1.0/29`	Adds an entry into the IPv4 ACL. This entry permits TCP traffic from the ServiceInfra interface on ports(16000 to 16003) to a host configured on the ServiceInfra subnet on the CGSE modules. The *serviceinfra_IP_address* parameter specifies the IP address assigned to the ServiceInfra interface. The *serviceinfra_subnet/mask* specifies the destination subnet for which traffic will be permitted.

	Command or Action	Purpose
Step 5	**30 permit tcp host** *serviceinfra_IP_address* **range 4000 4003** *serviceinfra_subnet/mask* **Example:** `RP/0/RP0/CPU0:router (config) # 30 permit tcp host 100.1.1.1 range 4000 4003 100.1.1.0/29`	Adds an entry into the IPv4 ACL. This entry permits TCP traffic from the ServiceInfra interface on ports(4000 to 4003) to a host configured on the ServiceInfra subnet on the CGSE modules. The *serviceinfra_IP_address* parameter specifies the IP address assigned to the ServiceInfra interface. The *serviceinfra_subnet/mask* specifies the destination subnet for which traffic will be permitted.
Step 6	**40 permit udp host** *serviceinfra_IP_address* **eq 5567** *serviceinfra_subnet/mask* **Example:** `RP/0/RP0/CPU0:router (config) # 40 permit udp host 100.1.1.1 eq 5567 100.1.1.0/29`	Adds an entry into the IPv4 ACL. This entry permits UDP traffic from the ServiceInfra interface on port 5567 to a host configured on the ServiceInfra subnet on the CGSE modules. The *serviceinfra_IP_address* parameter specifies the IP address assigned to the ServiceInfra interface. The *serviceinfra_subnet/mask* specifies the destination subnet for which traffic will be permitted.
Step 7	**50 permit icmp any** *serviceinfra_subnet/mask* **echo** **Example:** `RP/0/RP0/CPU0:router (config) # 50 permit icmp any 100.1.1.0/29 echo`	Adds an entry into the IPv4 ACL. This entry permits ICMP ping requests from any source host to the destination host on the *serviceinfra_subnet/mask* subnet.
Step 8	**60 permit icmp any** *serviceinfra_subnet/mask* **echo-reply** **Example:** `RP/0/RP0/CPU0:router (config) # 60 permit icmp any 100.1.1.0/29 echo-reply`	Adds an entry into the IPv4 ACL. This entry permits ICMP replies from any source host to the destination host on the *serviceinfra_subnet/mask* subnet.
Step 9	**70 deny IPv4 any any log** **Example:** `RP/0/RP0/CPU0:router (config) # 70 deny IPv4 any any log`	Adds an entry into the IPv4 ACL. This entry denies all access to the ServiceInfra interface except the access given in the previous steps.
Step 10	**commit**	

Applying the IPv4 ACL to the Service Infrastructure Interface

Perform the following steps to apply the IPv4 Access Control List (ACL) to the Service Infrastructure (ServiceInfra) interface.

SUMMARY STEPS

1. **configure**
2. **interface serviceinfra** *id*
3. **ipv4 access-group** *access_group_name* **egress**
4. **commit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure**	
Step 2	**interface serviceinfra** *id* **Example:** `RP/0/RP0/CPU0:router (config) # interface serviceinfra 1`	Enters the interface mode of the ServiceInfra interface. The *id* parameter specifies the number that was assigned to the ServiceInfra interface.
Step 3	**ipv4 access-group** *access_group_name* **egress** **Example:** `RP/0/RP0/CPU0:router (config) # ipv4 access-group acl egress`	Controls access to the ServiceInfra interface towards the outbound direction.
Step 4	**commit**	

Additional References

The following sections provide references related to implementing DDoS mitigation support on CGSE.

Related Documents

Related Topic	Document Title
DDoS Mitigation Support commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*DDoS Mitigation Support commands* on *System Security Command Reference for Cisco CRS Routers*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of 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/techsupport