

## Chapter 44

# Trigger Facility

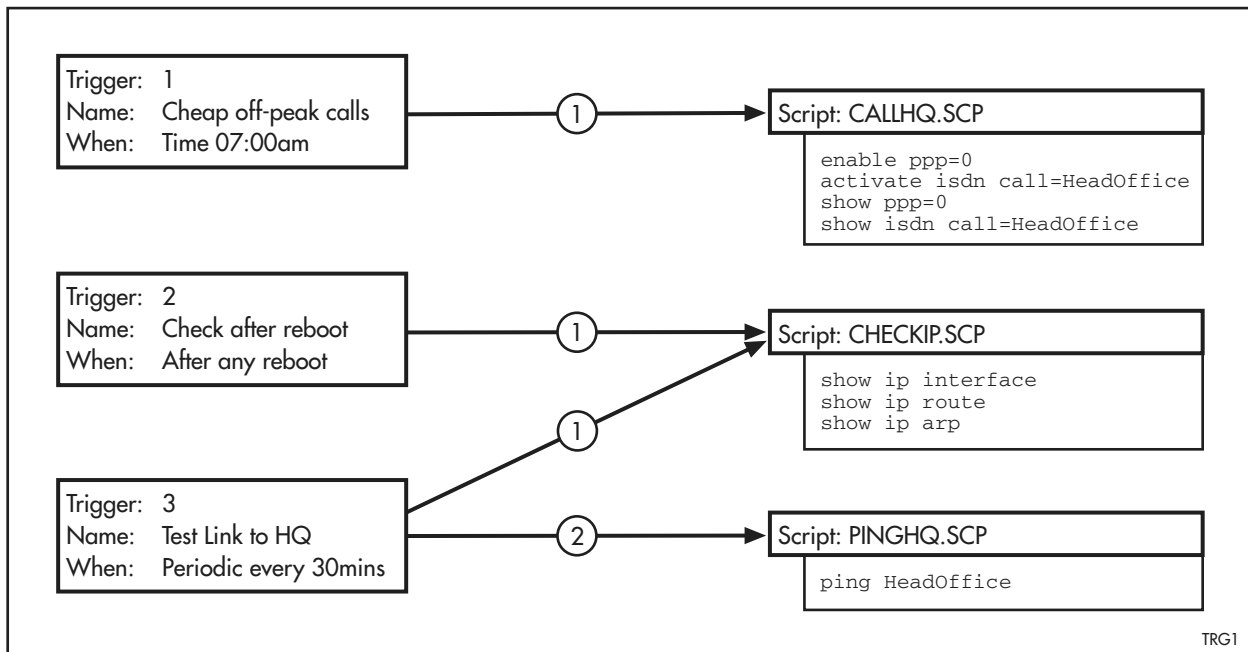
Introduction .....	44-2
Defining Triggers .....	44-3
Power Supply Units (PSUs) .....	44-3
Command Reference .....	44-5
activate trigger .....	44-5
add trigger .....	44-6
create trigger .....	44-7
delete trigger .....	44-12
destroy trigger .....	44-12
disable trigger .....	44-13
enable trigger .....	44-14
purge trigger .....	44-14
set trigger .....	44-15
show trigger .....	44-19

## Introduction

The Trigger facility provides a powerful mechanism for automatic and timed management of the switch by automating the execution of switch commands in response to certain events. For example, triggers can be configured to collect diagnostic information after a switch reboots.

A *trigger* is an ordered sequence of scripts to be executed when a certain event occurs. A *script* is a sequence of commands stored as a plaintext file in the switch's file system, either in NVS or flash memory. Each trigger may reference multiple scripts and any script may be used by any trigger. Various types of triggers are supported, each activated in a different way ([Figure 44-1](#)).

Figure 44-1: Triggers respond to events by performing a sequence of predefined scripts



When a trigger is activated, the scripts associated with it are executed in sequence by the switch. The trigger passes script parameters to the script that depend on the trigger type, and in the case of module triggers, the event that activates the trigger. These parameters are available to the script that is executed. Global script parameters that contain date, time, system name, and serial number of the switch are also available to scripts. See [“Script Parameters” on page 43-4 of Chapter 43, Scripting](#) for more information about passing parameters to scripts. The output from the scripts is passed to the Logging facility, and can be displayed with the [show log command on page 45-31 of Chapter 45, Logging Facility](#), or forwarded to another switch.

## Defining Triggers

Three main groups of parameters define each trigger. Trigger type parameters define the kind of event that activates a trigger. Some modules have their own module type triggers, with additional module-specific defining parameters described in the chapter for the individual module. There are also a group of general trigger parameters that can be used with any trigger type.

To create a trigger, use the [create trigger command on page 44-7](#).

To change a trigger, use the [set trigger command on page 44-15](#).

Each trigger can be assigned a descriptive name and up to five scripts can be executed. See [Chapter 43, Scripting](#) for more information about creating scripts. To add a script to any position in the sequence, use the [add trigger command on page 44-6](#).

To delete scripts from a trigger, use the [delete trigger command on page 44-12](#).

To destroy a trigger, use the [destroy trigger command on page 44-12](#).

By default, triggers are enabled when created. A trigger can be disabled by using the [disable trigger command on page 44-13](#).

To activate a trigger, first enable the trigger module if it is not already enabled, then enable the trigger. Both of these are done using the [enable trigger command on page 44-14](#). You must have the Trigger module enabled before triggers will activate.

A trigger can be explicitly activated (triggered) by using the [activate trigger command on page 44-5](#) regardless of whether it is currently enabled.

The [show trigger command on page 44-19](#) displays summary information about all triggers or details about a particular one.

## Power Supply Units (PSUs)

You can set triggers on switches with a hot-swappable PSUs installed. PSU monitoring is part of the **system** module. Use the Trigger Facility to automatically run specific scripts for the following PSU events:

- when a PSU is hot-swapped into the switch
- when a PSU is hot-swapped out of the switch

To create a PSU trigger, specify **module=system** in the command:

```
create trigger=trigger-id module=system event=event
```

<b>Module</b>	SYSTEM
<b>Event</b>	HOTSWAPIN
<b>Description</b>	A PSU has been hot-swapped into the switch.
<b>Parameters</b>	You cannot specify any command parameters in the <b>create trigger</b> command.
<b>Script arguments</b>	There are no arguments to pass to the script.
<b>Example</b>	To create trigger 5, which activates whenever a PSU is hot-swapped into the switch, use the command: <pre>create trigger=5 module=system event=hotswapi</pre>

<b>Event</b>	HOTSWAPOUT
<b>Description</b>	A PSU has been hot-swapped out of the switch.
<b>Parameters</b>	You cannot specify any command parameters in the <b>create trigger</b> command.
<b>Script arguments</b>	There are no arguments to pass to the script.

**Example** To create trigger 6, which activates whenever a PSU is hot-swapped out of the switch, use the command:

```
create trigger=6 module=system event=hotswapout
```

---

## Command Reference

---

This section describes the commands to configure and manage the trigger facility in the switch. The trigger facility requires that the switch's internal clock be set correctly. See [Chapter 4, Configuring and Monitoring the System](#) for descriptions of the commands required to configure the switch's internal clock.

The shortest valid command is denoted by capital letters in the Syntax section. See ["Conventions" on page xlix of About this Software Reference](#) in the front of this manual for details of the conventions used to describe command syntax. See [Appendix A, Messages](#) for a complete list of messages and their meanings.

---

### activate trigger

---

**Syntax**    ACTivate TRIGger=*trigger-id*

where *trigger-id* is a number from 1 to 250

**Description**    This command immediately activates a specific trigger even if it has been disabled with the **disable trigger** command. The scripts associated with the trigger are executed even if the **test** option is set. Normally, a **test** trigger generates only log entries when it triggers and does not invoke scripts.

The **trigger** parameter specifies the number of the trigger to activate. The specified trigger must already exist.

Triggers activated manually do not have their repeat counts decremented or their *"last triggered"* time updated, and do not result in updates to the *"time/periodic triggers today"* counters.

**Examples**    To activate trigger number 8, use the command:

```
act trig=8
```

**Related Commands**    [create trigger](#)  
[disable trigger](#)  
[enable trigger](#)  
[show trigger](#)

## add trigger

---

**Syntax** `ADD TRIGger=trigger-id SScript=filename... [NUMBER=index]`

where:

- *trigger-id* is a number from 1 to 250.
- *filename* is a file name in the format `[device]:filename.ext`. The *.ext* must be *.scp* or *.cfg*. Valid characters are lowercase and uppercase letters, digits (0–9), and the characters `~'!@#$%^&()-_{}`. Invalid characters are `*+= " | \ [ ] ; : ? / , < >`. Wildcards are not allowed. The *device* is optional and specifies the physical memory device on which the file is stored—either NVS or flash. If *device* is specified, it must be separated from the rest of the file name by a colon. If *device* is not specified, the default is flash.
- *index* is a number from 1 to *n*+1, where *n* is the number of scripts already assigned to the trigger.

**Description** This command adds a script to a trigger so that the script is executed when the trigger is activated.

The **trigger** parameter specifies the number of the trigger to which the script is to be added. The specified trigger must already exist.

The **script** parameter specifies the name of the script to be added. The **script** parameter may be repeated up to five times in one command in order to add up to five scripts (in the order specified) at once.

The **number** parameter specifies the position in the script list where the script is to be added. If **number** is specified, the new script occupies this position and all following scripts are pushed down one position. The default position is at the end of the list of scripts.

**Examples** To add scripts *snapshot.scp* and *callhq.scp* to trigger 1 at position 3, use the command:

```
add trig=1 sc=snapshot.scp sc=callhq.scp num=3
```

**Related Commands**

- [create trigger](#)
- [delete trigger](#)
- [disable trigger](#)
- [enable trigger](#)
- [set trigger](#)
- [show trigger](#)

## create trigger

**Syntax**

```

CREate TRIGger=trigger-id CPU=value [Direction={UP|DOWN|
ANY}] [AFter=hh:mm] [BEfore=hh:mm] [{Date=date|
DAYs=day-list}] [NAME=name] [REpeat={Yes|No|ONCe|
FORever|count}] [ScripT=filename...] [STate={ENabled|
Disabled}] [TEST={YES|NO|ON|OFF|True|False}]

CREate TRIGger=trigger-id INTERface=interface EVENT={UP|
DOWN|FAIL|ANY} [CP={APPLE|ATCP|BCP|CCP|DCP|DNCP|IPcp|
IPXcp|LCP}] [AFter=hh:mm] [BEfore=hh:mm] [{Date=date|
DAYs=day-list}] [NAME=name] [REpeat={Yes|No|ONCe|
FORever|count}] [ScripT=filename...] [STate={ENabled|
Disabled}] [TEST={YES|NO|ON|OFF|True|False}]

CREate TRIGger=trigger-id MEMOrY=value [Direction={UP|
DOWN|ANY}] [AFter=hh:mm] [BEfore=hh:mm] [{Date=date|
DAYs=day-list}] [NAME=name] [REpeat={Yes|No|ONCe|
FORever|count}] [ScripT=filename...] [STate={ENabled|
Disabled}] [TEST={YES|NO|ON|OFF|True|False}]

CREate TRIGger=trigger-id MODUle=module EVENT=event
[module-parameters...] [AFter=hh:mm] [BEfore=hh:mm]
[{Date=date|DAYs=day-list}] [NAME=name] [REpeat={Yes|
No|ONCe|FORever|count}] [ScripT=filename...]
[STate={ENabled|Disabled}] [TEST={YES|NO|ON|OFF|True|
False}]

CREate TRIGger=trigger-id MODUle=SWitch PORT=port
EVENT={LINKDOWN|LINKUP} [AFter=hh:mm] [BEfore=hh:mm]
[{Date=date|DAYs=day-list}] [NAME=name] [REpeat={Yes|
No|ONCe|FORever|count}] [ScripT=filename...]
[STate={ENabled|Disabled}] [TEST={YES|NO|ON|OFF|True|
False}]

CREate TRIGger=trigger-id PERiodic=minutes [AFter=hh:mm]
[BEfore=hh:mm] [{Date=date|DAYs=day-list}] [NAME=name]
[REpeat={Yes|No|ONCe|FORever|count}]
[ScripT=filename...] [STate={ENabled|Disabled}]
[TEST={YES|NO|ON|OFF|True|False}]

CREate TRIGger=trigger-id REBoot={REStart|CRASH|ALL}
[AFter=hh:mm] [BEfore=hh:mm] [{Date=date|
DAYs=day-list}] [NAME=name] [REpeat={Yes|No|ONCe|
FORever|count}] [ScripT=filename...] [STate={ENabled|
Disabled}] [TEST={YES|NO|ON|OFF|True|False}]

CREate TRIGger=trigger-id Time=hh:mm [{Date=date|
DAYs=day-list}] [ScripT=filename...] [NAME=name]
[REpeat={Yes|No|ONCe|FORever|count}] [STate={ENabled|
Disabled}] [TEST={YES|NO|ON|OFF|True|False}]

```

where:

- *trigger-id* is a number from 1 to 250.
- *module* is the name of a switch module as given in “[Module Identifiers and Names](#)” on page B-2 of Appendix B, Reference Tables.

- *event* is an event defined in the specified module
- *module-parameters* are one or more other required or optional parameters for the specified module and event, whose syntax is defined in the specified module.
- *value* is a number from 1 to 100.
- *interface* is a valid interface name.
- *port* is the port number. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port, including uplink ports.
- *minutes* is a number from 1 to 1439.
- *hh:mm* is a time in hours and minutes.
- *date* is a date in the format dd-mmm-yyyy, where *mmm* is the first three letters of the month name.
- *day-list* is one or more keywords for the day of the week.
- *filename* is a file name in the format [device]:filename.ext. The *.ext* must be .scp or .cfg. Valid characters are uppercase and lowercase letters, digits (0–9), and the characters ~ ' ! @ # \$ % ^ & ( ) \_ - { }. Invalid characters are \* + = " | \ [ ] ; : ? / , < > . Wildcards are not allowed. The *device* is optional and specifies the physical memory device on which the file is stored—either NVS or flash. If specified, it must be separated from the rest of the file name by a colon. If *device* is not specified, the default is flash.
- *name* is a string 1 to 40 characters long. If the string contains spaces, it must be in double quotes.
- *count* is a number from 1 to 4294967294 ( $2^{32}-2$ ).

**Description** This command creates a new trigger and defines events and conditions that activate it. It requires a user with security officer privilege when the switch is in security mode. Different trigger types are supported—CPU triggers, link triggers, memory triggers, module triggers, periodic triggers, port triggers, reboot triggers, and time triggers. Some of these have additional trigger-specific parameters. The **event** parameter and other parameters for module triggers are described in the chapter for the module that supports them. General trigger parameters can be specified for all trigger types: **after**, **before**, **days** or **date**, **script**, **name**, **repeat**, **state**, and **test**.

When a trigger is activated, it passes parameters to the scripts that it executes. The global script parameters containing the date (%D), time (%T), system name (%N) and serial number (%S) of the switch are passed to scripts executed by all triggers. Additional module-specific parameters passed by module-specific triggers are described in the chapter for the specific module.

The **trigger** parameter specifies the number of the trigger to create. The number references the trigger in other commands. The specified trigger must not already exist. The **trigger** parameter must immediately follow the **create** keyword, and must be followed immediately by either **module**, **cpu**, **interface**, **memory**, **periodic**, **reboot**, or **time**.

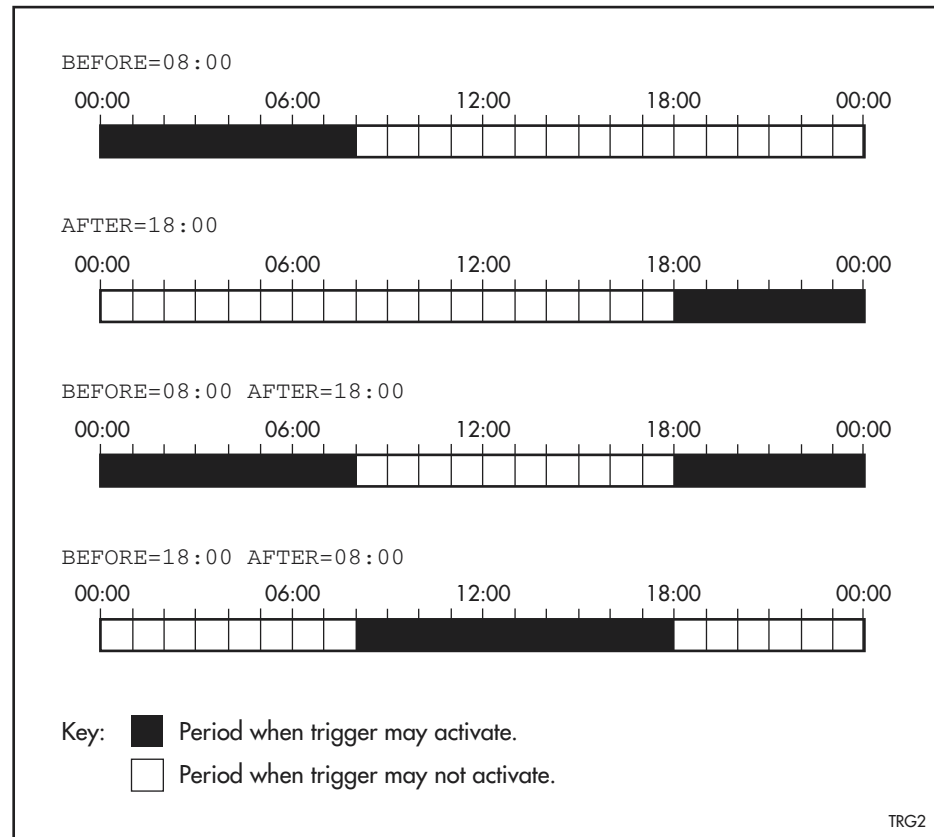
The **after** parameter is a general trigger parameter that can be used with all trigger types except for **time** triggers. It specifies the earliest time of day in hours and minutes that the trigger is activated. The trigger may be activated any time between the time specified and midnight.

The **before** parameter is a general trigger parameter that can be used with all trigger types except for **time** triggers. It specifies the latest time of day in hours and minutes that the trigger is activated. The trigger may be activated any time between midnight and the time specified.



If neither **after** nor **before** are specified, there is no restriction on when the trigger may activate the script. If both **after** and **before** are specified, and the time periods specified overlap, the trigger may activate it any time during the overlap period (Figure 44-2).

Figure 44-2: The effects of different combinations of the **after** and **before** parameters in the **create trigger** and **set trigger** commands



The **cp** parameter specifies a PPP control protocol to monitor. The trigger is activated when the condition defined by the combination of **interface**, **cp**, and **event** occurs. The **cp** parameter may only follow an **interface** parameter that specifies a PPP interface. When activated, the trigger passes three parameters to the trigger scripts—the PPP instance that caused the trigger, the control protocol, and the event state.

The **cpu** parameter defines a CPU utilisation trigger and specifies the CPU utilisation level at which the trigger is to be activated. The **direction** parameter, and the general trigger parameters may also be specified.

The **date** parameter is a general trigger parameter that can be used with all trigger types. It specifies a date on which the trigger may activate. The **days** and **date** parameters are mutually exclusive—use one or the other.

The **days** parameter is a general trigger parameter that can be used with all trigger types. It specifies a comma-separated list of days when the trigger is activated. The **weekday** option is a synonym for the list **mon, tue, wed, thu, fri**; **weekend** is a synonym for the list **sat, sun**. Any combination of these and names of days is acceptable. The default is **all**. The **days** and **date** parameters are mutually exclusive—use one or the other.

The **direction** parameter specifies how the CPU or memory utilisation threshold is reached to activate the trigger. If **up** is specified, the trigger is activated when CPU or memory utilisation increases to or exceeds the threshold. If **down** is specified, the trigger is activated when CPU or memory utilisation decreases to or falls below the threshold. If **any** is specified, the trigger is activated when CPU or memory utilisation equals or passes the threshold in either direction. The default is **any**.

The **event** parameter is valid if the **interface** parameter or the **module** parameter is specified. The **event** parameter has different meanings in these two contexts. The **event** parameter is required after the **module** parameter, and in this context specifies the event for a module specific trigger. Values for the **event** parameter are described in the chapter for the given module. The **event** parameter is also required after the **interface** parameter, and in this context it specifies a link (interface) status change event. If **up** is specified, the trigger is activated when the interface becomes operational. If **down**, it is activated when the interface closes. If **fail** is specified, the trigger is activated when the interface fails to open at all for any reason. If **any** is specified, it is activated when any of these events occurs.

The **interface** parameter defines an interface (link) trigger and specifies the interface to monitor. The **event** parameter is required for an **interface** trigger. The **interface** parameter must be followed by the **event** parameter. The **cp** parameter may be used if **interface** specifies a PPP interface. The general trigger parameters may also be specified. Valid interfaces are:

- eth (such as eth0)
- PPP (such as ppp0)
- VLAN (such as vlan1)

The interface must already exist. To see a list of all currently available interfaces, use the [show interface command on page 10-51 of Chapter 10, Interfaces](#).

The **memory** parameter defines a memory utilisation trigger and specifies the percentage of free memory at which the trigger activates. The **direction** parameter and general trigger parameters may also be specified. Use the [show buffer command on page 4-37 of Chapter 4, Configuring and Monitoring the System](#) to display the current percentage of free memory.

The **module** parameter defines a module-specific trigger and specifies the name of the module to which this trigger applies. The module should be a valid switch module as given in [“Module Identifiers and Names” on page B-2 of Appendix B, Reference Tables](#), and must be one for which module-specific triggers are defined. The **event** parameter must be specified, and other module parameters defining the module triggers may be valid or required for the given module. If the **module** parameter specified is **switch**, then the **port** parameter must be specified. The general trigger parameters may also be specified.

The **name** parameter is a general trigger parameter that can be used with all trigger types. It specifies a descriptive name for this trigger.

The **periodic** parameter defines a periodic trigger and specifies the period of the trigger in minutes. The general trigger parameters may also be specified.

The **port** parameter is valid if the **module** parameter is **switch**. The **port** parameter specifies the port where the event activates the trigger.

The **reboot** parameter defines a reboot trigger and specifies a list of reboot events that activates the trigger. If **crash** is specified, the trigger is activated by a switch crash. If **restart** is specified, the trigger is activated by any reboot other than a switch crash. If **all** is specified, the trigger is activated by any reboot event. The general trigger parameters may also be specified.

The **repeat** parameter is a general trigger parameter that can be used with all trigger types. It specifies whether the trigger repeats, and how many times the trigger repeats. If **yes** or **forever** is specified, the trigger repeats indefinitely, or until it is disabled. If **no** or **once** is specified, the trigger activates only once. If a numeric value is specified, the trigger repeats the set number of times. The default is **forever**. After the trigger has repeated for the specified number of times, the switch disables the trigger so it no longer activates. If you create a configuration script using the [create config command on page 5-22 of Chapter 5, Managing Configuration Files and Software Versions](#) while the trigger is in this disabled state, the configuration script will create the trigger but set its **state** to **disabled**. Use the [enable trigger command on page 44-14](#) to enable the trigger again.

The **script** parameter is a general trigger parameter that can be used with all trigger types. It specifies the name of a script to execute when the trigger is activated. A script is a predefined list of switch commands. The specified script must already exist. The **script** parameter may be repeated up to five times in one command, to add up to five scripts (in the order specified) at once. Additional scripts may be added using the [add trigger command on page 44-6](#).

The **state** parameter is a general trigger parameter that can be used with all trigger types. It specifies the initial state of the trigger. By default triggers are enabled when created. A trigger is activated automatically only when it is enabled. A trigger can be manually activated with the [activate trigger command on page 44-5](#) regardless of whether the trigger is enabled or disabled.

The **test** parameter is a general trigger parameter that can be used with all trigger types. It specifies whether this trigger is in **test** mode. When in test mode, it activates and logs the trigger but does not execute configured scripts. The default is **no**.

The **time** parameter defines a time trigger and specifies the time of day in hours and minutes when the trigger is to be activated. Resolutions of up to one minute with an accuracy of five seconds are supported. The trigger is activated at most five seconds after the specified minute. The type parameters **date** or **days**, and the general parameters **script**, **name**, **repeat**, **state**, and **test** may also be specified. The **after** and **before** parameters are not valid, but all other general trigger parameters may also be specified.

**Examples** To create trigger 1 that activates at 6am every weekday and initiates script offpeak.scp, use the command:

```
cre trig=1 ti=06:00 day=weekday sc=offpeak.scp rep=y
```

**Related Commands**

- [activate trigger](#)
- [add trigger](#)
- [destroy trigger](#)
- [disable trigger](#)
- [enable trigger](#)
- [set trigger](#)
- [show trigger](#)

## delete trigger

---

**Syntax** `DELEte TRIGger=trigger-id NUMber=index`

where:

- *trigger-id* is a number from 1 to 250.
- *index* is a number from 1 to *n*, where *n* is the number of scripts assigned to the trigger.

**Description** This command removes a script from a trigger. The **trigger** parameter specifies the number of the trigger from which the script is to be deleted.

The **number** parameter specifies the position in the script list of the script to be removed. The specified script must already exist in that position.

**Examples** To remove the third script from trigger 1, use the command:

```
del trig=1 num=3
```

**Related Commands** [add trigger](#)  
[destroy trigger](#)  
[set trigger](#)  
[show trigger](#)

## destroy trigger

---

**Syntax** `DESTroy TRIGger=trigger-id`

where *trigger-id* is a number from 1 to 250

**Description** This command destroys a previously-defined trigger. The **trigger** parameter specifies the number of the trigger to destroy.

**Examples** To destroy trigger 1, use the command:

```
dest trig=1
```

**Related Commands** [add trigger](#)  
[create trigger](#)  
[delete trigger](#)  
[disable trigger](#)  
[enable trigger](#)  
[purge trigger](#)  
[show trigger](#)

## disable trigger

---

**Syntax** `DISable TRIGger [=trigger-id]`

where *trigger-id* is a number from 1 to 250

**Description** This command disables the entire trigger facility if a trigger is not specified, or a specific trigger. The specified trigger will no longer be eligible for activation; however, you can still manually activate it with the **activate trigger** command.

The **trigger** parameter specifies the number of the trigger to disable. The specified trigger must already exist.

**Examples** To disable trigger 1, use the command:

```
dis trig=1
```

To disable the trigger module, use the command:

```
dis trig
```

**Related Commands**

- [activate trigger](#)
- [delete trigger](#)
- [destroy trigger](#)
- [enable trigger](#)
- [purge trigger](#)
- [show trigger](#)

## enable trigger

---

**Syntax** `ENABle TRIGGer [=trigger-id]`

where *trigger-id* is a number from 1 to 250

**Description** This command enables the entire trigger facility if a trigger is not specified, or a specific trigger. After you enable the trigger facility, you can specify an individual trigger. The specified trigger is then eligible for activation. All triggers are enabled by default when they are created. Except for manual activation with the **activate trigger** command, disabled triggers cannot be activated.

The **trigger** parameter specifies the number of the trigger to enable. The specified trigger must already exist.

**Examples** To enable the trigger module, use the command:

```
ena trig
```

To enable trigger 1, use the command:

```
ena trig=1
```

**Related Commands**

- [activate trigger](#)
- [delete trigger](#)
- [destroy trigger](#)
- [disable trigger](#)
- [purge trigger](#)
- [set trigger](#)
- [show trigger](#)

## purge trigger

---

**Syntax** `PURge TRIGGer`

**Description** This command erases the trigger facility configuration.

**Related Commands**

- [delete trigger](#)
- [destroy trigger](#)
- [disable trigger](#)
- [enable trigger](#)
- [set trigger](#)
- [show trigger](#)

## set trigger

**Syntax** SET TRIGger=*trigger-id* [CPU[=*value*]] [Direction={UP|DOWN|ANY}] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGGER=*trigger-id* [INTERface[=*interface*]] [Event={UP|DOWN|FAIL|ANY}] [CP={APPLE|ATCP|BCP|CCP|DCP|DNCP|IPcp|IPXcp|LCP}] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [MEMory[=*value*]] [Direction={UP|DOWN|ANY}] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [MODule] [*module-parameters...*] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [PERiodic[=*minutes*]] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [Port=*port*] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [REBoot[={REStart|CRASH|ALL}]] [AFTER=*hh:mm*] [BEFORE=*hh:mm*] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

SET TRIGger=*trigger-id* [Time[=*hh:mm*]] [{DATE=*date*|DAYS=*day-list*}] [NAME=*name*] [REPEAT={Yes|No|ONCe|FOREver|*count*}] [TEST={YES|NO|ON|OFF|True|False}]

where:

- *trigger-id* is a number from 1 to 250.
- *module* is the name of a switch module as given in “[Module Identifiers and Names](#)” on page B-2 of Appendix B, Reference Tables.
- *event* is an event defined in the specified module
- *module-parameters* are one or more other required or optional parameters for the specified module and event, whose syntax is defined in the specified module.
- *port* is the port number. Port numbers start at 1 and end at *m*, where *m* is the highest numbered Ethernet switch port.
- *value* is a number from 1 to 100.
- *interface* is a valid interface name.

- *minutes* is a number from 1 to 1439.
- *hh:mm* is a time in hours and minutes.
- *date* is a date in the format dd-mmm-yyyy, where *mmm* is the first three letters of the month name.
- *day-list* is one or more keywords for the day of the week.
- *name* is a string 1 to 40 characters long. If the string contains spaces, it must be in double quotes.
- *count* is a number from 1 to 4 294 967 294 ( $2^{32}-2$ ).

**Description** This command modifies the definition of a trigger. Different trigger types are supported—CPU triggers, link triggers, memory triggers, module triggers, periodic triggers, port triggers, reboot triggers, and time triggers. Some of these have additional trigger-specific parameters. Parameters for module triggers are described in the chapter for the module that supports them. General trigger parameters can be specified for all trigger types: **after**, **before**, **days** or **date**, **name**, **repeat**, and **test**. The type of trigger cannot be changed.

The **trigger** parameter specifies the number of the trigger to create. The number references the trigger in other commands. The specified trigger must not already exist. The **trigger** parameter must immediately follow the **create** parameter.

The **after** parameter is a general trigger parameter that can be used with all trigger types except for **time** triggers. It specifies the earliest time of day in hours and minutes that the trigger is activated. The trigger is activated any time between the time specified and midnight.

The **before** parameter is a general trigger parameter that can be used with all trigger types except for **time** triggers. It specifies the latest time of day in hours and minutes that the trigger is activated. The trigger is activated any time between midnight and the time specified.

If neither **after** nor **before** are specified, there is no restriction on when the trigger may activate the script. If both **after** and **before** are specified, and the time periods specified overlap, the trigger may activate it any time during the overlap period (Figure 44-2 on page 44-9).

The **cp** parameter specifies a PPP control protocol to monitor. The trigger activate when the condition defined by the combination of **interface**, **cp**, and **event** occurs. The **cp** parameter may only follow an **interface** parameter that specifies a PPP interface. When activated, the trigger passes three parameters to the trigger scripts—the PPP instance that caused the trigger, the control protocol, and the event state.

The **cpu** parameter defines a CPU utilisation trigger and specifies the CPU utilisation level at which the trigger is to be activated. The **direction** parameter and the general trigger parameters may also be specified.

The **date** parameter is a general trigger parameter that can be used with all trigger types. It specifies a date on which the trigger may activate. The **days** and **date** parameters are mutually exclusive—use one or the other.



The **days** parameter is a general trigger parameter that can be used with all trigger types. It specifies a comma-separated list of days when the trigger is activated. The **weekday** option is a synonym for the list **mon, tue, wed, thu, fri**; **weekend** is a synonym for the list **sat, sun**. Any combination of these and the day names is acceptable. The default is **all**. The **days** and **date** parameters are mutually exclusive—use one or the other.

The **direction** parameter specifies how the CPU or memory utilisation threshold is reached to activate the trigger. If **up** is specified, the trigger is activated when CPU or memory utilisation increases to or exceeds the threshold. If **down** is specified, the trigger is activated when CPU or memory utilisation decreases to or falls below the threshold. If **any** is specified, the trigger is activated when CPU or memory utilisation equals or passes the threshold in either direction. The default is **any**.

The **dlci** parameter specifies a Frame Relay DLC (Data Link Connection) to monitor. The trigger is activated when the condition defined by the combination of **interface**, **dlci**, and **event** occurs. The **dlci** parameter may only follow an **interface** parameter that specifies a Frame Relay interface. When activated, the trigger passes three parameters to the trigger scripts—the Frame Relay instance that caused the trigger, the DLCI, and the event state.

The **event** parameter after the **interface** parameter specifies a link (interface) status change event. If **up** is specified, the trigger is activated when the interface is operational. If **down**, it is activated when the interface closes. If **fail** is specified, the trigger is activated when the interface fails to open for any reason. If **any** is specified, it is activated when any of these events occur.

The **interface** parameter defines an interface (link) trigger and specifies the interface to monitor. The **event** parameter specifies a link (interface) status change event. The **cp** parameter may be used if **interface** specifies a PPP interface. The general trigger parameters may also be specified. The type of trigger cannot be changed. Valid interfaces are:

- eth (such as eth0)
- PPP (such as ppp0)
- VLAN (such as vlan1)

The **interface** parameter specifies a valid interface already assigned and configured. To see a list of current valid interfaces, use the [show interface command on page 10-51 of Chapter 10, Interfaces](#).

The **memory** parameter defines a memory utilisation trigger and specifies the percentage of free memory at which the trigger activates. The **direction** parameter and general trigger parameters may also be specified. Use the [show buffer command on page 4-37 of Chapter 4, Configuring and Monitoring the System](#) to display the current percentage of free memory.

The **module** parameter specifies that the trigger to be modified is a module type trigger. The **event** parameter is not valid, but other module parameters may be specified. The general trigger parameters may also be specified.

The **name** parameter is a general trigger parameter that can be used with all trigger types. It specifies a descriptive name for this trigger.

The **periodic** parameter defines a periodic trigger and specifies the period of the trigger in minutes. The general trigger parameters may also be specified. The type of trigger cannot be changed.

The **port** parameter specifies the port where the event activates the trigger.

The **reboot** parameter defines a reboot trigger and specifies a list of reboot events that activates the trigger. If **crash** is specified, the trigger is activated by a switch crash. If **restart** is specified, the trigger is activated by any reboot other than a switch crash. If **all** is specified, the trigger is activated by any reboot event. The general trigger parameters may also be specified. The type of trigger cannot be changed.

The **repeat** parameter is a general trigger parameter that can be used with all trigger types. It specifies whether the trigger repeats, and how many times the trigger repeats. If **yes** or **forever** is specified, the trigger repeats indefinitely, or until it is disabled. If **no** or **once** is specified, the trigger activates only once. If a numeric value is specified, the trigger repeats the set number of times. The default is **forever**. After the trigger has repeated for the specified number of times, the switch disables the trigger so it no longer activates. If you create a configuration script using the [create config command on page 5-22 of Chapter 5, Managing Configuration Files and Software Versions](#) while the trigger is in this disabled state, the configuration script will create the trigger but set its **state** to **disabled**. Use the [enable trigger command on page 44-14](#) to enable the trigger again.

The **test** parameter is a general trigger parameter that can be used with all trigger types. It specifies whether this trigger is in **test** mode. When in test mode, it activates and logs the trigger but does not execute configured scripts. The default is **no**.

The **time** parameter specifies the time of day in hours and minutes when the time trigger is to be activated. Resolutions of up to one minute with an accuracy of five seconds are supported. The trigger is activated at most five seconds after the specified minute. The **after** and **before** parameters are not valid but other general trigger parameters can be specified. The type of trigger cannot be changed.

**Examples** To modify time trigger 1 to activate at 8am every weekday, use the command:

```
set trig=1 ti=08:00 day=weekday rep=y
```

**Related Commands**

- [activate trigger](#)
- [add trigger](#)
- [create trigger](#)
- [destroy trigger](#)
- [disable trigger](#)
- [enable trigger](#)
- [show trigger](#)

## show trigger

**Syntax** `SHOW TRIGGER[=trigger-id] [{COUnTer|FULl|STAtus|SUMmary}]`

where *trigger-id* is a number from 1 to 250

**Description** This command displays information about all triggers that have been configured, a specific one, or general configuration information about the trigger facility. The **trigger** parameter specifies the number of the trigger to display.

If neither trigger nor parameter is specified, or the **summary** parameter is specified, summary information for all or the specified triggers is displayed (Figure 44-3, Table 44-1). If **full** is specified, or a trigger is specified without the **summary** parameter, detailed information about the specified triggers is displayed (Figure 44-4 on page 44-20, Table 44-2 on page 44-20).

The **status** parameter displays general configuration information for the trigger facility (Figure 44-5 on page 44-21, Table 44-3 on page 44-21). A trigger identifier may not be specified.

The **counter** parameter displays counters for the trigger facility (Figure 44-6 on page 44-22, Table 44-4 on page 44-22). A trigger identifier may not be specified.

Figure 44-3: Example output from the **show trigger** command

TR#	Type & Details	Name	En	Te	Rept	#Scr	Days/Date
001	Periodic (3 min)	Test Trigger	Y	Y	Yes	01	-TW-FSS
002	Time (10:00)	Call home	Y	N	Yes	01	23-Apr-2000
003	Reboot (Crash)	Get Debug Info	Y	N	Yes	01	MTWTFSS
004	Module (VRRP)	DOWNMASTER	Y	N	No	01	MTWTFSS

Table 44-1: Parameters in output of the **show trigger** command

Parameter	Meaning
TR#	Trigger identifier (ID).
Type & Details	Trigger type and details: Time - trigger time Periodic - period Reboot - either Crash, Reboot, or All Memory CPU Module - module, event
Name	Descriptive name of the trigger.
En	Whether the trigger is enabled.
Te	Whether the trigger is in test mode.
Rept	Whether the trigger repeats or a repeat count. The repeat count is decremented each time the trigger is activated automatically.
#Scr	Number of scripts associated with the trigger.
Days/Date	Days or date when the trigger is activated. For the days options, the days are shown as a seven character string representing Monday to Sunday. A hyphen indicates days when the trigger is not activated.

Figure 44-4: Example output from the **show trigger full** command

```

Trigger Configuration Details
-----
Trigger ..... 1
Name ..... Bring up Wellington link
Type and details ..... Time (13:45)
Days ..... All
Enabled ..... Enabled
Test ..... No
Repeat ..... No
Created/Modified ..... 08-Nov-1996 12:04:33
Number of Activations ..... 1
Last Activation ..... 08-Nov-1996 13:45:07
Number of scripts ..... 2

    callwgtn.scp
    idlegtn.scp

Trigger ..... 4
Name ..... VRRP master down action
Type and details ..... Module (VRRP) DOWNMASTER
Other Parameters ..... VRID=1
Days ..... Daily
Active TCP sessions ..... 0
Enabled ..... Enabled
Test ..... No
Repeat ..... No
Created/Modified ..... 25-Aug-2001 16:37:22
Number of Activations ..... 1
Last Activation ..... 08-Nov-1996 14:40:09
Number of scripts ..... 1

    downmast.scp

```

Table 44-2: Parameters in output of the **show trigger full** command

Parameter	Meaning
Trigger	Trigger identifier (ID).
Name	Descriptive name of the trigger.
Type and details	Trigger type and details Time - trigger time Periodic - period Reboot - either Crash, Reboot, or All Memory CPU Module - for module-specific triggers, the module name and the event on which the trigger is activated are included
Other parameters	Parameters for module-specific triggers, one per line, that define the trigger.
Days	List of the days when the trigger is activated, or either "Weekdays" (Monday to Friday), "Weekends" (Saturday and Sunday) or "Daily" (every day). Either "Days" or "Date" is displayed.
Date	Date or days when the trigger is activated.
Active TCP sessions	The total number of currently active TCP sessions on the switch.

Table 44-2: Parameters in output of the **show trigger full** command (Continued)

Parameter	Meaning
Enabled	Whether the trigger is enabled.
Test	Whether the trigger is in test mode.
Repeat	Whether the trigger repeats.
Created/Modified	Date and time the trigger was created or last modified.
Number of Activations	Number of times the trigger has been activated (triggered) since the last switch restart.
Last Activation	Date and time the trigger was last activated (triggered).
Number of scripts	Number of scripts assigned to the trigger, followed by a list of the script file names.

Figure 44-5: Example output from the **show trigger status** command

```

Trigger Module Configuration
-----
General
  Trigger Module ..... Enabled
  Triggers configured ..... 4
  Queued Commands ..... 0

Time Triggers
  Configured ..... 2
  Active ..... 2
  Activated today ..... 1

Periodic Triggers
  Configured ..... 1
  Active ..... 1
  Activated today ..... 0

Reboot Triggers
  Configured ..... 0

Interface Triggers
  Configured ..... 0

Resource Triggers
  Configured ..... 1
  Active ..... 1
  Activated today ..... 0

Module Triggers
  Configured ..... 2
  Activated today ..... 3

```

Table 44-3: Parameters in output of the **show trigger status** command

Parameter	Meaning
General	General information about the Trigger Facility.
Trigger Module	Whether the trigger module is enabled.
Triggers configured	Total number of triggers that have been configured.
Queued commands	Number of commands that are queued for execution.
Time Triggers	Information about time triggers.

Table 44-3: Parameters in output of the **show trigger status** command (Continued)

Parameter	Meaning
Periodic Triggers	Information about periodic triggers.
Reboot Triggers	Information about reboot triggers.
Interface Triggers	Information about interface triggers.
Resource Triggers	Information about CPU and memory resource triggers.
Module Triggers	Information about module triggers. Valid for x900-48FE switches only.
Configured	Number of triggers of the associated type that have been configured.
Active	Number of triggers of the associated type that are currently active (enabled).
Activated today	Number of times a trigger of the associated type has been activated (triggered) today.

Figure 44-6: Example output from the **show trigger counter** command

```

Trigger Module Counters
-----
Polls (05 sec timer) ..... 37
Idle loop entry count ..... 5
Time trigger checks ..... 2
Time trigger queue rebuilds ..... 1
Trigger activations ..... 1
Time triggers activated today ..... 1
Periodic triggers activated today .. 0
Interface triggers activated today . 0
Resource triggers activated today .. 0
Module triggers activated today .... 3

```

Table 44-4: Parameters in output of the **show trigger counter** command

Parameter	Meaning
Polls (05 sec timer)	Number of times the trigger module has polled for a trigger activation event.
Idle loop entry count	Number of times the trigger module has prepared commands for execution.
Time trigger checks	Number of times the trigger module has checked the list of time triggers for a trigger to activate.
Time trigger queue rebuilds	Number of times the time trigger queue has been rebuilt because time triggers have been added, deleted or modified, or because the time/date has been changed.
Trigger activations	Number of times a trigger has been activated.
Time triggers activated today	Number of times a time trigger has been activated today.
Periodic triggers activated today	Number of times a periodic trigger has been activated today.
Interface triggers activated today	Number of times an interface trigger has been activated today.

Table 44-4: Parameters in output of the **show trigger counter** command (Continued)

Parameter	Meaning
Resource triggers activated today	Number of times a CPU or memory resource trigger has been activated today.
Module triggers activated today	Number of times a module trigger has been activated today.

**Examples** To display summary information for trigger 3, use the command:

```
sh trig=3 sum
```

To display summary information for all triggers, use the command:

```
sh trig
```

To display a detailed description of trigger 3, use the command:

```
sh trig=3
```

To display a detailed description of all triggers, use the command:

```
sh trig ful
```

To display general configuration information for the trigger facility, use the command:

```
sh trig sta
```

To display counters for the trigger facility, use the command:

```
sh trig cou
```

**Related Commands**

- [activate trigger](#)
- [add trigger](#)
- [create trigger](#)
- [delete trigger](#)
- [destroy trigger](#)
- [disable trigger](#)
- [enable trigger](#)
- [purge trigger](#)
- [set trigger](#)

