

# How To Use the WebNS Command Scheduler on the CSS 11000 and 11500

Document ID: 5311

---

## Introduction

### Before You Begin

- Conventions

- Prerequisites

- Components Used

### Examples

- Record Output

- Execute a Command

- Collect Infos from CSS Operation Periodically

### Related Information

---

## Introduction

In the WebNS operating system, the **cmd-sched** command is used to schedule execution of CLI commands or playing of scripts. This document provides examples for how to use the command scheduler on the Content Services Switch (CSS) 11000 and 11500 series.

## Before You Begin

### Conventions

For more information on document conventions, see the Cisco Technical Tips Conventions.

### Prerequisites

There are no specific prerequisites for this document.

### Components Used

This document is not restricted to specific software and hardware versions.

The information presented in this document was created from devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If you are working in a live network, ensure that you understand the potential impact of any command before using it.

## Examples

### Record Output

In this example, the CSS records the output of the **show system-resources** command in a file. This information can be used for troubleshooting. The **show system-resources** command has to be executed every 30 minutes for three hours (for a total of six times). The command syntax is:

```
cmd-sched record name minute hour day month weekday "command..." {logfile_name}
```

For a more detailed explanation of each field in the command line, refer to the *Usage Guidelines* section of Global Configuration Mode Commands.

In this example, the command is executed today, for six times, every five minutes, starting at 12 o'clock. The output also needs to be saved in a specific file (output.log).

```
CS11503(config)#show clock

Date: 07-29-2004
Time: 12:00:07

TimeZone: EST: -05:-00:+00

CS11503(config)#cmd-sched record sched1 0,05 12-14 29 07 *
                "show system-resources" output.log

CS11503(config)#show cmd-sched
Cmd Scheduler:Enabled    2 cmd records currently configured.

Sched Rec:sched2 id:8a5bb640 next exec:AUG 1 11:00:00 executions:0
  minList:0
  hourList:11
  dayList:*
  monthList:*
  weekdayList:1
  cmd:clear log sys.log
Sched Rec:sched1 id:8a58acb0 next exec:JUL 29 12:05:00 executions:0
  minList:0 5
  hourList:12 13 14
  dayList:29
  monthList:7
  weekdayList:*
  cmd:show system-resources
  log:output.log

CSS(config)#show cmd-sched
Cmd Scheduler:Disabled    1 cmd records currently configured.

Sched Rec:sched1 id:854cf160 next exec:UnScheduled executions:0
  minList:0 30
  hourList:10 11 12
  dayList:13
  monthList:10
  weekdayList:*
  cmd:show system-resources
  log:output.log
```

If the scheduler process was not activated the Cmd Scheduler states is shown as Disabled, as shown in the command output above.

```
C:\WINNT\system32\command.com
monthList:7
weekdayList:*
cmd:show system-resources
log:output.log

CS11503(config)# show cmd-sched
Cmd Scheduler:Enabled      2 cmd records currently configured.

Sched Rec:sched2 id:8a5bb640 next exec:AUG  1 11:00:00 executions:0
minList:0
hourList:11
dayList:*
monthList:*
weekdayList:1
cmd:clear log sys.log
Sched Rec:sched1 id:8a58acb0 next exec:JUL 29 14:00:00 executions:3
minList:0 5
hourList:12 13 14
dayList:29
monthList:7
weekdayList:*
cmd:show system-resources
log:output.log

CS11503(config)#
```

To view the log entry created, issue the **show log output.log** command:

```
C:\WINNT\system32\command.com
CS11503(config)# show log output.log
show system-resources
System Resources for 1/1:
Installed Memory:  268,435,456 <256 MB>
Free Memory:      137,831,872 <131 MB>
CPU:              0%

Buffer Statistics:
Buffer Pool: 0
  Size:2048  Total:3072  Available:2832  Failures: 0  Low Buffer Count: 2832

Buffer Pool: 1
  Size:2048  Total:3072  Available:2812  Failures: 0  Low Buffer Count: 2800

Buffer Pool: 2
  Size:2048  Total:2048  Available:2016  Failures: 0  Low Buffer Count: 2016

System Resources for 3/1:
Installed Memory:  134,217,728 <128 MB>
Free Memory:      29,747,840 <28 MB>
CPU:              0%

Buffer Statistics:
Buffer Pool: 0
  Size:2048  Total:3072  Available:2812  Failures: 0  Low Buffer Count: 2808

Buffer Pool: 1
  Size:2048  Total:3072  Available:2808  Failures: 0  Low Buffer Count: 2800

Buffer Pool: 2
  Size:2048  Total:2048  Available:2024  Failures: 0  Low Buffer Count: 2024

show system-resources
System Resources for 1/1:
Installed Memory:  268,435,456 <256 MB>
```

Remember, because there is not an entry for the year, after the six scheduled executions, the next one is set for next July 29th at 12 o'clock. This means that if you do not remove the scheduled command **sched1** from the configuration, the CSS records the output of the command **show system-resources** six times every July 29th.

**Note:** To remove the **cmd-sched** command record, issue the **no cmd-sched record sched1** command.

## Execute a Command

In this example, the command **clear log sys.log** is to be executed every Sunday at 11:00 AM. You do not need to record the command output. The day of the week is expressed in numbers. Valid numbers are from 1 to 7, and Sunday is 1. To specify that you want the command to be repeated every day and month, you need to use an asterisk (\*) as the wildcard. Even if the command is executed only on Sundays, you still need to set an asterisk for the day.

```
CSS(config)#cmd-sched record sched2 0 11 * * 1 "clear log sys.log"
```

```
CSS(config)#show cmd-sched
```

```
Sched Rec:sched1 id:854cf160 next exec:OCT 13 10:30:00 executions:1
  minList:0 30
  hourList:10 11 12
  dayList:13
  monthList:10
  weekdayList:*
  cmd:show system-resources
  log:output.log
Sched Rec:sched2 id:85571840 next exec:OCT 14 11:00:00 executions:0
  minList:*
  hourList:11
  dayList:*
  monthList:*
  weekdayList:1
  clear log sys.log
```

```
CSS(config)#show running-config
```

```
!Generated on 10/13/2001 09:34:36
!Active version: ap0501005
```

```
configure
```

```
!***** GLOBAL *****
```

```
dns primary 144.254.6.77
```

```
ip route 0.0.0.0 0.0.0.0 172.17.241.1 1
```

```
cmd-sched record sched1 0,30 10-12 13 10 * "show system-resources" output.log
```

```
cmd-sched record sched2 * 11 * * 1 "clear log sys.log"
```

```
cmd-sched
```

```
!***** INTERFACE *****
```

```
interface e2
```

```
bridge vlan 2
```

```
!***** CIRCUIT *****
```

```
circuit VLAN1
```

```
ip address 172.17.241.128 255.255.255.0
```

```
circuit VLAN2
```

```
ip address 10.10.10.120 255.255.255.0
```

```
!***** SERVICE *****
```

```
service server1
```

```
ip address 10.10.10.150
```

```
active
```

```
!***** OWNER *****
```

```
owner test

content test
vip address 172.17.241.235
add service server1
protocol tcp
port 80
active
```

## Collect Infos from CSS Operation Periodically

This example uses the **cmd-sched** command to periodically capture information about the status and operations of the CSS, and log it into a file in the logs directory. The command syntax is:

```
css# script record ciscoinfo
```

This is a list of commands that are relevant:

```
show summary
show service summary
show ether-errors
show system-resources
show mibii
show mibii-32
show bridge status
show bridge forwarding
show redundant-interfaces
show redundant-vips
show critical-services
llama
flow stat
exit
```

**Note:** The **script play showtech** command is available in the latest WebNS release. This command produces a list of command outputs that are considered relevant by the Cisco Technical Assistance Center (TAC) for initial troubleshooting.

Type **<control-c>** to end the recording of the script. Please note that if you make a mistake when entering the command and get an error, you need to start the script record again. Afterwards, check that the script is recorded by issuing the **show script ciscoinfo** command. You then need to schedule it, as shown here:

```
css# conf t
css(config)# cmd-sched
css(config)# cmd-sched record ciscotac 30 * * * * "script play ciscoinfo" ciscotac.txt
```

Issuing this command runs the script at X:30 of each hour, and appends the output to the file ciscotac.txt in the log director of the CSS. After the first 30 minutes past the hour, if you issue the **sh log ?** command, you should see the ciscotac.txt file there. Please check this log to make sure it is complete by issuing the command shown below.

```
show log ciscotac.txt
```

Every time a command is executed, a pseudo shell is temporarily opened for it. If you are scheduling scripts that take some time to be executed, you can view the pseudo shell running by issuing this command:

```
CSS# show lines
Line      User      Login      Idle      ; Location
----      -
* console admin    0 days 00:00:52  0 days 00:00:00  local
```

pty31 exeRec\_sched4 0 days 00:00:01 0 days 00:00:00 system

**Note:** To terminate the execution of a shell and stop the scheduled script, you can issue the **disconnect** command, followed by the session identifier (the line):

```
CSS#disconnect pty31
```

---

## Related Information

- [Application Networking Services Support Resources](#)
  - [Technical Support – Cisco Systems](#)
- 

All contents are Copyright © 2006–2007 Cisco Systems, Inc. All rights reserved. Important Notices and Privacy Statement.

---

Updated: Aug 03, 2007

Document ID: 5311

---