

WatchDog Commands

The WatchDog is responsible for bootstrapping the MPLS VPN Solution and starting the necessary set of server processes. In addition, the WatchDog monitors the health and performance of each server to ensure it is functioning properly. In the event of a software error that causes a server to fail, the WatchDog automatically restarts the errant server.

The WatchDog is a background daemon process that is automatically installed as part of the installation procedure for MPLS VPN Solution. After the installation procedure has been completed, you can execute the **startwd** command to run the WatchDog for the first time. The WatchDog is automatically started any time the machine is rebooted.

This chapter provides the description, syntax, and arguments (listed alphabetically) for the following WatchDog commands:

- startwd Command, page 2-1
- stopwd Command, page 2-2
- wdclient Command, page 2-2
- wdgui Command, page 2-9
- wdperf Command, page 2-14

startwd Command

This section provides the description and syntax for the startwd command.

Description

The **startwd** command starts the WatchDog and all MPLS VPN Solution processes. Running this manually is necessary after installing new software, after changing the **csm.properties** file, or when restarting after issuing a **stopwd** command. The **startwd** command is run automatically when the machine is rebooted.



The Orbix daemon must be running for the **startwd** command to operate correctly. If the Orbix daemon is not running, you will receive a message indicating that.

startwd



The **startwd** command has no arguments.



The location of **startwd** is: *<MPLS VPN Directory>/bin*.

stopwd Command

This section provides the description and syntax for the stopwd command.

Description

The **stopwd** command stops the WatchDog and all MPLS VPN Solutionprocesses. Normally this will only be necessary before installing new versions of MPLS VPN Solution or changing the **csm.properties** file. When stopping and restarting the WatchDog, the **csm.properties** file is reread.

Syntax

stopwd [-y]

where: $-\mathbf{y}$ indicates not to prompt before shutdown. If $-\mathbf{y}$ is not specified, you are prompted with the following message: "Are you absolutely sure you want to stop the watchdog and all of its servers? Other users may be using this system as well. No activity (e.g.: collections, performance monitoring, provisioning) will occur until the system is restarted." You are then prompted to reply **yes** or **no**.



The location of **stopwd** is: *<MPLS VPN Directory>/bin*.

wdclient Command

This section provides the description, syntax, and options (listed alphabetically) for the **wdclient** subcommands. These subcommands are diagnostic tools. This section also describes the column format of the output of each of the subcommands.



The location of **wdclient** is: *<MPLS VPN Directory>/bin*.

The following are the **wdclient** subcommands:

- wdclient group Subcommand, page 2-3
- wdclient groups Subcommand, page 2-3
- wdclient log Subcommand, page 2-4

- wdclient logs Subcommand, page 2-4
- wdclient restart Subcommand, page 2-5
- wdclient start Subcommand, page 2-5
- wdclient status Subcommand, page 2-6
 - Information Produced: Name Column, page 2-6
 - Information Produced: State Column, page 2-7
 - Information Produced: Gen Column, page 2-8
 - Information Produced: Exec Time Column, page 2-8
 - Information Produced: Success Column, page 2-8
 - Information Produced: Missed Column, page 2-8
- wdclient stop Subcommand, page 2-9

wdclient group Subcommand

This section provides the description and syntax for the wdclient group subcommand.

Description

The **wdclient group** subcommand lists the servers in the specified server group. Server groups provide a convenient way to start or stop a group of servers with a single command.

Syntax

wdclient [-host <hostname>] group <group_name>

where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

<*group_name*> is the name of a server group chosen from the list displayed by the **wdclient groups** command.

wdclient groups Subcommand

This section provides the description and syntax for the wdclient groups subcommand.

Description

The wdclient groups subcommand lists all the active server groups.

wdclient [-host <hostname>] groups

where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

wdclient log Subcommand

This section provides the description and syntax for the wdclient log subcommand.

Description

The wdclient log subcommand displays the specified number of lines of the specified server log.

Syntax

wdclient [-host <hostname>] [-poll <seconds>] log <log name> [<lines>]

where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

-poll *<seconds>* is an optional parameter. *<seconds>* is the number of seconds. A number other than zero indicates that when new status data is available it will be displayed every *<seconds*> seconds, where *<seconds>* is the specified number of seconds. The default **-poll** value is zero (0).

<log_name> is the name of a server log displayed by the wdclient logs command.

log. The default number of lines is 100.

```
Note
```

The complete history (log) file of all WatchDog servers is in the watchdog subdirectory of the temporary directory as configured in the csm.properties file. This temporary directory is specified during system configuration. If the WatchDog is stopped and restarted, each log file is renamed from server.<log_name> to server.<log_name>.<time_stamp>, where <log_name> is the same as specified in the wdclient log subcommand and <time_stamp> is a time indicator of when this file was created. The new logs are then collected in server.<log_name>. If the WatchDog is not stopped and restarted within a 24-hour period, the log file is automatically renamed with a *<time_stamp>* and a new file is started. Also, any log file more than a week old is automatically deleted.

wdclient logs Subcommand

This section provides the description and syntax for the wdclient logs subcommand.

Description

The wdclient logs subcommand lists the names of all the logs.

wdclient [-host <hostname>] logs

where: -host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

wdclient restart Subcommand

This section provides the description and syntax for the wdclient restart subcommand.

Description

The wdclient restart subcommand restarts one or more servers. Any dependent servers will also be restarted.

Note

It is not necessary to restart servers in a properly functioning system. The **wdclient restart** command should only be run under the direction of Cisco Support. Restarting an individual server will not read in changes in the **csm.properties** file. For changes in the **csm.properties** file to be effective, stop the WatchDog and restart it.

Syntax

wdclient [-host <hostname>] restart {<server_name> | group <group_name> | all}

where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

You must choose one of the following arguments:

<server_name> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 2-1, "Servers and Their Functions," for server descriptions.

group <group_name> is the term group followed by the name of a server group chosen from the list displayed by the wdclient groups command.

all is all servers.

wdclient start Subcommand

This section provides the description and syntax for the wdclient start subcommand.

Description

The **wdclient start** subcommand starts one or more servers. Other servers that depend on the specified server(s) may also start.

<u>Note</u>

It is not necessary to stop and start servers in a properly functioning system. The **wdclient start** command should only be run under the direction of Cisco Support.

Syntax

```
wdclient [-host <hostname>] start {<server_name> | group <group_name> | all}
```

where:

-host <hostname> is an optional parameter.<hostname> is the name of the remote host on which the WatchDog is running.

You must choose one of the following three arguments.

<*server_name*> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 2-1, "Servers and Their Functions," for server descriptions.

group <group_name> is the name of a server group chosen from the list displayed by the wdclient groups command.

all is all servers.

wdclient status Subcommand

This section provides the description, syntax, and information produced for the wdclient status subcommand.

Description

The **wdclient status** subcommand lists all the servers and their states. See Table 2-2 on page 2-8, "Valid States," for the list of all the states.

Syntax

wdclient [-host <hostname>] [-poll <seconds>] status

where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

-poll <*seconds*> is an optional parameter. <*seconds*> is the number of seconds. A number other than zero indicates that when new status data is available it will be displayed every <*seconds*> seconds, where <*seconds*> is the specified number of seconds. The default -poll value is zero (0).

Information Produced: Name Column

The **Name** column provides the name of each of the servers. Table 2-1 provides a list of the servers and a description of the function that each server provides.

Server	Function
DataSetServer	Provides a CORBA front end for SA Agent and Accounting APIs.
EventGateway	Gateways events from the TIBCO domain to the CORBA domain.
LayoutServer	Provides topology layout recomputation services for web topology, which is used when selecting certain topology views.
ReportServerFactory	Launches and manages ReportServer processes that generate and provide access to dynamic web reports.
ResourceMgr	Handles device locking so a router's configuration is not modified by multiple service requests at the same time, and allocates Telnet Gateway Servers in the system for download/upload requests.
TGServer	Provides a CORBA API to download configlets, upload configuration files, and send IOS commands to the router.
TaskServer	Provides a CORBA front end to the MPLS VPN Solution task repository.
TemplateServer	Provides a CORBA front end to the Template Provisioning System.
VerifyReportServer	Back end that generates Verify Reports.
VpnInvServer	Provisioning API CORBA server.
aggregator	Aggregates collected datasets periodically.
httpd	Web server.
journal	Listens to all repository events and saves them into journal files. Also archives the journal files periodically.
lock_manager	Handles locking for the internal database.
log	Makes the output of tasks available to you in a browsable format.
poller	Gets requests from other data collectors and forwards the requests to the device. Gets the response and sends it to appropriate collectors.
rmiregistry	Underlying communication process necessary for ReportServerFactory and LayoutServer to communicate with web clients.
scheduler	Enables you to schedule tasks immediately or later in time, for one-time or repeated execution.
trapcatcher	Catches configuration change traps from routers.
watchdog_perf	Tracks performance for the system itself. The data is collected and stored in the internal database only. The data is useful for diagnostics.

Table 2-1	Servers	and	Their	Functions
-----------	---------	-----	-------	-----------

Information Produced: State Column

The **State** column provides the current state of the server. Table 2-2 provides a description of each of the states in normal progression order.

Stata	Description
סומוש	nescuhunu
start_depends	This server has been asked to start, but is waiting for servers it depends on to start. Once all dependent servers have started, this server will transition to the state of starting.
starting	This server is currently starting. Once a successful heartbeat occurs, this server will transition to the state of started.
started	This server is currently started and running.
stop_depends	This server is supposed to be stopped, but it is waiting for servers it depends on to be stopped first.
stopping_gently	This server is in the process of stopping in a gentle fashion. That is, it was notified that it is to stop.
stopping_hard	This server is in the process of being killed because either it did not have a way to stop gently or because the gentle stop took too long.
stopped	This server is stopped. The WatchDog will either start it again or disable it if it has been frequently dying.
disabled_dependent	This server is disabled because one or more servers it depends on are disabled. If all servers it depends on are started, this server will automatically start.
disabled	This server is disabled and must be manually restarted.
restart_delay	This server is delaying before restarting. There is a short delay after a server stops and before it is restarted again.

Table 2-2Valid States

Information Produced: Gen Column

The **Gen** column provides the generation of the server. Each time the server is started, the generation is incremented by 1.

Information Produced: Exec Time Column

The Exec Time column provides the date and time the server was last started.

Information Produced: PID Column

The PID column provides the UNIX process identifier for each server.

Information Produced: Success Column

The **Success** column provides the number of successful heartbeats since the server was last started. Heartbeats are used to verify that servers are functioning correctly.

Information Produced: Missed Column

The Missed column provides the number of missed heartbeats since the server was last started.

A few missed heartbeats could simply indicate the system was busy. However, more than a couple of missed heartbeats per day could indicate a problem. See the logs to diagnose the reason. If a server misses three heartbeats in a row, the server is automatically restarted.

<u>Note</u>

Three missed heartbeats in a row is the default for restarting the server. The default number can be reset in the **csm.properties** file. After three failed attempts to restart in a row, the server is disabled.

wdclient stop Subcommand

This section provides the description and syntax for the wdclient stop subcommand.

Description

The wdclient stop subcommand stops one or more servers. Other servers that depend on the specified servers will also stop.

Note

It is not necessary to stop servers in a properly functioning system. The **wdclient stop** command should only be run under the direction of Cisco Support.

Syntax

wdclient [-host <hostname>] stop {<server_name> | group <group_name> | all}
where:

-host <hostname> is an optional parameter. <hostname> is the name of the remote host on which the WatchDog is running.

You must choose one of the following arguments.

<server_name> is the name of a server chosen from the list displayed by the **wdclient status** command. See Table 2-1, "Servers and Their Functions," for server descriptions.

group <group_name> is the name of a server group chosen from the list displayed by the wdclient groups command.

all is all servers.

wdgui Command

This section provides the description and syntax for the **wdgui** command. This graphical interface to the WatchDog is a diagnostic tool that combines the functionality of the **wdclient status** and **wdclient log** commands. This section also describes the column format of the output when you click each of the tabs.

Description

The **wdgui** command activates the WatchDog user interface. See Figure 2-1, "VPN Solutions Center—Watch Dog."

The top of the screen provides a list of the names of servers. You can drag and drop the columns of information to rearrange them. The columns of information about the servers are described in the following sections:

- Name Column, page 2-11
- State Column, page 2-12
- Generation Column, page 2-13
- Exec Time Column, page 2-13
- Pid Column, page 2-13
- Success Column, page 2-14
- Missed Column, page 2-14

The bottom of the screen provides tabs for each of the servers. Click the tab of the server that you want to track and you will get up to the most current 250 lines of detailed log information.

Syntax

wdgui [&]



The **wdgui** command has no arguments. To run it as a background process, use the optional **&**.

-		VPN Soluti	ons Center – Wa	tch Dog			
Name	State	Generation	Exec Tir	me	Pic	Succe	ss Missed
DataSetServer	started	1	Fri Jan 12 14:52:25 I	PST 2001	20963	2310	0
EventGateway	started	1	Fri Jan 12 14:51:53	PST 2001	20892	2363	0
LayoutServer	started	1	Fri Jan 12 14:52:04 I	PST 2001	20925	2297	0
ReportServerFac	tory started	1	Fri Jan 12 14:52:56 I	PST 2001	21103	2365	0
ResourceMgr	started	1	Fri Jan 12 14:52:26 I	PST 2001	20975	2190	0
TGServer	started	1	Fri Jan 12 14:52:36 I	PST 2001	20991	2238	0
TaskServer	started	1	Fri Jan 12 14:52:26 I	PST 2001	20976	2291	0
TemplateServer	started	1	Fri Jan 12 14:51:53	PST 2001	20897	2211	0
VerifyReportSer	ver started	1	Fri Jan 12 14:52:57 I	PST 2001	21104	2294	0
VpnInvServer	started	1	Fri Jan 12 14:52:26 I	PST 2001	20984	2326	0
aggregator	started	1	Fri Jan 12 14:52:25	PST 2001	20959	2203	0
httpd	started	1	Fri Jan 12 14:52:24 I	PST 2001	20954	2190	0
journal	started	1	Fri Jan 12 14:52:25 I	PST 2001	20956	2302	0
lock_manager	started	1	Fri Jan 12 14:51:54 I	PST 2001	20900	2332	0
log	started	1	Fri Jan 12 14:52:25 I	PST 2001	20971	2291	0
poller	started	1	Fri Jan 12 14:51:54 I	PST 2001	20910	2339	0
rmiregistry	started	1	Fri Ian 12 14:51:54 I	PST 2001	20907	2370	0
scheduler	started	1	Fri Jan 12 14:52:25 J	PST 2001	20967	2210	0
trancatcher	started	1	Fri Ian 12 14:51:53 I	PST 2001	20895	2354	0
watchdog nerf	started	1	Fri Jan 12 14:51:53 J	PST 2001	20891	2229	0
aggregator tr	ancatcher TCServer	lock manager Ver	rifvRenortServer rn	niregistry	noller		
iournal	DataSetServer	LavoutServer	scheduler	Resour	ceMar	httnd Ta	skServer log
journal WatchDog	DataSetServer EventGateway	LayoutServer TemplateServer	scheduler watchdog n	Resour erf	ceMgr RenortServe	httpd Ta: erFactory	skServer log VnninvServer
journal WatchDog 2001/01/12 14	DataSetServer EventGateway	LayoutServer TemplateServer	scheduler watchdog_p	Resour erf	ceMgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway :51:50.353 PST Create :51:50.392 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server	scheduler watchdog_p	Resour Resour	ceMgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway :51:50.353 PST Create :51:50.392 PST Create :51:50.434 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d TGServer server	scheduler watchdog_p	Resour erf	ceMgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway :51:50.353 PST Create :51:50.392 PST Create :51:50.434 PST Create :51:50.477 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d TGServer server d httpd server	scheduler watchdog_p	Resour erf	ceMgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 PST Create 51:50.434 PST Create 51:50.434 PST Create 51:50.477 PST Create 51:50.517 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d TGServer server d httpd server d DataSetServer serv	scheduler watchdog_po r	Resour erf	ceMgr ReportServe	httpd Ta: erFactory	skServer log VpnlnvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 PST Create 51:50.392 PST Create 51:50.434 PST Create 51:50.477 PST Create 51:50.571 PST Create 51:50.558 PST Create	LayoutServer TemplateServer d lock_manager server d schedhuler server d fGServer server d httpd server d DataSetServer serv d log server	scheduler watchdog_p r	Resour erf	ceMgr ReportServe	httpd Ta: erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataStEServer EventGateway 51:50.353 PST Create 51:50.354 PST Create 51:50.434 PST Create 51:50.477 PST Create 51:50.517 PST Create 51:50.518 PST Create 51:50.50 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d foServer server d httpd server d bataSetServer serve d log server d EventGateway serve	scheduler watchdog_pu r er	Resour erf	ceMgr ReportServe	httpd Ta rFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway :51:50:353 PST Create :51:50:37 PST Create :51:50.434 PST Create :51:50.47 PST Create :51:50.517 PST Create :51:50.558 PST Create :51:50.650 PST Create :51:50.650 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d ToServer server d btpd server d baaSetServer serv d log server d BventGateway serve d trapcatcher server d ministrice server	scheduler watchdog_pr r er	Resour erf	ce Mgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 PST Create 51:50.352 PST Create 51:50.434 PST Create 51:50.477 PST Create 51:50.558 PST Create 51:50.558 PST Create 51:50.600 PST Create 51:50.692 PST Create 51:50.693 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d ftGserver server d bata5etServer server d bata5etServer server d EventGateway server d trapcatcher server d minegistry server d provtServer*extor	scheduler watchdog_pr er	Resour erf	ce Mgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 PST Create 51:50.392 PST Create 51:50.434 PST Create 51:50.471 PST Create 51:50.517 PST Create 51:50.508 PST Create 51:50.600 PST Create 51:50.650 PST Create 51:50.733 PST Create 51:50.59 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d bassetServer serve d log server d kvantGateway serve d trapcatcher server d miregistry server d ReportServerFactor d poller server	scheduler watchdog_pr er y server	Resour erf	ceMgr ReportServe	httpd Ta erFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 551:50.329 251:50.341 251:50.434 PST Create 51:50.517 51:50.517 251:50.517 251:50.617 251:50.617 251:50.617 251:50.610 251:50.650 251:50.650 251:50.650 251:50.650 251:50.692 251:50.805	LayoutServer TemplateServer d lock_manager server d scheduler server d bttpd server d basSetServer server d brentGateway server d twentGateway server d trapatcher server d miregistry server d ReportServerFactor d poller server d LayoutServer server	scheduler watchdog_pr r r r y server r	Resour erf	ReportServe	httpd Ta PrFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14 2001/01/12 14	DataSterier EventGateway 51:50.353 PST Create 51:50.352 PST Create 51:50.434 PST Create 51:50.437 PST Create 51:50.558 PST Create 51:50.558 PST Create 51:50.509 PST Create 51:50.692 PST Create 51:50.895 PST Create 51:50.895 PST Create 51:50.886 PST Create	LayoutServer TemplateServer d lock_manager served d scheduler server d tScheduler server d totasfetServer server d batafetServer server d EventGateway server d trapcatcher server d ReportServerServer d ReportServer serve d Jopoller server d JopotServer serve	scheduler watchdog_pr r r y server r r	Resour erf	ReportServe	httpd Ta rFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 51:50.352 51:50.434 PST Create 51:50.477 51:50.477 PST Create 51:50.474 PST Create 51:50.600 FST Create 51:50.600 FST Create 51:50.600 FST Create 51:50.600 51:50.600 PST Create 51:50.600 FST Create 51:50.942 51:50.942 PST Create 51:50.944 51:50.945 51:50.947 PST Create 51:50.944 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d batpd server d batpd server d kventGateway serve d trapcatcher server d miregistry server d RopotServerFactor d poller server d layoutServer serve d VentJGerver serve d VentJGerver serve d VentJGerver serve	scheduler watchdog_pr r er y server r r s server	Resour erf	ReportServe	httpd Ta:	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventCateway 51:50.353 51:50.352 51:50.379 FST Create 51:50.434 FST Create 51:50.517 51:50.517 FST Create 51:50.637 51:50.637 FST Create 51:50.650 51:50.650 FST Create 51:50.650 51:50.650 FST Create 51:50.805 51:50.805 51:50.805 51:50.942 51:50.942 51:50.974 51:50.975 51:50.975 51:50.975 51:50.975 51:50.975 51:50.975 51:50.975	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d bassetServer server d brentGateway server d transpatcher server d märegistry server d märegistry server d märegistry server d poller server d layoutServer server d VenInvServer server d TaskServer server	scheduler watchdog_pr r er y server r r s server	Resour erf	ReportServe	httpd Ta rFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 51:50.352 51:50.372 51:50.375 51:50.375 51:50.375 51:50.377 51:50.378 51:50.378 51:50.538 51:50.538 51:50.537 51:50.538 51:50.373 51:50.373 51:50.373 51:50.373 51:50.380 51:50.974 51:50.974 51:50.974 51:50.974 51:50.974 51:50.974 51:51.50.974 51:51.50.77 51:51.50.77 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:51.50.78 51:	LayoutServer TemplateServer d lock manager served d scheduler server d to Scheduler server d to BataSetServer server d bataSetServer server d EventGateway server d EventGateway server d EventGateway server d ReportServerFactor d poller server d LayoutServer served d YonINServer server d MonINServer server d Server Server d Server Server d aggregator server	scheduler watchdog_pr r er y server r r r s server	Resour	ReportServe	httpd Ta rFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 51:50.352 51:50.372 FST Create 51:50.434 FST Create 51:50.477 51:50.477 FST Create 51:50.600 FST Create 51:50.600 51:50.600 FST Create 51:50.600 51:50.600 FST Create 51:50.601 51:50.602 FST Create 51:50.603 51:50.604 FST Create 51:50.932 51:50.942 FST Create 51:50.914 51:50.914 FST Create 51:50.914 51:50.914 FST Create 51:51.915 51:50.914 FST Create 51:51.907 51:51.909 51:51:50.914 FST Create 51:51.907 51:51:50.914 FST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d httpd server d bated server d bated server d bated server d kventGateway serve d trapcatcher server d miregistry server d RopotServerSactor d poller server d LayoutServer server d VerifyRopotServer d Server d Server server d aggregator server d watchdog perf serv	scheduler watchdog_pr r er y server r r : server	Resour	ReportServe	httpd Ta:	skServer log VpnInvServer
journal watchDog 2001/01/12 14 2001/01/12 14 20	DataSetServer EventGateway 51:50.353 PST Create 51:50.353 PST Create 51:50.434 PST Create 51:50.434 PST Create 51:50.517 PST Create 51:50.508 PST Create 51:50.650 PST Create 51:50.650 PST Create 51:50.805 PST Create 51:50.805 PST Create 51:50.942 PST Create 51:50.942 PST Create 51:50.974 PST Create 51:51.50.975 PST Create 51:51.50.975 PST Create 51:51.50.975 PST Create 51:51.50.975 PST Create 51:51.50.975 PST Create 51:51.50.975 PST Create 51:51.23 PST Create 51:51.	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d basefver server d brendfacteway server d trapatcher server d miregistry server d miregistry server d miregistry server d poller server d layoutServer server d VenifyReportServer d TaskServer server d watchdog perf server d yatoutserver server	scheduler watchdog_pr r er y server r r server er	Resources of the second s	ReportServe	httpd Ta:	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14 20	DataSteller DataSteller EventGateway 51:50.353 PST Create 51:50.353 PST Create 51:50.434 PST Create 51:50.477 PST Create 51:50.558 PST Create 51:50.558 PST Create 51:50.692 PST Create 51:50.893 PST Create 51:50.895 PST Create 51:50.896 PST Create 51:50.974 PST Create 51:50.974 PST Create 51:51.974 PST Create 51:51.974 PST Create 51:51.974 PST Create 51:51.92 PST Create 51:51.92 PST Create	LayoutServer TemplateServer d lock manager server d scheduler server d totsheduler server d bataSetServer server d bataSetServer server d trapcatcher server d mingdistry server d ReportServerServer d univeServer serve d VonInvServer server d MonInvServer server d TaskServer server d TaskServer server d aggregator server d soutschog perf serv d sjournal server	scheduler watchdog_pr r er y server r r r s server	Resour	ReportServe	httpd Ta rFactory	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.353 51:50.352 51:50.372 FST Create 51:50.434 FST Create 51:50.47 51:50.47 FST Create 51:50.47 51:50.47 FST Create 51:50.60 51:50.617 FST Create 51:50.620 51:50.631 FST Create 51:50.642 51:50.650 FST Create 51:50.661 51:50.672 FST Create 51:50.932 51:50.942 FST Create 51:51.957 51:51.957 FST Create 51:51.97 51:51.97 FST Create 51:51.123 FST Create 51:51.124 FST Create 51:51.206 51:51.120 FST Create 51:51.206 51:51.206 <t< td=""><td>LayoutServer TemplateServer d lock_manager server d scheduler server d thtpd server d battpd server d battpd server d battpd server d kventGateway serve d targoatcher server d miregistry server d RopotServer server d RopotServer server d VerlifkpotServer server d aggregator server d sachdog perf serv d journal server d Resourcägr server d TemplateServer server d TemplateServer server</td><td>scheduler watchdog_pr r er y server r r server er er ver</td><td>Resour</td><td>ReportServe</td><td>httpd Ta:</td><td>skServer log VpnInvServer</td></t<>	LayoutServer TemplateServer d lock_manager server d scheduler server d thtpd server d battpd server d battpd server d battpd server d kventGateway serve d targoatcher server d miregistry server d RopotServer server d RopotServer server d VerlifkpotServer server d aggregator server d sachdog perf serv d journal server d Resourcägr server d TemplateServer server d TemplateServer server	scheduler watchdog_pr r er y server r r server er er ver	Resour	ReportServe	httpd Ta:	skServer log VpnInvServer
journal watchDog 2001/01/12 14 2001/01/12 14 20	DataSetServer EventGateway 51:50.353 51:50.352 51:50.352 51:50.372 51:50.374 PST Create 51:50.371 51:50.371 PST Create 51:50.371 51:50.373 PST Create 51:50.373 PST Create 51:50.305 51:50.305 PST Create 51:50.305 51:50.305 PST Create 51:50.305 51:50.305 PST Create 51:50.305 51:50.305 PST Create 51:50.307 51:50.307 PST Create 51:50.307 51:50.307 PST Create 51:51.007 51:51.007 PST Create 51:51.208 51:51.212 PST Create 51:51.214 FST Create 51:51.228 51:51.228	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d baseserver d baseserver d brentGateway server d trapatcher server d märegistry server d märegistry server d märegistry server d layoutServer server d layoutServer server d verifyReportServer d verifyReportServer d verifyReportServer d satskerver server d satchdor perf server d manal server manan reveitor	scheduler watchdog_pr r er y server r r server er ver er	Resour	Deta Set Serve	httpd Ta: rFactory	skServer log VpnInvServer
Journal journal WatchDog 2001/01/12	DataSetServer EventGateway 51:50.35 P6T Create 51:50.37 P6T Create 51:50.434 P6T Create 51:50.47 P6T Create 51:50.47 P6T Create 51:50.47 P6T Create 51:50.517 P6T Create 51:50.600 P5T Create 51:50.650 P5T Create 51:50.650 P5T Create 51:50.650 P5T Create 51:50.650 P5T Create 51:50.902 P5T Create 51:50.912 P5T Create 51:50.9142 P5T Create 51:50.9142 P5T Create 51:51.915 P5T Create 51:51.916 P5T Create 51:51.144 P5T Create 51:51.2105 P5T Create	LayoutServer TemplateServer d lock_manager server d scheduler server d scheduler server d bitpd server d bag server d EventGateway serve d trapatcher server d miregistry server d miregistry server d poller server d aportServer server d upotServer server d upotServer server d VerifyRpotServer d aggregator server d aggregator server d sagter server d server server d server server d server server d server server d rasident server d server server d manal server d TamplateServer ser mg server groups group repository_us	scheduler watchdog_pr r r r r server r server er ver ers contains: sched	Resour	a DataSetSer	httpd Ta: rFactory /	skServer log VpnInvServer
journal WatchDog 2001/01/12 14 2001/01/12 14	DataSetServer EventGateway 51:50.352 51:50.352 51:50.372 51:50.434 PST Create 51:50.47 51:50.47 PST Create 51:50.47 51:50.47 PST Create 51:50.517 51:50.60 PST Create 51:50.60 51:50.60 PST Create 51:50.912 51:50.60 PST Create 51:50.91 51:50.92 PST Create 51:51.93 51:50.94 PST Create 51:51.95 51:51.97 PST Create 51:51.12 PST Create 51:51.21 PST Create 51:51.22 51:51.22 PST Create 51:51.22 51:51.22 PST Create 51:51.22 51:51.22 PST Create	LayoutServer TemplateServer d lock_manager server d scheduler server d thtpd server d battpd server d battpd server d battpd server d battpd server d battpd server server d krentGateway server d tarpoatCher server d miregistry server d amiregistry server d poller server server d poller server server d verifyRpontServer d verifyRpontServer d verifyRpontServer d askServer server d askServer server d askServer server d askServer server d askServer server d manual server server groups group repository_us er aggregator me all servers	scheduler watchdog_pr r er r y server r r server er ver ers contains: sched	Resour	DataSetSer	httpd Ta: rFactory ver log Report:	skServer log VpnInvServer
Journal yatch Dog 2001/01/12 2001/01/12	DataSetServer EventGateway 51:50.353 51:50.352 51:50.352 51:50.372 51:50.374 PST Create 51:50.371 51:50.371 PST Create 51:50.371 51:50.371 PST Create 51:50.600 51:50.650 PST Create 51:50.373 PST Create 51:50.050 51:50.050 PST Create 51:50.050 51:50.050 PST Create 51:50.071 51:50.097 PST Create 51:50.097 51:50.097 PST Create 51:51.007 51:51.007 PST Create 51:51.007 51:51.208 PST Create 51:51.213 PST Create 51:51.228 FST Create 51:51.228 FST Create 51:51.2328 <td>LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d bassetServer server d brandSetServer server d maregistry server d maregistry server d maregistry server d maregistry server d maregistry server d layoutServer server d layoutServer server d layoutServer server d laskServer server d marefyReportServer d marefyReportServer d marefyReportServer group repository_us er aggregator mg all servers</td> <td>scheduler watchdog_pr r er y server r r r r r er er ver er er ver ers contains: sched</td> <td>Resour</td> <td>n DataSetSer</td> <td>httpd Ta rFactory ver log Report:</td> <td>skServer log VpnInvServer</td>	LayoutServer TemplateServer d lock_manager server d scheduler server d bitpd server d bassetServer server d brandSetServer server d maregistry server d maregistry server d maregistry server d maregistry server d maregistry server d layoutServer server d layoutServer server d layoutServer server d laskServer server d marefyReportServer d marefyReportServer d marefyReportServer group repository_us er aggregator mg all servers	scheduler watchdog_pr r er y server r r r r r er er ver er er ver ers contains: sched	Resour	n DataSetSer	httpd Ta rFactory ver log Report:	skServer log VpnInvServer

Figure 2-1 VPN Solutions Center – Watch Dog

Name Column

The **Name** column provides the name of each of the servers. Table 2-3 provides a list of the servers and a description of the function that each server provides.



To sort alphabetically, click the column header Name. Uppercase sorts prior to lowercase.

Table 2-3 Servers and Their Function	ıs
--------------------------------------	----

Server	Function
DataSetServer	Provides a CORBA front end for SA Agent and Accounting APIs.
EventGateway	Gateways events from the TIBCO domain to the CORBA domain.
LayoutServer	Provides topology layout recomputation services for web topology, which is used when selecting certain topology views.

Server	Function
ReportServerFactory	Launches and manages ReportServer processes that generate and provide access to dynamic web reports.
ResourceMgr	Handles device locking so a router's configuration is not modified by multiple service requests at the same time, and allocates Telnet Gateway Servers in the system for download/upload requests.
TGServer	Provides a CORBA API to download configlets, upload configuration files, and send IOS commands to the router.
TaskServer	Provides a CORBA front end to the MPLS VPN Solution task repository.
TemplateServer	Provides a CORBA front end to the Template Provisioning System.
VerifyReportServer	Back end that generates Verify Reports.
VpnInvServer	Provisioning API CORBA server.
aggregator	Aggregates collected datasets periodically.
httpd	Web server.
journal	Listens to all repository events and saves them into journal files. Also archives the journal files periodically.
lock_manager	Handles locking for the internal database.
log	Makes the output of tasks available to you in a browsable format.
poller	Gets requests from other data collectors and forwards the requests to the device. Gets the response and sends it to appropriate collectors.
rmiregistry	Underlying communication process necessary for ReportServerFactory and LayoutServer to communicate with web clients.
scheduler	Enables you to schedule tasks immediately or later in time, for one-time or repeated execution.
trapcatcher	Catches configuration change traps from routers.
watchdog_perf	Tracks performance for the system itself. The data is collected and stored in the internal database only. The data is useful for diagnostics.

 Table 2-3
 Servers and Their Functions (continued)

State Column

The **State** column provides the current state. Table 2-4 provides a description of each of the states in normal progression order.

State	Description
start_depends	This server has been asked to start, but is waiting for servers it depends on to start. Once all dependent servers have started, this server will transition to the state of starting.
starting	This server is currently starting. Once a successful heartbeat occurs, this server will transition to the state of started.
started	This server is currently started and running.
stop_depends	This server is supposed to be stopped, but it is waiting for servers it depends on to be stopped first.
stopping_gently	This server is in the process of stopping in a gentle fashion. That is, it was notified that it is to stop
stopping_hard	This server is in the process of being killed because either it did not have a way to stop gently or because the gentle stop took too long.
stopped	This server is stopped. The WatchDog will either start it again or disable it if it has been frequently dying.
disabled_dependent	This server is disabled because one or more servers it depends on are disabled. If all servers it depends on are started, this server will automatically start.
disabled	This server is disabled and must be manually restarted.
restart_delay	This server is delaying before restarting. There is a short delay after a server stops and before it is restarted again.

Table 2-4 Valid States

Generation Column

The **Generation** column provides the generation of the server. Each time the server is started, the generation is incremented by 1.

Exec Time Column

The Exec Time column provides the date and time that the server was last started.



To sort from the earliest to the latest date and time, click the column header Exec Time.

Pid Column

The Pid column provides the UNIX process identifier for each server.

Success Column

The **Success** column provides the number of successful heartbeats since the server was last started. Heartbeats are used to verify that servers are functioning correctly.



To sort from the least number of successful heartbeats to the greatest number of successful heartbeats, click the column header **Success**.

Missed Column

The Missed column provides the number of missed heartbeats since the server was last started.

A few missed heartbeats could indicate that the system was busy. However, more than a couple of missed heartbeats per day could indicate a problem. See the logs to diagnose the reason. If a server misses three heartbeats in a row, the server is automatically restarted.

Note

Three missed heartbeats in a row is the default for restarting the server. The default number can be reset in the **csm.properties** file. After three failed attempts to restart in a row, the server is disabled.

wdperf Command

This section provides the description, syntax, and report information for the **wdperf** command. This section also describes the reports that are generated by executing this command and the common information in these reports:

- Average, Minimum, and Maximum % CPU Utilization per Time Period, page 2-17
- Average, Minimum, and Maximum % Memory Usage per Time Period, page 2-18
- Average, Minimum, and Maximum Virtual Memory Usage per Time Period, page 2-18

This graphical interface to the WatchDog provides information about system performance and resource utilization.



The default for netsys.watchdog.server.watchdog_perf.enable in the csm.properties file is **false**, which disables data gathering for the **wdperf**. To enable this function, set netsys.watchdog.server.watchdog_perf.enable to **true** and relaunch the WatchDog.

Description

The **wdperf** command is a monitoring tool for MPLS VPN Solution that provides reports indicating the % CPU utilization, the % Memory usage, and the amount of virtual memory used by each of the system's servers and user-defined tasks. The reported values are based on performance data gathered by the WatchDog.

```
wdperf [%cpu | %mem | vmem] [&]
```

or

```
wdperf {%cpu | %mem | vmem} [<date> | start] [&]
```

where:

% **cpu** is a parameter that causes the Average % CPU Utilization per Hour report to be displayed. This is the default option.

% mem is a parameter that causes the Average % Memory Utilization per Hour report to be displayed.

vmem is a parameter that causes the Average Virtual Memory Utilization per Hour report to be displayed.

<date> is an optional parameter that specifies the date for which performance data will be displayed. The default date is the current date. The format of the date is either: *mmlddlyy* or *mmlddlyyyy*, where:

- *mm* is the month, specified as **01** to **12**.
- *dd* is the day, specified as **01** to **31**.
- yy or yyyy is the year, specified in two-character or four-character year designations.

start is an optional parameter that causes the earliest available performance data to be displayed (that is, the repository creation date).

& is an optional parameter that causes wdperf to be run as a background process.

Note

The location of **wdperf** is: *<MPLS VPN Directory>/bin*.

Report Information

For a description of the reports created by the **wdperf** command, first see explanations of the generic report fields in the "Status Row" and "Filter Information" sections in Chapter 14, "Reports Overview." Additionally, each report has the following information:

- Results Area
- Detail Area
- Bottom Task Bar

Results Area

The columns of information are as follows:

- Process. This column lists the names of all the servers and task processes managed by WatchDog.
- The data displayed in each of the other columns depends on the current display level. The **Daily** display level displays data for each hour of the selected day, **00** to **23**. These columns start at midnight (**00:00**) and go to 11:00 p.m (**23:00**). The **Hours** display level displays data for each minute of the selected hour, **00** to **59**. To switch between the display levels, see the "Bottom Task Bar" section on page 2-16. The color of each cell depends on the value contained in the cell. A blue cell indicates the server was restarted in the designated time period. Note that all blue cells have a minus sign preceding the cell value.

Detail Area

The information in this area is:

```
pid = <####>
```

where: <####> is the Process identifier of the server or task (process) that you highlight in the Results Area.

start time = localized date, time, and time zone when the server or task (process) that you highlight in the Results Area started.

Note

If the highlighted server or task restarts, multiple lines will be displayed in the Detail Area, one line for each time the server or task starts.

Bottom Task Bar

From left to right, the bottom task bar includes the following items:

- <= is a button that you click to display the previous day's data when at the **Daily** display level and the previous hour's data when at the **Hours** level.
- => is a button that you click to display the next day's data when at the **Daily** display level and the next hour's data when at the **Hours** level.



If you want to view a report for a specific date, you may want to re-enter the **wdperf** command with the desired date. This may be preferable to using the <= and => buttons, which only display adjacent days one day at a time.

- Metric is a drop-down list with the following choices:
 - % cpu displays the percentage of the CPU that is being occupied by each of the WatchDog's processes. Values below 20% are displayed in green, those between 20% and 50% are displayed in yellow, and those above 50% are displayed in red.
 - % mem displays the percentage of the machine's physical memory that is being used by each of the WatchDog's processes. Values below 20% are displayed in green, those between 20% and 50% are displayed in yellow, and those above 50% are displayed in red.
 - vmem displays the amount of virtual memory (in kilobytes) allocated to each of the WatchDog's processes. Values are displayed in various color shades to highlight memory usage trends.
- Aggregate is a drop-down list with the following choices:
 - Average displays the average value for the selected metric during the applicable time period (for example, one hour or one minute, depending on the current display level).
 - **Maximum** displays the maximum value for the selected metric during the applicable time period (for example, one hour or one minute, depending on the current display level).
 - **Minimum** displays the minimum value for the selected metric during the applicable time period (for example, one hour or one minute, depending on the current display level).
- Hours is a drop-down list that displays a selection of hours from which to choose (00 to 23). When you select an hour from the list, you switch to the Hours display level for the selected hour, which displays the data aggregated per minute. From the Hours level, you can return to the Daily level by clicking on the Daily button in the bottom task bar.

L

Average, Minimum, and Maximum % CPU Utilization per Time Period

These reports display the percentage of the CPU that is being occupied by each of the WatchDog's processes. Values less than 20% are displayed in green, those between 20% and 50% are displayed in yellow, and those greater than 50% are displayed in red.

The Average % CPU Utilization per Hour report for the current date is the default report if you do not specify another **Metric** on the command line, as specified in the "Syntax" section, and maintain the default **Aggregate** selection on the bottom task bar.

See a sample of the % CPU Utilization report, as shown in Figure 2-2, "% CPU Utilization Report.

Some processes in MPLS VPN Solution launch children processes as an **Unknown Process**. The command line command that launches the **Unknown Process** can be selected in the top window, and its related arguments (args) are listed in the **Detail** pane.

From this report, you can use the controls in the bottom task bar to navigate to reports displaying other metrics, aggregates, and display periods.

— 🔰 🚽 Average % CPL	J Uti	lizat	ion	per	Minu	ute f	or W	/ed I	Feb (07, 2	2001	10	:00		•
Status: Ready	Re	fresh	I [Nev	v Vie	w	Pri	nt	Ba	ack	No) Con	npari	son I	Perfor
Results															
Process	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14
SERVERS:															
DataSetServer	0.0	0.0	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
EventGateway	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
LayoutServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
ResourceMgr	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TGServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TaskServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TemplateServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
VerifyReportServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
VpnInvServer	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
aggregator	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
httpd	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
journal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
log	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
poller	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
rmiregistry	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
scheduler	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1
trapcatcher	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
watchdog	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
watchdog_perf	0.1	0.0	0.1	0.0	0.0	0.0	0.0	0.1	0.0	0.1	0.1	0.1	0.0	0.0	0.0
UNKNOWN PROCESSES:															
Unknown Process (230	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
• 3000000000000000000000000000000000000		32													•
Filter:								22	/22 0	ispla	ayed	Ac	lvanc	ed Fi	lter
Detail															
															_
				-											_
<= => Metric •	· Ag	jgreg	ate 🔻	· I	Daily										

Figure 2-2 % CPU Utilization Report

Average, Minimum, and Maximum % Memory Usage per Time Period

These reports display the percentage of the machine's physical memory that is being used by each of the WatchDog's processes. Values less than 20% are displayed in green, those between 20% and 50% are displayed in yellow, and those greater than 50% are displayed in red.

The Average % Memory Utilization per Hour report for the current date is the report that is displayed if you specify **% mem** on the command line and maintain the other defaults on the command line, as specified in the "Syntax" section, and the default **Aggregate** selection on the bottom task bar.

See a sample of the % Memory Utilization report, as shown in Figure 2-3, "% Memory Utilization Report."

From this report, you can use the controls in the bottom task bar to navigate to reports displaying other metrics, aggregates, and display periods.

– 🔰 🚽 Average % Me	moŋ	γUs	age	per	Minu	ute f	or W	/ed I	Feb (97,2	2001	10	:00		· 🗌
tatus: Ready	Re	fresh	1	Nev	v Vie	w	Pri	int	B	ack	N	o Cor	npari	son I	Perfor.
Results															
Process	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14
SERVERS:															
DataSetServer	4.0	4.0	4.0	4.0	4.0	4.1	4.1	4.1	4.1	4.1	4.1	4.1	4.1	4.1	4.1
EventGateway	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2
LayoutServer	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2	1.2
ResourceMgr	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
TGServer	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
TaskServer	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
TemplateServer	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9	0.9
VerifyReportServer	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2
VpnInvServer	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1
aggregator	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3	2.3
httpd	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
journal	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2	2.2
log	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8	2.8
poller	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1	1.1
rmiregistry	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
scheduler	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1
trapcatcher	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8	1.8
watchdog	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
watchdog_perf	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
UNKNOWN PROCESSES:															
Unknown Process (230	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
		996 1997													•
Filter:								22	/22 [ispl	ayed	Ac	lvand	ed Fi	lter
Detail															
				-		_									
<= => Metric •	r Ag	greg	ate 🔻	· _	Daily										

Figure 2-3 % Memory Utilization Report

Average, Minimum, and Maximum Virtual Memory Usage per Time Period

These reports display the amount of virtual memory (in kilobytes) allocated to each of the WatchDog's processes. Values are displayed in various color shades to highlight memory usage trends.

The Average Virtual Memory Utilization per Hour report for the current date is the report that is displayed if you specify **vmem** on the command line and maintain the other defaults on the command line, as specified in the "Syntax" section, and the default **Aggregate** selection on the bottom task bar.

See a sample of the Virtual Memory Utilization report, as shown in Figure 2-4, "Virtual Memory Utilization Report."

From this report, you can use the controls in the bottom task bar to navigate to reports displaying other metrics, aggregates, and display periods.

— Average Virtual M	/lemory	Usage	per Mi	nute fo	r Wed	Feb 07,	2001	10:00	· 🗆			
Status: Ready	Refres	h P	New View	/ Pr	int	Back	No Con	nparison	Perfor			
Results												
Process	00	01	02	03	04	05	06	07	08			
SERVERS:												
DataSetServer	54488	54488	54488	54488	54488	54488	54488	54488	54488			
EventGateway	13456	13456	13456	13456	13456	13456	13456	13456	13456			
LayoutServer	15152	15152	15152	15152	15152	15152	15152	15152	15152			
ResourceMgr	9328	9328	9328	9328	9328	9328	9328	9328	9328			
TGServer	8024	8024	8024	8024	8024	8024	8024	8024	8024			
TaskServer	19952	19952	19952	19952	19952	19952	19952	19952	19952			
TemplateServer	11296	11296	11296	11296	11296	11296	11296	11296	11296			
VerifyReportServer	36984	36984	36984	36984	36984	36984	36984	36984	36984			
VpnInvServer	28976	28976	28976	28976	28976	28976	28976	28976	28976			
aggregator	36736	36736	36736	36736	36736	36736	36736	36736	36736			
httpd	1648	1648	1648	1648	1648	1648	1648	1648	1648			
journal	35760	35760	35760	35760	35760	35760	35760	35760	35760			
log	42208	42208	42216	42216	42224	42224	42232	42232	42240			
poller	12280	12280	12280	12280	12280	12280	12280	12280	12280			
rmiregistry	10552	10552	10552	10552	10552	10552	10552	10552	10552			
scheduler	45064	45064	45064	45064	45064	45064	45064	45064	45064			
trapcatcher	35000	35000	35000	35000	35000	35000	35000	35000	35000			
watchdog	10357	10357	10357	10357	10357	10357	10357	10357	10357			
watchdog_perf	5184	5184	5184	5184	5184	5184	5184	5184	5184			
UNKNOWN PROCESSES:												
Unknown Process (230	11016	11016	11016	11016	11016	11016	11016	11016	11016			
Filter:					22/22	Display	/ed Ad	vanced F	ilter			
Detail												
<= => Metric •	Aggre	gate 🔻	Daily									

Figure 2-4 Virtual Memory Utilization Report