

watchdogPing: Watchdog monitoring using pings to IP targets

Issue: 1.3

Date: 12 November 2014

1	Introduction	3
2	Configuring the watchdogPing script.....	4
2.1	Script overview.....	4
2.2	Recovery procedure	4
2.3	Script parameters	4
2.4	Configuring the script	5
2.4.1	Pasting the script into the script editor	5
2.4.2	Scheduling the script to run on boot	6
3	Debugging commands	8
4	Script events	9
5	watchdogPing script	10
6	Script history	16

1 Introduction

The watchdogPing script is designed to alleviate issues which can result in an interface staying down and not recovering: stale L2TP sessions, interoperability issues between vendor ADSL chipsets, undetectable 3G network issues, Ethernet DHCP issues.

The script periodically pings IP targets and takes the following recovery steps on consecutive ping failure.

- Resets the physical interface
- Optionally reboots the router

The watchdogPing script is commonly used in a scenario shown in the figure below. The WAN connection in the example is via a GSM link using PPP.

Note: this script is designed for a single WAN interface. It will only reset one physical interface and you can only run it once. It can be used in a multi-WAN scenario, such as ADSL with 3G backup, but if you do this the script must be stopped and started dynamically on routing switchover, or ping targets must be routable via all interfaces. Depending on the multi-WAN configuration, you may also have to configure static routes and block filters to stop the pings from being routed out to an incorrect interface.

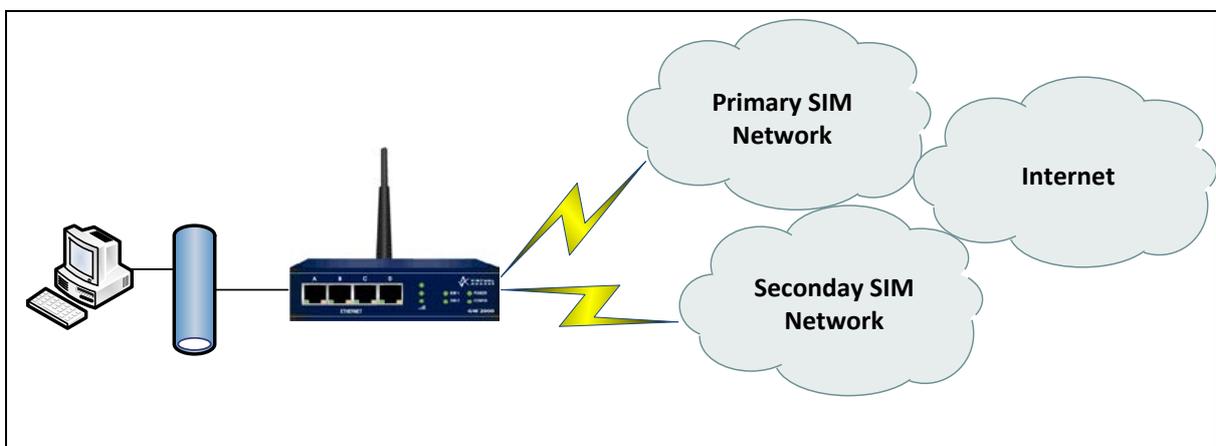


Figure 1: Possible network architecture

2 Configuring the watchdogPing script

2.1 Script overview

You can only run the script once. It is designed to be run on boot. On boot, the script does the following:

- Waits 60 seconds to allow for time for interfaces to connect
- Sends pings to target destinations at configurable durations. If only one target destination is configured then only one ping is sent to this target. Pings are sourced from the eth-0 interface to allow for routing across a VPN tunnel if required and allow for a reply wait time to be specified
- The script optionally waits for at least one ping success before running to avoid continuous reloading in the event of a valid network outage.
- A ping from either target signifies that router is operating as normal
- When a number of configurable consecutive ping failures are detected the script will start implementing its recovery procedure. Read Section 2.2.
Note: if two ping targets are configured, pings must fail to both targets consecutively.
- On pings success the recovery procedure is terminated.

2.2 Recovery procedure

1. The router resets the physical interface for adsl-x and modem-x interaces. For eth-x interfaces the router will release and renew the IP address.
2. An INFO event is generated for visibility.
3. On further continuous consecutive ping failure the router optionally reloads.
4. An INFO event is generated for visibility.

2.3 Script parameters

The script name is watchdogPing and it takes in five required parameters and a further four optional parameters:

```
watchdogPing [phy port] [ping target1] [ping target2] [ping wait] [maximum fail]  
              [ping reply wait] [reload] [ping_success_before_start]
```

These parameters are described in the example and table below.

```
watchdogPing modem-0, 1.1.1.1, 0, 30, 10, 2, 1, 1, eth-0.
```

Parameter	Type	Description
modem-0	Required	The physical interface to reset when pings fail. Available interfaces are adsl-x; modem-x, eth-x.
1.1.1.1	Required	The first ping target
2.2.2.2	Required	The second ping target. Set to 0 to signify no second ping target.
30	Required	The wait between pings in seconds. Pings are sent to both ping targets at the same time.
5	Required	The number of consecutive pings failures to cause recovery procedure
2	Optional	The time to wait for a ping reply in seconds (default: 2)
1	Optional	Whether to reload on ping failure (default: 1; 0 for no reload)
1	Optional	Whether to wait for ping success before start checks (default: 1; 0 to not wait for a ping success on boot)
eth-0	Optional	The interface or IP to source the pings from (default: eth-0; 0 to use outbound interface IP)

Table 1: watchdogPing parameter descriptions

2.4 Configuring the script

This script was introduced into firmware in versions 9.09.21 and 10.00.15. To use the script on older firmware versions first paste the script from Section 5 'watchdogPing script' into the script editor and then use the scheduler to run the script at boot up.

From the start page, click **Advanced** to open the Expert View menu.

2.4.1 Pasting the script into the script editor

If you are using 9.09.xx firmware, in the Expert View menu, click **system > scripts->script editor**. The Script Editor page appears.

If you are using 10.00.xx firmware, in the Expert View menu, click **system > management > scripts > script editor**. The Script Editor page appears.

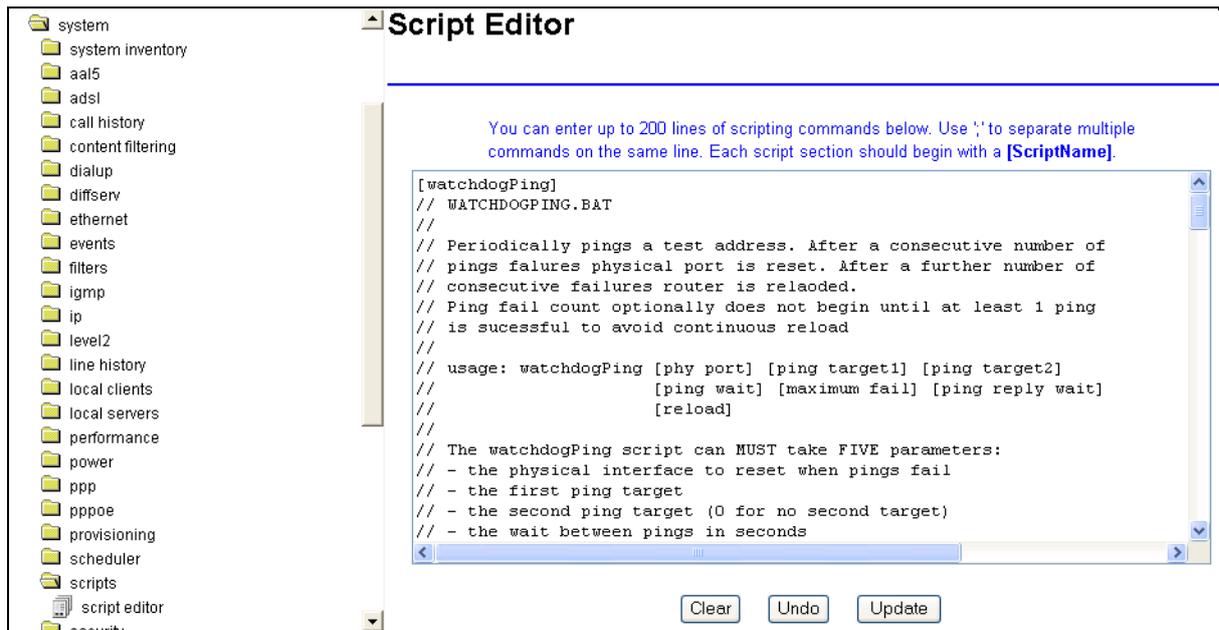


Figure 2: The script editor page in version 9.09.xx

Paste in the script from Section 5 'watchdogPing script' from this document. The first line of the script should begin with the script name in square brackets, [simSwitchPing]. This name will be used to call the script using the scheduler.

If the number of script lines needs to be reduced, you can omit any line beginning with '//' as this denotes a comment tag. Also, you can enter multiple script lines onto the same script editor line separated by ';' (semi colon).

When you have completed the script, click **Update**.

2.4.2 Scheduling the script to run on boot

If you are using 9.09.xx firmware, in the Expert View menu, click **system > scheduler > scheduler tasks**. The Scheduler Task Entry page appears.

Click **add** in the Operation column of the list. The Scheduler Task form appears.

If you are using 10.00.xx firmware, in the Expert View menu, click **system > management > scheduler > scheduler tasks**. The Scheduler Task Entry page appears.

Click **add** in the Operation column of the list. The Scheduler Task form appears.

Scheduler Task Entry 1

Enabled:

Name:

Date: (dd-mm-yyyy)

Time: (hh:mm)

Frequency:

Window: secs

Script:

Figure 3: The scheduler task entry page in version 9.09.xx

Field	Description
Enabled	Enables or disables a particular schedule. Set to Yes .
Name	The name associated with the schedule. Enter a descriptive name
Date	The date the script initiates. This field is ignored when frequency is set to start up. Leave as default .
Time	The time the script initiates. This field is ignored when frequency is set to start up. Leave as default .
Frequency	Sets the frequency the script executes. Set to startup .
Window	This parameter sets how long the system will wait if it is busy before executing the script. For example if the script is set to execute at 10:00 and the window is set to 30 seconds, the system will try executing the script within this window only. Set to 30 .
Script	The name of the script to be executed. Enter the script name, followed by the relevant parameters as shown in the above image. Separate the parameters by commas. Example: watchdogPing modem-0, 1.1.1.1, 0, 30, 10, 2, 1, 1, eth-0

Table 2: The scheduler task fields and their descriptions

3 Debugging commands

Useful debug commands via the command line are described in the table below.

Diagnostic Command	Description
Show tasks	Displays all running tasks.
Show task <tasknum>	Displays running task. Also indicates position task is currently at.
Show task vars <tasknum>	Displays variables and variable values associated with task.
Show ip route	Shows routing table.
Show stats adsl	Shows ADSL stats.
Show modem interface status modem-x	Displays GSM stats.
show modem interface gsm sim status modem-x	Displays GSM SIM status.
Show dhcp client info [eth-x waneth-x]	Displays DHCP client information
Show events	Displays event log.
Show change log	Displays recent configuration changes.
Dir scripts	Displays all scripts embedded in the firmware.
Show config script ALL	Displays all scripts in the script editor.
Show config script <scriptname>	Displays the <scriptname> script as configured in script editor. Includes line numbers.
Show config script -n <scriptname>	Displays the <scriptname> script as configured in the script editor. Does not include line numbers.

Table 3: Debug command lines and their descriptions

Useful trace commands via the command line are described in the table below.

Trace command	Description
++All 6	Traces all INFO events
++ip	Traces IP traffic
++ip:icmp	Traces ICMP IP traffic
++modem	Traces modem events
++script	Traces script events
--script	Stops script event tracing
--	Stops all event tracing
Trace on <script_name>	Traces each line in a script as it executes
Trace off <script_name>	Turns off tracing for script

Table 4: Trace command lines and their descriptions

4 Script events

Severity	Class	Subclass	Text
INFO	49	40	watchdogPing script error invalid ping targets<ping targets>
INFO	49	40	simSwitchPing target <ping targets> via <phy port> every <time between pings> secs
INFO	49	40	watchdogPing <number> ping fails after <phy port> reset - rebooting
INFO	49	40	watchdogPing <number> ping fails - resetting <phy port>
DEBUG	49	9999	watchdogPing <number> ping fails - resetting <phy port> (only generated after INFO event above until ping success)

Table 5: Script events

5 watchdogPing script

```
[watchdogPing]
// WATCHDOGPING.BAT
//
// Periodically pings a test address. After a consecutive number of
// pings failures physical port is reset. After a further number of
// consecutive failures router is reloaded.
// Ping fail count optionally does not begin until at least 1 ping
// is successful to avoid continuous reload
//
// usage: watchdogPing [phy port] [ping target1] [ping target2]
//                [ping wait] [maximum fail] [ping reply wait]
//                [reload] [ping_success_before_start]
//                [ping source]
//
// The watchdogPing script can MUST take FIVE parameters:
// - the physical interface to reset when pings fail
//   (adsl-x, modem-x, eth-x)
// - the first ping target
// - the second ping target (0 for no second target)
// - the wait between pings in seconds
// - the number of consecutive failures before recovery procedure
//
// It can optionally take FOUR parameters:
// - the time to wait for a ping reply in seconds (default: 2)
// - whether to reload on ping failure (def: 1, 0 for no reload)
// - whether to wait for ping success before start checks
//   (default: 1, 0 to not wait for a ping success on boot)
// - the interface or IP to source the pings from
//   (def: eth-0, 0 to use outbound port IP)
//
// EXAMPLES
// -----
// watchdogPing modem-0, 1.1.1.1, 0, 30, 10, 2, 1, 0, eth-1
// (Ping 1.1.1.1 every 30 seconds with a source of eth-1 interface.
```

```
// Waits 2 secs for ping reply. After 10 consecutive failures reset
// modem-0. After a further 10 consecutive ping failures reload
// the router. Do not wait at least one ping success on boot before
// staring checks

!echo off

!arg phyPort, pingTarget1, pingTarget2, pingWait, pingFails

$pingReplyWait = $6
$routerReload = $7
$waitForPingSuccess = $8
$pingSource = $9

//defaults
!if pingReplyWait = ``
    $pingReplyWait = 2
!endif
!if routerReload = ''
    $routerReload = 1
!endif
!if waitForPingSuccess = ''
    $waitForPingSuccess = 1
!endif
!if pingSource = ''
    $pingSource = eth-0
!endif

//set defaults
$i = 1
$ping_reply_wait_msec = 1000
!while $i < $pingReplyWait
    !add ping_reply_wait_msec, 1000
    !inc i
!endwhile
```

```
$phy_reset = 0
$restart_checks_event = 0
!if $waitForPingSuccess <> 0
    $boot_ping_success = 0
!else
    $boot_ping_success = 1
!endif

//checking
!if pingTarget1 = ''
    $pingTarget1 = 0
!endif
!if pingTarget2 = ''
    $pingTarget2 = 0
!endif

//logging
$logpingstr = $pingTarget1
!if $pingTarget2 <> 0
    $logpingstr = $logpingstr/$pingTarget2
!endif

!if $pingTarget1 = 0
    !if $pingTarget2 = 0
        !log watchdogPing error invalid ping targets $logpingstr
        !exit
    !else
        $pingTarget1 = $pingTarget2
        $pingTarget2 = 0
    !endif
!endif

!if $pingTarget2 = 0
    $rtIndex2 = 0
!endif
```

```
!log watchdogPing target $loggingstr via $phyPort every $pingWait secs

!pause 60

!while 1
    $fail_count = 0
    !while $fail_count < $pingFails

        $z = `set ping results reset`

        !if $pingSource <> 0
            $z = `quiet ping $pingTarget1 -s $pingSource -w
$ping_reply_wait_msec`
            !if $pingTarget2 <> 0
                $z = `quiet ping $pingTarget2 -s $pingSource -w
$ping_reply_wait_msec`
            !endif
        !else
            $z = `quiet ping $pingTarget1 -w $ping_reply_wait_msec`
            !if $pingTarget2 <> 0
                $z = `quiet ping $pingTarget2 -w $ping_reply_wait_msec`
            !endif
        !endif
        !pause $pingReplyWait

        $result = `sh ping replies`

        !if $result > 0
            $boot_ping_success = 1
            $fail_count = 0
            $phy_reset = 0
            $restart_checks_event = 0
        !else
            !if $boot_ping_success <> 0
                !inc fail_count
            !endif
        !endif
    !endif
```

```
    !pause $pingWait
!endwhile

!if $phy_reset <> 0
    !if $routerReload <> 0
        !log watchdogPing $pingFails ping fails after $phyPort reset -
rebooting
        !pause 2
        reload
    !else
        $phy_reset = 0
    !endif

!else
    //only generate reset event as INFO once until ping success so do not
fill event log
    !if $restart_checks_event <> 0
        !event script.9999 watchdogPing $pingFails ping fails - resetting
$phyPort
    !else
        !log watchdogPing $pingFails ping fails - resetting $phyPort
        $restart_checks_event = 1
    !endif

//ADSL reset
!if $phyPort = "*adsl-"
    $z = `reset adsl $phyPort `
    !pause 90
!endif

//3G modem reset
!if $phyPort = "*modem-"
    $z = `reset modem $phyPort `
    !pause 10
!endif
```

```
//DHCP release and renew
!if $phyPort = "*eth-"
    $z = `dhcp release $phyPort
    !pause 2
    $z = `dhcp renew $phyPort
    !pause 2
!endif

    $phy_reset = 1
!endif

!endwhile
```

6 Script history

Version	Changes	Firmware version changes introduced
1.2	Addition for attempted recovery of a DHCP Client Ethernet interfaces.	9.08.29 10.00.25
1.3	Addition for optional configuration for the interface or IP address to source the pings from	9.08.29 10.00.25