

# Re: Script to ping

---

*Source:*

<http://www.tech-archive.net/Archive/Windows/microsoft.public.windows.server.scripting/2006-08/msg00301.html>

---

- *From:* "Alexander Trofimov" <[a\\_trofimov@xxxxxxxx](mailto:a_trofimov@xxxxxxxx)>
  - *Date:* Thu, 10 Aug 2006 09:33:34 +0400
- 

Great solution, Jason. But unfortunately it needs in \*nix box or installing perl in win machine. So, maybe you could rewrite it in vbscript??? =)

--

Regards,  
Alexander Trofimov

"Jason Gurtz" <[jasonNOgurtz@xxxxxxxxxxxxxxxxxxxx](mailto:jasonNOgurtz@xxxxxxxxxxxxxxxxxxxx)> wrote in message  
<news:esdqfZ%23uGHA.4512@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

Glenn wrote:

I'd like to create a script that could ping my switches and servers and send me an e-mail as soon as one goes down.

I did this at my last job using perl to monitor and report on the availability of an interstate frame relay WAN link. Unfortunately for you, the script ran on a Linux box and some of the perl modules are not translatable to the Win32 world under activestate perl so I can not offer any code that would work without a bit of modification.

Nonetheless, here's psuedocode logic for what it did (I leave out many details that I used generating/formatting the report and error checking).

START main

```
store date and time #1
ping host once
store ping1 result (fail or response time in ms)
if ping1 is successful
sendEmail()
quit()
else
store ping one fail result
```

Re: Script to ping

```
sleep 60 seconds
store date and time #2
ping host once
store ping2 result (fail or response time in ms)
endif
END main
```

START sendEmail

```
case ping1 is successful
send success email contents
```

```
case ping1 failure and ping2 successful
send warning email about blip in availability
```

```
case ping2 failure
send error email about total interruption of service
```

END sendEmail

I think you could do all this in a plain batch script using the blat.exe program to send the emails and local environment variables. you can "ping -n %seconds% 127.0.0.1" as a workable sleep function. ping.exe will set %ERRORLEVEL% to zero if it is successful, one if unsuccessful, and a really small number if ping is ^C.

I found that simply pinging the remote host once generated a lot of false alarms due to momentary drops not affecting LOB, the line being saturated with data, or etc... Therefore I implemented the delayed follow up second ping idea. I ran the script via cron every half hour which is akin to using the task scheduler.

~Jason

--