Bash Shellshock Mitigation Using ASM Signatures

Nir Zigler, 2014-27-09

Update: The signature mentioned in this article have been released as part of an Attack Signature Update.
You may head to https://downloads.f5.com to download the file manually, or use the automatic update feature in ASM.

This week we've seen new vulnerabilities with massive damage potential come to light – CVE-2014-6271, CVE-2014-6277 and CVE-2014-7169 - named quite appropriately "Shellshock".

Background

You can find details regarding this bash vulnerability on the Red Hat security blog:

https://securityblog.redhat.com/2014/09/24/bash-specially-crafted-environment-variables-code-injection-attack/

In a typical exploit, the payload is sent through a header (typically Cookie, Referrer or User-Agent) and takes advantage of the way the web server saves the data in that request to environment variables.

A malicious request will attempt to fool the bash parser by sending a payload that will invoke a system command, for instance:

```
GET /home.php HTTP/1.1
Host: example.com
User-Agent: () { :;}; /bin/bash -c "ls"
```

The string "`() { :;};`" means it is a function declaration.
The string is followed by various shell commands – in our case it is execution of the "ls" command.

Mitigation using F5 ASM Attack Signatures

The following signature will catch attempts to exploit this CVE:

```
headercontent:"() {";
```

This signature is compatible with all BIG-IP versions.

To prevent any other potential exploitation attempts via the URL or a parameter, two additional signatures can be used:

```
uricontent:"() {"; objonly;
```

```
valuecontent:"() {"; norm;
```

Note: The signatures have been updated to catch exploit attempts in all their variations.

It is important to note, that all attempts to exploit this vulnerability via HTTP parameters and several known exploits via the HTTP header are already mitigated using existing "command execution" and "predictable resource location" signatures.
Exploits via the Cookie header will encounter the "Cookie not RFC-compliant" violation.
You need to make sure they are enabled and are not in staging.

To protect your application, create those user-defined signatures and associate them with the relevant security policy.
Make sure that the signatures are not in staging.

F5 Networks, Inc.
Corporate Headquarters
info@f5.com

F5 Networks
Asia-Pacific
apacinfo@f5.com

F5 Networks Ltd.
Europe/Middle-East/Africa
emeainfo@f5.com

F5 Networks
Japan K.K.
f5j-info@f5.com