Defending Computer Networks

*Lecture 15: More HTTP/NIDS*

Stuart Staniford
Adjunct Professor of Computer Science
Logistics

• Quiz2
• Guest lecture Tuesday
  – Cornell ITSO staff
  – Content part of curriculum, fair game on quizzes
• HW3.
  – Working on it...
Clinton Email Server Open to Cyber Attack

Clinton’s server, which handled her personal and State Department correspondence, appeared to allow users to connect openly over the Internet to control it remotely, according to detailed records compiled in 2012. Experts said the Microsoft remote desktop service wasn’t intended for such use without additional protective measures, and was the subject of U.S. government and industry warnings at the time over attacks from even low-skilled intruders.

Records show that Clinton additionally operated two more devices on her home network in Chappaqua, New York, that also were directly accessible from the Internet. One contained similar remote-control software that also has suffered from security vulnerabilities, known as Virtual Network Computing, and the other appeared to be configured to run websites.

"That's total amateur hour," said Marc Maiffret, who has founded two cybersecurity companies. I remote access connections directly over the Internet would be the result of someone choosing a
Assigned Reading

• Javascript Tutorial
  – [http://www.w3schools.com/js/](http://www.w3schools.com/js/)
  – Recommend you play around with at least the basics to get a feel for it
Main Goals for Today

• HTTP responses
• NIDS in Context of HTTP Server Attacks
• Start on web-client side attacks
HTTP Request Basics

• Text lines separated by \r\n
  – Servers often accept “\n” only, but protocol is “\r\n”

• Header is terminated by a blank line (\r\n\r\n)

• Initial request line
  – GET /dumprequest HTTP/1.1\r\n    • Other methods include POST, CONNECT, HEAD, DELETE, etc.
    • Focus on GET for now

• Followed by headers of form
  – Header: Value...\r\n
  – No request headers are actually required
HTTP Request

GET /dumprequest HTTP/1.1\r\nHost: djce.org.uk\r\nConnection: keep-alive\r\nAccept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_8_5) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/30.0.1599.101 Safari/537.36\r\nDNT: 1\r\nReferer: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=3&ved=0CD4QFjAC&url=http%3A%2F%2Fdjce.org.uk
%2Fdumprequest&ei=835IUpjEM5Xb4APEgIGoDA&usg=AFQjCNEeAn5wSZM%M2_p_y_oTmOKonq482sS9A&sig2=pSajtDK-YYIvE4HFDqmRfA&bvm=bv.54934254,d.dmg\r\nAccept-Language: en-US,en;q=0.8\r\n\r\nTry it at http://www.procato.com/my+headers/
A Few Popular Request Headers

• Host:
  – Used to specify domain (server might have several).

• User-Agent:
  – Gives browser specifics (allows server to customize responses to browser)

• Referer:
  – What page (etc) sent us here

• Accept-Language:
  – We speak English, or...

• Accept:
  – media formats we accept (eg text/html)
HTTP Response

HTTP/1.1 404 Not Found
Content-Type: text/html; charset=UTF-8
X-Content-Type-Options: nosniff
Date: Mon, 21 Oct 2013 19:37:20 GMT
Server: sffe
Content-Length: 946
X-XSS-Protection: 1; mode=block
Alternate-Protocol: 80:quic

<!DOCTYPE html>
...

HTTP Response Basics

• Text lines separated by \r\n
• Header is terminated by a blank line (\r\n\r\n)

• Initial response line
  – HTTP/1.1 404 Not Found\r\n    • Indicates status of request.

• Followed by headers of form
  – Header: Value...\r\n  – No response headers are actually required
    • Though hard to get much done without them...
Important Response Codes

• 200 OK
• 301 Moved Permanently
• 304 Not Modified
• 400 Bad Request
• 404 Not Found
• 500 Internal Server Error
A Few Popular Response Headers

• Content-Type:
  – Media-type of entity attached after header

• Content-Length:
  – Length of same (in bytes)

• Content-encoding:
  – ‘gzip’ means compression applied

• Date:

• Server: software being run on the server
Let’s try it

• telnet on 80 to a few popular websites
  – www.cnn.com
  – www.yahoo.com
  – www.nytimes.com
Entity Body

• Follows header
  – either request or response, but more consistently in response direction
  – Can be any media type:
    • text/html, text/plain, image/jpeg, audio/mpeg
    • http://www.iana.org/assignments/media-types
  – Three methods to delineate length:
    • Content-length
    • Transfer-encoding: chunked
    • Connection: close
Detecting Attacks on Web Servers

• Has been a major industry for 15+ years
• Exploits on the servers themselves
• Exploits on cgi scripts,
  – other server-side plugins
• SQL Injection
• Cross-site scripting
• Also HTTP command-and-control
  – Similar issues of detecting bad HTTP requests
Top Snort Rule Files

Stuarts-MacBook-Pro:rules stuarts$ du -s -k *.rules |sort -n -r |head -10
6152  deleted.rules
1216  browser-plugins.rules
792   malware-cnc.rules
688   blacklist.rules
568   server-webapp.rules
392   file-identify.rules
348   file-office.rules
344   server-other.rules
328   pua-adware.rules
316   browser-ie.rules
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"MALWARE-CNC Win.Trojan.Zbot variant in.php outbound connection"; flow:to_server,established; urilen:7; content:"/in.php"; http_uri; content:".ru|0D 0A|User-Agent|3A 20|Mozilla/4.0|0D 0A|"; fast_pattern:only; http_header; content:"|0A|Content-Length|3A 20|"; http_header; metadata:policy balanced-ips drop, policy security-ips drop, ruleset community, service http; reference:url,zeustracker.abuse.ch/monitor.php?ipaddress=195.22.26.231; classtype:trojan-activity; sid:26023; rev:3;)
Snort Example 5


89861 : D-Link Multiple Router command.php cmd Parameter Remote Command Execution

Printer | http://osvdb.org/89861 | Email This | Edit Vulnerability

<table>
<thead>
<tr>
<th>Views This Week</th>
<th>Views All Time</th>
<th>Added to OSVDB</th>
<th>Last Modified</th>
<th>Modified (since 2008)</th>
<th>Percent Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>574</td>
<td>9 months ago</td>
<td>about 1 month ago</td>
<td>14 times</td>
<td>100%</td>
</tr>
</tbody>
</table>

Timeline

Disclosure Date
2013-02-04

Time to Exploit | Time to Vendor Response
52 days | 7 days

Description

Multiple D-Link routers contain a flaw that is triggered when input passed via the ‘cmd’ parameter is not properly sanitized before being used in the command.php script. This may allow a remote attacker to execute arbitrary commands.

Classification

Location: Remote / Network Access
Attack Type: Input Manipulation
Impact: Loss of Integrity
Solution: Solution Unknown
Exploit: Exploit Public
Disclosure: Vendor Disputed, Third-party Verified
OSVDB: Web Related

Solution

OSVDB is not currently aware of a solution for this vulnerability.

Products

<table>
<thead>
<tr>
<th>Products</th>
<th>DIR-300</th>
<th>DIR-600</th>
</tr>
</thead>
<tbody>
<tr>
<td>D-Link Corporation/D-Link Systems, Inc.</td>
<td>2.12b02</td>
<td>2.13</td>
</tr>
<tr>
<td></td>
<td>2.13b01</td>
<td>2.14b01</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.13</td>
</tr>
</tbody>
</table>
HTTP Level Evasions

• HTTP is a very complex protocol
  – Many important sub-protocols/formats
    • URIs
    • Character sets
    • Media types of entities

• As a result
  – Hard to inspect
  – Very evasion prone
  – Extensive work required in IDS to deal with issues

• We will start to work on URI issues...
Obscure HTTP Methods

• “HEAD” instead of “GET”.
• RFC 2616:

9.4 HEAD

The HEAD method is identical to GET except that the server MUST NOT return a message-body in the response. The metainformation contained in the HTTP headers in response to a HEAD request SHOULD be identical to the information sent in response to a GET request. This method can be used for obtaining metainformation about the entity implied by the request without transferring the entity-body itself. This method is often used for testing hypertext links for validity, accessibility, and recent modification.
Pipelining of Requests

• If IDS doesn’t properly reassemble TCP and parse protocol:

  GET foo.html HTTP/1.1\n  \n  GET bar.html HTTP/1.1\n
• Could miss the “bar.html”
• Have seen commercial products with this issue recently...
Directory Type Evasions

• Suppose IDS looking for “/servlet/command.php” in URL

• So attackers might try:
  – /servlet//command.php
  – /servlet///command.php
  – /servlet/.command.php
  – /servlet/./command.php
  – /servlet/././command.php
  – /servlet/subdir/..command.php

• On Windows based web servers:
  – /servlet\command.php
URL Encoding

• RFC 2396 specifies URL format:

  2.4.1. Escaped Encoding

  An escaped octet is encoded as a character triplet, consisting of the
  percent character "%" followed by the two hexadecimal digits
  representing the octet code. For example, "%20" is the escaped
  encoding for the US-ASCII space character.

  escaped = "%" hex hex
  hex = digit | "A" | "B" | "C" | "D" | "E" | "F" | "a" | "b" | "c" | "d" | "e" | "f"

• And RFC 2616 says:

  The Request-URI is transmitted in the format specified in section 3.2.1. If the Request-URI is encoded using
  the "% HEX HEX" encoding [42], the origin server MUST decode the Request-URI in order to properly
  interpret the request. Servers SHOULD respond to invalid Request-URIs with an appropriate status code.

• So IDS must do the same...
Double Percent Encoding

• %25 is ‘%’ in ASCII
• %41 is ‘A’
• So if you write %2541 and decode once
  – you get %41
• Decode again
  – you get ‘A’
• Unbelievably, IIS did this...
  – IDS must follow...
Double Nibble Hex Encoding

- %%34%31
- On first decoding goes to %41
- On second decoding goes to A
- Again, Microsoft IIS supported this encoding
- Also variations like %341 and %4%31
  - Also get correctly transformed to A
- Not a current issue by default
Loose Implementations

• RFC says:
  – Method  <space> URI  <space> HTTP/  Version CRLF CRLF

• But some Apache versions allow
  – Method  <tab> URI  <tab> HTTP/  Version CRLF CRLF

• IDS must follow implementations exactly, or attacker can fool
Case Insensitivity of Windows

• /SerVLeT/ComMaNd.Phpe
• May well work fine if underlying OS is case insensitive
• IDS must match behavior of target
Web Drive-By Download Attacks

• Have been the main action since then.
• Attacker response to firewall/IPS technology.
  – Largely circumvents those defenses.
  – Took a while to develop useful defenses.
    • Still a very active arms-race.
Two Main Schemas

• Scan/Compromise legit websites
  – Eg SQL Injection attacks
  – Insert <iframe>s into site
  – Iframes include content from an exploit server

• Malverts
  – Malicious ads bought through chains of middle men
  – Redirects to malicious content (often swf (Flash))
Either Way

• Exploit server runs an exploit kit
• Exploit kit tests nature of browser/plugins
  – Java
  – PDF
  – Flash
• Picks one or more exploit objects
• Takes control of browser/plugins
• Installs malware/trojans
• Command and control for instructions
• Exploit kits often have extensive management infrastructure
• Profit!
Also, social engineering

• Trick humans into installing/running malware
• Also works pretty well
• Let’s look at a few examples
Fake AV

Microsoft Security Warning

Antivirus 360 Web Scanner detected dangerous spyware on your system!

Detected malicious programs can damage your computer and compromise your privacy. It is **strongly recommended** to remove them immediately.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Risk level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spyware.IEMonster.b</td>
<td>Spyware</td>
<td>CRITICAL</td>
</tr>
<tr>
<td>Zlob.PornAdvertiser.Xplisit</td>
<td>Spyware</td>
<td>High</td>
</tr>
<tr>
<td>Trojan.InfoStealer.Banker.s</td>
<td>Trojan</td>
<td>Medium</td>
</tr>
</tbody>
</table>
More Social Engineering

And More