HTTP Fingerprinting

Table of Contents

1. Introduction
2. HTTP Headers
   a. Server Tag
   b. Cookie Value
   c. Error pages
   d. X-Powered By
   e. Page names
   f. Banner Grabbing
3. Fingerprinting Tools
   a. httprint
   b. Nmap
   c. Amap
   d. Netcraft
   e. Passive Fingerprinting Using P0f
   f. Passive Fingerprinting using Google
4. Preventing Fingerprinting
   a. Banner String Obfuscation
   b. IIS
   c. Apache

Introduction
Fingerprinting Web Servers

Every company with a web presence opens TCP Port 80/HTTP on their firewalls to the Internet for web-based applications. Web servers can leak useful bits of information that attackers can use to refine their attack plan. Information like what version of web server (IIS, Apache, etc...) you’re running, operating system, patch levels, and names and versions of web applications (PHP, SSL, SQL) your site may be utilizing.

Since security vulnerabilities are dependent on software vendor and version, blindly attacking may lead to detection, denial of request/service or in severe cases systems being temporarily taken off line.

Knowing a web server’s version and operating system details can greatly increase the probability and efficiency of an attack. If an attacker can accurately use available exploits, the chances of successful exploitation increase significantly. For an attacker to be able to accurately identify the version of your web server opens yourself to attacks both manual and automated (worms).
HTTP Headers

Server Tag

*From RFC 2616 Section 14.38*

[http://www.w3.org/Protocols/rfc2616/rfc2616.html](http://www.w3.org/Protocols/rfc2616/rfc2616.html)

The Server response-header field contains information about the software used by the origin server to handle the request. The field can contain multiple product tokens and comments identifying the server and any significant sub products. The product tokens are listed in order of their significance for identifying the application.

```
Server = "Server" ":" 1* ( product | comment )
```

Server Tags

Examples:

```
Server: Microsoft-IIS/5.0
Server: Apache/1.3.33 (Unix) PHP/4.3.10
Server: Sun-ONE-Web-Server/6.1
Server: Oracle-Application-Server-10g OracleAS-Web-Cache-10g/9.0.4.1.0
```

If the response is being forwarded through a proxy, the proxy application MUST NOT modify the Server response-header. Instead, it SHOULD include a Via field.

Note: Revealing the specific software version of the server might allow the server machine to become more vulnerable to attacks against software that is known to contain security holes. Server implementers are encouraged to make this field a configurable option.

Cookie Values

*From RFC 2109*  [http://rfc.net/rfc2109.html](http://rfc.net/rfc2109.html)

A piece of information sent by a Web server to a user's browser. Cookies may include information such as login or registration identification, user preferences, online "shopping cart" information, etc. The browser saves the information, and sends it back to the Web server whenever the browser returns to the Web site.

Fingerprint from Cookie Values

Examples:

[https://www.learnsecurityonline.com/](https://www.learnsecurityonline.com/)
A cookie with ASP is a dead giveaway we are on some sort of a Windows Box
Where a cookie with JSP tells us that some sort of Java is at work.

Error Pages

If a malformed request is send to the server, it may reply back with an Error Code and
Software version and information. So even if you have fixed your header information,
calling a non-existent page may give you an error message with useful information.
Using netcat or telnet to call nonexistent pages can give you information as well.

400 - Bad Request
401 - Unauthorized Request
403 - Forbidden
404 - Not Found
500 - Internal error
503 - Service Unavailable

Apache Server
When Apache Server encounters an error, it displays a designated error message that's
prebuilt into the server. For example, if you request a page that Apache can't find or that
doesn't exist. Apache returns a 404 (page not found) error and provides a Web page that
indicates the error.

Not Found
The requested URL /grr was not found on this server.
Object not found!

The requested URL was not found on this server. If you entered the URL manually please check your spelling and try again.

If you think this is a server error, please contact the webmaster.

Error 404

www.windgazer.nl
Tue Sep 26 06:02:28 2006
Apache/2.2.2 (Linux/SUSE)

Apache Error Page revealing its software version information

Apache draws this information from the data stored in the httpd.conf configuration file.

IIS Error Page

The page cannot be found

The page you are looking for might have been removed, had its name changed, or is temporarily unavailable.

Please try the following:

- Make sure that the Web site address displayed in the address bar of your browser is spelled and formatted correctly.
- If you reached this page by clicking a link, contact the Web site administrator to alert them that the link is incorrectly formatted.
- Click the Back button to try another link.

HTTP Error 404 - File or directory not found.

Internet Information Services (IIS)

Technical Information (for support personnel)

- Go to Microsoft Product Support Services and perform a title search for the words "HTTP and 404."
- Open IIS Help, which is accessible in IIS Manager (inetmgr), and search for topics titled "Web Site Setup, Common Administrative Tasks, and About Custom Error Messages."
X-Powered By

ASP.net and PHP adds its own banner to your server tags “X-Powered by.” This allows an attacker to fingerprint what version of PHP you are running.

X-Powered-By server tag

Examples:

X-Powered-By: PHP/4.3.10
X-Powered-By: ASP.NET
X-Powered-By: JSP/2.0

Turning the “X-Powered-By” tag off:

Locate in php.ini the variable expose_php and turn it off. In your php.ini (based on your Linux distribution this can be found in various places, like /etc/php.ini, /etc/php5/apache2/php.ini, etc.) locate the line containing “expose_php On” and set it to Off.

Page Names


Example:
microsoft.com/technet/security/bulletin/FQ05-010.asp

Microsoft Active Server Pages (ASP) is a server-side scripting environment that you can use to create and run dynamic, interactive Web server applications. With ASP, you can combine HTML pages, script commands, and COM components to create interactive Web pages and powerful Web-based applications that are easy to develop and modify.
.jsp - Sun JSP [http://java.sun.com/products/jsp/]

Example:
java.sun.com/index.jsp

JavaServer Pages (JSP) technology enables Web developers and designers to rapidly develop and easily maintain, information-rich, dynamic Web pages that leverage existing business systems. As part of the Java technology family, JSP technology enables rapid development of Web-based applications that are platform independent. JSP technology separates the user interface from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content.

.php [http://www.php.net]

Example:
www.lso.com/index.php?id=1

PHP is an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly.

.cfm - Macromedia Cold Fusion Server [http://www.macromedia.com/software/coldfusion]

Example:
macromedia.com/cfusion/resource/rc_driver.cfm?pagename=cfmx%20updater

ColdFusion MX makes Internet application development and deployment faster and easier than any other solution available today. Easily extend or integrate with Java or .NET applications, connect to enterprise data and applications, create or consume web services, or interface with SMS on mobile devices or instant messaging clients. Add powerful application services for business reporting, rich-forms generation, printable document generation, full-text search, and graphing and charting.

.aspx

This is part of .Net/J2EE frameworks resource for web services and web services can be developed/deployed using this type of resource. Hence, by just glancing at the set of characters containing the .asmx extension we can fingerprint this resource to .Net.

.jws

Java Web Services runs with .jws extension on a few platforms. By looking at this extension we can guess about the underlying backend technologies. Axis integrated with tomcat can be identified because of the .jws extension.
WSDL (web services definition language) is the file in which web services’ access information resides.

**Banner Grabbing**

The simplest and most basic form of identifying HTTP servers is to look at the Server field in the HTTP response header. Using a TCP client like **netcat** or even **telnet**, it is possible to send an HTTP request to return the HTTP response header of the server, as shown below:

```
$ nc 192.168.0.56 80
HEAD / HTTP/1.0
```

```
HTTP/1.1 200 OK
Date: Mon, 16 Jun 2003 02:53:29 GMT
Server: Apache/1.3.3 (Unix)  (Red Hat/Linux)
ETag: "1813-49b-361b4df6"
Accept-Ranges: bytes
Content-Length: 1179
Connection: close
Content-Type: text/html
```

**Example 1: Apache 1.3.3 on Red Hat Linux**

```
HTTP/1.1 200 OK
Date: Sun, 15 Jun 2003 17:10:49 GMT
Server: Apache/1.3.23
Last-Modified: Thu, 27 Feb 2003 03:48:19 GMT
ETag: "32417-c4-3e5d8a83"
Accept-Ranges: bytes
Content-Length: 196
Connection: close
Content-Type: text/html
```

**Example 2: Apache 1.3.23**

```
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Expires: Tue, 17 Jun 2003 01:41:33 GMT
Date: Mon, 16 Jun 2003 01:41:33 GMT
Content-Type: text/html
Accept-Ranges: bytes
ETag: "b0aac0542e25c31:89d"
Content-Length: 7369
```

**Example 3: IIS 5.0**
Example 4: Netscape Enterprise 4.1

```
$nc www.microsoft.com 80
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Connection: close
Date: Tue, 26 Sep 2006 03:19:57 GMT
Server: Microsoft-IIS/6.0
P3P: CP="ALL IND DSP COR ADM CONo CUR CUso IVa0 IVDo PSA PSD TElO OUR SAMo C NT COM INT NAV ONL PHY PRE PUR UNI"
X-Powered-By: ASP.NET
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: text/html; charset=utf-8
Content-Length: 30606
```

Example 4: IIS 6.0

```
$nc www.sun.com 80
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Connection: close
Date: Tue, 26 Sep 2006 03:17:09 GMT
Server: Sun-Java-System-Web-Server/6.1
Content-type: text/html;charset=UTF-8
P3p: policyref="http://www.sun.com/p3p/Sun_P3P_Policy.xml", CP="CAO DSP COR CUR ADMa DEVa TAIa PSAa PSDa CONi TELi OUR SAMi PUBi IND PHY ONL PUR COM NAV INT DE M CNT STA POL PRE GOV"
X-powered-by: Servlet/2.4,JSP/2.0
Connection: close
Set-cookie: Starload=star-fep2; Path=/
Set-cookie: JSESSIONID=e8203f5cc16a34547426b9d909c5; Path=/
Set-cookie: JROUTE=V6Yg; Path=/
```

Example 5: Sun One Web server

```
HTTP/1.1 302 Found
Location: http://www.oracle.com/index.html
Content-Type: text/html; charset=iso-8859-1
Server: Oracle-Application-Server-10g OracleAS-Web-Cache-10g/10.1.2.0.2 (N;ecid=216174166961840885,0)
Date: Thu, 28 Sep 2006 14:11:36 GMT
Connection: Keep-Alive
```
Example 6: Oracle Application Server

**Fingerprinting tools**

**httprint** [http://net-square.com/httprint/](http://net-square.com/httprint/)

httprint is a web server fingerprinting tool. It relies on web server characteristics to accurately identify web servers, despite the fact that they may have been obfuscated by changing the server banner strings, or by plug-ins such as *mod_security* or *servermask*.

![httprint GUI](htpcprint.png)
httprint Command Line

httprint can also be used to detect web enabled devices which do not have a server banner string, such as wireless access points, routers, switches, and cable modems. httprint uses text signature strings and it is very easy to add signatures to the signature database.

Example httprint Signature Strings:

```
# 30/07/03
Microsoft-IIS/5.0
CD2698FD6E3C295E4B1653082C1D64050C5D2594DF1BD04276E4BB811C9DC5
0D7645B811C9DC5A2A00B84C9D690316014C217811C9DC5811C9DC52655F350
FCCCS53BE2CE69237E2CE6923FCD6E61AE2CE69272576B769E2CE6926CD2698FD
6ED3C295E2CE692009DB9B3E811C9DC5811C9DC56ED3C295E2CE6923
6ED3C295E2CE69235811C9DC56BE2CE69276ED3C295
icon: iis4_5.gif

# 30/07/03 - unverified
SunONE WebServer 6.0
811C9DC568D17AAE811C9DC5811C9DC5811C9DC594DF1BD0811C9DC5184C9B2
7FCBD95811C9DC52A20DB84C9D64C09C811C9DC5811C9DC5655F350
FCCCS53BE2CE69237E2CE6923FCD6E61AE2CE69272576B769E2CE6926CD2698FD
6ED3C295E2CE692009DB9B3E811C9DC5811C9DC56ED3C295E2CE6923
6ED3C295E2CE69235811C9DC56BE2CE69276ED3C295
icon: sun.gif
```

To get more information on how httprint works read the paper here:

https://www.learnsecurityonline.com/
Nmap (-sV Version Scan) http://insecure.org/nmap/

Nmap Version Scan

Nmap tries to determine the service protocol (e.g. ftp, ssh, telnet, http), the application name (e.g. ISC Bind, Apache httpd, Solaris telnetd), the version number, and sometimes miscellaneous details like whether an X server is open to connections or the SSH protocol version). If Nmap was compiled with OpenSSL support, it will connect to SSL servers to deduce the service listening behind the encryption.

Nmap –version-trace –d options

For more information read the Nmap Man page:

https://www.learnsecurityonline.com/
http://insecure.org/nmap/man/
http://insecure.org/nmap/vscan/

THC amap http://thc.org.segfault.net/thc-amap/

Amap (application map) is a next-generation scanning tool for pen testers. It attempts to identify applications even if they are running on a different port than normal. It also identifies non-ascii based applications. This is achieved by sending trigger packets, and looking up the responses in a list of response strings.

NetCraft http://www.netcraft.com/

Netcraft will report a site's operating system, web server, and netblock owner together with, if available, a graphical view of the time since last reboot for each of the computers serving the site.
Passive Fingerprinting Using Google [http://www.google.com](http://www.google.com)

If you want to locate “types” of web servers, you can use google and “googleprint”

```
site:netcraft.com intitle:That.Site.Running Apache
site:netcraft.com intitle:That.Site.Running "Windows 98"
site:netcraft.com intitle:That.Site.Running "Windows NT"
site:netcraft.com intitle:That.Site.Running "Sun One"
site:netcraft.com intitle:That.Site.Running "Netscape-Enterprise/3.6"
site:netcraft.com intitle:That.Site.Running Apache Freebsd
site:netcraft.com intitle:That.Site.Running Apache Linux
```

“intitle:Under.Construction "Disabling Dynamic" shows IIS 6.0 on W2K3

We can take the information that netcraft stores and use Google to query that information to look for certain types of web servers.
Preventing Fingerprinting

Banner String Obfuscation

Banner String Obfuscation is simply changing the string that the server returns for the "Server: " value. We’ll cover how to do this for Apache and IIS below.

IIS


URLScan is an ISAPI filter that allows Web site administrators to restrict the kind of HTTP requests that the server will process. By blocking specific HTTP requests, the URLScan filter prevents potentially harmful requests from reaching the server and causing damage.


The IIS Lockdown Tool functions by turning off unnecessary features, thereby reducing attack surface available to attackers.

ISAPI Filters

If your Kung Fu is good or just paranoid, you can create a custom Internet Server Application Program (ISAPI) filter or a dynamic link library (DLL) your IIS server calls each time it responds to a client request. The filter application sits between the network connection to the client and the HTTP server, allowing administrators to control the data
exchange the way headers are composed in HTTP responses) between the IIS and the client to help stop attackers from fingerprinting the server.

You might also choose to change the application mappings on your server to hide the file extensions which reveal your server is IIS. Wayne Berry explains how to map .asp extensions to .html [http://www.asp101.com/articles/wayne/pryingeyes/default.asp](http://www.asp101.com/articles/wayne/pryingeyes/default.asp)

**Server mask** [http://www.port80software.com/products/servermask](http://www.port80software.com/products/servermask)

Server Mask modifies your Web server's "finger print" by removing unnecessary HTTP response data, modifying cookie values and adjusting other response information thus obscuring the identity of your server. Successful obfuscation can confuse hackers and make it more likely they try the wrong exploits first and thus are identified by an intrusion detection system.

---

**Apache**

**Source modifications**

**Apache Source Altering**

Include/httpd.h

Define SERVER_BASEVENDOR “Apache Group” ➕Change these values
Define SERVER_PRODUCTVENDOR “Apache” ➕Change these values
Define SERVER_BASEVERSION “1.3.26” ➕Change these values

**Limit Directive Method Restrictions**

Apache httpd.conf
mod_headers [http://httpd.apache.org/docs/mod/mod_headers.html](http://httpd.apache.org/docs/mod/mod_headers.html)

http://httpd.apache.org/docs/2.2/mod/mod_headers.html

The Apache Mod_Headers.c module provides directives to control and modify HTTP request and response headers. Headers can be merged, replaced or removed.


ModSecurity(TM) is an open source intrusion detection and prevention engine for web applications. It can also be called a web application firewall. It operates embedded into the web server, acting as a powerful umbrella, shielding applications from attacks.

ModSecurity integrates with the web server, increasing your power to deal with web attacks. Some of its features worth mentioning are:

- Request filtering; incoming requests are analyzed as they come in, and before they get handled by the web server or other modules. (Strictly speaking, some processing is done on the request before it reaches ModSecurity but that is unavoidable in the embedded mode of operation.)
- Anti-evasion techniques; paths and parameters are normalized before analysis takes place in order to fight evasion techniques.
- Understanding of the HTTP protocol; since the engine understands HTTP, it performs very specific and fine granulated filtering. For example, it is possible to look at individual parameters, or named cookie values.
- POST payload analysis; the engine will intercept the contents transmitted using the POST method, too.
- Audit logging; full details of every request (including POST) can be logged for forensic analysis later.
- HTTPS filtering; since the engine is embedded in the web server, it gets access to request data after decryption takes place.
- Compressed content filtering; same as above, the security engine has access to request data after decompression takes place.
References and Resources

servermask: [http://www.port80software.com/support/articles/maskyourwebserver](http://www.port80software.com/support/articles/maskyourwebserver)
URLscan: [http://support.microsoft.com/kb/307608](http://support.microsoft.com/kb/307608)
Writing and ISAPI Filter: [http://www.graphcomp.com/info/specs/ms/httpfilt.htm](http://www.graphcomp.com/info/specs/ms/httpfilt.htm)