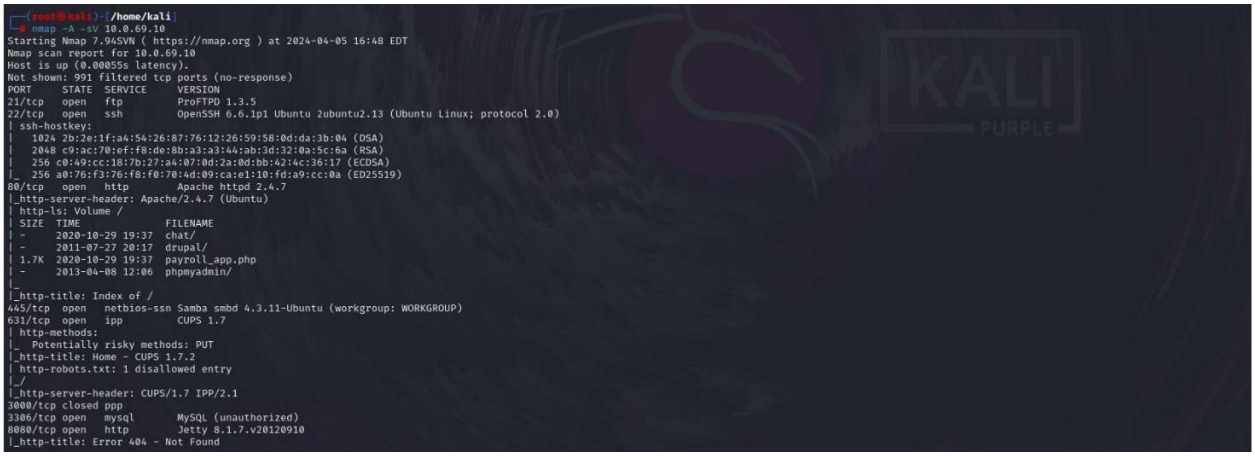
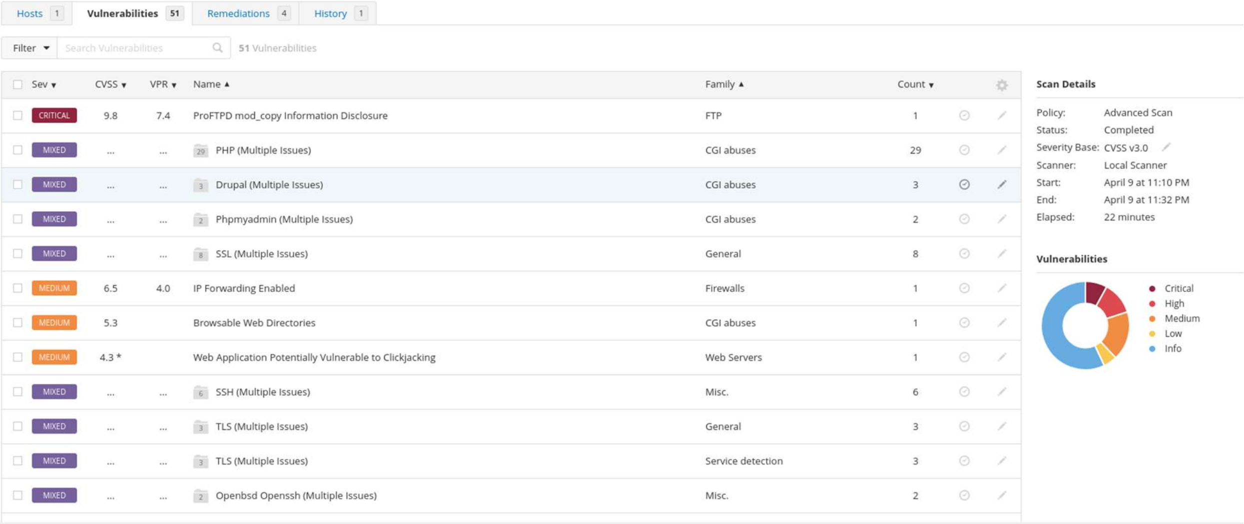
This report details how I tried to implement the necessary security fixes and the steps taken to fix the vulnerabilities observed through the Nessus Advanced and Nmap scans. The Nmap scan helps discover the open ports and the services running on the ports. In contrast, the Nessus scan of the Metasploitable machine (10.0.69.10) helped find outdated software, required configuration settings, and weak patches on the machine. Found the open ports using nmap scan : -A -sV 10.0.69.10



```
┌──(root㉿kali)-[/home/kali]
└─# nmap -A -sV 10.0.69.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-05 16:48 EDT
Nmap scan report for 10.0.69.10
Host is up (0.00055s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 2b:2e:1f:a4:54:26:87:76:12:26:59:58:0d:da:3b:04 (DSA)
|   2048 c9:ac:70:ef:f8:de:8b:a3:a3:44:ab:3d:32:0a:5c:6a (RSA)
|   256 c0:49:cc:18:7b:27:a4:07:0d:2a:0d:bb:42:4c:36:17 (ECDSA)
|_  256 a0:76:f3:76:f8:f0:70:4d:09:ca:e1:10:fd:a9:cc:0a (ED25519)
80/tcp   open   http        Apache httpd 2.4.7
|_http-server-header: Apache/2.4.7 (Ubuntu)
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2020-10-29 19:37  chat/
| -     2011-07-27 20:17  drupal/
| 1.7K  2020-10-29 19:37  payroll_app.php
| -     2013-04-08 12:06  phpmyadmin/
|_
|_http-title: Index of /
445/tcp  open   netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
| http-methods:
|_  Potentially risky methods: PUT
|_http-title: Home - CUPS 1.7.2
| http-robots.txt: 1 disallowed entry
|_/
|_http-server-header: CUPS/1.7 IPP/2.1
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
8080/tcp open   http        Jetty 8.1.7.v20120910
|_http-title: Error 404 - Not Found
```

And found vulnerabilities using the Advanced Scan using Nessus (*Metasploitable scan_Vulnerability Pre-Fix Scan*, n.d.) as per the screenshot provided below.



51 Vulnerabilities

□ Sev ▾	CVSS ▾	VPR ▾	Name ▴	Family ▴	Count ▾
□ CRITICAL	9.8	7.4	ProFTPD mod_copy Information Disclosure	FTP	1
□ MIXED	...	...	29 PHP (Multiple Issues)	CGI abuses	29
□ MIXED	...	...	3 Drupal (Multiple Issues)	CGI abuses	3
□ MIXED	...	...	2 Phpmyadmin (Multiple Issues)	CGI abuses	2
□ MIXED	...	...	8 SSL (Multiple Issues)	General	8
□ MEDIUM	6.5	4.0	IP Forwarding Enabled	Firewalls	1
□ MEDIUM	5.3		Browsable Web Directories	CGI abuses	1
□ MEDIUM	4.3 *		Web Application Potentially Vulnerable to Clickjacking	Web Servers	1
□ MIXED	...	...	6 SSH (Multiple Issues)	Misc.	6
□ MIXED	...	...	3 TLS (Multiple Issues)	General	3
□ MIXED	...	...	3 TLS (Multiple Issues)	Service detection	3
□ MIXED	...	...	2 Openbsd Openssh (Multiple Issues)	Misc.	2

Scan Details

Policy:	Advanced Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	April 9 at 11:10 PM
End:	April 9 at 11:32 PM
Elapsed:	22 minutes

Vulnerabilities

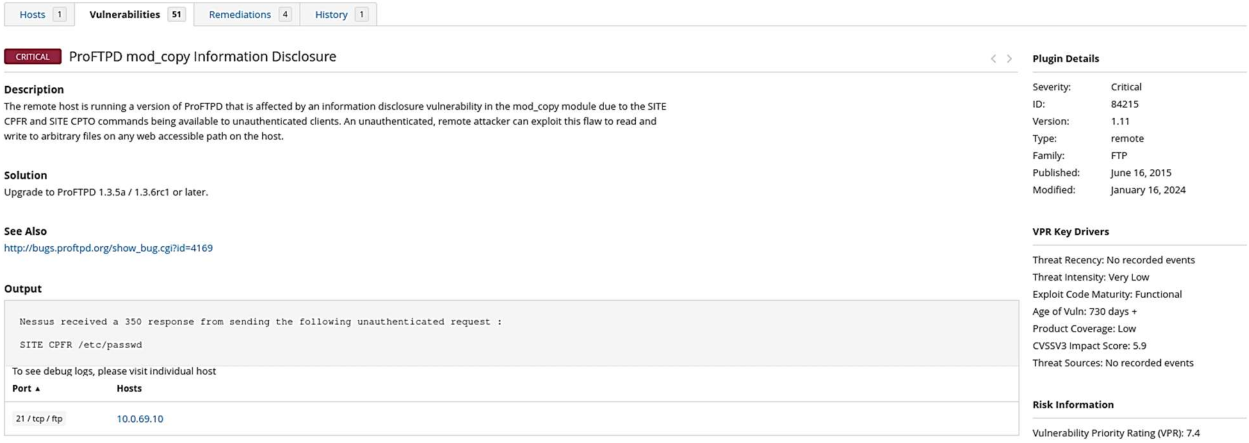
I managed to fix a few vulnerabilities related to

- Network services: FTP, SSH
- Web applications: SQL injection, Drupal Coder deserialization RCE.
- System services: SMB, IP forwarding.

- Outdated software versions: Php, drupal version
- Browsable Web Directories,Web Application Clickjacking, Php outdated software

Network Services:

FTP: ProFTPD mod_copy Information Disclosure



CRITICAL ProFTPD mod_copy Information Disclosure

Description

The remote host is running a version of ProFTPD that is affected by an information disclosure vulnerability in the mod_copy module due to the SITE CPFR and SITE CPTO commands being available to unauthenticated clients. An unauthenticated, remote attacker can exploit this flaw to read and write to arbitrary files on any web accessible path on the host.

Solution

Upgrade to ProFTPD 1.3.5a / 1.3.6rc1 or later.

See Also

http://bugs.proftpd.org/show_bug.cgi?id=4169

Output

```
Nessus received a 350 response from sending the following unauthenticated request :

SITE CPFR /etc/passwd
```

To see debug logs, please visit individual host

Port ▲	Hosts
21 / tcp / ftp	10.0.69.10

Plugin Details

Severity:	Critical
ID:	84215
Version:	1.11
Type:	remote
Family:	FTP
Published:	June 16, 2015
Modified:	January 16, 2024

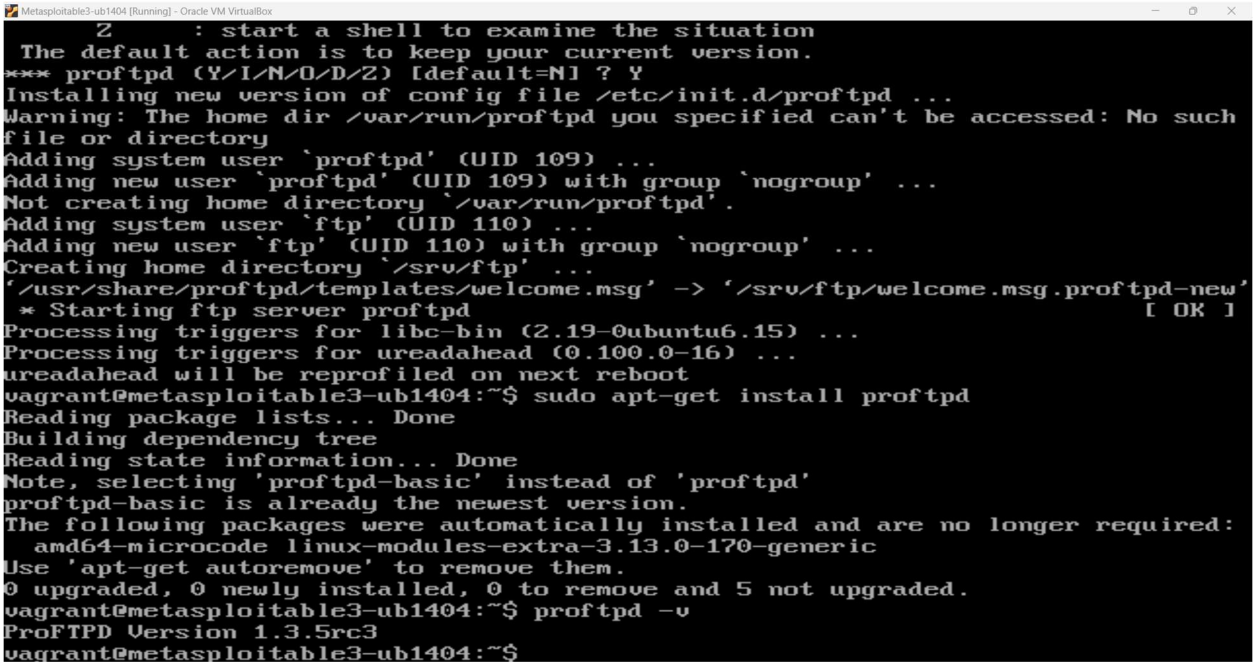
VPR Key Drivers

Threat Recency: No recorded events
Threat Intensity: Very Low
Exploit Code Maturity: Functional
Age of Vuln: 730 days +
Product Coverage: Low
CVSSV3 Impact Score: 5.9
Threat Sources: No recorded events

Risk Information

Vulnerability Priority Rating (VPR): 7.4

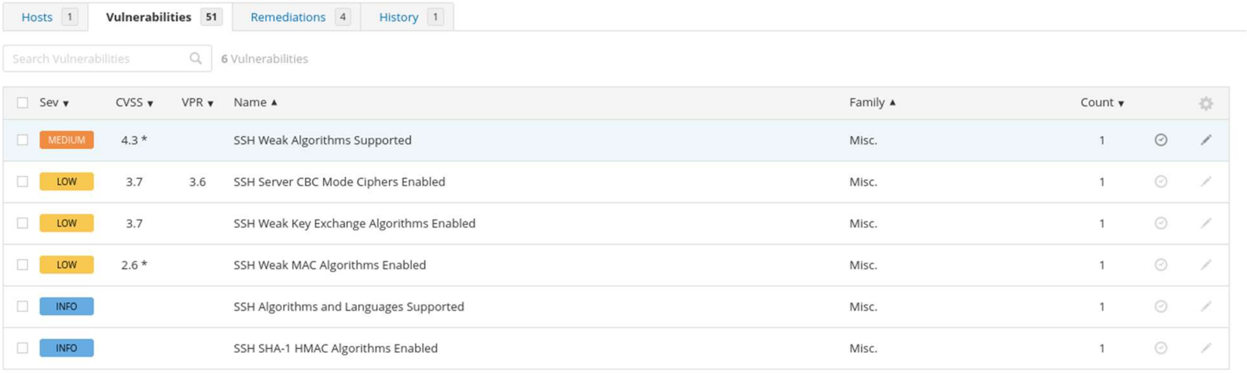
The solution to fix this vulnerability has been to upgrade the proftpd to a more recent version of proftpd 1.3.5a/1.3.6 or later.I used the command **sudo apt-get install proftpd**



```
     Z      : start a shell to examine the situation
 The default action is to keep your current version.
*** proftpd (Y/I/N/O/D/Z) [default=N] ? Y
Installing new version of config file /etc/init.d/proftpd ...
Warning: The home dir /var/run/proftpd you specified can't be accessed: No such
file or directory
Adding system user `proftpd' (UID 109) ...
Adding new user `proftpd' (UID 109) with group `nogroup' ...
Not creating home directory `/var/run/proftpd'.
Adding system user `ftp' (UID 110) ...
Adding new user `ftp' (UID 110) with group `nogroup' ...
Creating home directory `/srv/ftp' ...
'/usr/share/proftpd/templates/welcome.msg' -> '/srv/ftp/welcome.msg.proftpd-new'
 * Starting ftp server proftpd                                           [ OK ]
Processing triggers for libc-bin (2.19-0ubuntu6.15) ...
Processing triggers for ureadahead (0.100.0-16) ...
ureadahead will be reprofiled on next reboot
vagrant@metasploitable3-ub1404:~$ sudo apt-get install proftpd
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'proftpd-basic' instead of 'proftpd'
proftpd-basic is already the newest version.
The following packages were automatically installed and are no longer required:
  amd64-microcode linux-modules-extra-3.13.0-170-generic
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
vagrant@metasploitable3-ub1404:~$ proftpd -v
ProFTPD Version 1.3.5rc3
vagrant@metasploitable3-ub1404:~$
```

The above screenshot shows that the proftpd version is now updated which fixed the vulnerability related to FTP.

SSH: SSH Weak Algorithms supported

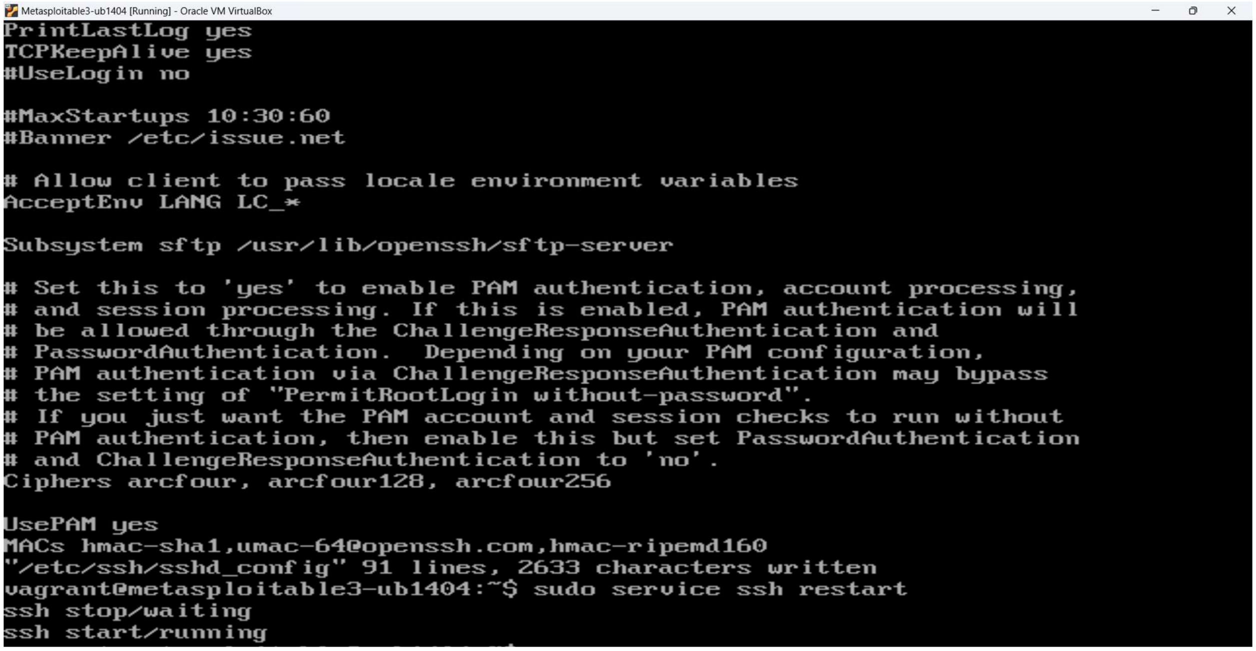


☐	Sev ▾	CVSS ▾	VPR ▾	Name ▴	Family ▴	Count ▾
☐	MEDIUM	4.3 *		SSH Weak Algorithms Supported	Misc.	1
☐	LOW	3.7	3.6	SSH Server CBC Mode Ciphers Enabled	Misc.	1
☐	LOW	3.7		SSH Weak Key Exchange Algorithms Enabled	Misc.	1
☐	LOW	2.6 *		SSH Weak MAC Algorithms Enabled	Misc.	1
☐	INFO			SSH Algorithms and Languages Supported	Misc.	1
☐	INFO			SSH SHA-1 HMAC Algorithms Enabled	Misc.	1

After Nessus detected that the remote SSH server was using weak or no cipher for encryption, including the Arcfour stream cipher, I fixed this issue by configuring the SSH server to use strong ciphers and MACs (message authentication codes). I opened the SSH configuration file at **/etc/ssh/sshd_config** and added.

Ciphers arcfour, arcfour256, arcfour128

MACs hmac-sha1, umac-64@openssh.com, hmac-ripemd160.

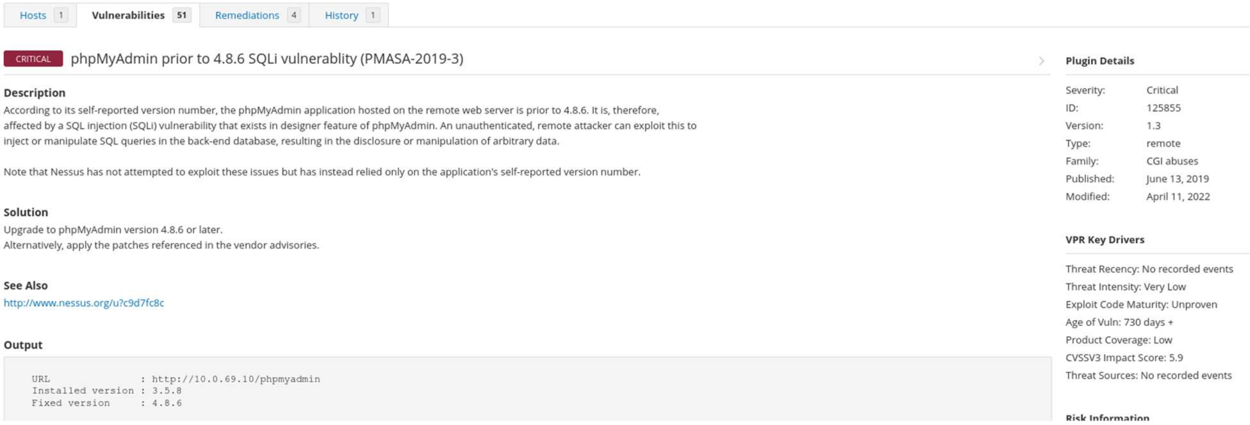


```
PrintLastLog yes
TCPKeepAlive yes
#UseLogin no

#MaxStartups 10:30:60
#Banner /etc/issue.net

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

Subsystem sftp /usr/lib/openssh/sftp-server

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
Ciphers arcfour, arcfour128, arcfour256

UsePAM yes
MACs hmac-sha1,umac-64@openssh.com,hmac-ripemd160
"/etc/ssh/sshd_config" 91 lines, 2633 characters written
vagrant@metasploitable3-ub1404:~$ sudo service ssh restart
ssh stop/waiting
ssh start/running
```

Once the configuration was updated, I saved the file and restarted the SSH service to apply the changes.

Web applications:

SQL injection: PhpMyAdmin



phpMyAdmin prior to 4.8.6 SQLi vulnerablity (PMASA-2019-3)

Description

According to its self-reported version number, the phpMyAdmin application hosted on the remote web server is prior to 4.8.6. It is, therefore, affected by a SQL injection (SQLi) vulnerability that exists in designer feature of phpMyAdmin. An unauthenticated, remote attacker can exploit this to inject or manipulate SQL queries in the back-end database, resulting in the disclosure or manipulation of arbitrary data.

Note that Nessus has not attempted to exploit these issues but has instead relied only on the application's self-reported version number.

Solution

Upgrade to phpMyAdmin version 4.8.6 or later.
Alternatively, apply the patches referenced in the vendor advisories.

See Also

http://www.nessus.org/u?c9d7fc8c

Output

```
URL               : http://10.0.69.10/phpmyadmin
Installed version : 3.5.8
Fixed version     : 4.8.6
```

Plugin Details

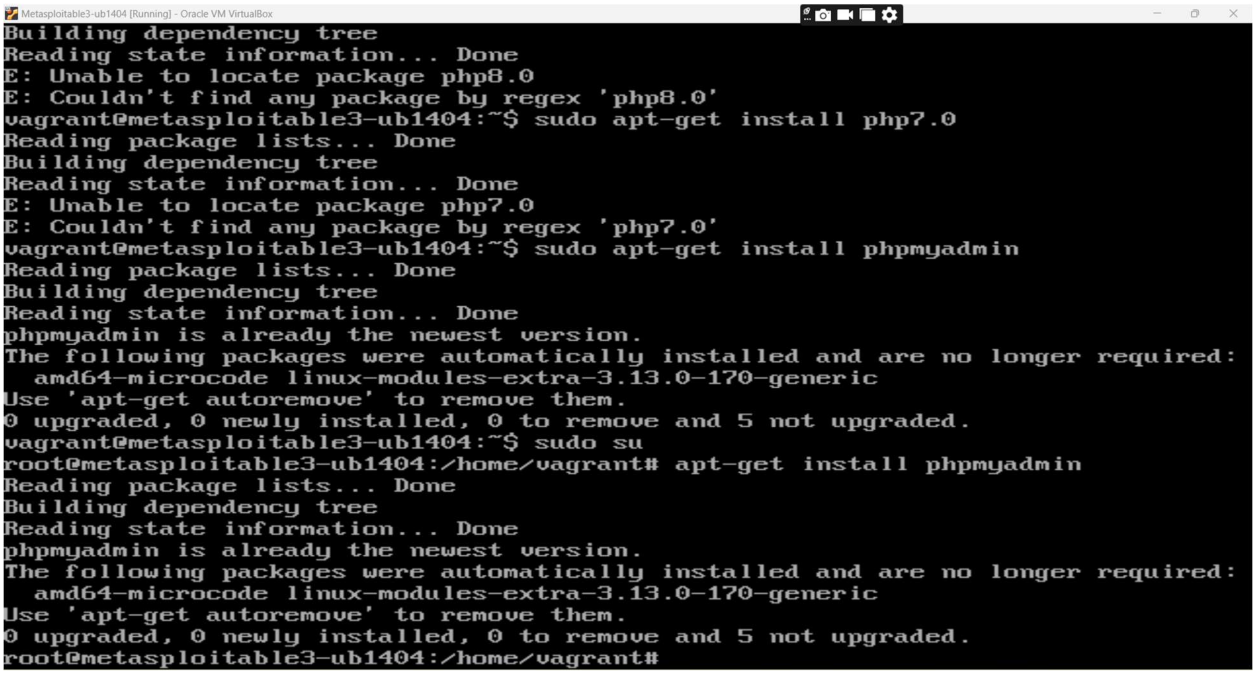
Severity:	Critical
ID:	125855
Version:	1.3
Type:	remote
Family:	CGI abuses
Published:	June 13, 2019
Modified:	April 11, 2022

VPR Key Drivers

Threat Recency: No recorded events
Threat Intensity: Very Low
Exploit Code Maturity: Unproven
Age of Vuln: 730 days +
Product Coverage: Low
CVSSV3 Impact Score: 5.9
Threat Sources: No recorded events

Risk Information

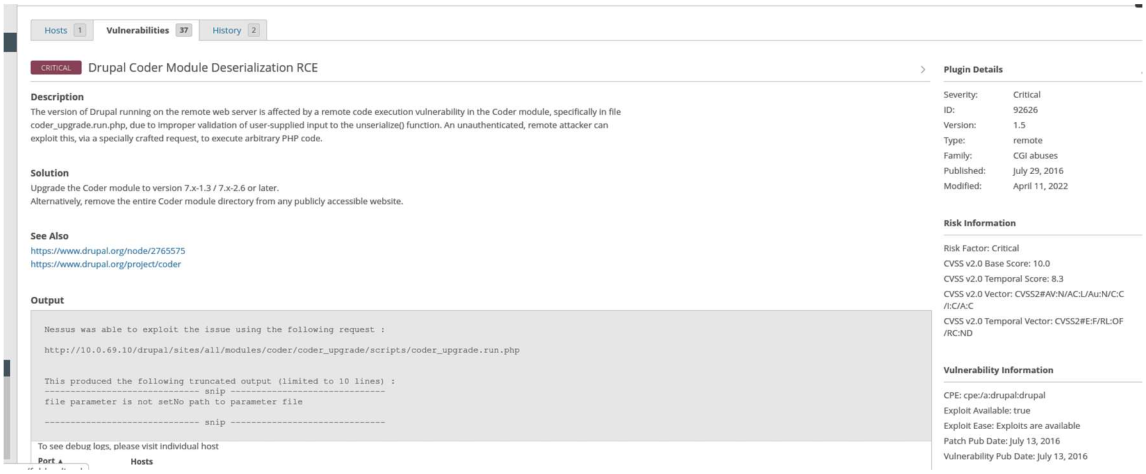
I updated phpMyAdmin to the most recent version in order to fix the SQL injection vulnerability present in versions of phpMyAdmin lower than 4.8.6. I downloaded the most recent phpMyAdmin package and unzipped the files, then made a backup of the current installation and database.



```
Building dependency tree
Reading state information... Done
E: Unable to locate package php8.0
E: Couldn't find any package by regex 'php8.0'
vagrant@metasploitable3-ub1404:~$ sudo apt-get install php7.0
Reading package lists... Done
Building dependency tree
Reading state information... Done
E: Unable to locate package php7.0
E: Couldn't find any package by regex 'php7.0'
vagrant@metasploitable3-ub1404:~$ sudo apt-get install phpmyadmin
Reading package lists... Done
Building dependency tree
Reading state information... Done
phpmyadmin is already the newest version.
The following packages were automatically installed and are no longer required:
  amd64-microcode linux-modules-extra-3.13.0-170-generic
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
vagrant@metasploitable3-ub1404:~$ sudo su
root@metasploitable3-ub1404:/home/vagrant# apt-get install phpmyadmin
Reading package lists... Done
Building dependency tree
Reading state information... Done
phpmyadmin is already the newest version.
The following packages were automatically installed and are no longer required:
  amd64-microcode linux-modules-extra-3.13.0-170-generic
Use 'apt-get autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 5 not upgraded.
root@metasploitable3-ub1404:/home/vagrant#
```

1.

Drupal Deserialization RCE:



Drupal Coder Module Deserialization RCE

Description

The version of Drupal running on the remote web server is affected by a remote code execution vulnerability in the Coder module, specifically in file coder_upgrade.run.php, due to improper validation of user-supplied input to the unserialize() function. An unauthenticated, remote attacker can exploit this, via a specially crafted request, to execute arbitrary PHP code.

Solution

Upgrade the Coder module to version 7.x-1.3 / 7.x-2.6 or later.
Alternatively, remove the entire Coder module directory from any publicly accessible website.

See Also

https://www.drupal.org/node/2765575
https://www.drupal.org/project/coder

Output

```
Nessus was able to exploit the issue using the following request :

http://10.0.69.10/drupal/sites/all/modules/coder/coder_upgrade/scripts/coder_upgrade.run.php


This produced the following truncated output (limited to 10 lines) :
------------------------------ snip ------------------------------
file parameter is not setNo path to parameter file

------------------------------ snip ------------------------------
```

To see debug logs, please visit individual host

Port ▲ Hosts

Plugin Details

Severity: Critical
ID: 92626
Version: 1.5
Type: remote
Family: CGI abuses
Published: July 29, 2016
Modified: April 11, 2022

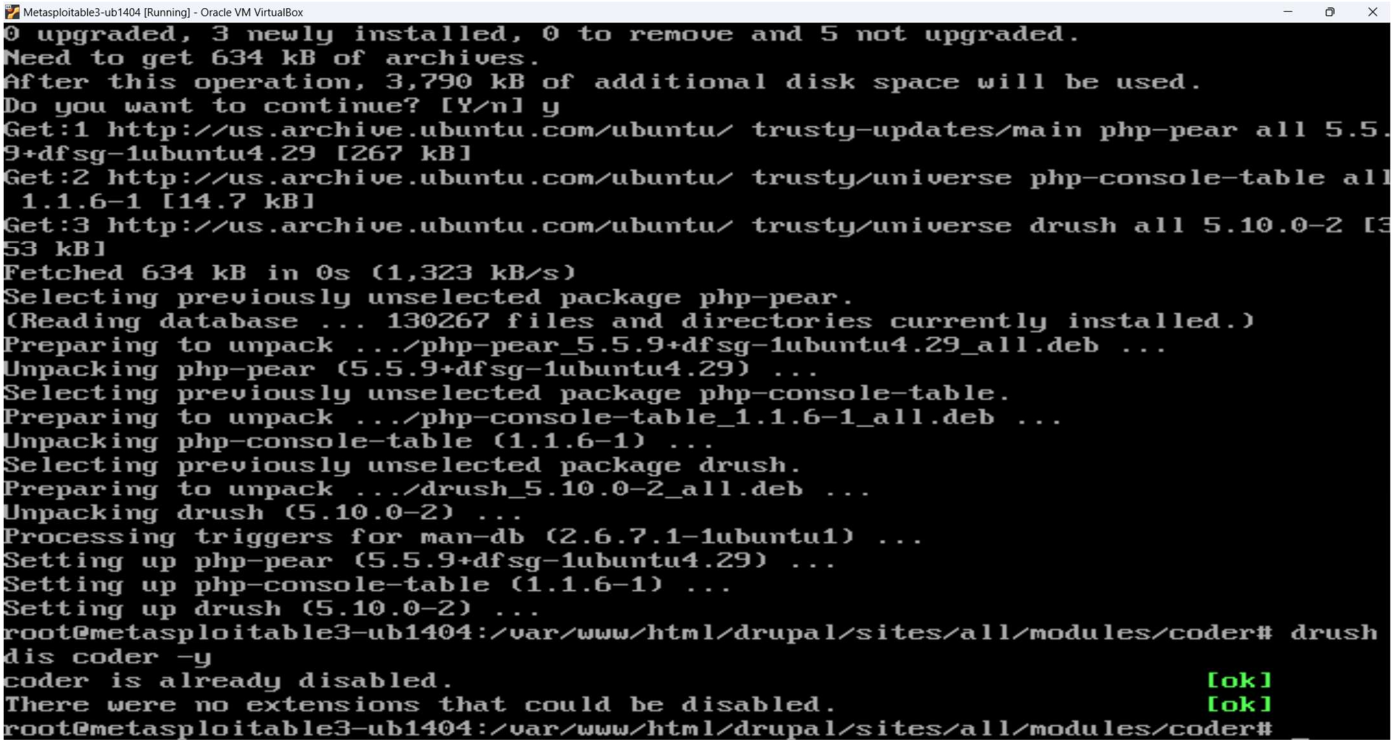
Risk Information

Risk Factor: Critical
CVSS v2.0 Base Score: 10.0
CVSS v2.0 Temporal Score: 8.3
CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:C/I:C/A:C
CVSS v2.0 Temporal Vector: CVSS2#E:F/RL:OF/RC:ND

Vulnerability Information

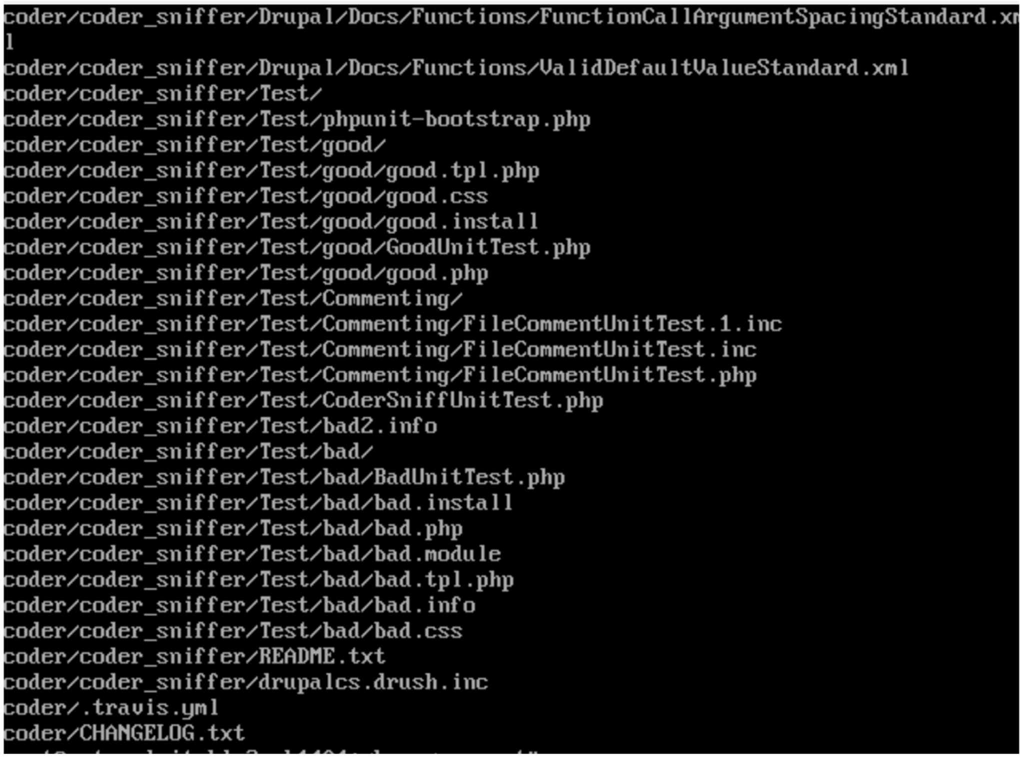
CPE: cpe:/a:drupal:drupal
Exploit Available: true
Exploit Ease: Exploits are available
Patch Pub Date: July 13, 2016
Vulnerability Pub Date: July 13, 2016

I disabled the existing coder module and downloaded the tar file for the most recent version of the Coder module, which is 7.x-2.6, using the wget command. After downloading, I used the tar -xzf command to extract the tar file; then I went to the /var/www/html/drupal/sites/all/modules directory to access the Drupal coder installation.



```
0 upgraded, 3 newly installed, 0 to remove and 5 not upgraded.
Need to get 634 kB of archives.
After this operation, 3,790 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu/ trusty-updates/main php-pear all 5.5.
9+dfsg-1ubuntu4.29 [267 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu/ trusty/universe php-console-table all
 1.1.6-1 [14.7 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu/ trusty/universe drush all 5.10.0-2 [3
53 kB]
Fetched 634 kB in 0s (1,323 kB/s)
Selecting previously unselected package php-pear.
(Reading database ... 130267 files and directories currently installed.)
Preparing to unpack .../php-pear_5.5.9+dfsg-1ubuntu4.29_all.deb ...
Unpacking php-pear (5.5.9+dfsg-1ubuntu4.29) ...
Selecting previously unselected package php-console-table.
Preparing to unpack .../php-console-table_1.1.6-1_all.deb ...
Unpacking php-console-table (1.1.6-1) ...
Selecting previously unselected package drush.
Preparing to unpack .../drush_5.10.0-2_all.deb ...
Unpacking drush (5.10.0-2) ...
Processing triggers for man-db (2.6.7.1-1ubuntu1) ...
Setting up php-pear (5.5.9+dfsg-1ubuntu4.29) ...
Setting up php-console-table (1.1.6-1) ...
Setting up drush (5.10.0-2) ...
root@metasploitable3-ub1404:/var/www/html/drupal/sites/all/modules/coder# drush
dis coder -y
coder is already disabled.                                             [ok]
There were no extensions that could be disabled.                       [ok]
root@metasploitable3-ub1404:/var/www/html/drupal/sites/all/modules/coder# _
```

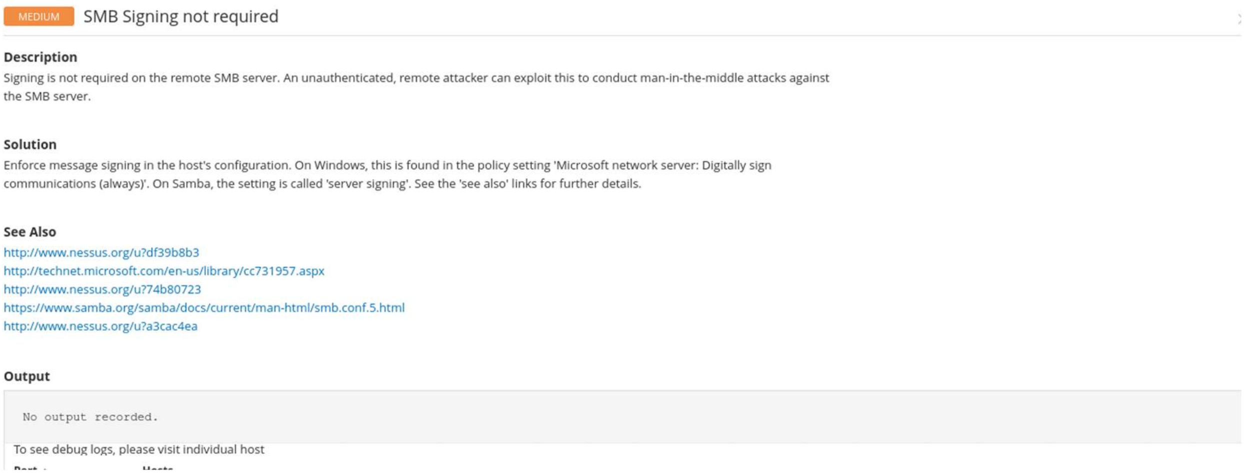
Lastly, I moved the Coder module files that had been extracted into my Drupal installation's modules directory of /var/www/html/drupal/sites/all/modules/coder



```
coder/coder_sniffer/Drupal/Docs/Functions/FunctionCallArgumentSpacingStandard.xm
l
coder/coder_sniffer/Drupal/Docs/Functions/ValidDefaultValueStandard.xml
coder/coder_sniffer/Test/
coder/coder_sniffer/Test/phpunit-bootstrap.php
coder/coder_sniffer/Test/good/
coder/coder_sniffer/Test/good/good.tpl.php
coder/coder_sniffer/Test/good/good.css
coder/coder_sniffer/Test/good/good.install
coder/coder_sniffer/Test/good/GoodUnitTest.php
coder/coder_sniffer/Test/good/good.php
coder/coder_sniffer/Test/Commenting/
coder/coder_sniffer/Test/Commenting/FileCommentUnitTest.1.inc
coder/coder_sniffer/Test/Commenting/FileCommentUnitTest.inc
coder/coder_sniffer/Test/Commenting/FileCommentUnitTest.php
coder/coder_sniffer/Test/CoderSniffUnitTest.php
coder/coder_sniffer/Test/bad2.info
coder/coder_sniffer/Test/bad/
coder/coder_sniffer/Test/bad/BadUnitTest.php
coder/coder_sniffer/Test/bad/bad.install
coder/coder_sniffer/Test/bad/bad.php
coder/coder_sniffer/Test/bad/bad.module
coder/coder_sniffer/Test/bad/bad.tpl.php
coder/coder_sniffer/Test/bad/bad.info
coder/coder_sniffer/Test/bad/bad.css
coder/coder_sniffer/README.txt
coder/coder_sniffer/drupalcs.drush.inc
coder/.travis.yml
coder/CHANGELOG.txt
```

System services:

SMB: SMB Signing not required.



MEDIUM SMB Signing not required

Description

Signing is not required on the remote SMB server. An unauthenticated, remote attacker can exploit this to conduct man-in-the-middle attacks against the SMB server.

Solution

Enforce message signing in the host's configuration. On Windows, this is found in the policy setting 'Microsoft network server: Digitally sign communications (always)'. On Samba, the setting is called 'server signing'. See the 'see also' links for further details.

See Also

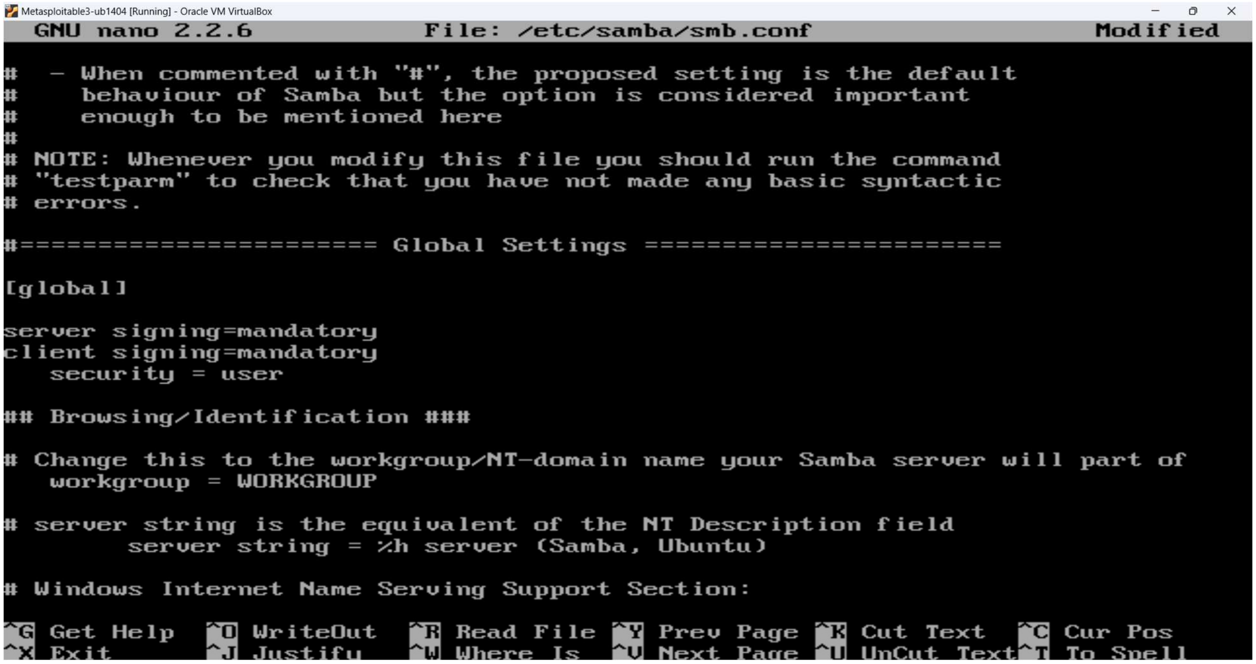
http://www.nessus.org/u?df39b8b3
http://technet.microsoft.com/en-us/library/cc731957.aspx
http://www.nessus.org/u?74b80723
https://www.samba.org/samba/docs/current/man-html/smb.conf.5.html
http://www.nessus.org/u?a3cac4ea

Output

```
No output recorded.
```

To see debug logs, please visit individual host

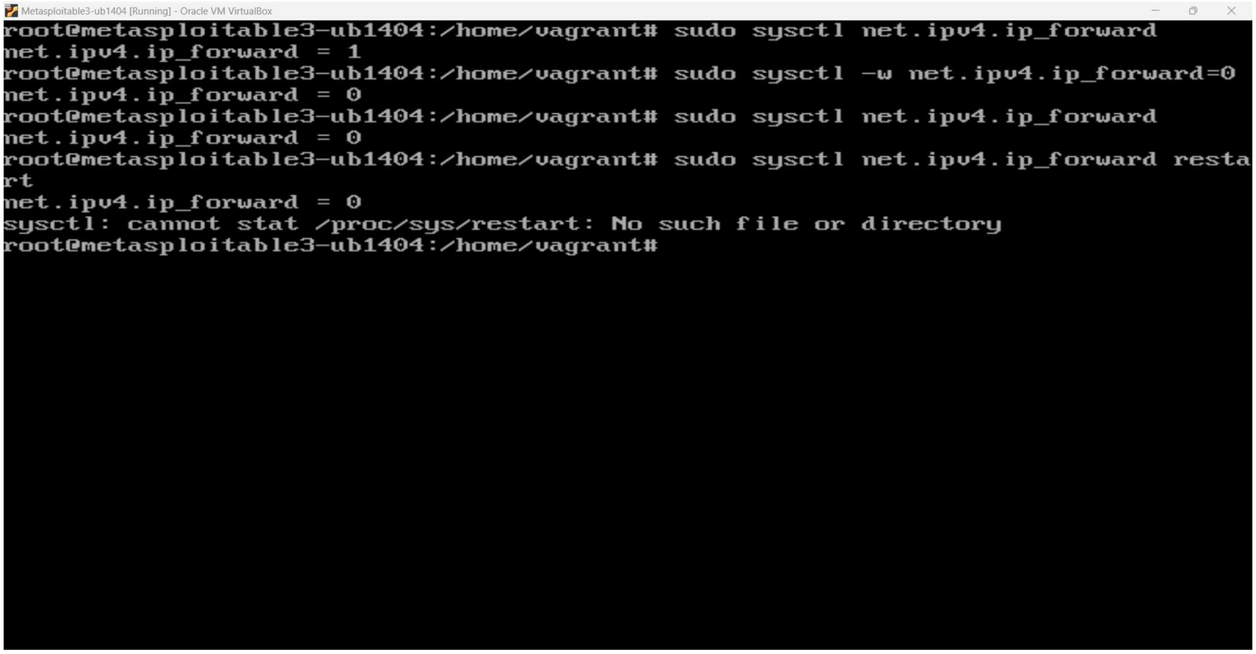
I first checked to see if I have samba services installed or not using “apt install samba -y” and once ensuring it is available I checked the service to start and the status using the command “service samba start/status” in super user mode and then used edited the samba.conf using “nano /etc/samba/smb.conf” command and then In global, I entered lines with “server signing = mandatory and client signing =mandatory.” And saved and restarted the samba service as shown below this solved the issue of signing not being required.



```
  GNU nano 2.2.6           File: /etc/samba/smb.conf                    Modified

#  - When commented with "#", the proposed setting is the default
#    behaviour of Samba but the option is considered important
#    enough to be mentioned here
#
# NOTE: Whenever you modify this file you should run the command
# "testparm" to check that you have not made any basic syntactic
# errors.

#======================= Global Settings =======================

[global]

server signing=mandatory
client signing=mandatory
   security = user

## Browsing/Identification ###

# Change this to the workgroup/NT-domain name your Samba server will part of
   workgroup = WORKGROUP

# server string is the equivalent of the NT Description field
        server string = %h server (Samba, Ubuntu)

# Windows Internet Name Serving Support Section:

^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page ^U UnCut Text^T To Spell
```

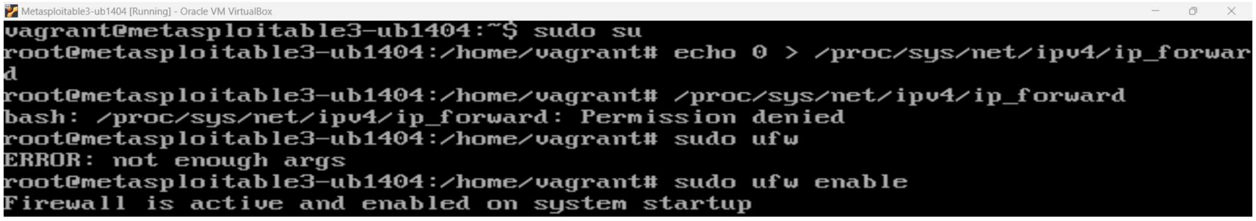
IP Forwarding:

Using the command sudo sysctl net.ipv4.ip_forward, I verified the status of the IP forwarding configuration.



```
root@metasploitable3-ub1404:/home/vagrant# sudo sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 1
root@metasploitable3-ub1404:/home/vagrant# sudo sysctl -w net.ipv4.ip_forward=0
net.ipv4.ip_forward = 0
root@metasploitable3-ub1404:/home/vagrant# sudo sysctl net.ipv4.ip_forward
net.ipv4.ip_forward = 0
root@metasploitable3-ub1404:/home/vagrant# sudo sysctl net.ipv4.ip_forward resta
rt
net.ipv4.ip_forward = 0
sysctl: cannot stat /proc/sys/restart: No such file or directory
root@metasploitable3-ub1404:/home/vagrant#
```

Next, I used the command sudo sysctl -w net.ipv4.ip_forward=0 to deactivate IP forwarding by setting the value of net.ipv4.ip_forward to 0. I updated the /etc/sysctl.conf file with the configuration to make sure it would hold up through reboots.

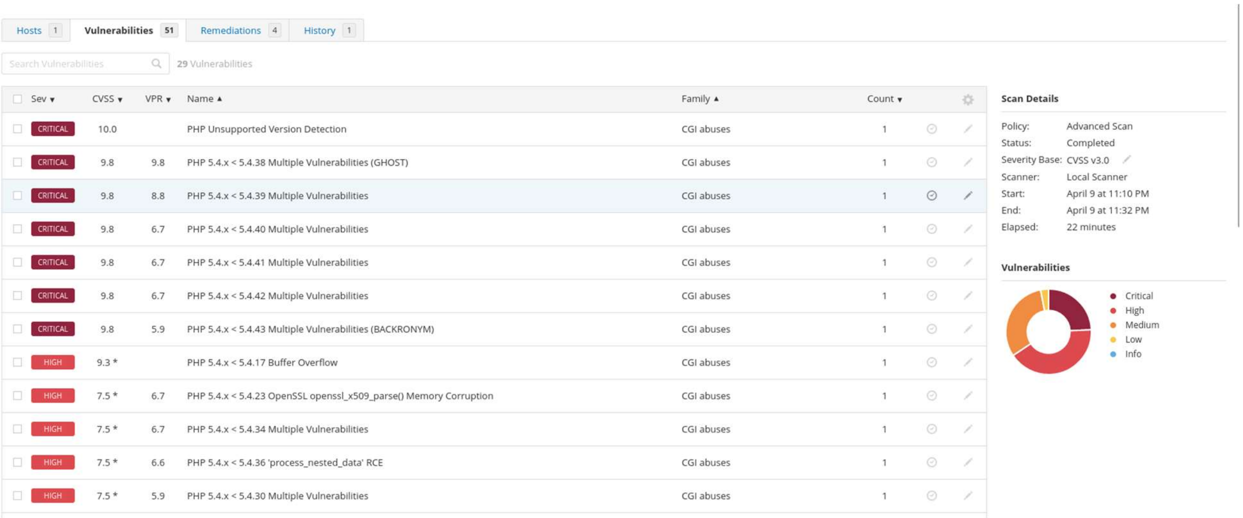


```
vagrant@metasploitable3-ub1404:~$ sudo su
root@metasploitable3-ub1404:/home/vagrant# echo 0 > /proc/sys/net/ipv4/ip_forwar
d
root@metasploitable3-ub1404:/home/vagrant# /proc/sys/net/ipv4/ip_forward
bash: /proc/sys/net/ipv4/ip_forward: Permission denied
root@metasploitable3-ub1404:/home/vagrant# sudo ufw
ERROR: not enough args
root@metasploitable3-ub1404:/home/vagrant# sudo ufw enable
Firewall is active and enabled on system startup
```

Using the command sudo ufw enable, I also turned on the Uncomplicated Firewall (UFW) for further network security which solves this vulnerability

Outdated software versions:

Php: Outdated PHP version leading to several vulnerabilities.



29 Vulnerabilities

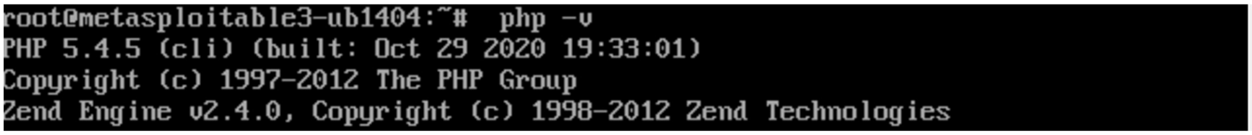
☐ Sev ▾	CVSS ▾	VPR ▾	Name ▴	Family ▴	Count ▾
☐ CRITICAL	10.0		PHP Unsupported Version Detection	CGI abuses	1
☐ CRITICAL	9.8	9.8	PHP 5.4.x < 5.4.38 Multiple Vulnerabilities (GHOST)	CGI abuses	1
☐ CRITICAL	9.8	8.8	PHP 5.4.x < 5.4.39 Multiple Vulnerabilities	CGI abuses	1
☐ CRITICAL	9.8	6.7	PHP 5.4.x < 5.4.40 Multiple Vulnerabilities	CGI abuses	1
☐ CRITICAL	9.8	6.7	PHP 5.4.x < 5.4.41 Multiple Vulnerabilities	CGI abuses	1
☐ CRITICAL	9.8	6.7	PHP 5.4.x < 5.4.42 Multiple Vulnerabilities	CGI abuses	1
☐ CRITICAL	9.8	5.9	PHP 5.4.x < 5.4.43 Multiple Vulnerabilities (BACKRONYM)	CGI abuses	1
☐ HIGH	9.3 *		PHP 5.4.x < 5.4.17 Buffer Overflow	CGI abuses	1
☐ HIGH	7.5 *	6.7	PHP 5.4.x < 5.4.23 OpenSSL openssl_x509_parse() Memory Corruption	CGI abuses	1
☐ HIGH	7.5 *	6.7	PHP 5.4.x < 5.4.34 Multiple Vulnerabilities	CGI abuses	1
☐ HIGH	7.5 *	6.6	PHP 5.4.x < 5.4.36 'process_nested_data' RCE	CGI abuses	1
☐ HIGH	7.5 *	5.9	PHP 5.4.x < 5.4.30 Multiple Vulnerabilities	CGI abuses	1

Scan Details

Policy:	Advanced Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	April 9 at 11:10 PM
End:	April 9 at 11:32 PM
Elapsed:	22 minutes

Vulnerabilities

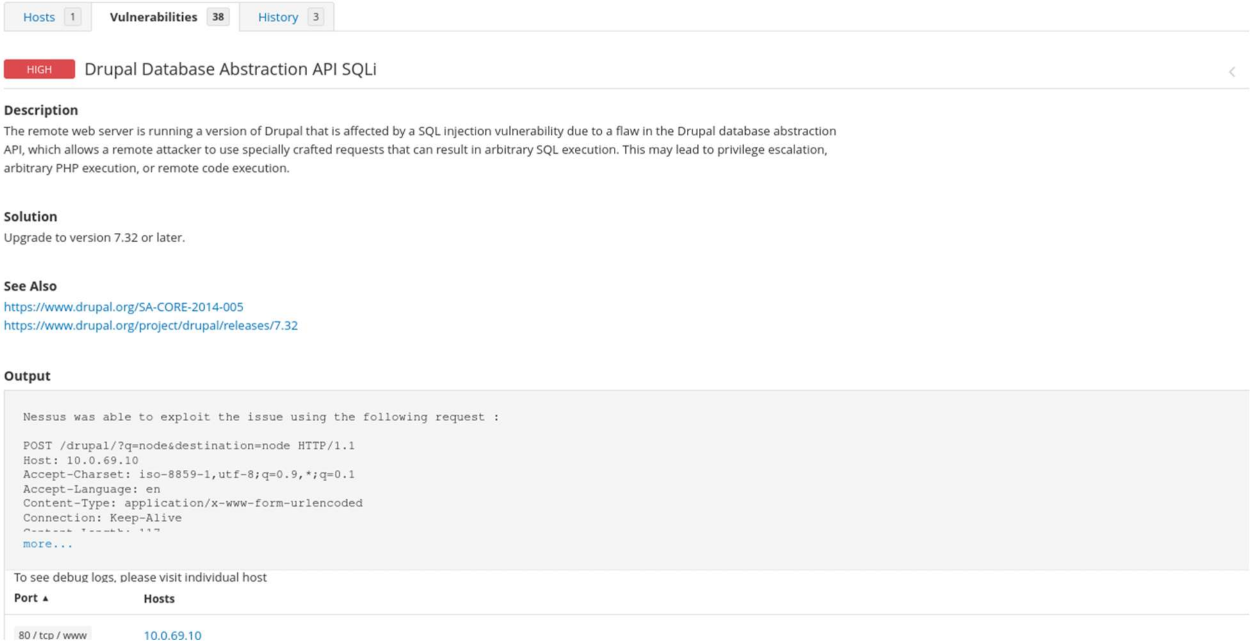
Using the command “php -v”, I first verified the version of PHP that was running. To get the most recent details on the packages that were available, I then used “sudo apt update” to update the package lists along with “Sudo apt install software-properties-common“ I then used “sudo apt upgrade php” to update PHP to the most recent version. I used “php -v” to check the installed PHP version after updating to make sure everything went well.



```
root@metasploitable3-ub1404:~#  php -v
PHP 5.4.5 (cli) (built: Oct 29 2020 19:33:01)
Copyright (c) 1997-2012 The PHP Group
Zend Engine v2.4.0, Copyright (c) 1998-2012 Zend Technologies
```

Finally, I used “sudo systemctl apache2 restart” to restart the webserver to implement the modifications and guarantee that the upgraded PHP version would function properly and remove all the vulnerabilities associated with it

Drupal version update:



Hosts 1 | Vulnerabilities 38 | History 3

HIGH Drupal Database Abstraction API SQLi

Description

The remote web server is running a version of Drupal that is affected by a SQL injection vulnerability due to a flaw in the Drupal database abstraction API, which allows a remote attacker to use specially crafted requests that can result in arbitrary SQL execution. This may lead to privilege escalation, arbitrary PHP execution, or remote code execution.

Solution

Upgrade to version 7.32 or later.

See Also

https://www.drupal.org/SA-CORE-2014-005
https://www.drupal.org/project/drupal/releases/7.32

Output

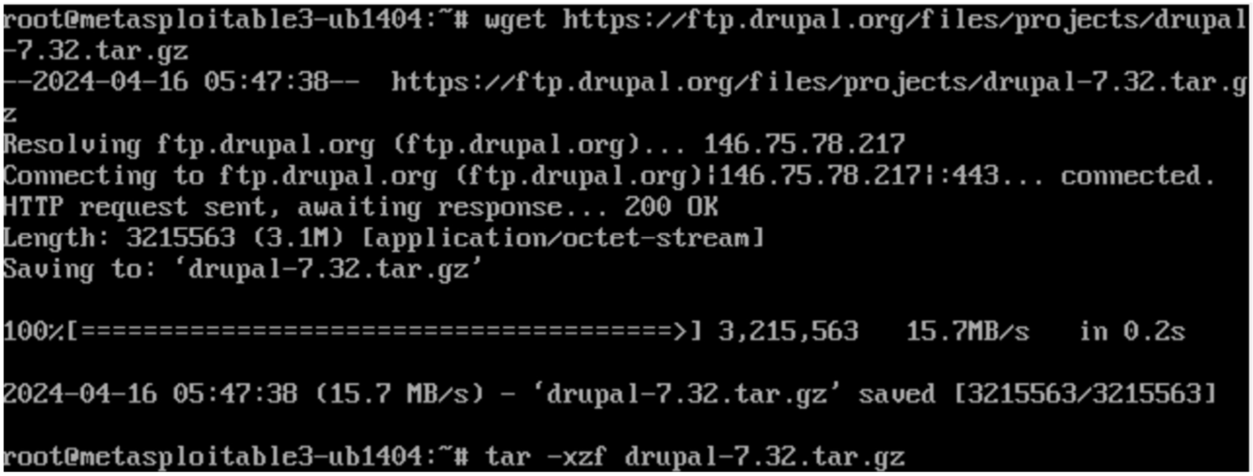
```
Nessus was able to exploit the issue using the following request :

POST /drupal/?q=node&destination=node HTTP/1.1
Host: 10.0.69.10
Accept-Charset: iso-8859-1,utf-8;q=0.9,*;q=0.1
Accept-Language: en
Content-Type: application/x-www-form-urlencoded
Connection: Keep-Alive
Content-Length: 117
more...
```

To see debug logs, please visit individual host

Port ▲	Hosts
80 / tcp / www	10.0.69.10

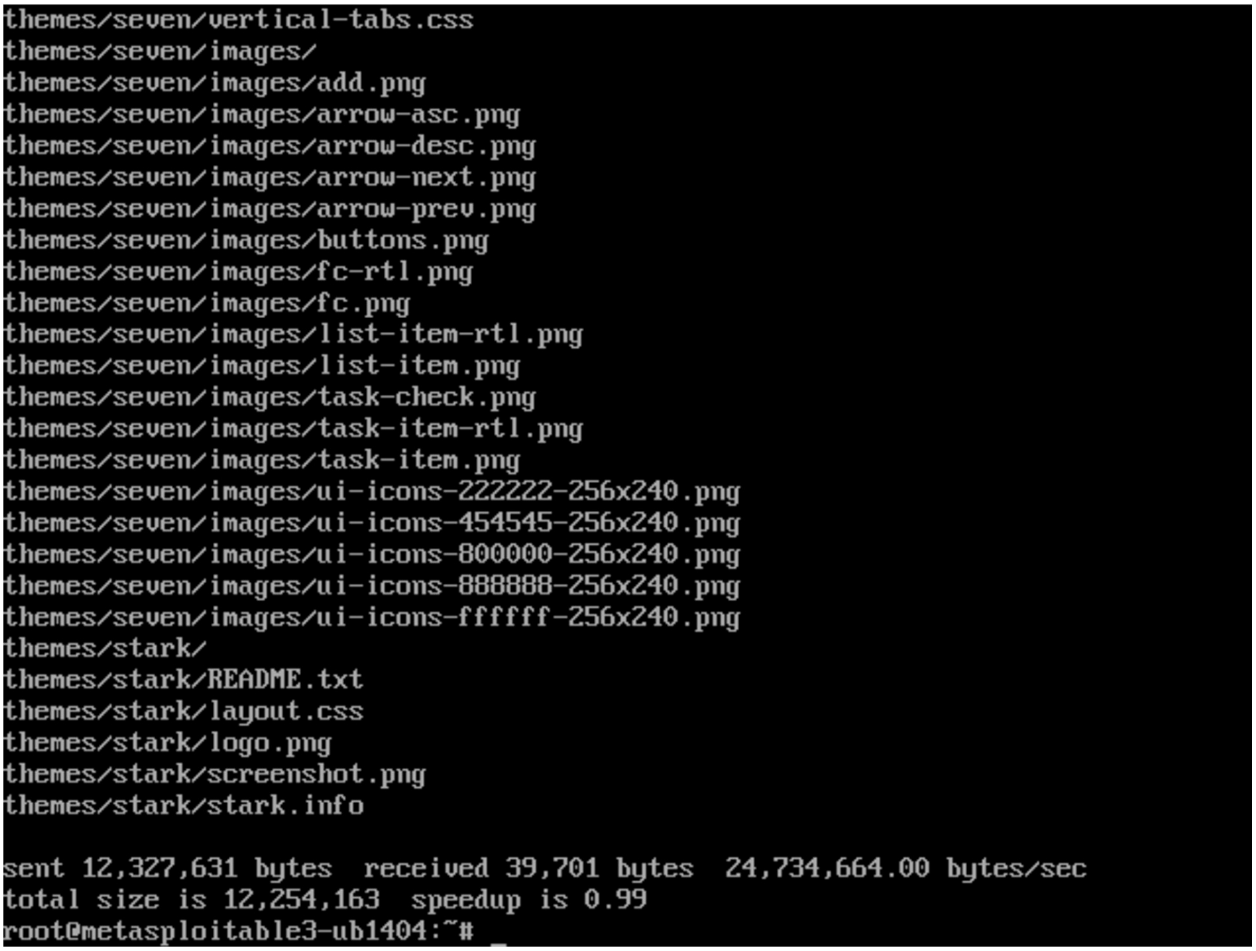
Using the wget command, I started by downloading the most recent version from the official phpMyAdmin FTP server: wget https://ftp.drupal.prg/files/projects/drupal-7.32.tar.gz

After the download had finished, I used tar -xzvf phpMyAdmin-7.32-all-languages.tar.gz to extract the file.



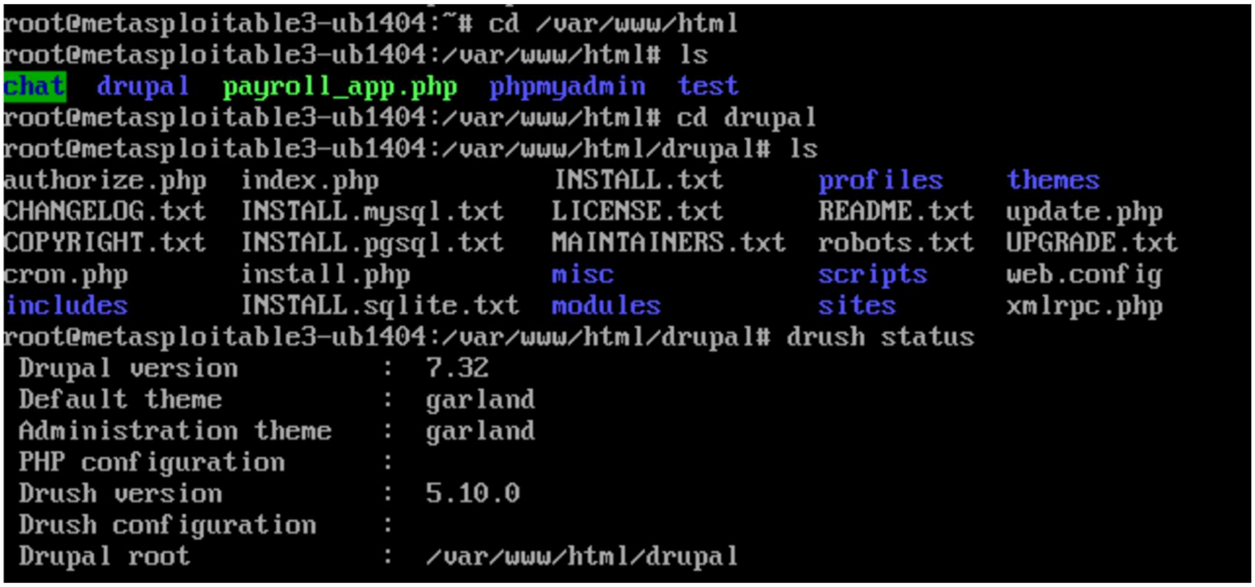
```
root@metasploitable3-ub1404:~# wget https://ftp.drupal.org/files/projects/drupal
-7.32.tar.gz
--2024-04-16 05:47:38--  https://ftp.drupal.org/files/projects/drupal-7.32.tar.g
z
Resolving ftp.drupal.org (ftp.drupal.org)... 146.75.78.217
Connecting to ftp.drupal.org (ftp.drupal.org)|146.75.78.217|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3215563 (3.1M) [application/octet-stream]
Saving to: ‘drupal-7.32.tar.gz’

100%[======================================>] 3,215,563   15.7MB/s   in 0.2s

2024-04-16 05:47:38 (15.7 MB/s) - ‘drupal-7.32.tar.gz’ saved [3215563/3215563]

root@metasploitable3-ub1404:~# tar -xzf drupal-7.32.tar.gz
```

I then moved the existing drupal and replaced it with the downloaded version of drupal



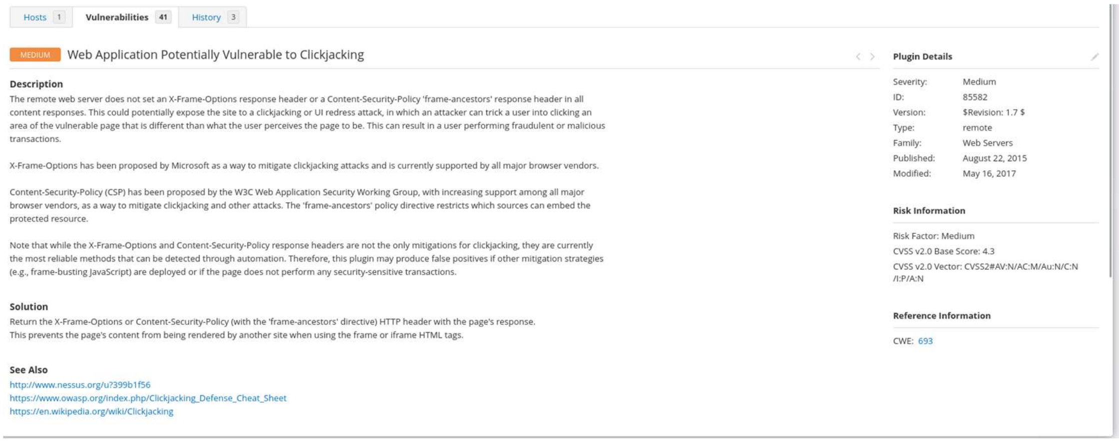
```
themes/seven/vertical-tabs.css
themes/seven/images/
themes/seven/images/add.png
themes/seven/images/arrow-asc.png
themes/seven/images/arrow-desc.png
themes/seven/images/arrow-next.png
themes/seven/images/arrow-prev.png
themes/seven/images/buttons.png
themes/seven/images/fc-rtl.png
themes/seven/images/fc.png
themes/seven/images/list-item-rtl.png
themes/seven/images/list-item.png
themes/seven/images/task-check.png
themes/seven/images/task-item-rtl.png
themes/seven/images/task-item.png
themes/seven/images/ui-icons-222222-256x240.png
themes/seven/images/ui-icons-454545-256x240.png
themes/seven/images/ui-icons-800000-256x240.png
themes/seven/images/ui-icons-888888-256x240.png
themes/seven/images/ui-icons-ffffff-256x240.png
themes/stark/
themes/stark/README.txt
themes/stark/layout.css
themes/stark/logo.png
themes/stark/screenshot.png
themes/stark/stark.info

sent 12,327,631 bytes  received 39,701 bytes  24,734,664.00 bytes/sec
total size is 12,254,163  speedup is 0.99
root@metasploitable3-ub1404:~# _
```

. I checked the configuration file to make sure everything was set correctly before completing the update and restarting the web server by running sudo systemctl restart apache2 to make the changes take effect.



```
root@metasploitable3-ub1404:~# cd /var/www/html
root@metasploitable3-ub1404:/var/www/html# ls
chat  drupal  payroll_app.php  phpmyadmin  test
root@metasploitable3-ub1404:/var/www/html# cd drupal
root@metasploitable3-ub1404:/var/www/html/drupal# ls
authorize.php  index.php          INSTALL.txt      profiles    themes
CHANGELOG.txt  INSTALL.mysql.txt  LICENSE.txt      README.txt  update.php
COPYRIGHT.txt  INSTALL.pgsql.txt  MAINTAINERS.txt  robots.txt  UPGRADE.txt
cron.php       install.php        misc             scripts     web.config
includes       INSTALL.sqlite.txt modules          sites       xmlrpc.php
root@metasploitable3-ub1404:/var/www/html/drupal# drush status
 Drupal version          :  7.32
 Default theme           :  garland
 Administration theme    :  garland
 PHP configuration       :
 Drush version           :  5.10.0
 Drush configuration     :
 Drupal root             :  /var/www/html/drupal
```

Web application clickjacking:



MEDIUM Web Application Potentially Vulnerable to Clickjacking

Description

The remote web server does not set an X-Frame-Options response header or a Content-Security-Policy 'frame-ancestors' response header in all content responses. This could potentially expose the site to a clickjacking or UI redress attack, in which an attacker can trick a user into clicking an area of the vulnerable page that is different than what the user perceives the page to be. This can result in a user performing fraudulent or malicious transactions.

X-Frame-Options has been proposed by Microsoft as a way to mitigate clickjacking attacks and is currently supported by all major browser vendors.

Content-Security-Policy (CSP) has been proposed by the W3C Web Application Security Working Group, with increasing support among all major browser vendors, as a way to mitigate clickjacking and other attacks. The 'frame-ancestors' policy directive restricts which sources can embed the protected resource.

Note that while the X-Frame-Options and Content-Security-Policy response headers are not the only mitigations for clickjacking, they are currently the most reliable methods that can be detected through automation. Therefore, this plugin may produce false positives if other mitigation strategies (e.g., frame-busting JavaScript) are deployed or if the page does not perform any security-sensitive transactions.

Solution

Return the X-Frame-Options or Content-Security-Policy (with the 'frame-ancestors' directive) HTTP header with the page's response.
This prevents the page's content from being rendered by another site when using the frame or iframe HTML tags.

See Also

http://www.nessus.org/u?399b1f56
https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet
https://en.wikipedia.org/wiki/Clickjacking

Plugin Details

Severity:	Medium
ID:	85582
Version:	$Revision: 1.7 $
Type:	remote
Family:	Web Servers
Published:	August 22, 2015
Modified:	May 16, 2017

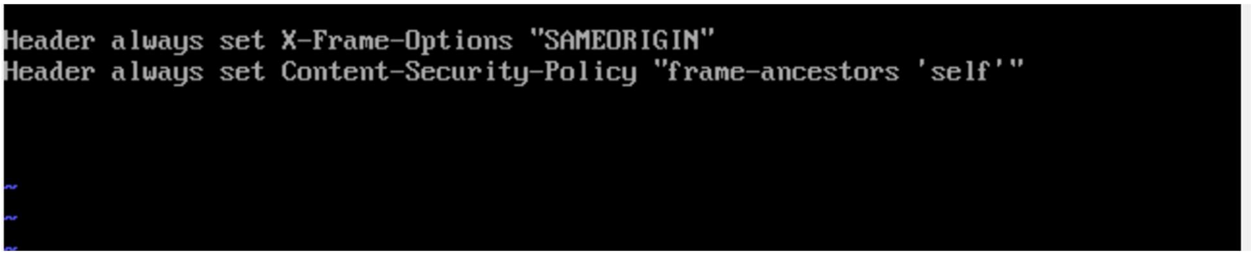
Risk Information

Risk Factor: Medium
CVSS v2.0 Base Score: 4.3
CVSS v2.0 Vector: CVSS2#AV:N/AC:M/Au:N/C:N
/I:P/A:N

Reference Information

CWE: 693

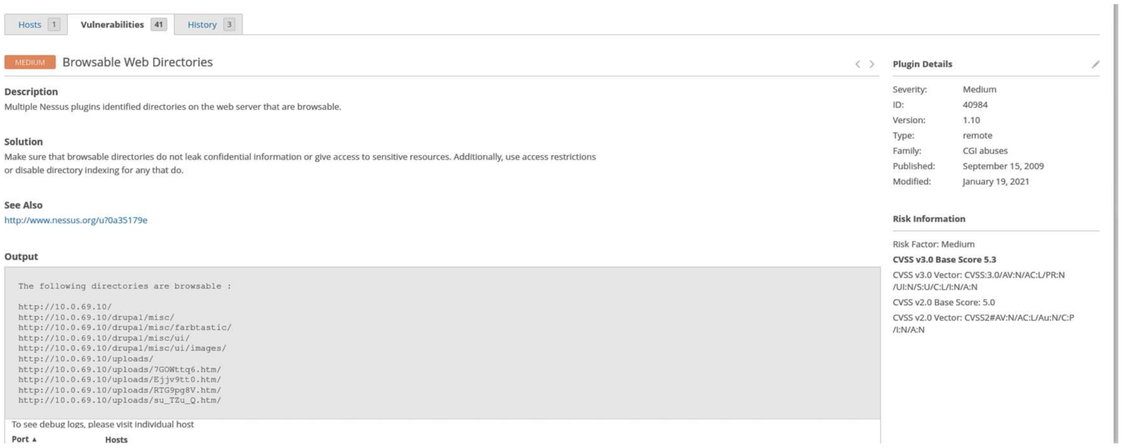
I updated the.htaccess file in the /var/www/html/ directory with security headers to address the web application clickjacking vulnerability. I went to the /var/www/html/ directory and used vi.htaccess to open the.htaccess file for editing in order to reduce this danger. To prevent the web application from being embedded in iframes, I added the lines Header always set X-Frame-Options "SAMEORIGIN" and Header always set Content-Security-Policy "frame-ancestors 'self'" to the.htaccess file.



```
Header always set X-Frame-Options "SAMEORIGIN"
Header always set Content-Security-Policy "frame-ancestors 'self'"



~
~
~
```

I saved the file and closed the editor after modifying. By blocking attempts to embed the program within iframes, this setting successfully defends the web application from clickjacking assaults.

Browsable Web directories:



Browsable Web Directories

Description

Multiple Nessus plugins identified directories on the web server that are browsable.

Solution

Make sure that browsable directories do not leak confidential information or give access to sensitive resources. Additionally, use access restrictions or disable directory indexing for any that do.

See Also

http://www.nessus.org/u?0a35179e

Output

```
The following directories are browsable :

http://10.0.69.10/
http://10.0.69.10/drupal/misc/
http://10.0.69.10/drupal/misc/farbtastic/
http://10.0.69.10/drupal/misc/ui/
http://10.0.69.10/drupal/misc/ui/images/
http://10.0.69.10/uploads/
http://10.0.69.10/uploads/7GOWttq6.htm/
http://10.0.69.10/uploads/Ejjv9tt0.htm/
http://10.0.69.10/uploads/RTG9pg8V.htm/
http://10.0.69.10/uploads/su_TZu_Q.htm/
```

To see debug logs, please visit individual host

Port ▲ Hosts

Plugin Details

Severity:	Medium
ID:	40984
Version:	1.10
Type:	remote
Family:	CGI abuses
Published:	September 15, 2009
Modified:	January 19, 2021

Risk Information

Risk Factor: Medium

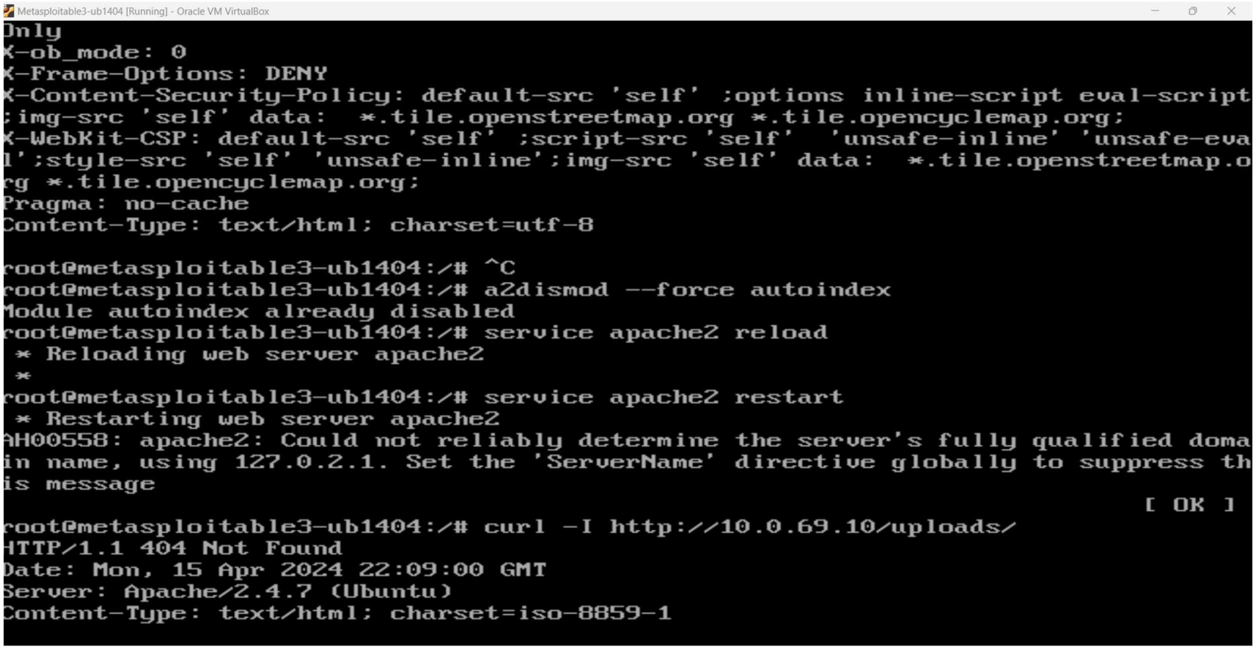
CVSS v3.0 Base Score 5.3

CVSS v3.0 Vector: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

CVSS v2.0 Base Score: 5.0

CVSS v2.0 Vector: CVSS2#AV:N/AC:L/Au:N/C:P/I:N/A:N

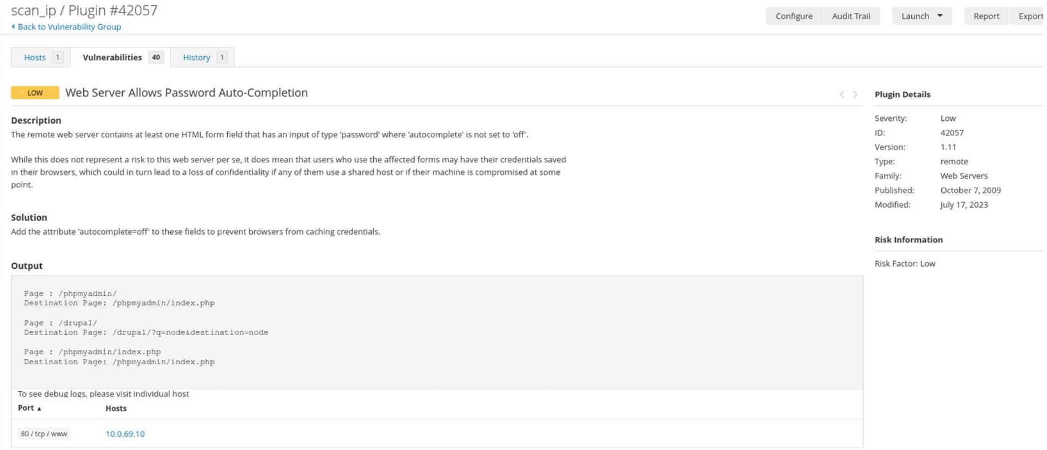
To fix browsable web directories and improve web server security, I turned off the autoindex module, which stops the server from showing directory listings. To firmly disable the autoindex module, I used the “sudo a2dismod --force autoindex” command.



```
Only
X-ob_mode: 0
X-Frame-Options: DENY
X-Content-Security-Policy: default-src 'self' ;options inline-script eval-script
;img-src 'self' data:  *.tile.openstreetmap.org *.tile.opencyclemap.org;
X-WebKit-CSP: default-src 'self' ;script-src 'self'  'unsafe-inline' 'unsafe-eva
l';style-src 'self' 'unsafe-inline';img-src 'self' data:  *.tile.openstreetmap.o
rg *.tile.opencyclemap.org;
Pragma: no-cache
Content-Type: text/html; charset=utf-8

root@metasploitable3-ub1404:/# ^C
root@metasploitable3-ub1404:/# a2dismod --force autoindex
Module autoindex already disabled
root@metasploitable3-ub1404:/# service apache2 reload
 * Reloading web server apache2
 *
root@metasploitable3-ub1404:/# service apache2 restart
 * Restarting web server apache2
AH00558: apache2: Could not reliably determine the server's fully qualified doma
in name, using 127.0.2.1. Set the 'ServerName' directive globally to suppress th
is message
                                                                        [ OK ]
root@metasploitable3-ub1404:/# curl -I http://10.0.69.10/uploads/
HTTP/1.1 404 Not Found
Date: Mon, 15 Apr 2024 22:09:00 GMT
Server: Apache/2.4.7 (Ubuntu)
Content-Type: text/html; charset=iso-8859-1
```

Using the command sudo service apache2 restart, I restarted the web server after disabling the module to ensure the modifications took effect, and the web directory is not browsable as shown in the figure above where we get 404 not found, which solves the issue.

Web Server Allows Password Auto-Completion:



scan_ip / Plugin #42057

‹ Back to Vulnerability Group

LOW Web Server Allows Password Auto-Completion

Description

The remote web server contains at least one HTML form field that has an input of type 'password' where 'autocomplete' is not set to 'off'.

While this does not represent a risk to this web server per se, it does mean that users who use the affected forms may have their credentials saved in their browsers, which could in turn lead to a loss of confidentiality if any of them use a shared host or if their machine is compromised at some point.

Solution

Add the attribute 'autocomplete=off' to these fields to prevent browsers from caching credentials.

Output

```
Page : /phpmyadmin/
Destination Page: /phpmyadmin/index.php

Page : /drupal/
Destination Page: /drupal/?q=node&destination=node

Page : /phpmyadmin/index.php
Destination Page: /phpmyadmin/index.php
```

To see debug logs, please visit individual host

Port ▲	Hosts
80 / tcp / www	10.0.69.10

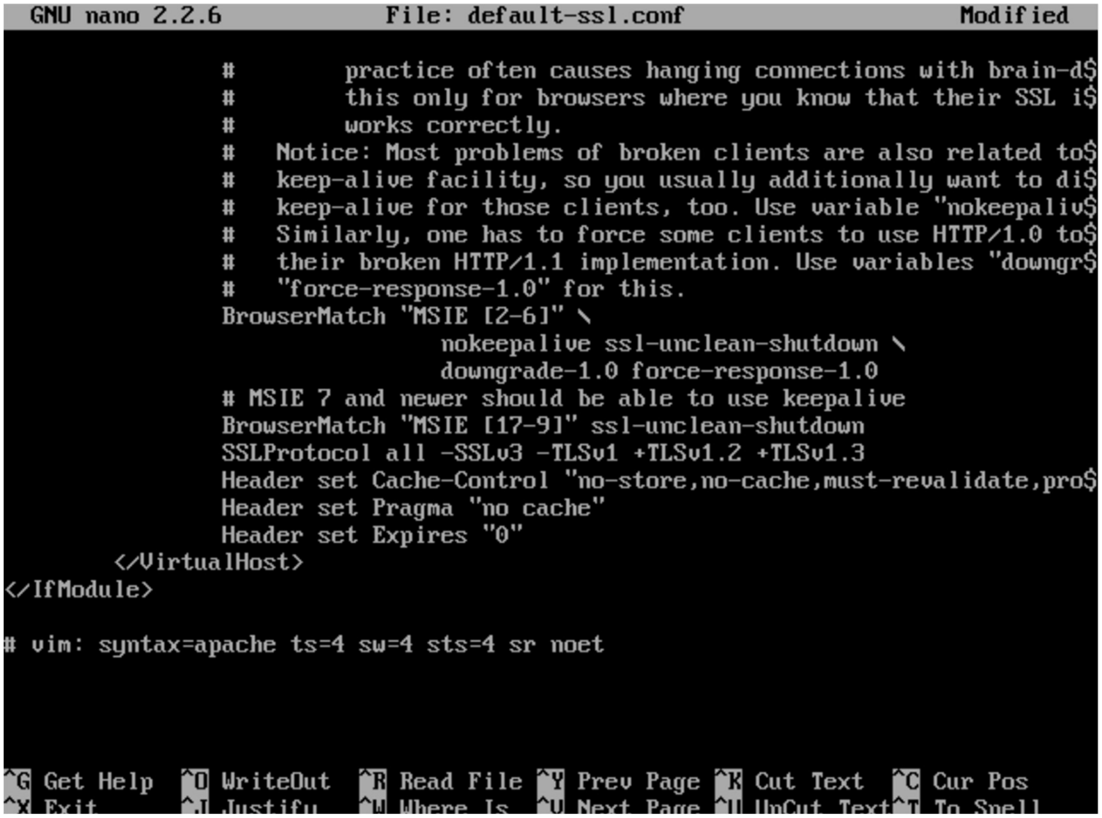
Plugin Details

Severity:	Low
ID:	42057
Version:	1.11
Type:	remote
Family:	Web Servers
Published:	October 7, 2009
Modified:	July 17, 2023

Risk Information

Risk Factor: Low

I managed to resolve this vulnerability using sudo nano default-ssl.conf
And added the Header lines inside the <Virtual Host> block which managed to remove the issue.



```
  GNU nano 2.2.6              File: default-ssl.conf                    Modified

                #       practice often causes hanging connections with brain-d$
                #       this only for browsers where you know that their SSL i$
                #       works correctly.
                #  Notice: Most problems of broken clients are also related to$
                #  keep-alive facility, so you usually additionally want to di$
                #  keep-alive for those clients, too. Use variable "nokeepaliv$
                #  Similarly, one has to force some clients to use HTTP/1.0 to$
                #  their broken HTTP/1.1 implementation. Use variables "downgr$
                #  "force-response-1.0" for this.
                BrowserMatch "MSIE [2-6]" \
                                nokeepalive ssl-unclean-shutdown \
                                downgrade-1.0 force-response-1.0
                # MSIE 7 and newer should be able to use keepalive
                BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
                SSLProtocol all -SSLv3 -TLSv1 +TLSv1.2 +TLSv1.3
                Header set Cache-Control "no-store,no-cache,must-revalidate,pro$
                Header set Pragma "no cache"
                Header set Expires "0"
        </VirtualHost>
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet



^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Conclusion:

I have successfully fixed the security flaws in the Metasploitable 3 virtual machine, but I encountered SSL/TLS vulnerabilities because of compatibility issues with Metasploitable version 3. After fixing the vulnerabilities, I ran a scan again through Nessus to check the vulnerabilities and managed to secure my machine to a great extent.(*Post Fix scan_Metaspolitable*, n.d.)

References and Appendix:

Metasploitable scan_Vulnerability pre-fix scan. (n.d.).

Post fix scan_Metaspolitable. (n.d.).