

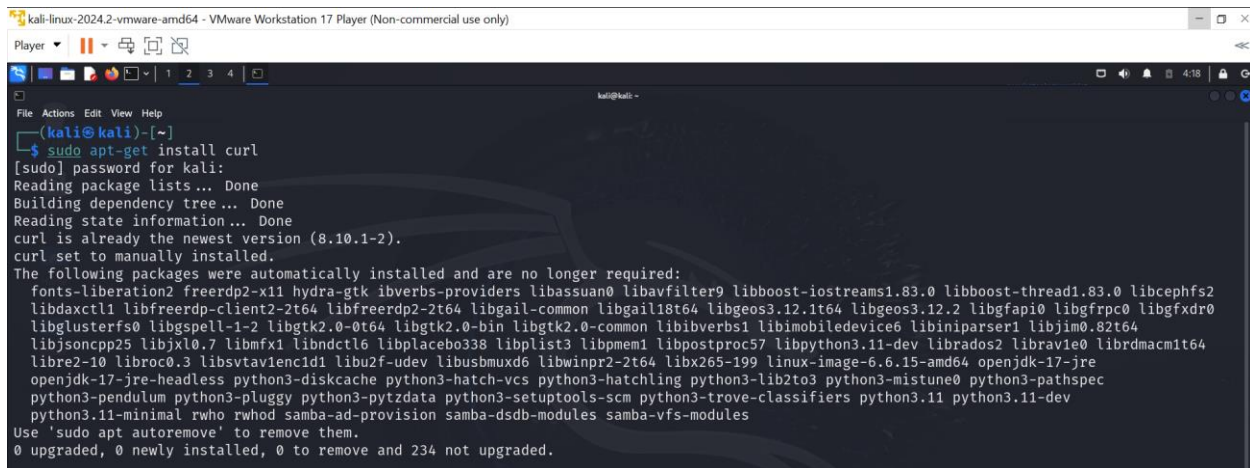
# MANUAL INFORMATION GATHERING

## TOOL: CURL TOOL

The curl tool is a command-line utility used to transfer data to or from a server using various protocols, including HTTP, HTTPS, FTP, and others. It's widely used for making web requests, testing APIs, downloading files, and more.

### TASK 1

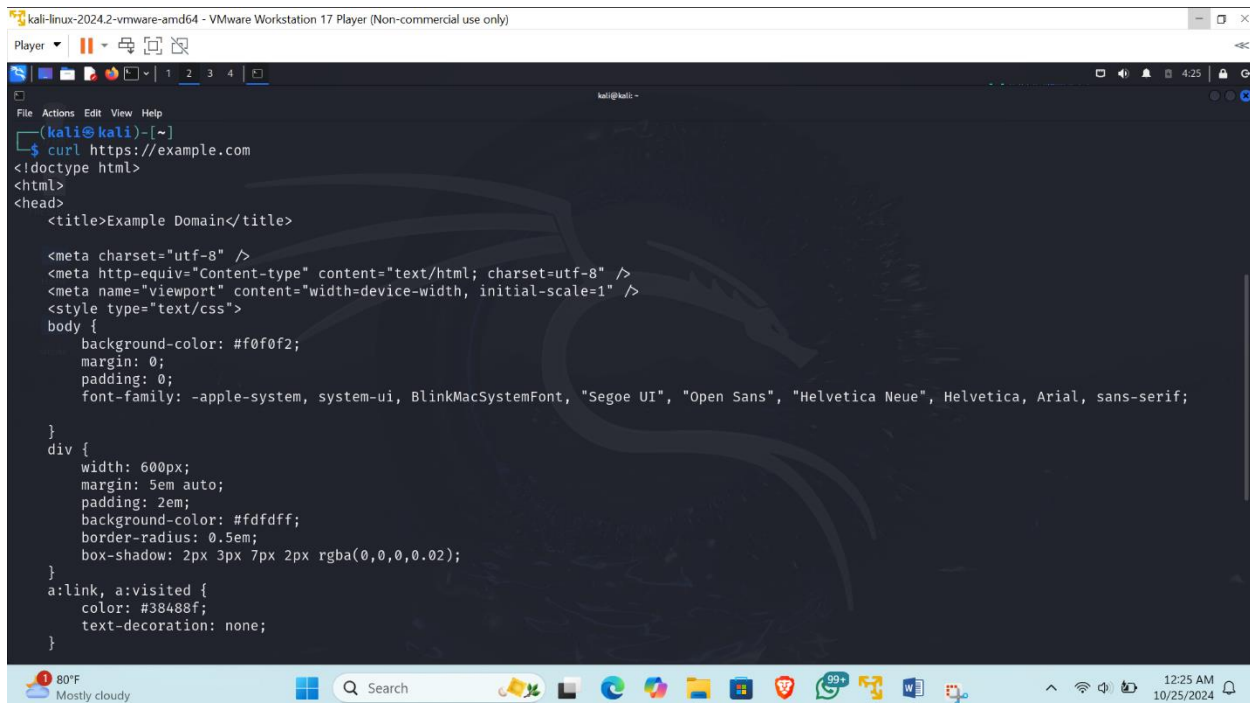
Installation of “curl” in Linux using this command: **sudo apt-get install curl**



```
kali@kali:~$ sudo apt-get install curl
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
curl is already the newest version (8.10.1-2).
curl set to manually installed.
The following packages were automatically installed and are no longer required:
 fonts-liberation2 freerdp2-x11 hydra-gtk ibverbs-providers libassuan0 libavfilter9 libboost-iostreams1.83.0 libboost-thread1.83.0 libcephfs2
 libdaxctl1 libfreerdp-client2-2t64 libfreerdp2-2t64 libgail-common libgail18t64 libgeos3.12.1t64 libgeos3.12.2 libgfsapi0 libgfrpc0 libgfxdr0
 libglusterfs0 libgspell-1-2 libgtk2.0-0t64 libgtk2.0-bin libgtk2.0-common libibverbs1 libimobiledevice6 libiniparser1 libjim0.82t64
 libjsoncpp25 libjxl0.7 libmfx1 libndctl6 libplacebo338 libplist3 libpmem1 libpostproc57 libpython3.11-dev librados2 librav1e0 librdmacm1t64
 libre2-10 libroc0.3 libsvtav1enc1d1 libu2f-udev libusbmuxd6 libwinpr2-2t64 libx265-199 linux-image-6.6.15-amd64 openjdk-17-jre
 openjdk-17-jre-headless python3-diskcache python3-hatch-vcs python3-hatchling python3-lib2to3 python3-mistune0 python3-pathspect
 python3-pendulum python3-pluggy python3-pytzdata python3-setuputils-scm python3-trove-classifiers python3.11 python3.11-dev
 python3.11-minimal rwho rhod samba-ad-provision samba-dsdb-modules samba-vfs-modules
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 234 not upgraded.
```

### TASK 2

We are getting the source code of a site by typing “curl <https://example.com>”



```
kali@kali:~$ curl https://example.com
<!doctype html>
<html>
<head>
<title>Example Domain</title>

<meta charset="utf-8" />
<meta http-equiv="Content-type" content="text/html; charset=utf-8" />
<meta name="viewport" content="width=device-width, initial-scale=1" />
<style type="text/css">
body {
background-color: #f0f0f2;
margin: 0;
padding: 0;
font-family: -apple-system, system-ui, BlinkMacSystemFont, "Segoe UI", "Open Sans", "Helvetica Neue", Helvetica, Arial, sans-serif;
}
div {
width: 600px;
margin: 5em auto;
padding: 2em;
background-color: #fdfdff;
border-radius: 0.5em;
box-shadow: 2px 3px 7px 2px rgba(0,0,0,0.02);
}
a:link, a:visited {
color: #38488f;
text-decoration: none;
}
```

This are the brief statistic data on this output after typing “curl -o output.txt <https://example.com>” to save your output below

```
(kali㉿kali)-[~]
$ curl -o output.txt https://example.com
```

% Total	% Received	% Xferd	Average Speed		Time		Time		Current
			Dload	Upload	Total	Spent	Left	Speed	
100	1256	100	1256	0	0	0:00:01	0:00:01	--:--:-- 1201	

## TASK 3

Curl also provides you with the ability to download multiple files at once. To do this, use multiple -O options, followed by the URL of the file you want to download. For example:

**curl -O <https://arxiv.org/ftp/arxiv/papers/1610/1610.05971.pdf> -O**

```
(kali㉿kali)-[~]
$ curl -O https://arxiv.org/ftp/arxiv/papers/1610/1610.05971.pdf -O https://arxiv.org/pdf/2103.08624.pdf
```

% Total	% Received	% Xferd	Average Speed		Time		Time		Current
			Dload	Upload	Total	Spent	Left	Speed	
100	846k	100	846k	0	0	0:00:01	0:00:01	--:--:-- 538k	

% Total	% Received	% Xferd	Average Speed		Time		Time		Current
			Dload	Upload	Total	Spent	Left	Speed	
100	249	100	249	0	0	--:--:--	--:--:--	--:--:-- 905	

## TASK 4

When you want to resume download, you will make use of this command: “**curl -C- -O <https://arxiv.org/pdf/2103.08624.pdf>**”

```
(kali㉿kali)-[~]
$ curl -C- -O https://arxiv.org/pdf/2103.08624.pdf
** Resuming transfer from byte position 249
```

% Total	% Received	% Xferd	Average Speed		Time		Time		Current
			Dload	Upload	Total	Spent	Left	Speed	
0	249	0	0	0	0	--:--:--	--:--:--	--:--:-- 0	

When testing a site, **curl** is useful for downloading HTTP header by typing this “**curl -I <https://example.com>**”

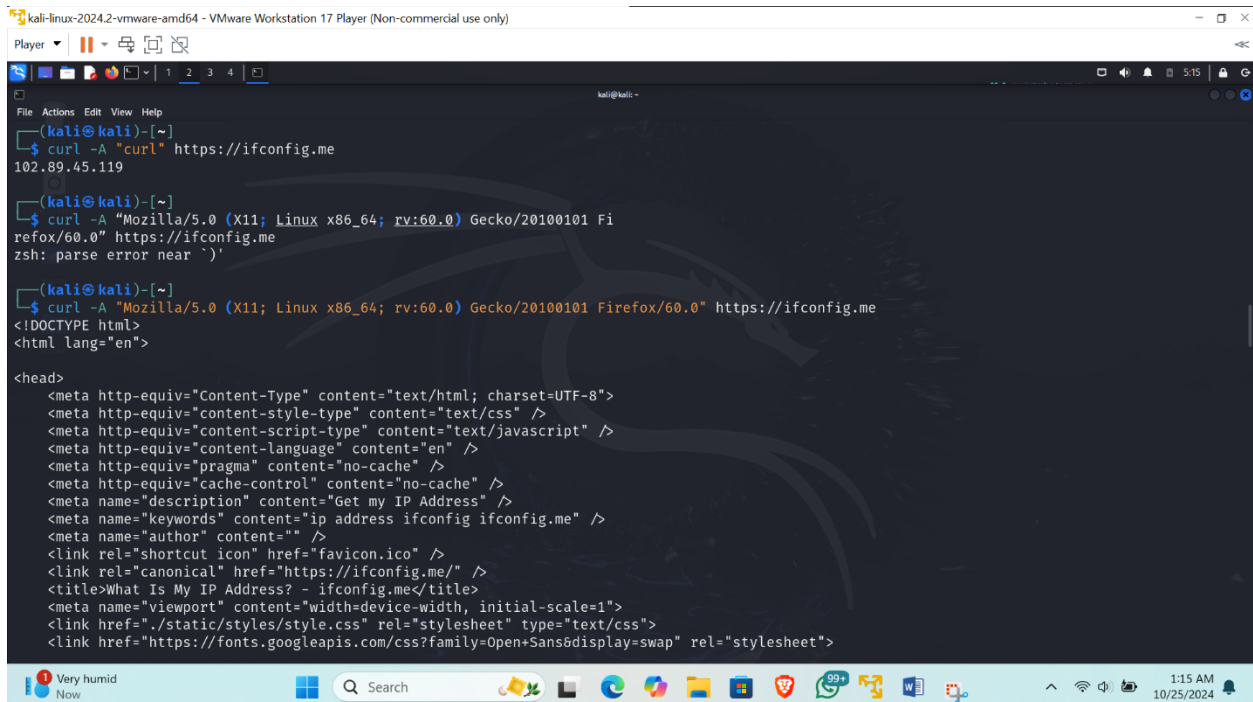
This will display many useful pieces of information, such as server info, content type, and content encoding.

```
(kali㉿kali)-[~]
$ curl -I https://example.com
HTTP/2 200
content-encoding: gzip
accept-ranges: bytes
age: 17845
cache-control: max-age=604800
content-type: text/html; charset=UTF-8
date: Thu, 24 Oct 2024 23:56:08 GMT
etag: "3147526947+gzip"
expires: Thu, 31 Oct 2024 23:56:08 GMT
last-modified: Thu, 17 Oct 2019 07:18:26 GMT
server: ECAcc (dcd/7D82)
x-cache: HIT
content-length: 648
```

## TASK 5

When attempting to download a file or gather other information using curl, you may discover that the target site may be designed to block curl. In this case, it is useful to emulate a browser, such as Firefox, to return the information you are looking for. To do this, use the following command:

**curl -A "Mozilla/5.0 (X11; Linux x86\_64; rv:60.0) Gecko/20100101 Firefox/60.0"**  
<https://ifconfig.me>



```
kali-linux-2024.2-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)
Player
File Actions Edit View Help
(kali@kali)-[~]
$ curl -A "curl" https://ifconfig.me
102.89.45.119

(kali@kali)-[~]
$ curl -A "Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0" https://ifconfig.me
zsh: parse error near `)'

(kali@kali)-[~]
$ curl -A "Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0" https://ifconfig.me
<!DOCTYPE html>
<html lang="en">

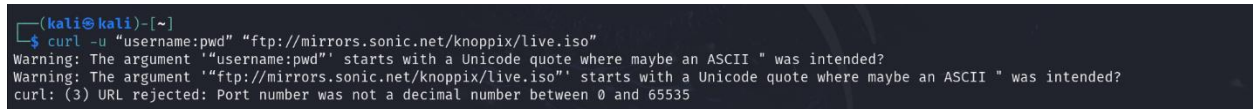
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta http-equiv="content-style-type" content="text/css" />
<meta http-equiv="content-script-type" content="text/javascript" />
<meta http-equiv="content-language" content="en" />
<meta http-equiv="pragma" content="no-cache" />
<meta http-equiv="cache-control" content="no-cache" />
<meta name="description" content="Get my IP Address" />
<meta name="keywords" content="ip address ifconfig ifconfig.me" />
<meta name="author" content="" />
<link rel="shortcut icon" href="favicon.ico" />
<link rel="canonical" href="https://ifconfig.me/" />
<title>What Is My IP Address? - ifconfig.me</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="/static/styles/style.css" rel="stylesheet" type="text/css">
<link href="https://fonts.googleapis.com/css?family=Open+Sans&display=swap" rel="stylesheet">
```

In this example, remote site <https://ifconfig.me> answers with different messages according to clients' user-agent strings.

## TASK 6

To access a protected FTP server, use the -u option to specify the username and password:

**curl -u "username:pwd" <ftp://mirrors.sonic.net/knoppix/live.iso>**



```
(kali@kali)-[~]
$ curl -u "username:pwd" "ftp://mirrors.sonic.net/knoppix/live.iso"
Warning: The argument "username:pwd" starts with a Unicode quote where maybe an ASCII " was intended?
Warning: The argument "ftp://mirrors.sonic.net/knoppix/live.iso" starts with a Unicode quote where maybe an ASCII " was intended?
curl: (3) URL rejected: Port number was not a decimal number between 0 and 65535
```

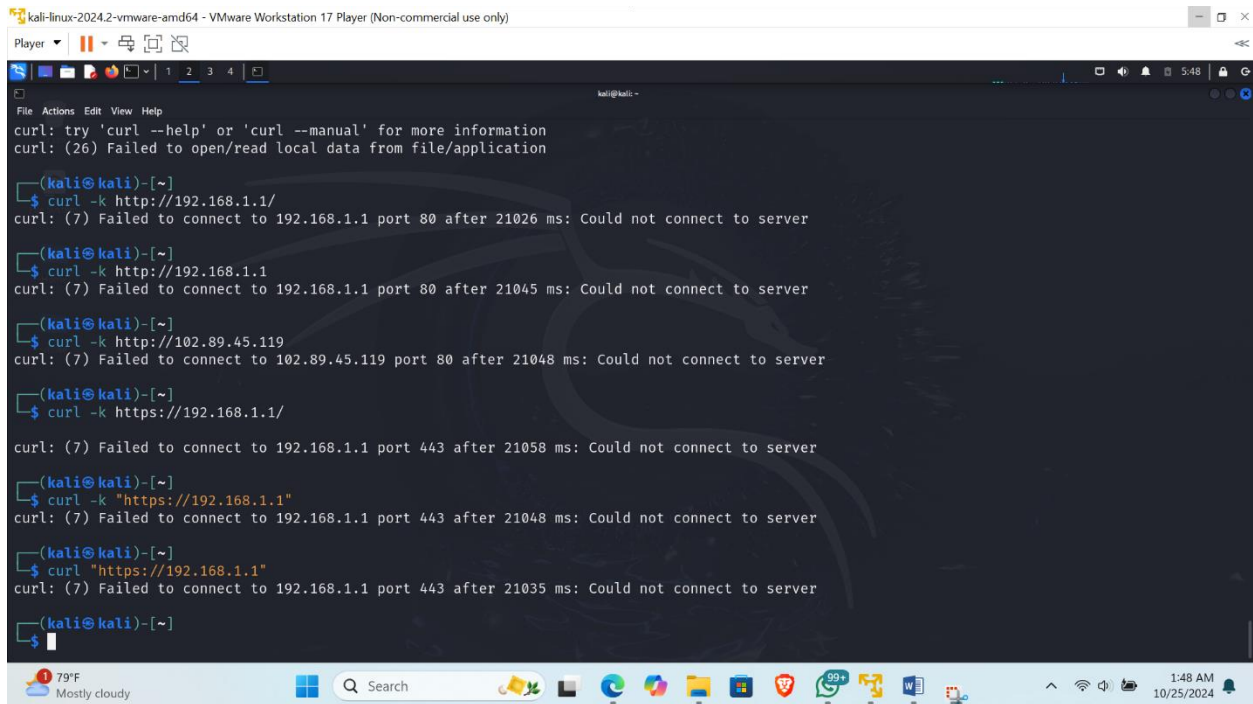
To upload a file to the server, we can use the -T option:

**curl -T file.zip -u "username:password" <ftp://mirrors.sonic.net/>**

```
(kali@kali)-[~]
$ curl -T file.zip -u "username:password" ftp://mirrors.sonic.net/
Warning: The argument 'username:password' starts with a Unicode quote where maybe an ASCII " was intended?
curl: cannot open 'file.zip'
curl: try 'curl --help' or 'curl --manual' for more information
curl: (26) Failed to open/read local data from file/application
```

## TASK 7

curl -k <http://192.168.1.1>

A screenshot of a Kali Linux terminal window. The terminal shows several curl commands being executed. The first command is curl -T file.zip -u "username:password" ftp://mirrors.sonic.net/, which fails with error (26). The second command is curl -k http://192.168.1.1/, which fails with error (7). The third command is curl -k http://192.168.1.1, which also fails with error (7). The fourth command is curl -k http://102.89.45.119, which fails with error (7). The fifth command is curl -k https://192.168.1.1/, which fails with error (7). The sixth command is curl -k "https://192.168.1.1", which fails with error (7). The seventh command is curl "https://192.168.1.1", which also fails with error (7). The terminal window has a title bar that says "kali-linux-2024.2-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)". The bottom of the window shows a taskbar with various icons and a system tray with the date and time "1:48 AM 10/25/2024".

```
kali-linux-2024.2-vmware-amd64 - VMware Workstation 17 Player (Non-commercial use only)
Player
File Actions Edit View Help
kali@kali: ~
curl: try 'curl --help' or 'curl --manual' for more information
curl: (26) Failed to open/read local data from file/application

(kali@kali)-[~]
$ curl -T file.zip -u "username:password" ftp://mirrors.sonic.net/
Warning: The argument 'username:password' starts with a Unicode quote where maybe an ASCII " was intended?
curl: cannot open 'file.zip'
curl: try 'curl --help' or 'curl --manual' for more information
curl: (26) Failed to open/read local data from file/application

(kali@kali)-[~]
$ curl -k http://192.168.1.1/
curl: (7) Failed to connect to 192.168.1.1 port 80 after 21026 ms: Could not connect to server

(kali@kali)-[~]
$ curl -k http://192.168.1.1
curl: (7) Failed to connect to 192.168.1.1 port 80 after 21045 ms: Could not connect to server

(kali@kali)-[~]
$ curl -k http://102.89.45.119
curl: (7) Failed to connect to 102.89.45.119 port 80 after 21048 ms: Could not connect to server

(kali@kali)-[~]
$ curl -k https://192.168.1.1/
curl: (7) Failed to connect to 192.168.1.1 port 443 after 21058 ms: Could not connect to server

(kali@kali)-[~]
$ curl -k "https://192.168.1.1"
curl: (7) Failed to connect to 192.168.1.1 port 443 after 21048 ms: Could not connect to server

(kali@kali)-[~]
$ curl "https://192.168.1.1"
curl: (7) Failed to connect to 192.168.1.1 port 443 after 21035 ms: Could not connect to server

(kali@kali)-[~]
$
```

## TASK 8

Curl can also be configured to use a proxy. To do this, use the -x option followed by the proxy URL. For example:

curl -x 192.168.0.1:8080 <http://example.com/>