CMP5329 Cyber Security

Cyber Security Coursework

Name: Kacper Popis

ID: 23161791

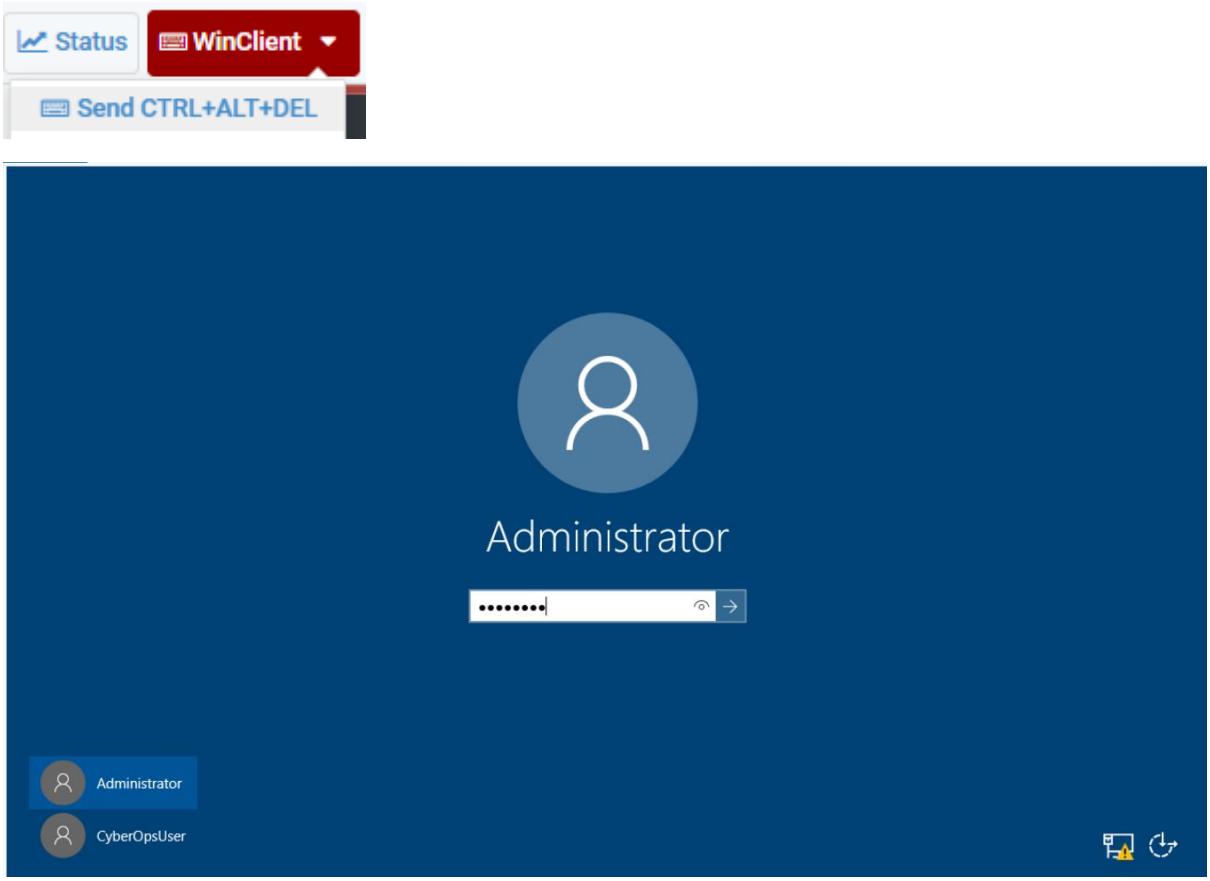
Contents

Module 7: Windows Labs

7.2.11 Lab - Exploring Processes, Threads, Handles, and Windows Registry

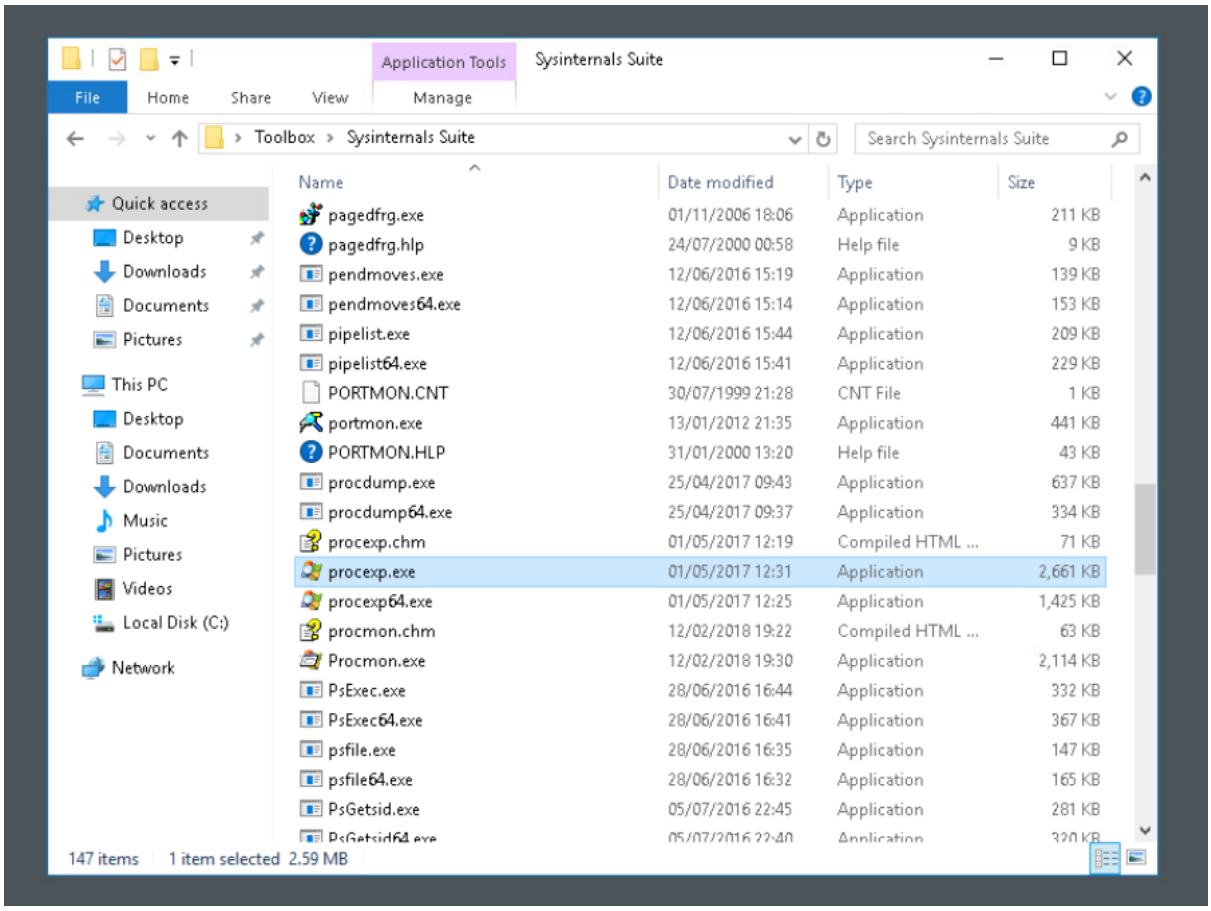
Part 1

Step 1

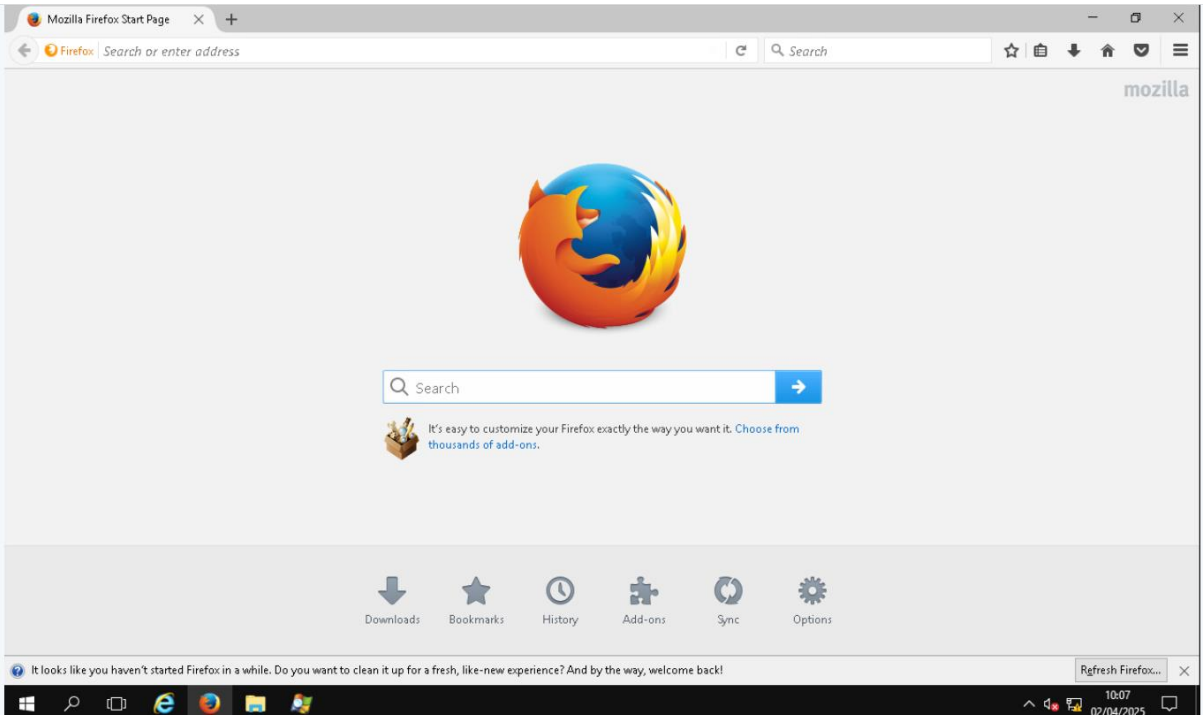


Accessing the VM on netlab.

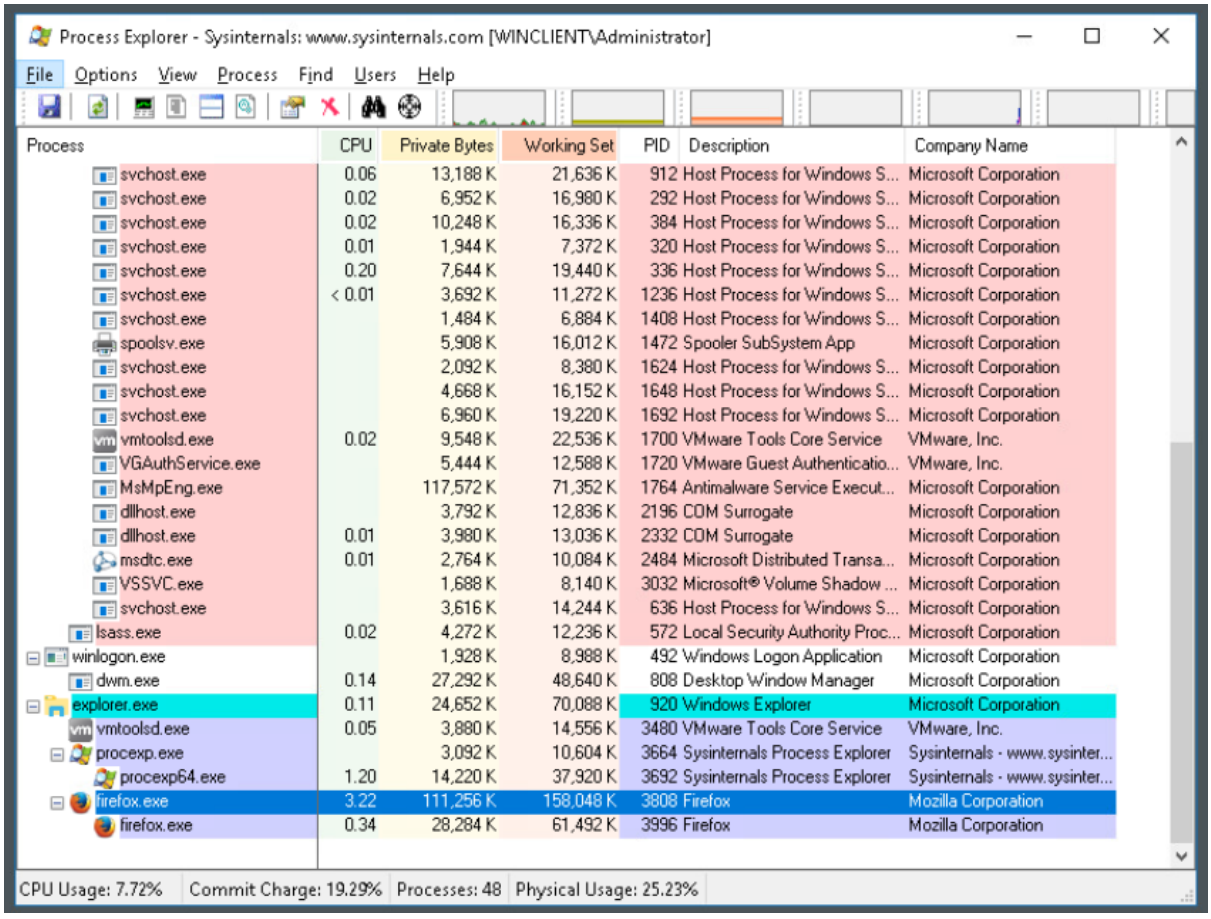
Step 2



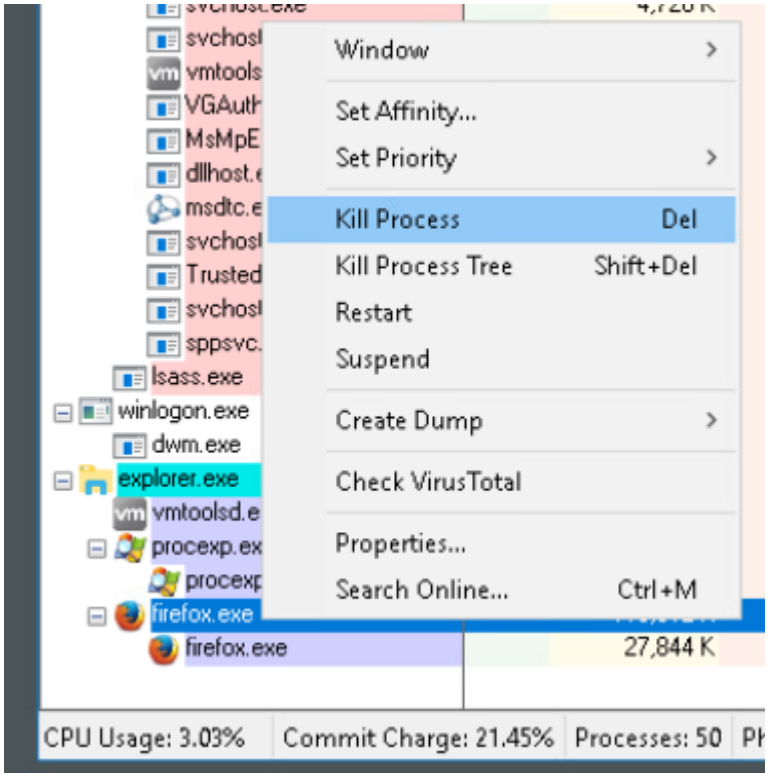
Finding and locating the **Sysinternals Suite** folder on the desktop and locating the **procexp.exe** inside of the folder and starting it, then accepting the licence agreement to be able to use it.



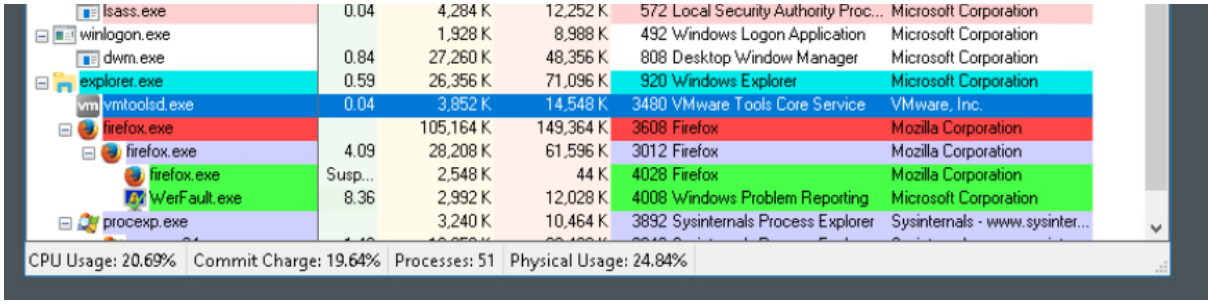
Opening **Firefox** browser to be used in this task.



Using the **Find Window's Processes** (inside process explorer) by drag and dropping it onto the **Firefox** icon in the taskbar, it will locate the process inside of the process list



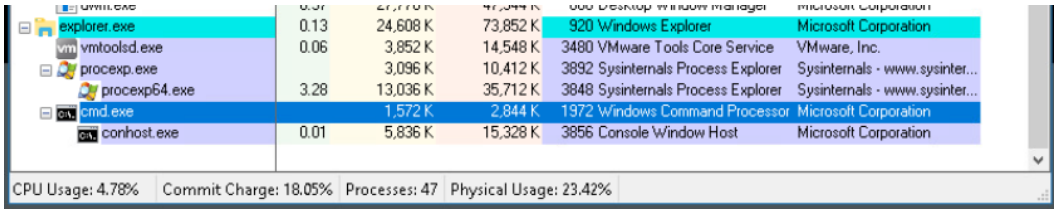
Right clicking the **firefox.exe** process will open a menu where the process can be terminated.



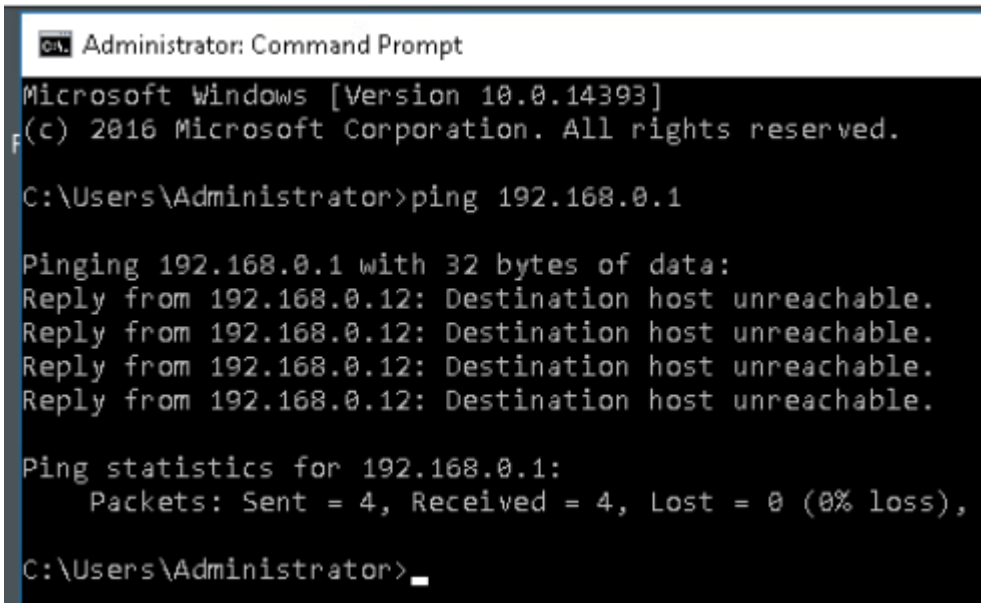
After killing the process, the executable gets suspended, progressively stops using processing power (CPU usage as an example) and eventually turns red before disappearing from the process list.

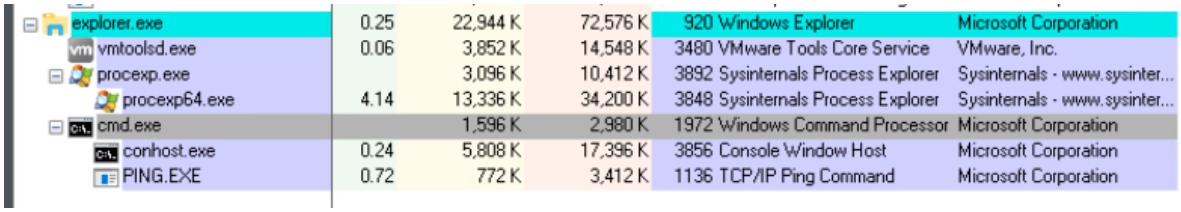
Step 3

Opening **Command Prompt** application from search bar.



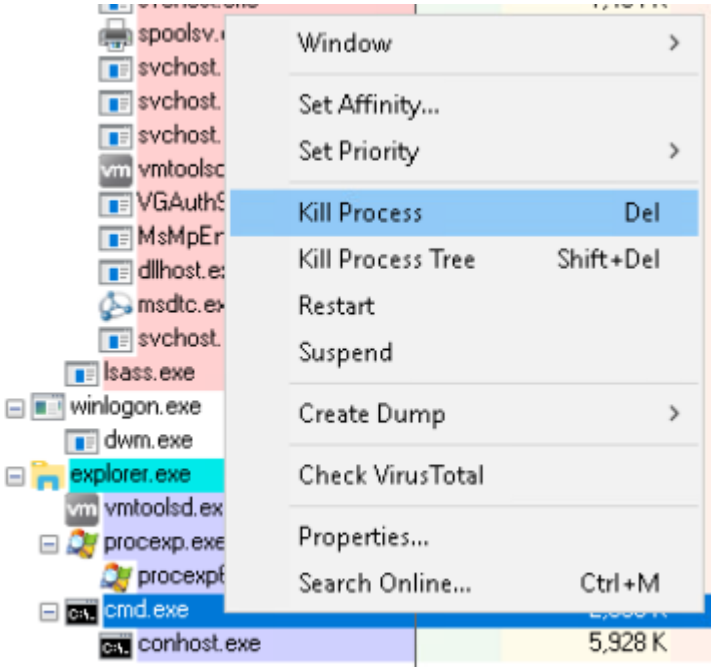
Using the **Find Window's Processes** (inside process explorer) by drag and dropping it onto the **Command Prompt** icon in the taskbar, it will locate the process inside of the process list

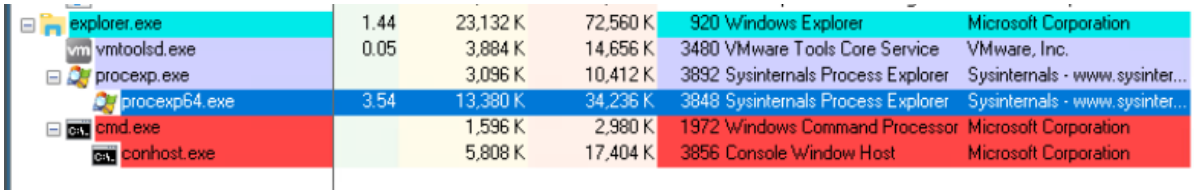




explorer.exe	0.25	22,944 K	72,576 K	920 Windows Explorer	Microsoft Corporation
vmtoolsd.exe	0.06	3,852 K	14,548 K	3480 VMware Tools Core Service	VMware, Inc.
procexp.exe		3,096 K	10,412 K	3892 Sysinternals Process Explorer	Sysinternals - www.sysinter...
procexp64.exe	4.14	13,336 K	34,200 K	3848 Sysinternals Process Explorer	Sysinternals - www.sysinter...
cmd.exe		1,596 K	2,980 K	1972 Windows Command Processor	Microsoft Corporation
conhost.exe	0.24	5,808 K	17,396 K	3856 Console Window Host	Microsoft Corporation
PING.EXE	0.72	772 K	3,412 K	1136 TCP/IP Ping Command	Microsoft Corporation

After typing the **ping** command with an appropriate IP address (in this case its **192.168.0.**1) in **cmd** your device will send a total of 4 packets to the desired IP address and will give a breakdown of how many packets were sent, received and lost. Additonally when executing the **ping** command, in the **Process Explorer** a child process will appear called **PING.EXE**.

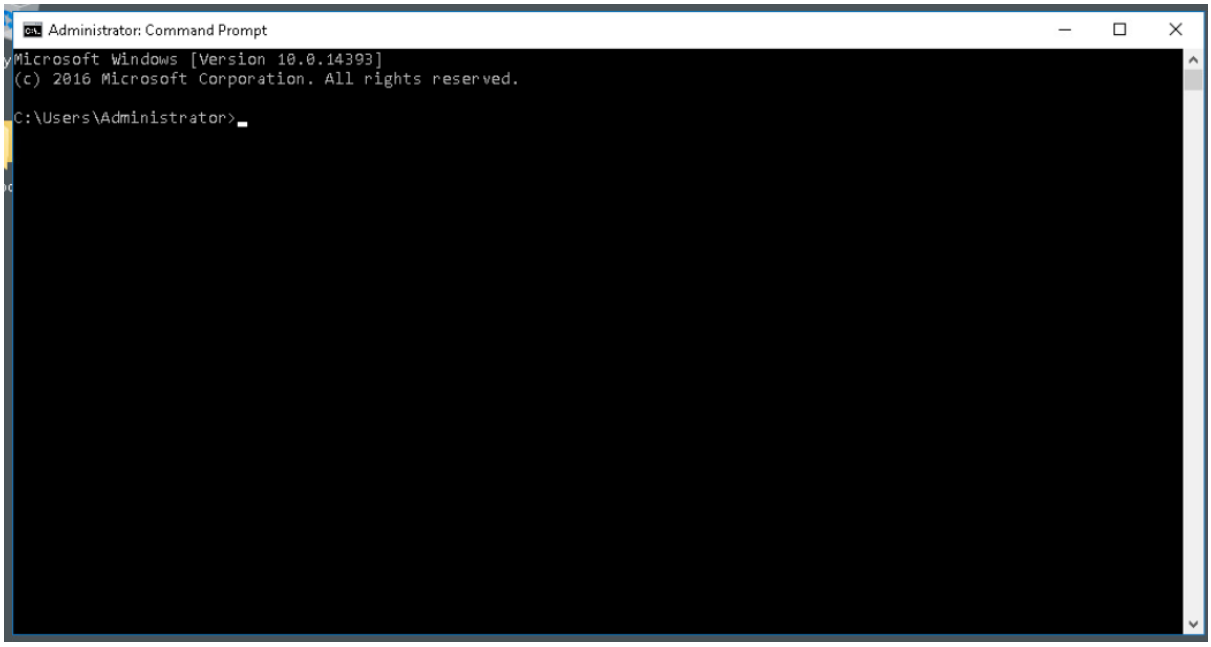




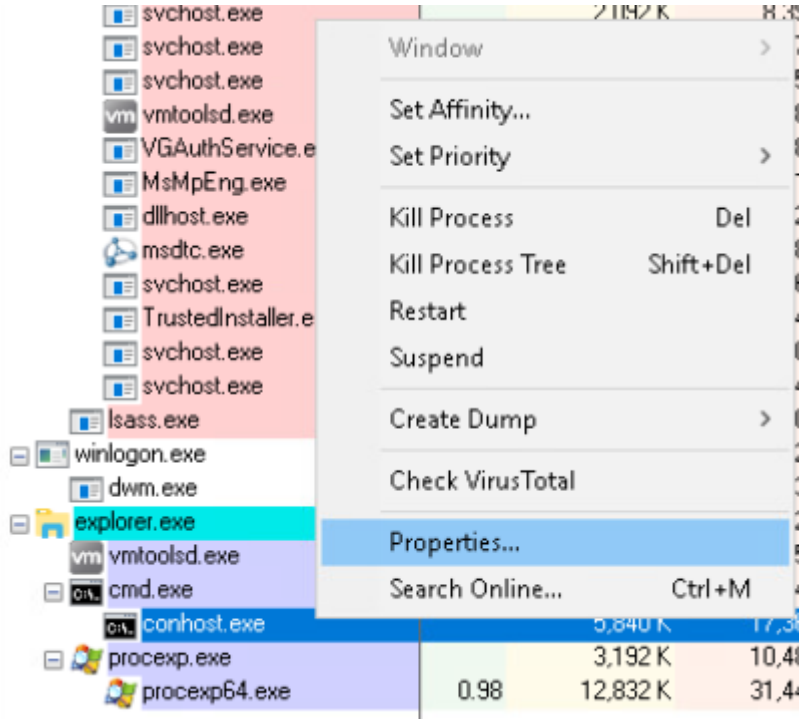
After killing the process, the executable progressively stops using processing power (CPU usage as an example) and eventually turns red before disappearing from the process list as it has been terminated.

Part 2

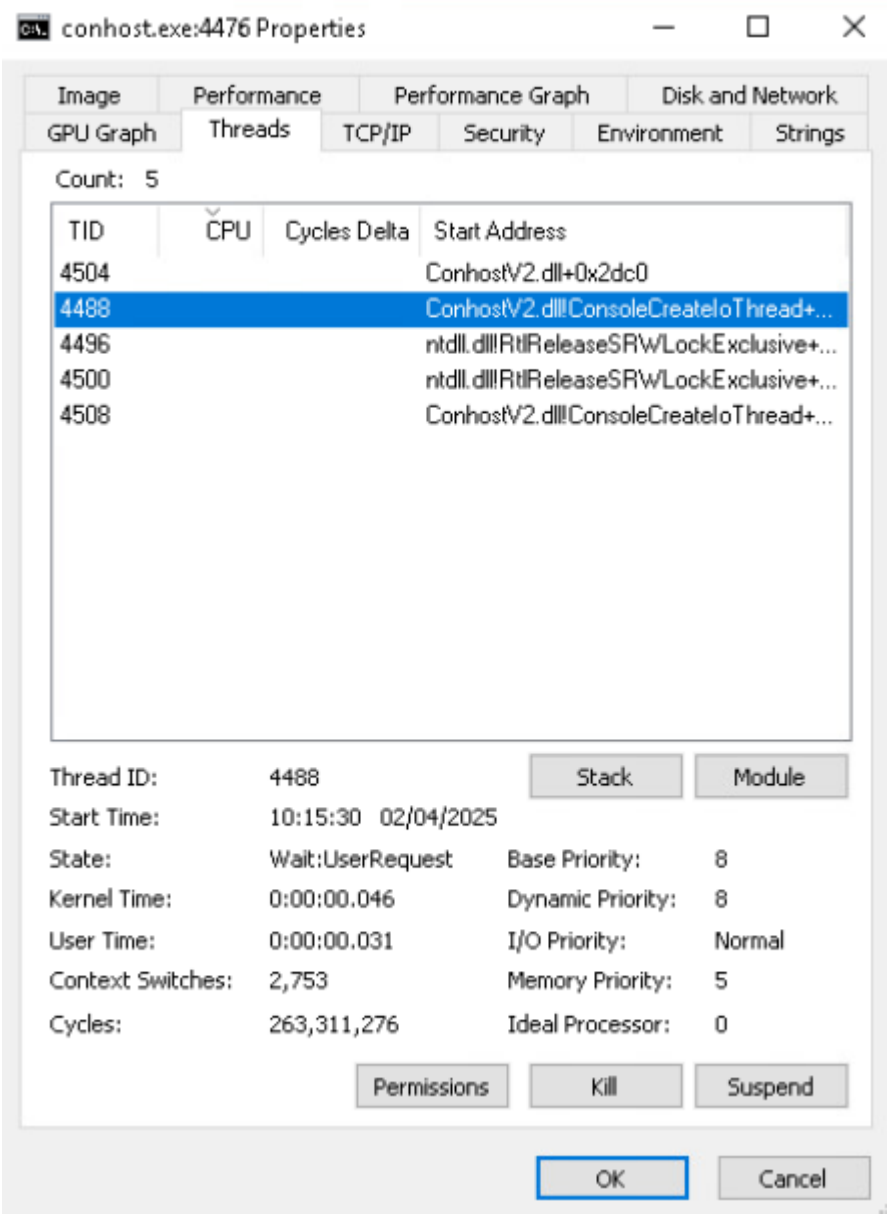
Step 1

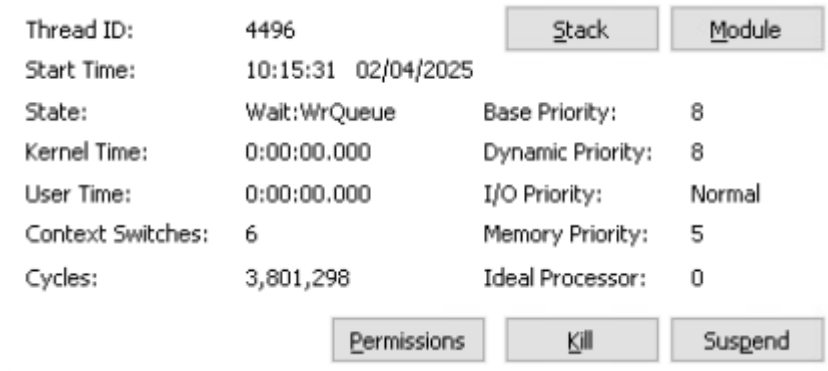


Opening a fresh **Command Prompt**



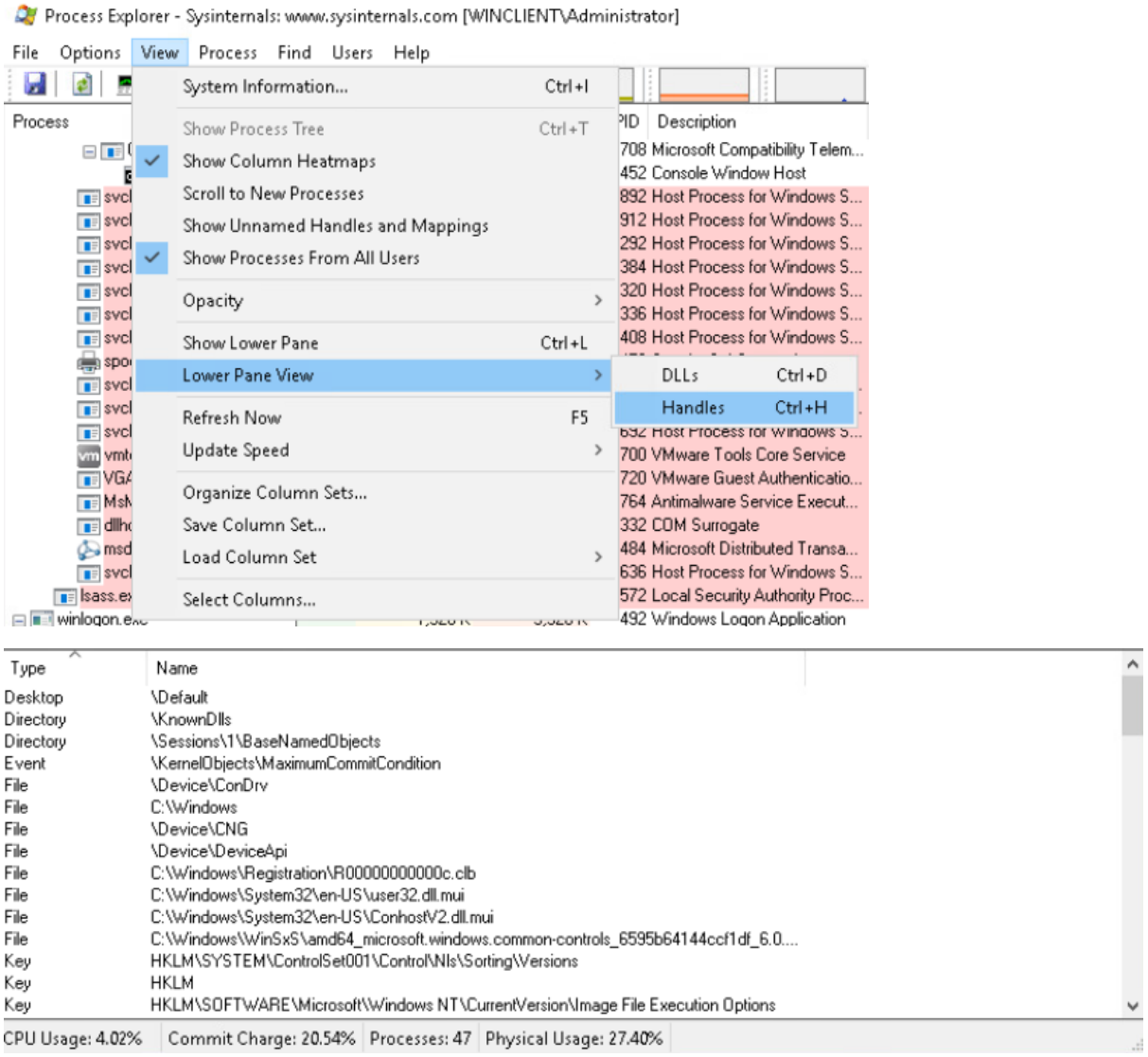
Tabbing back into the **Process Explorer**, finding the **cmd** process using the **Find Window's Processes** tool and expanding it to be able to see **conhost.exe** process. After right clicking it and clicking on properties a property window will open.





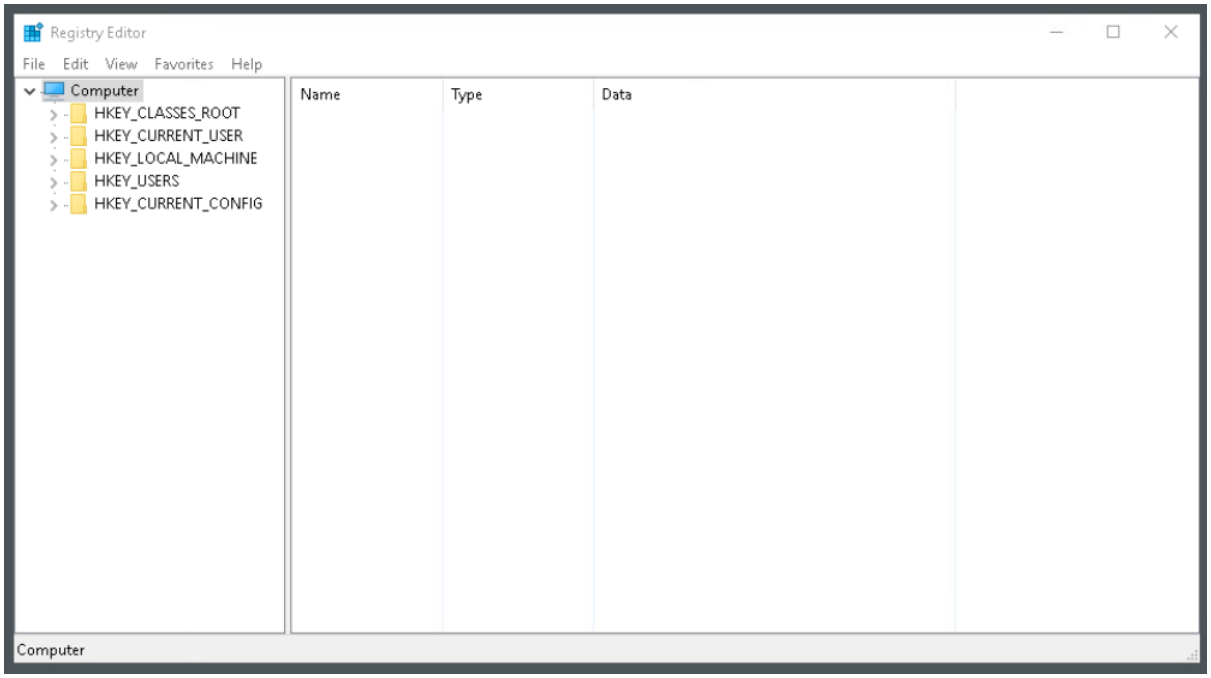
Inside the **Thread** tab each thread is selectable and once selected it will different information about each one (like memory priority and state)

Step 2

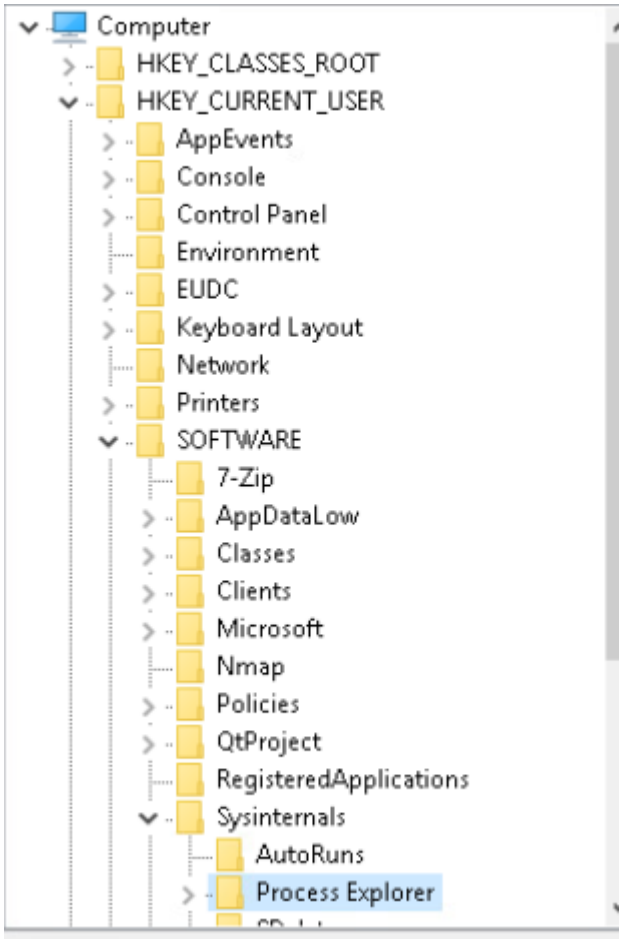


Opening the **Handles** panel allows to see all associated files that are used on the process (in this example its for **cmd**).

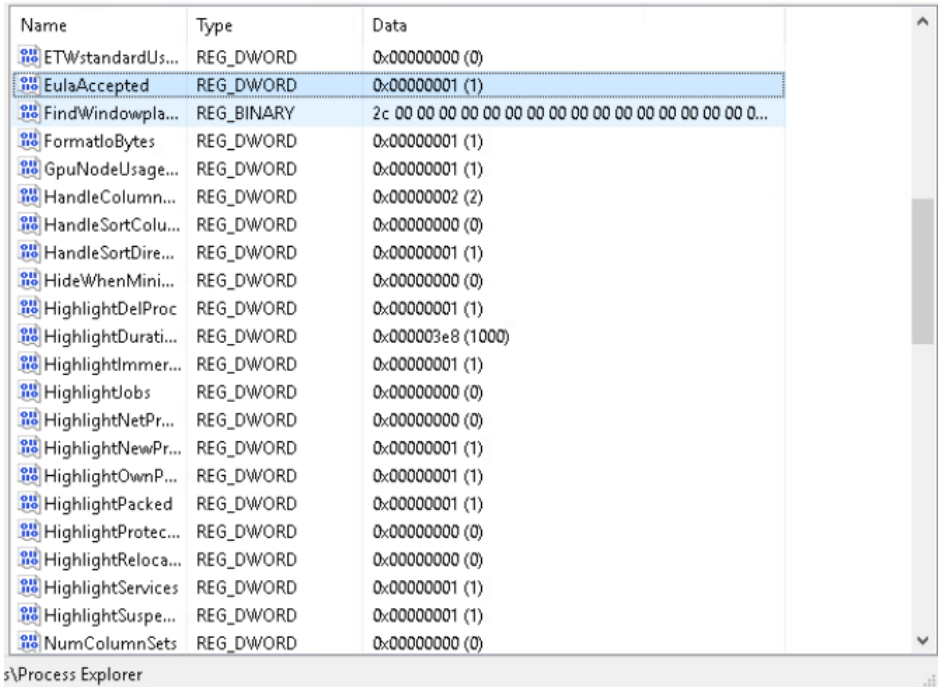
Part 3



Opening **Registry Editor**



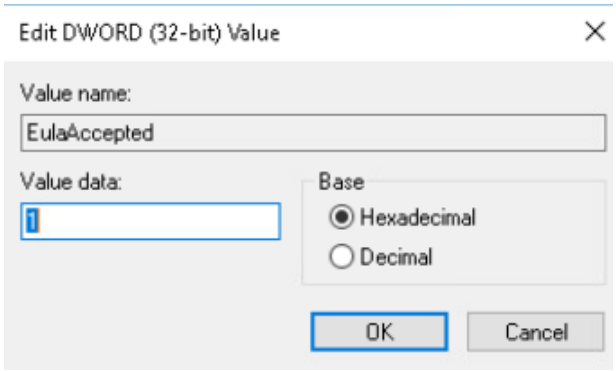
Following and expanding the hierarchy to locate the **Process Explorer's** configuration

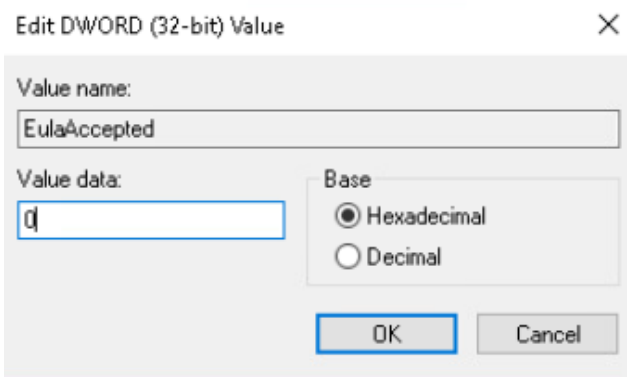


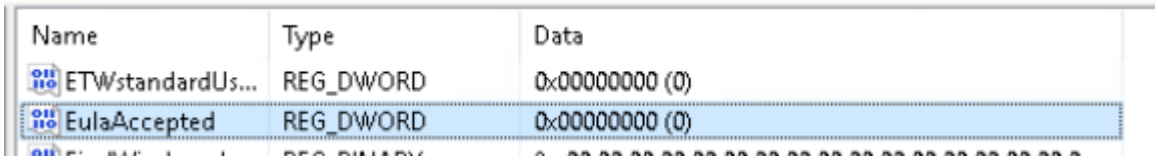
Name	Type	Data
ETWstandardUs...	REG_DWORD	0x00000000 (0)
EulaAccepted	REG_DWORD	0x00000001 (1)
FindWindowpla...	REG_BINARY	2c 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 0...
FormatIoBytes	REG_DWORD	0x00000001 (1)
GpuNodeUsage...	REG_DWORD	0x00000001 (1)
HandleColumn...	REG_DWORD	0x00000002 (2)
HandleSortColu...	REG_DWORD	0x00000000 (0)
HandleSortDire...	REG_DWORD	0x00000001 (1)
HideWhenMini...	REG_DWORD	0x00000000 (0)
HighlightDelProc	REG_DWORD	0x00000001 (1)
HighlightDurati...	REG_DWORD	0x000003e8 (1000)
HighlightImmer...	REG_DWORD	0x00000001 (1)
HighlightJobs	REG_DWORD	0x00000000 (0)
HighlightNetPr...	REG_DWORD	0x00000000 (0)
HighlightNewPr...	REG_DWORD	0x00000001 (1)
HighlightOwnP...	REG_DWORD	0x00000001 (1)
HighlightPacked	REG_DWORD	0x00000001 (1)
HighlightProtec...	REG_DWORD	0x00000000 (0)
HighlightReloca...	REG_DWORD	0x00000000 (0)
HighlightServices	REG_DWORD	0x00000001 (1)
HighlightSuspe...	REG_DWORD	0x00000001 (1)
NumColumnSets	REG_DWORD	0x00000000 (0)

s\Process Explorer

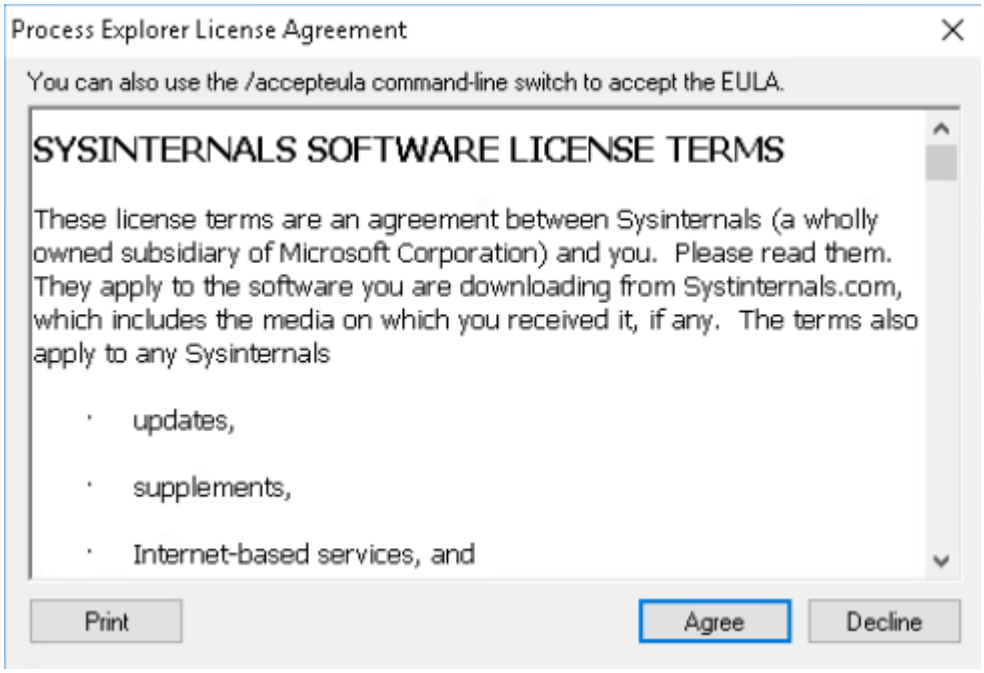
Highlighted is the **key** value behind the condition of "**EulaAccepted**", and the value is 0x00000001 (1) - this prevents from the EULA window opening every time when opening the application







This key's value can be edited, and for this task it will be changed to the value **0**, and saved by pressing **OK**. Once this is done the value of the **Data column** for this key will change to 0x00000000 (0), showing that the value has been changed.

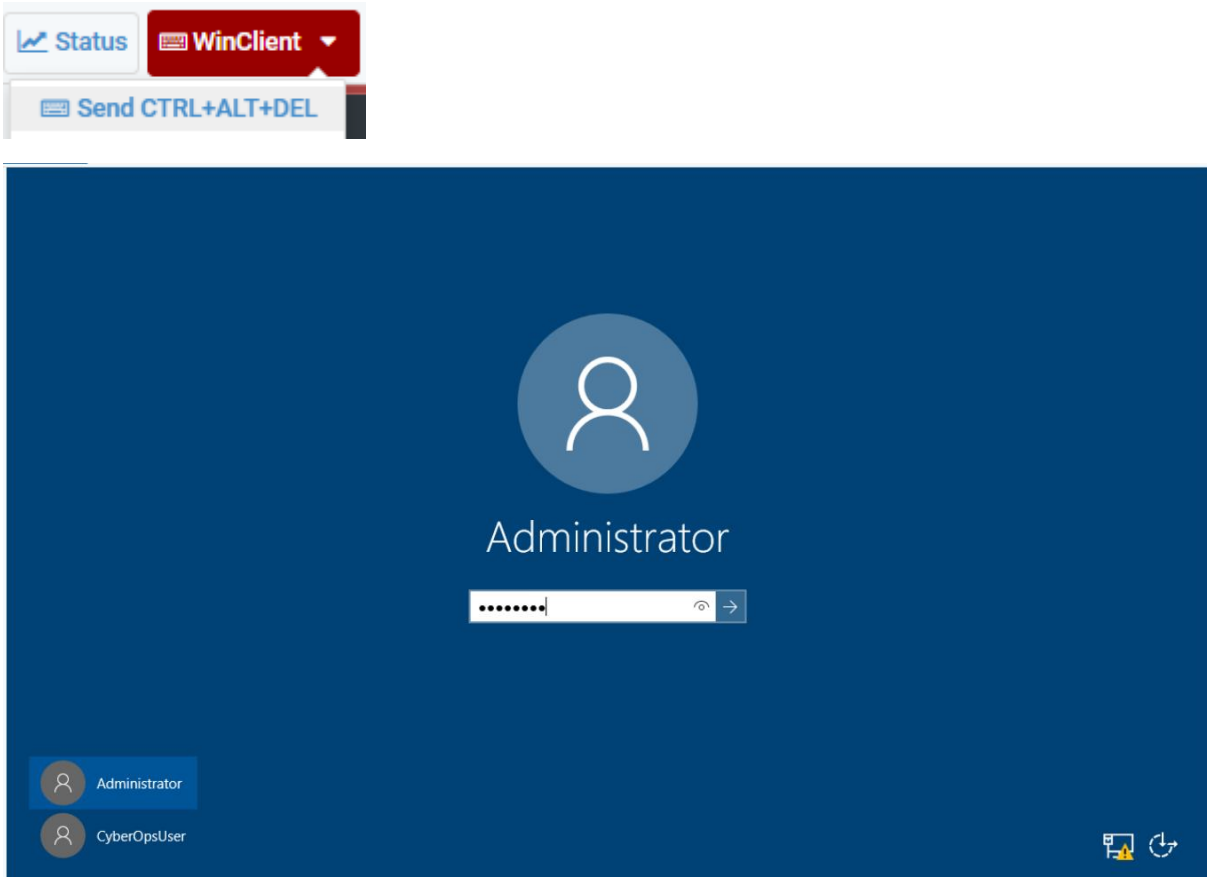


When opening the **Process Explorer** application again, the EULA window will open and will prompt the user to accept it before using the app.

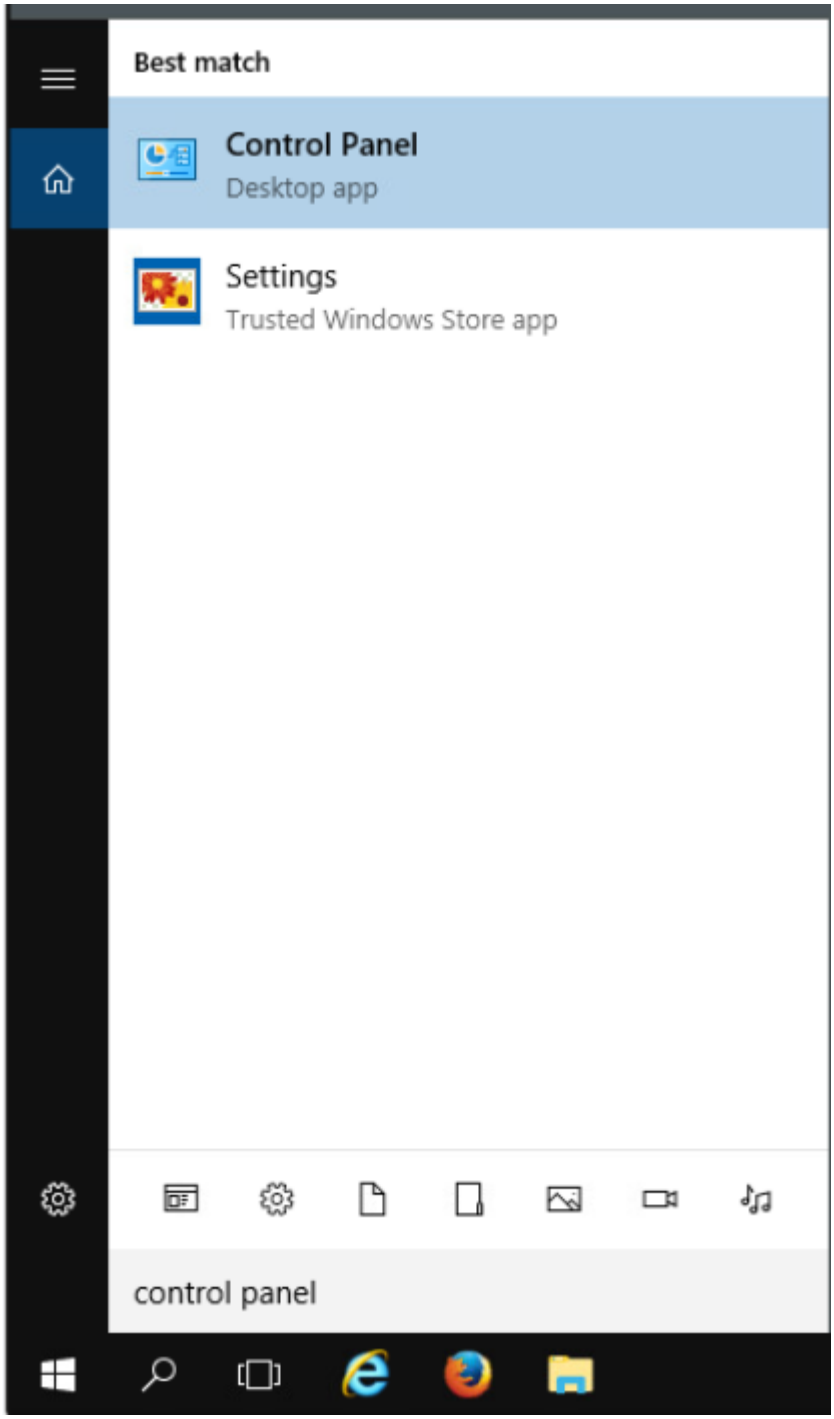
7.3.10 Lab - Create User Accounts

Part 1

Step 1

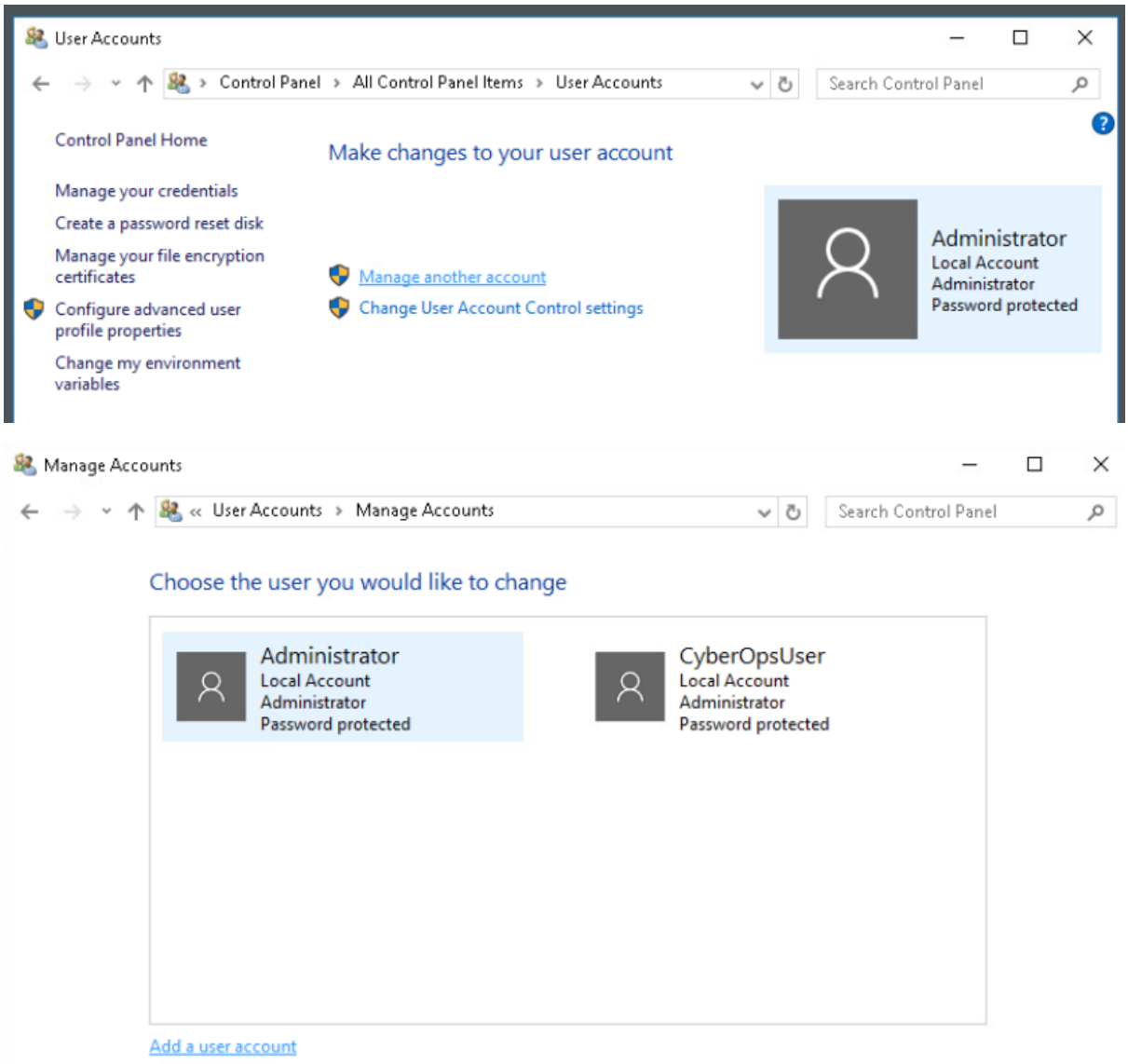


Accessing the VM on netlab

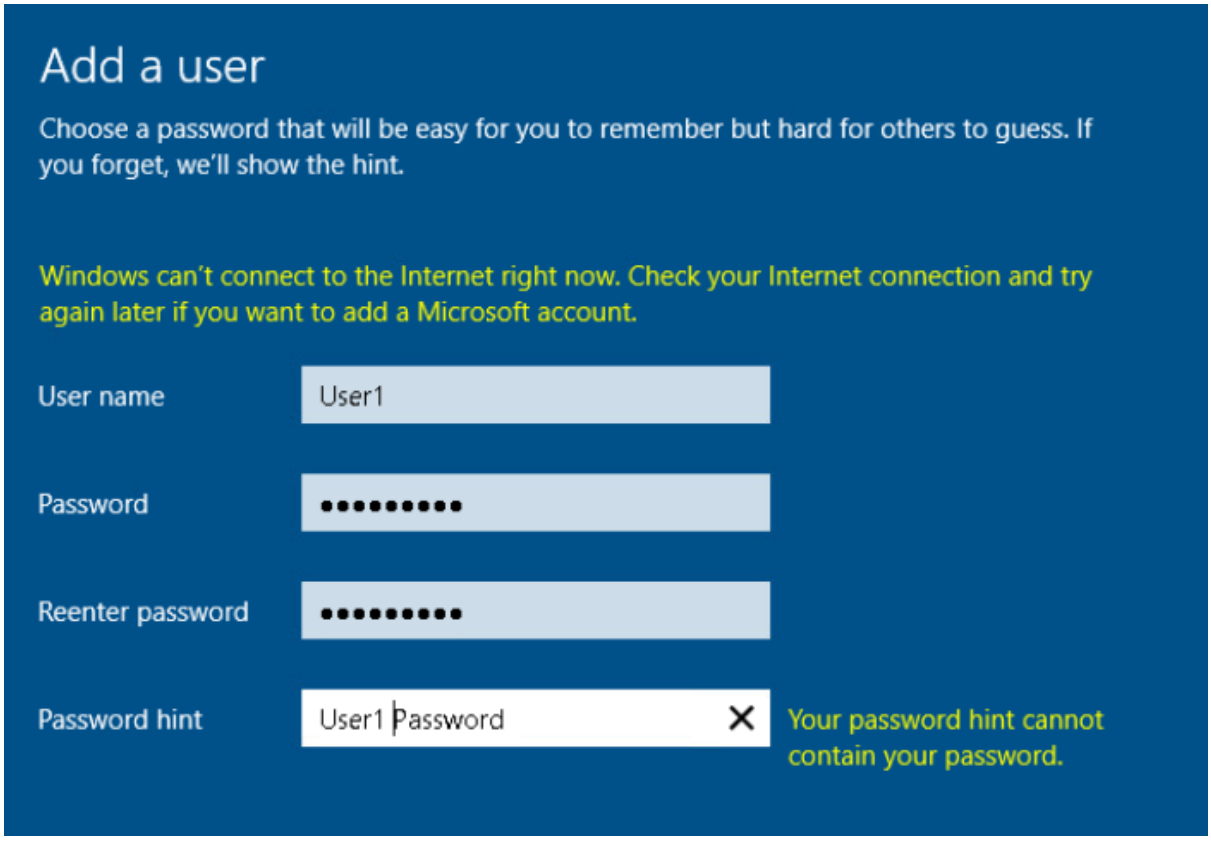


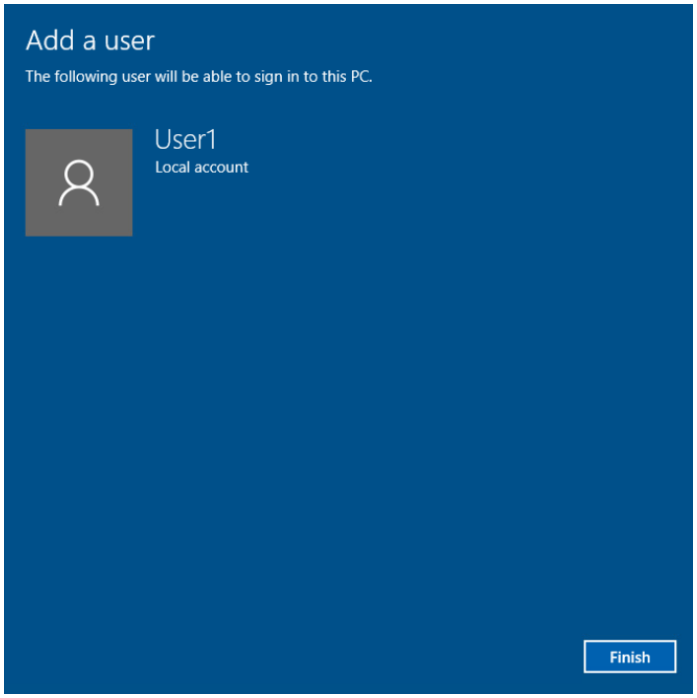
Opening **Control Panel**

Step 2

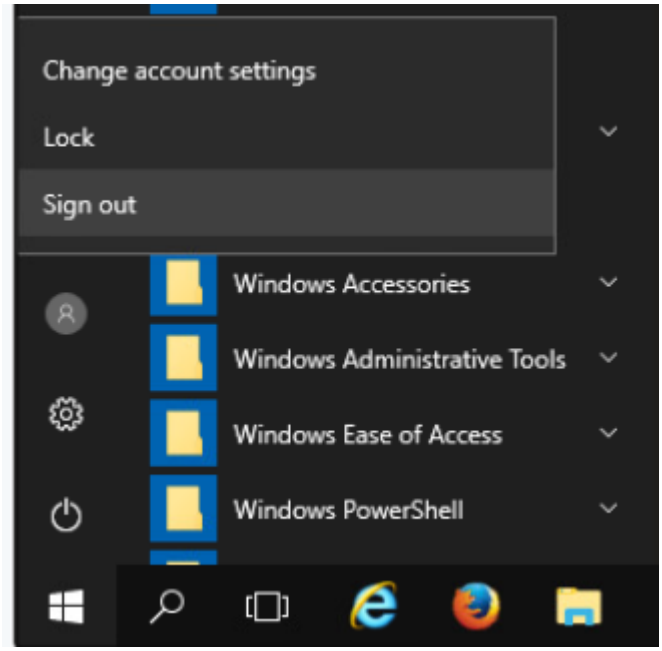


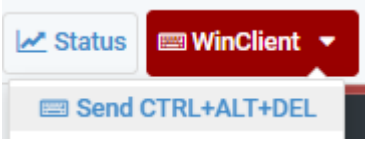
Accessing the **User Accounts** window and selecting "**Manage another account**" > "**Add a user account**"

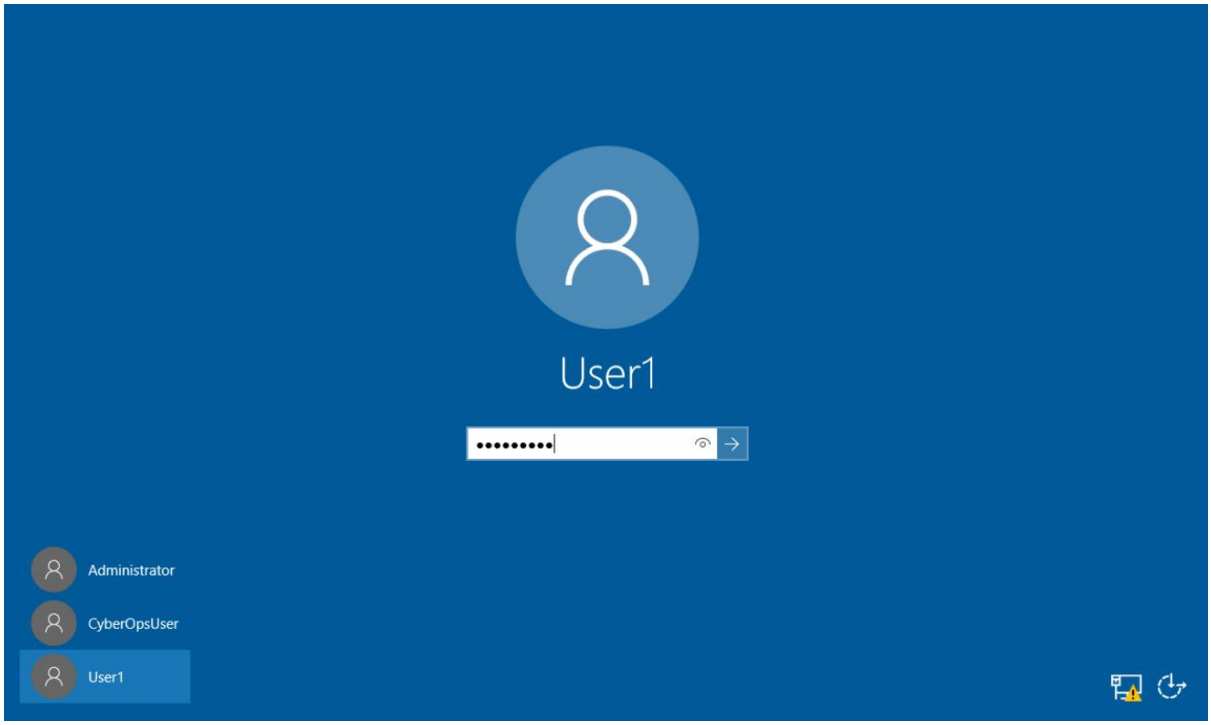




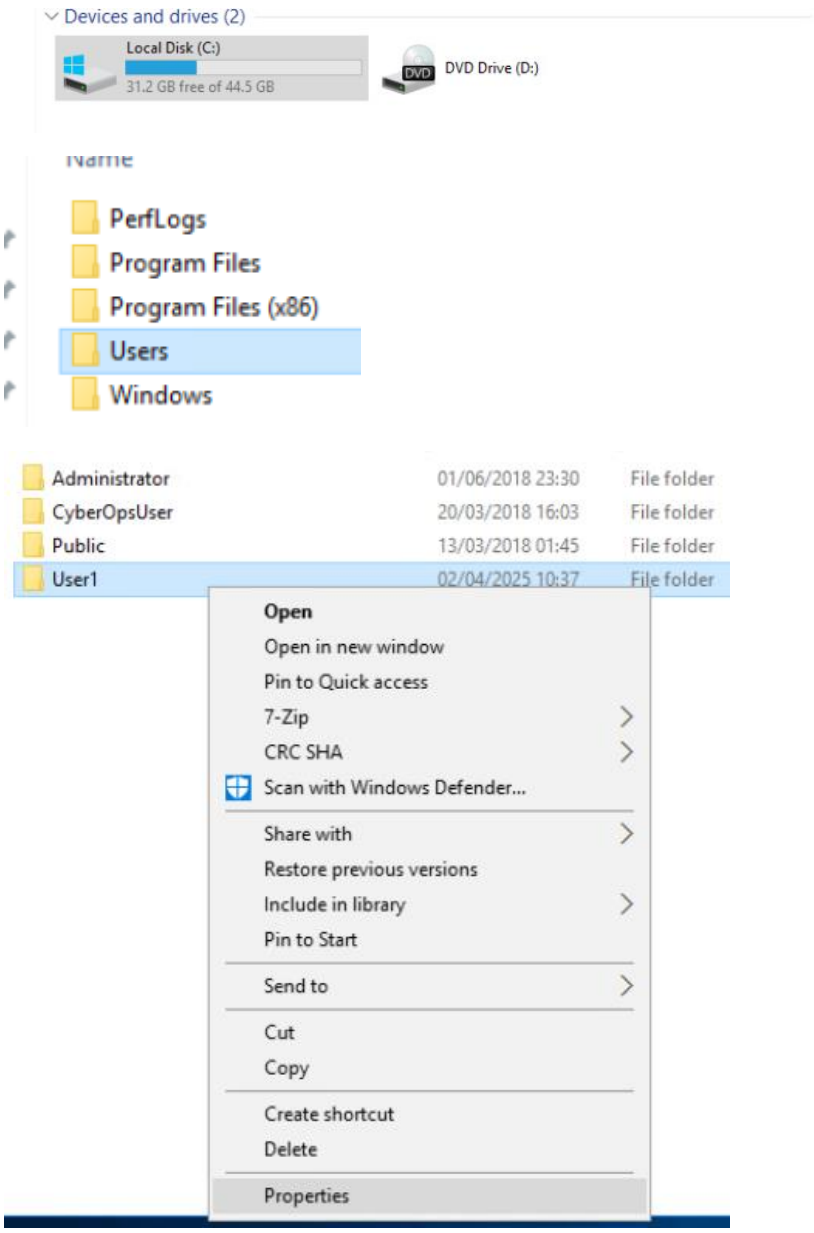
Creating a **new** user with mentioned parameters then selecting ”**Next**” > ”**Finish**”

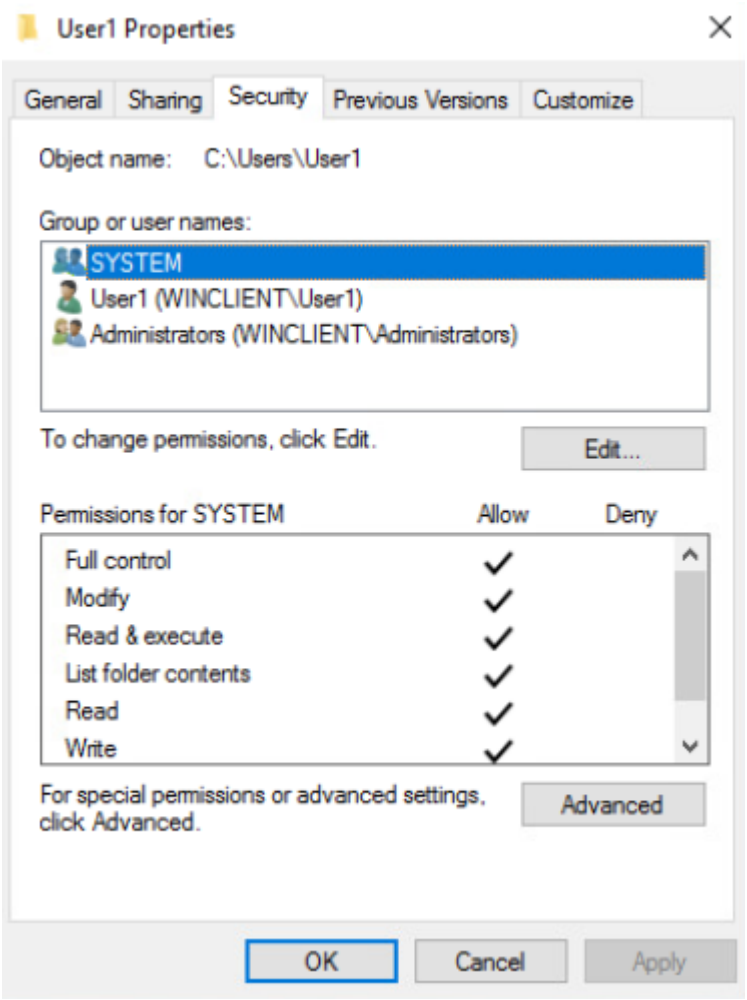




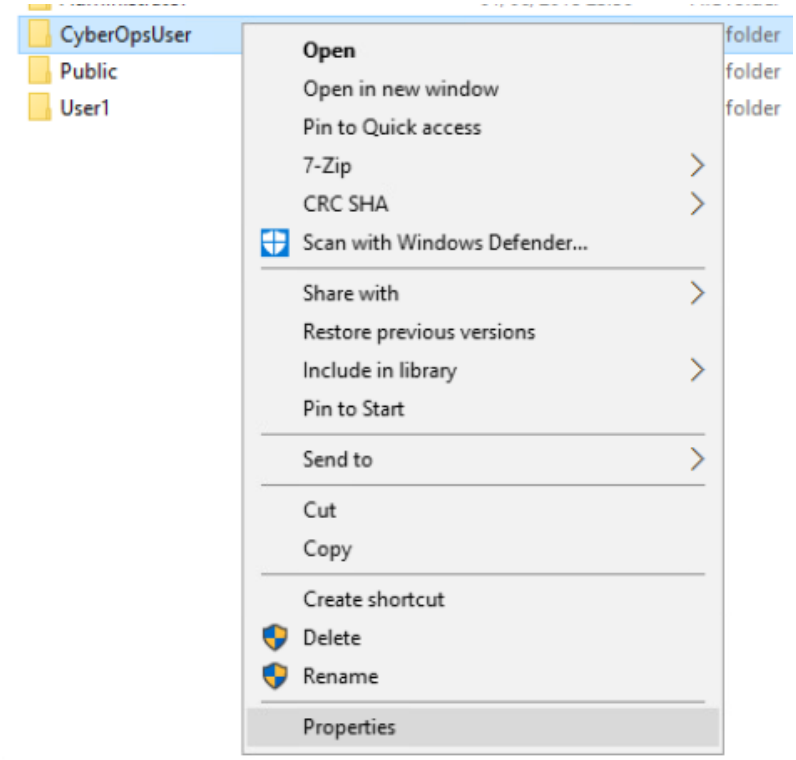


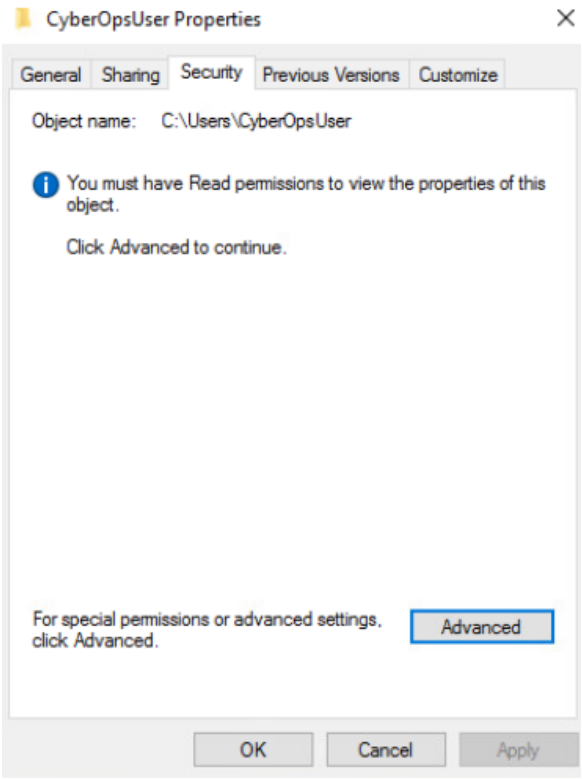
Logging out of the current account and logging into the new **User1** account


Devices and drives (2)
Local Disk (C:)
31.2 GB free of 44.5 GB
DVD Drive (D:)
Name
PerfLogs
Program Files
Program Files (x86)
Users
Windows
Administrator 01/06/2018 23:30 File folder
CyberOpsUser 20/03/2018 16:03 File folder
Public 13/03/2018 01:45 File folder
User1 02/04/2025 10:37 File folder
Open
Open in new window
Pin to Quick access
7-Zip
CRC SHA
Scan with Windows Defender...
Share with
Restore previous versions
Include in library
Pin to Start
Send to
Cut
Copy
Create shortcut
Delete
Properties

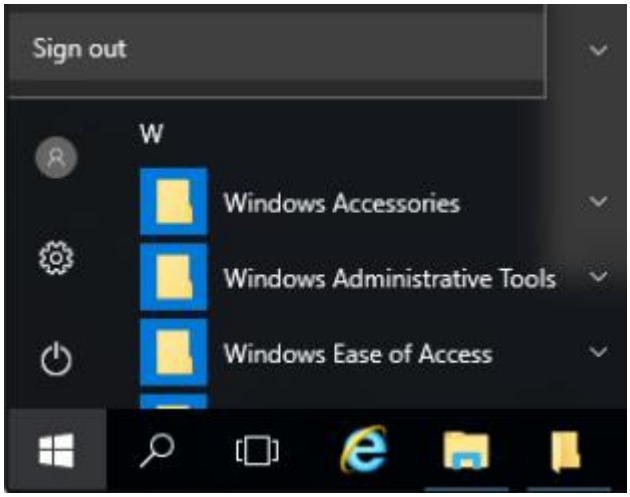


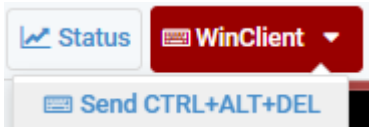
Navigating "**C:**" > "**Users**" > "**User1**" > "**Properties**" > "**Security**" will show the groups & users and the amount of access they have to the folder – in this case the groups **System** & **Administrators** have full access to the folder



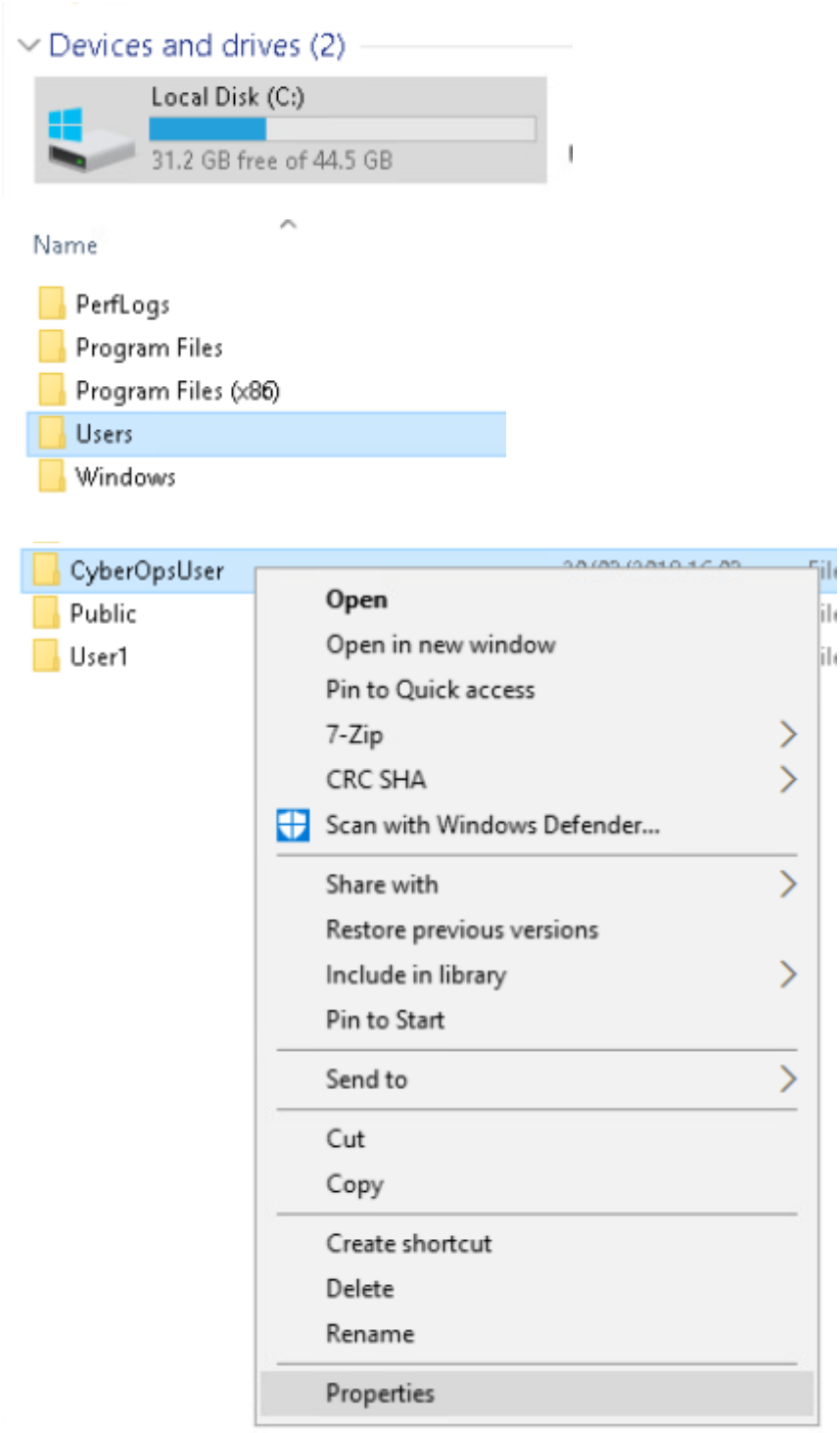


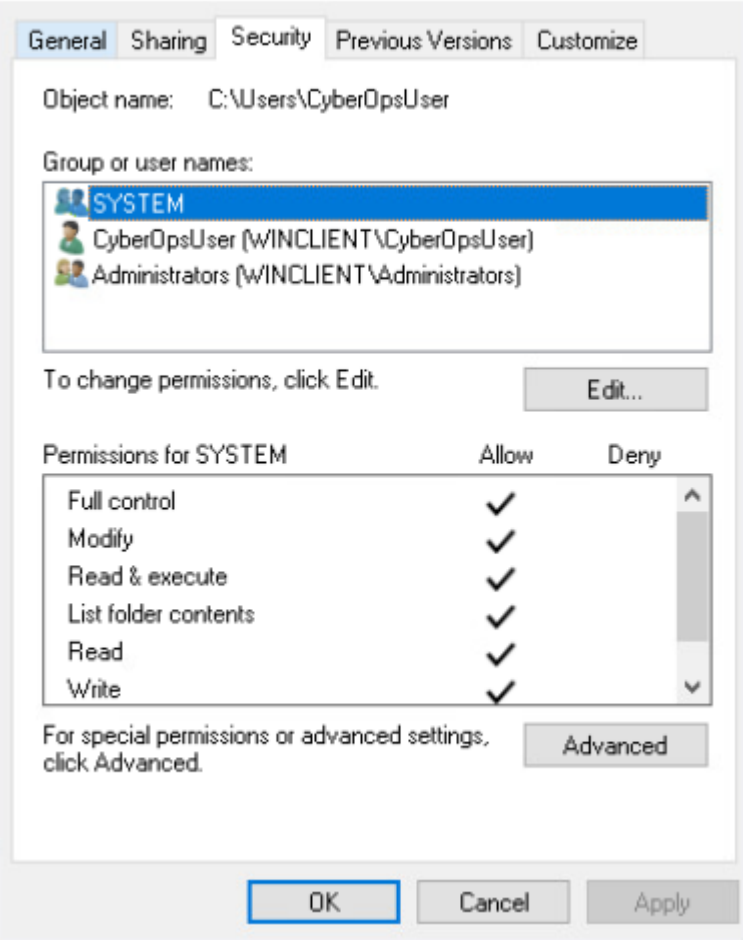
While logged in as **User1** and attempting to look at the **Security** tab of the **CyberOpsUser** folder, you will be denied access to view that data as you do not have the permission for that





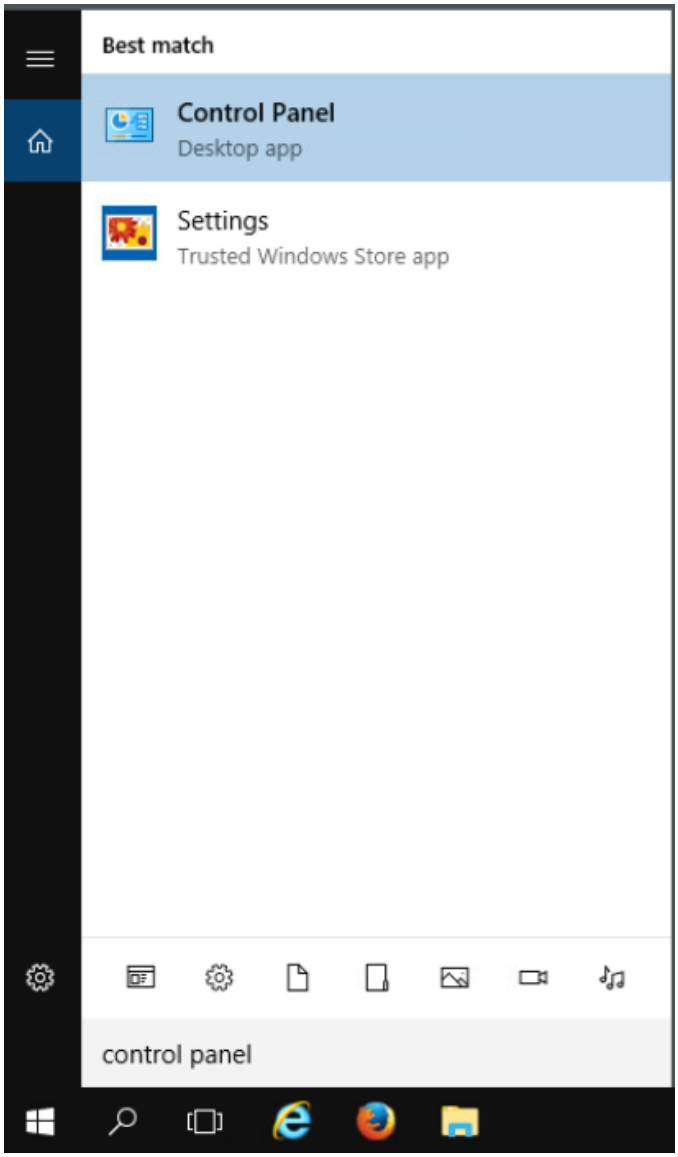
Logging in as **CyberOpsUser**

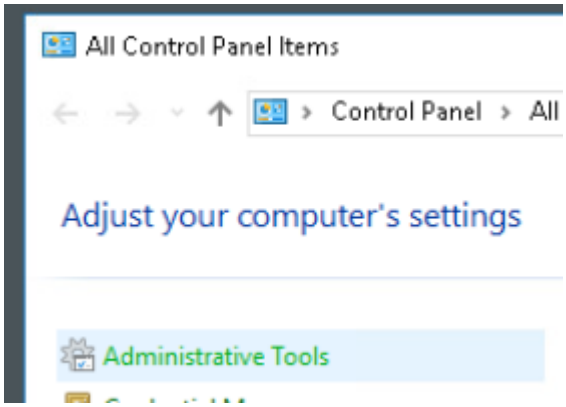

Devices and drives (2)
Local Disk (C:)
31.2 GB free of 44.5 GB
Name
PerfLogs
Program Files
Program Files (x86)
Users
Windows
CyberOpsUser
Public
User1
Open
Open in new window
Pin to Quick access
7-Zip
CRC SHA
Scan with Windows Defender...
Share with
Restore previous versions
Include in library
Pin to Start
Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties

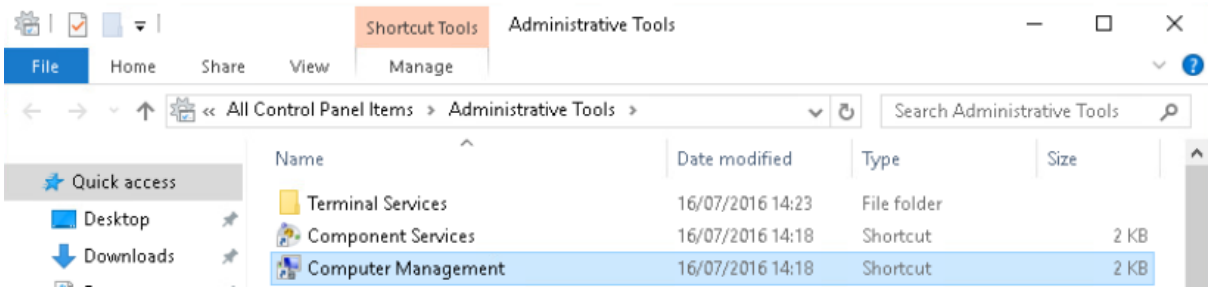


Navigating to the folder "**C:\Users**" and checking the **Security** tab of that folder will show that the system and administrator groups have full control over it

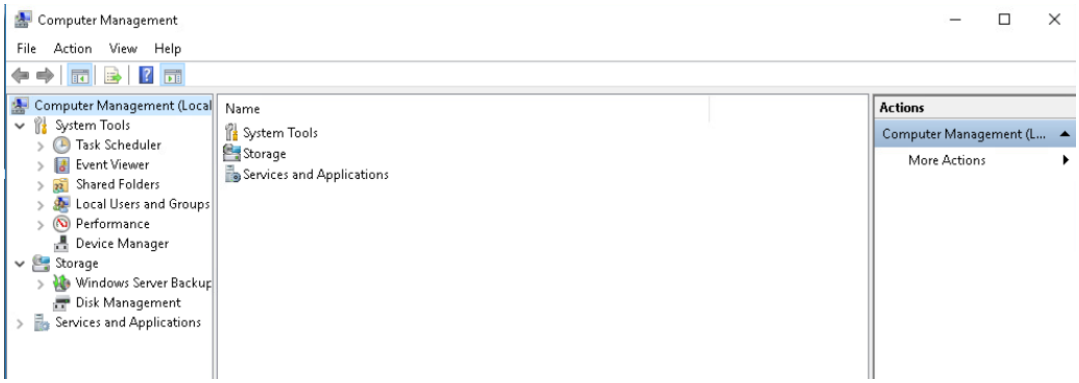
Part 2

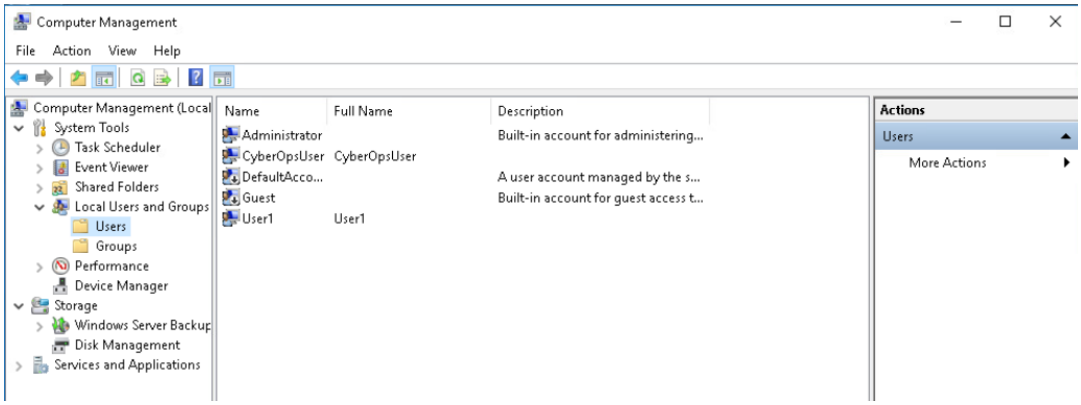

Best match
Control Panel
Desktop app
Settings
Trusted Windows Store app
control panel

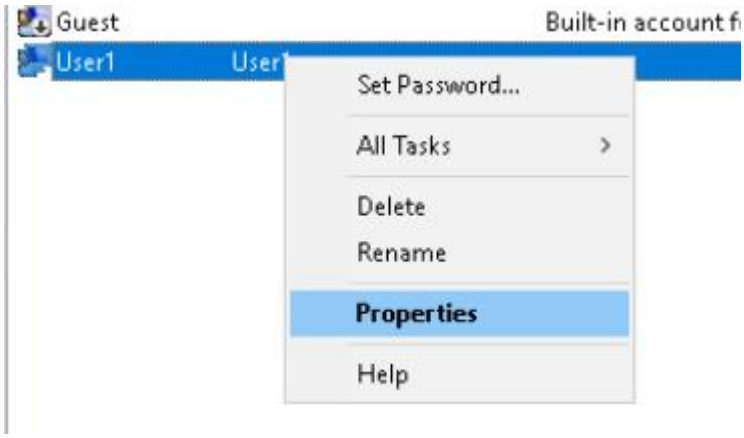

All Control Panel Items
Control Panel › All
Adjust your computer's settings
Administrative Tools



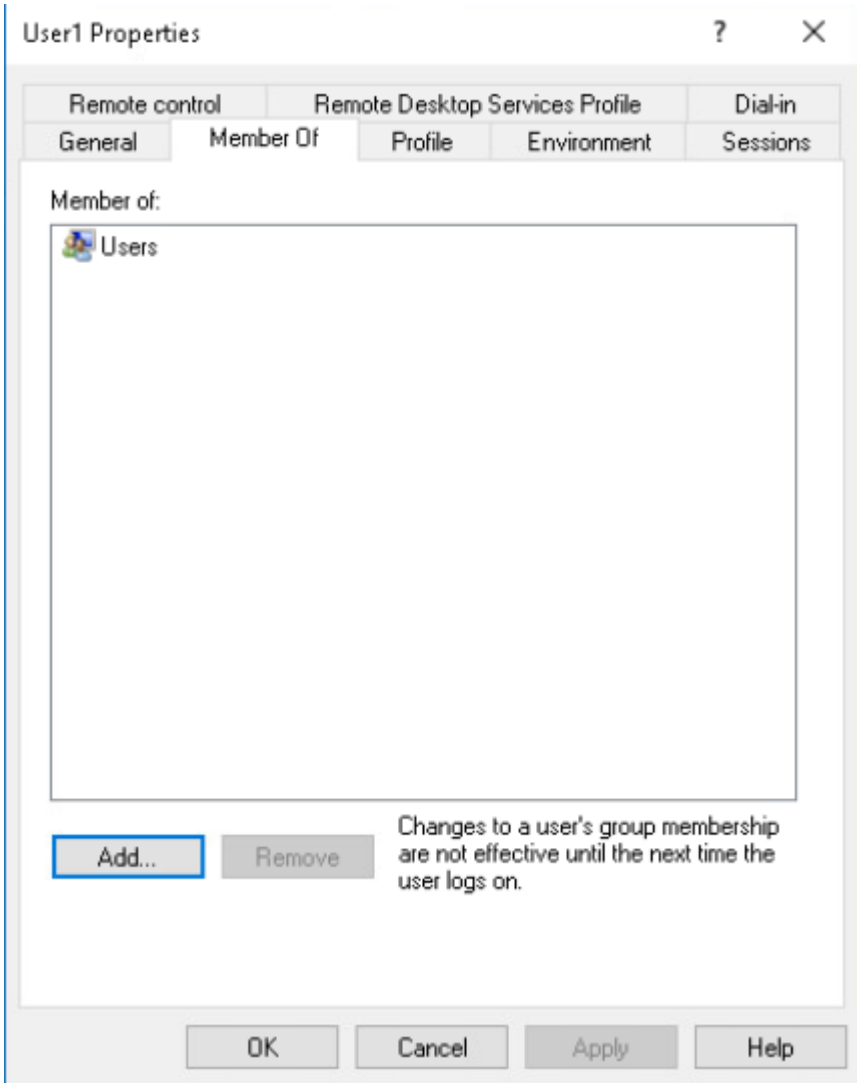
Opening and navigating: **"Control Panel"** > **"Administrative Tools"** > **"Computer Management"**



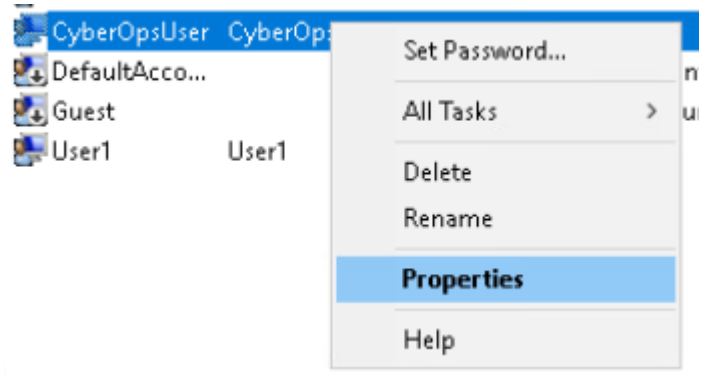


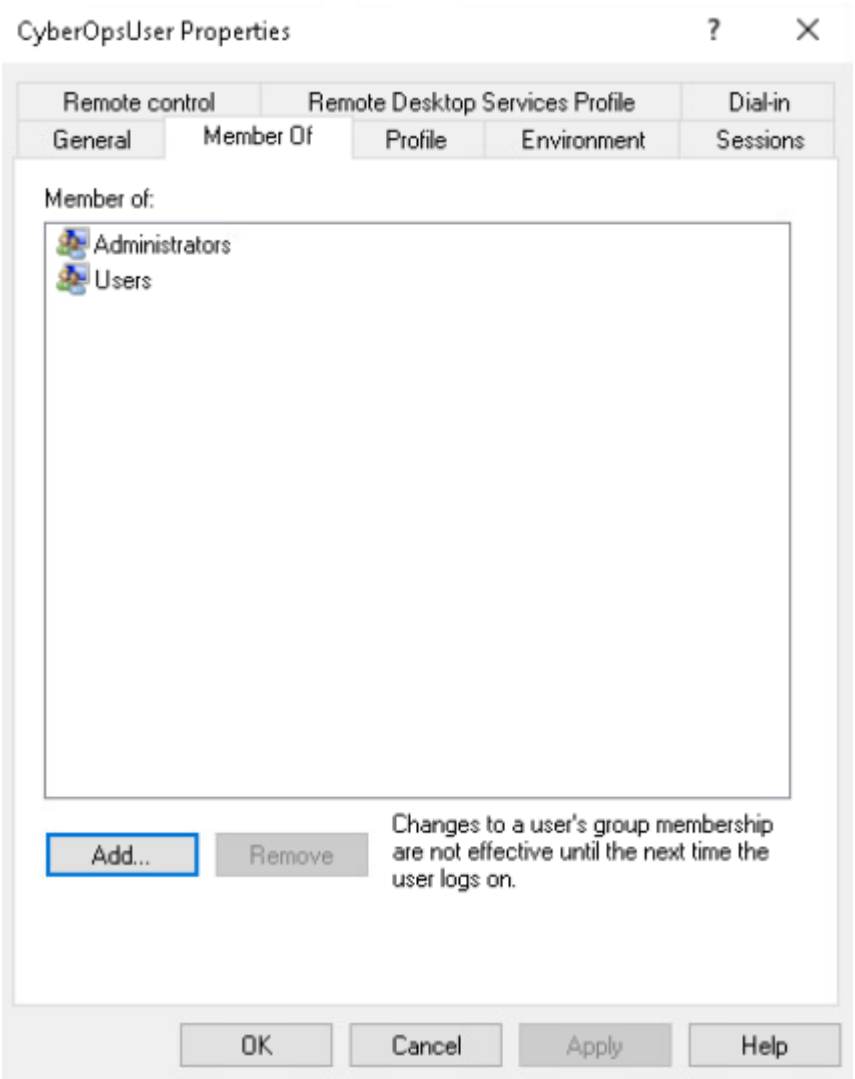


Navigating: **"Local Users and Groups"** > **"Users"** > **"Right click on User1"** > **"Properties"**



User1 is a member of the group **"Users"**

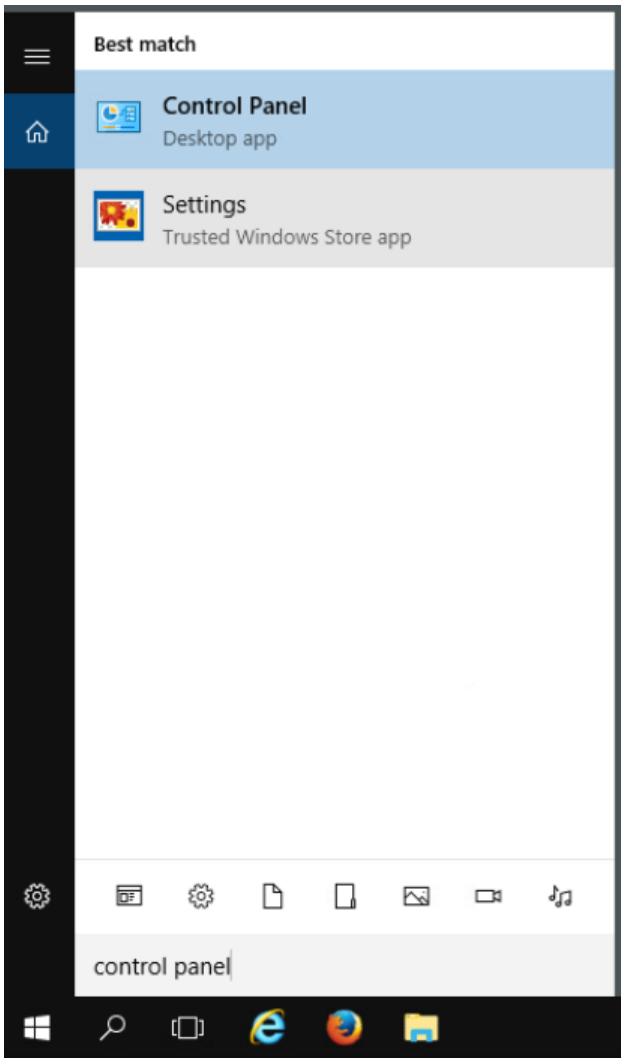


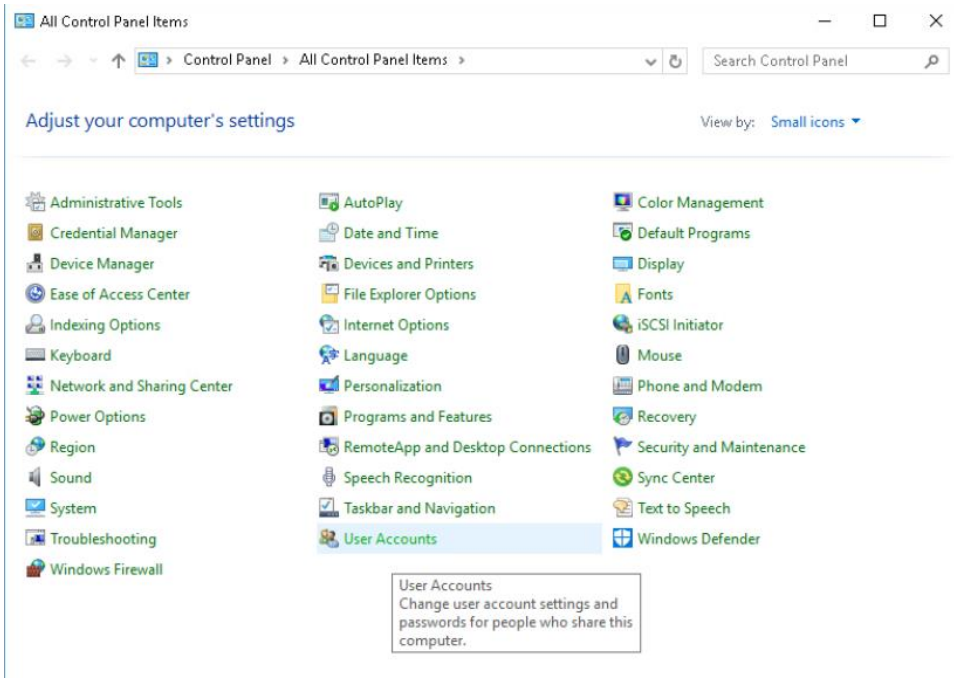


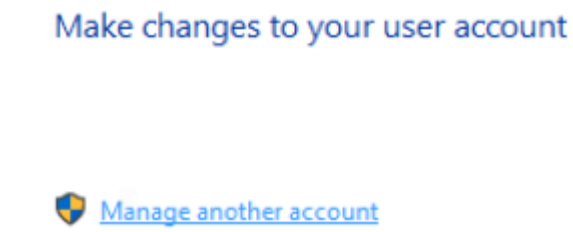
CyberOpsUser is a member of the groups "**Administrators**" and "**Users**"

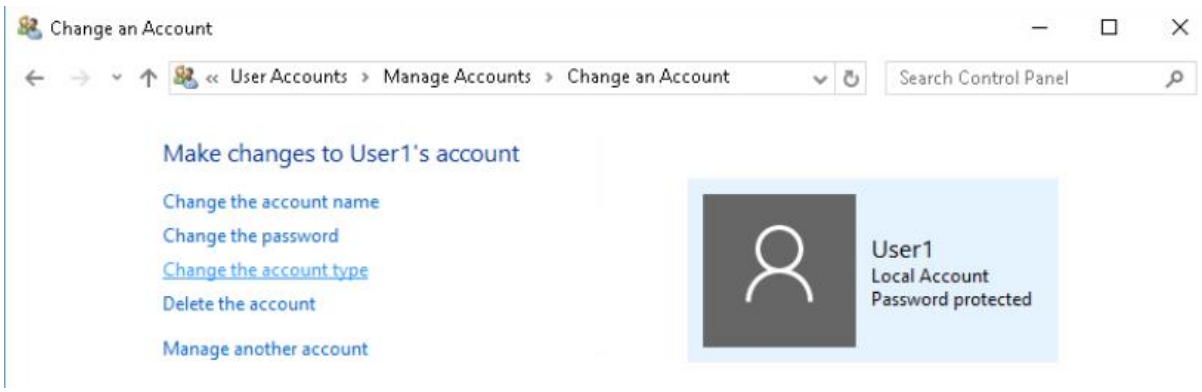
Part 3

Step 1

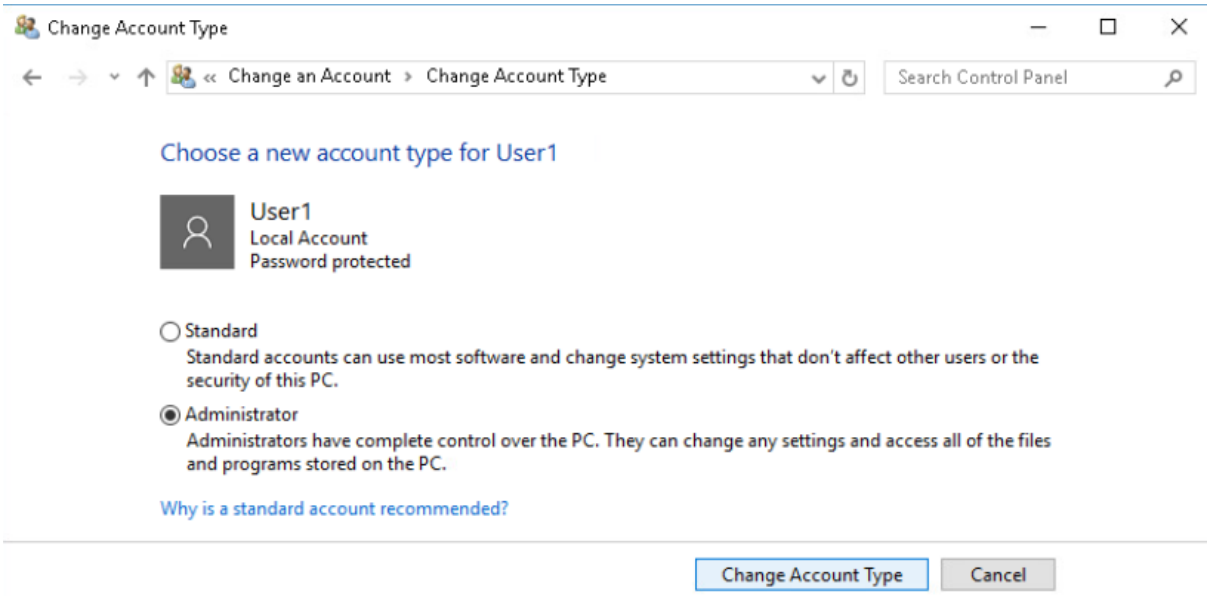

Best match
Control Panel
Desktop app
Settings
Trusted Windows Store app
control panel



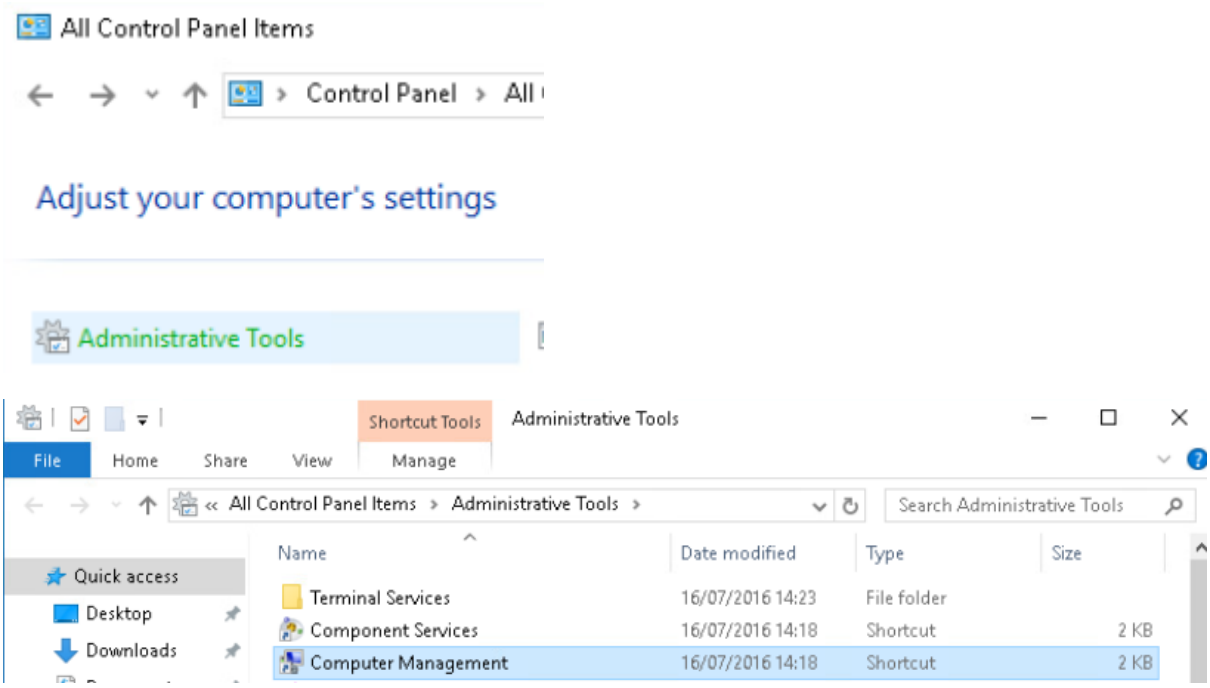




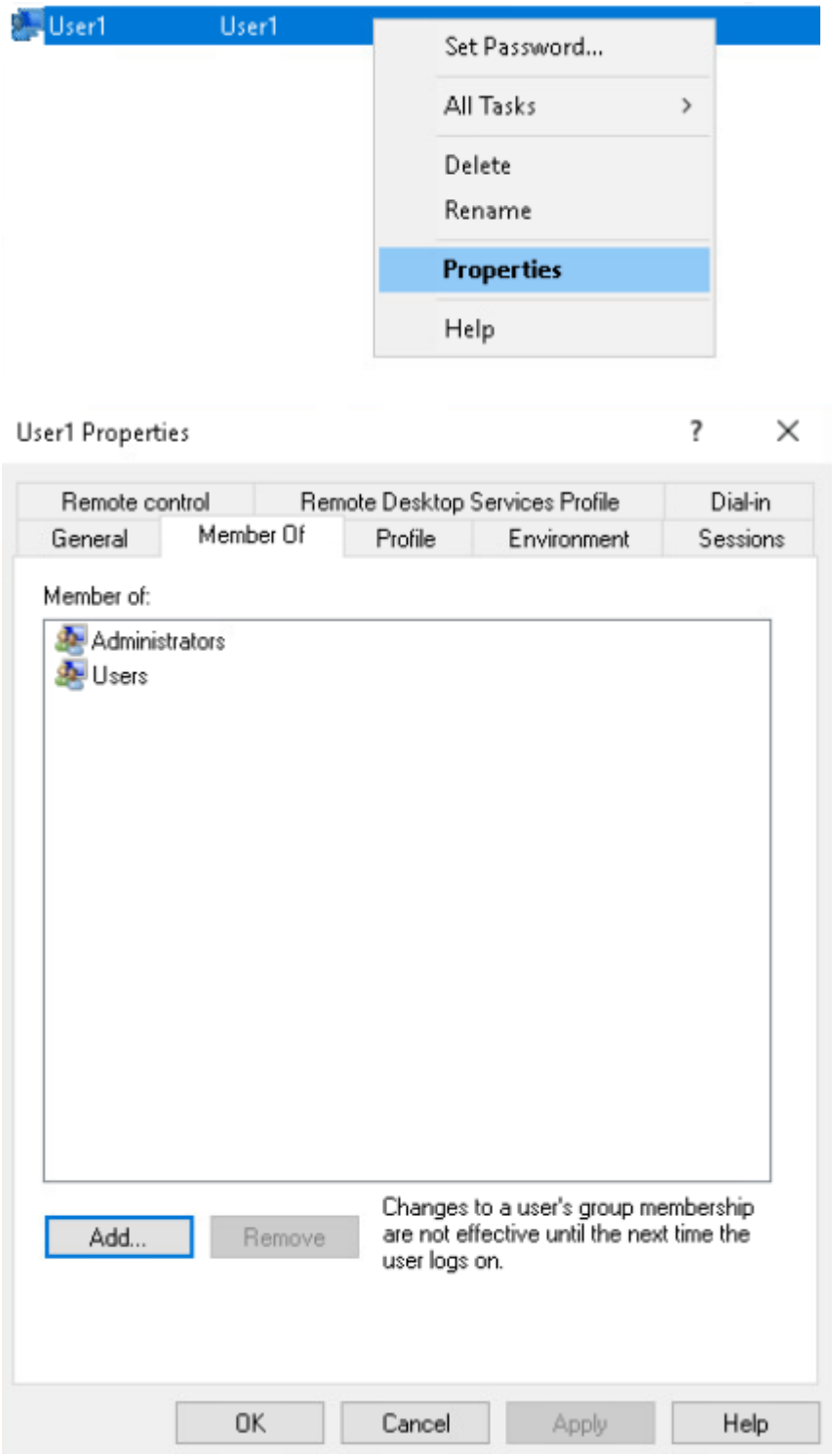
Opening and navigating: **”Control Panel”** > **”User Accounts”** > **”Manage another account”** > **”Change the account type”**



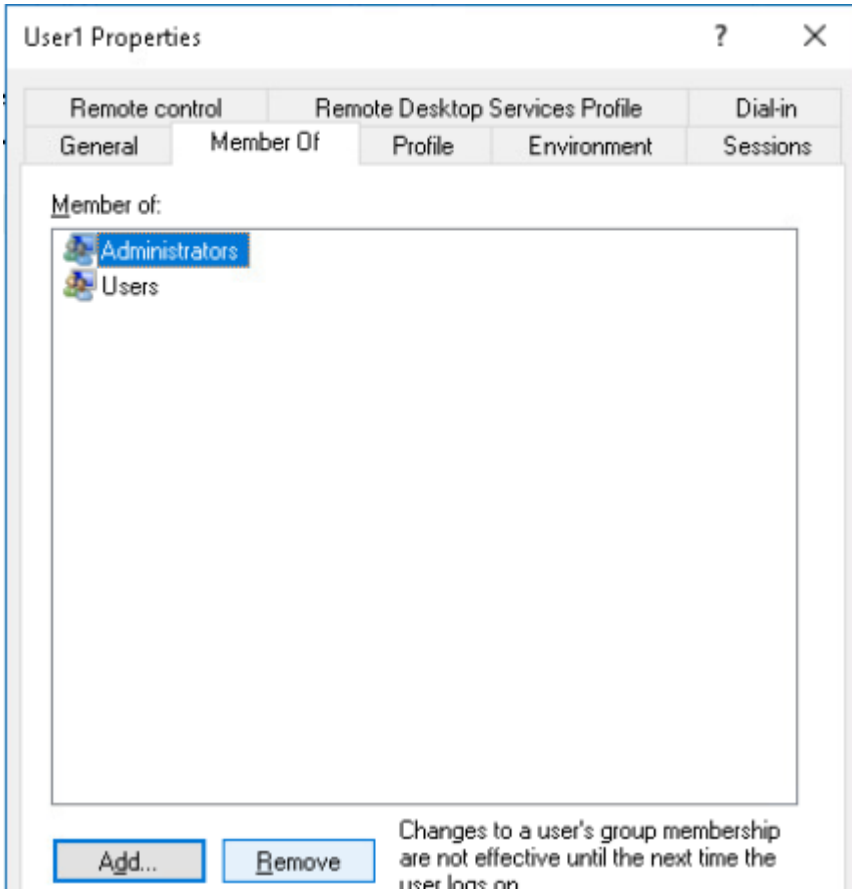
Selecting the **"Administrator"** radio button and pressing **"Change Account Type"** button will give User1 **administrative rights**.



Navigating: **"Control Panel"** > **"Administrative Tools"** > **"Computer Management"**

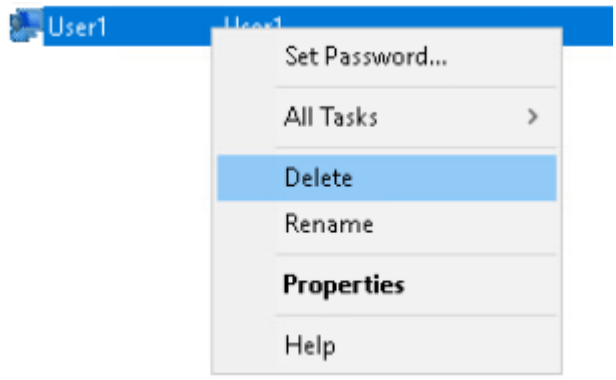


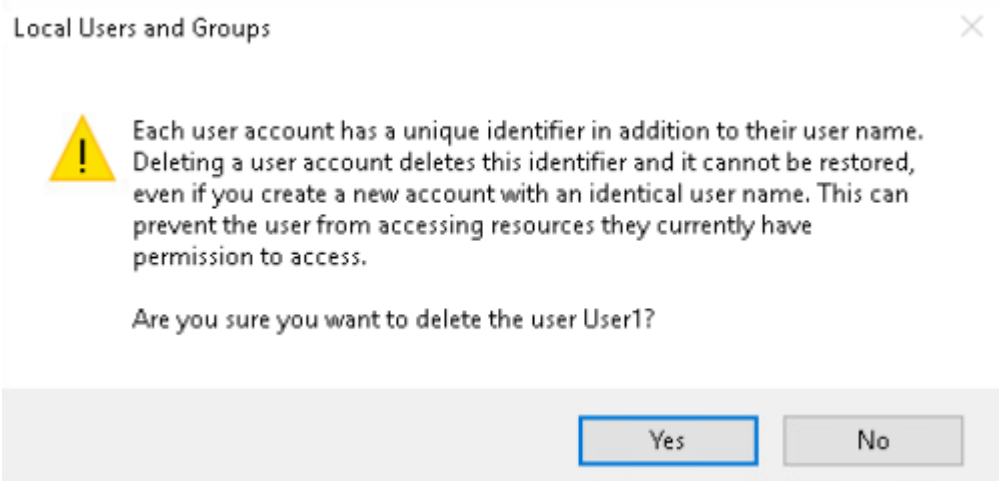
Navigating: "**Right click User1**" > "**Properties**" > "**Member Of**" will show that **User1** is a part of **Administrators** group



Navigating: **"Click Administrators"** > **"Remove"** > **"OK"** will remove **User1** from the group **Administrators**

Step 2



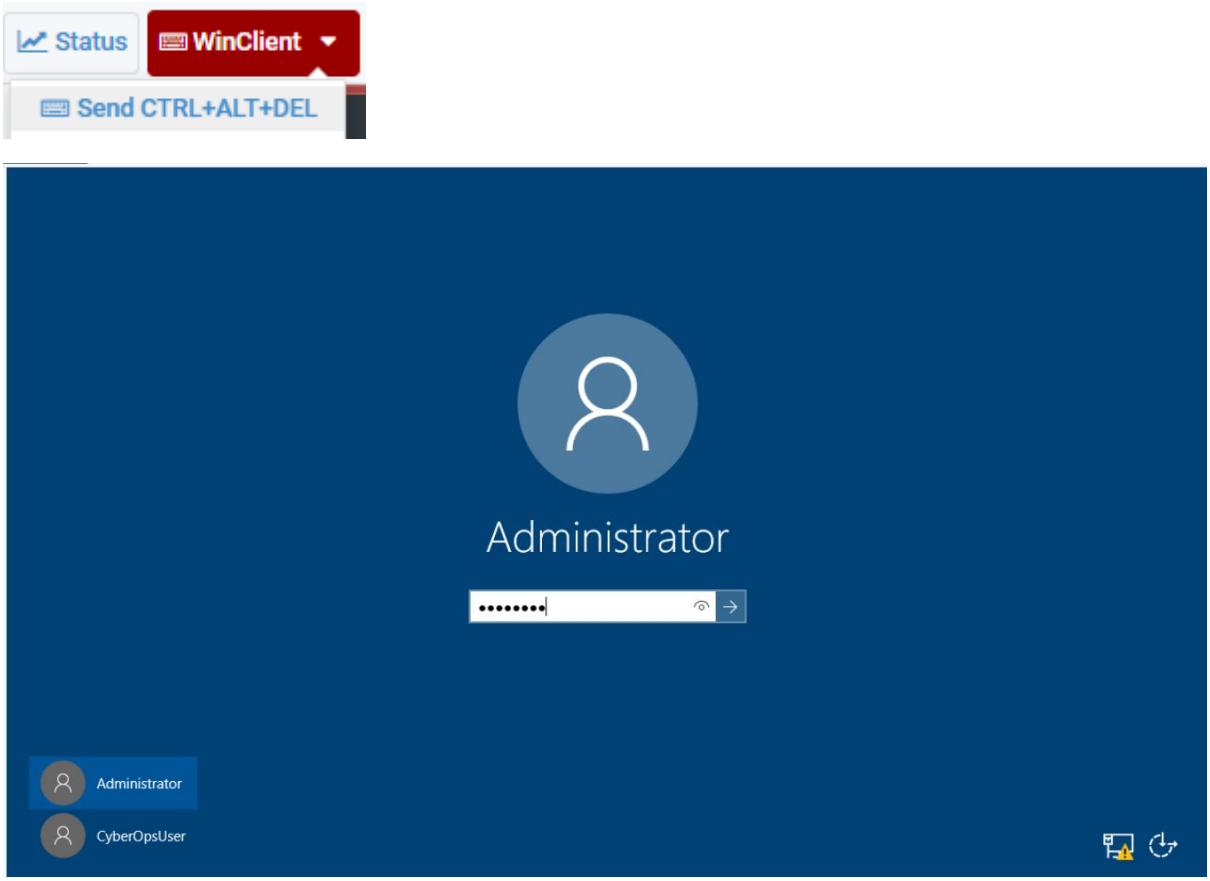


Navigate: "**Right click on User1**" > "**Delete**" > "**OK**" > "**Yes**" will delete the account from the device. Another way to delete a user account would be doing so via the Settings app

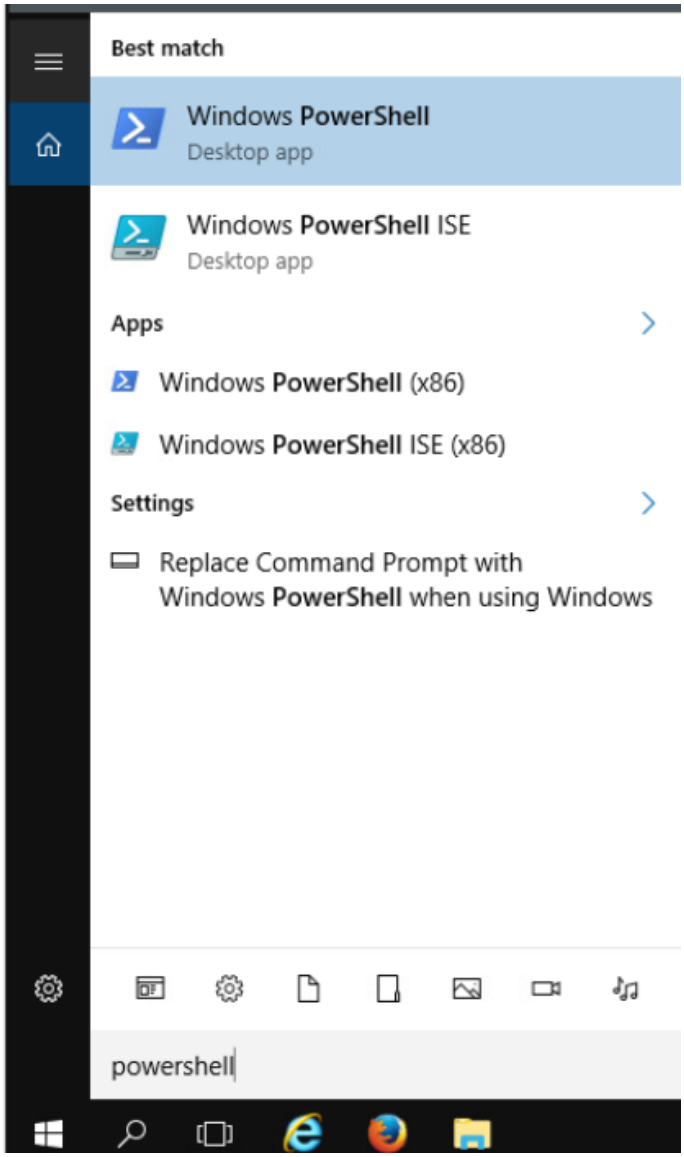
7.3.11 Lab - Using Windows PowerShell

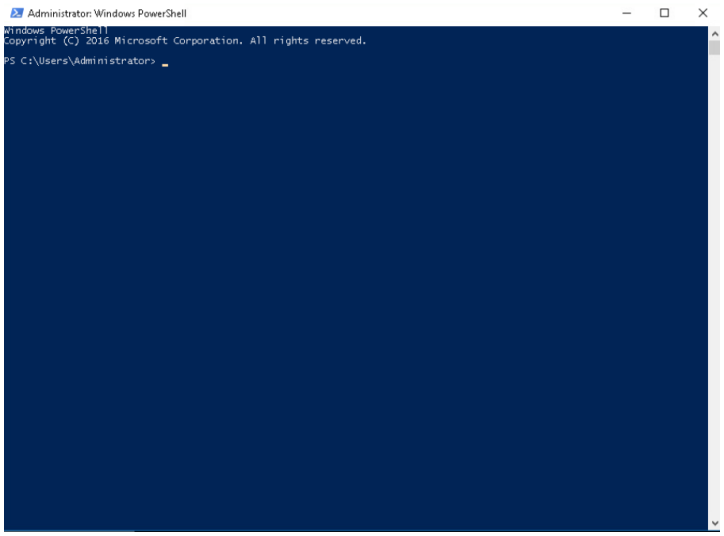
Part 1

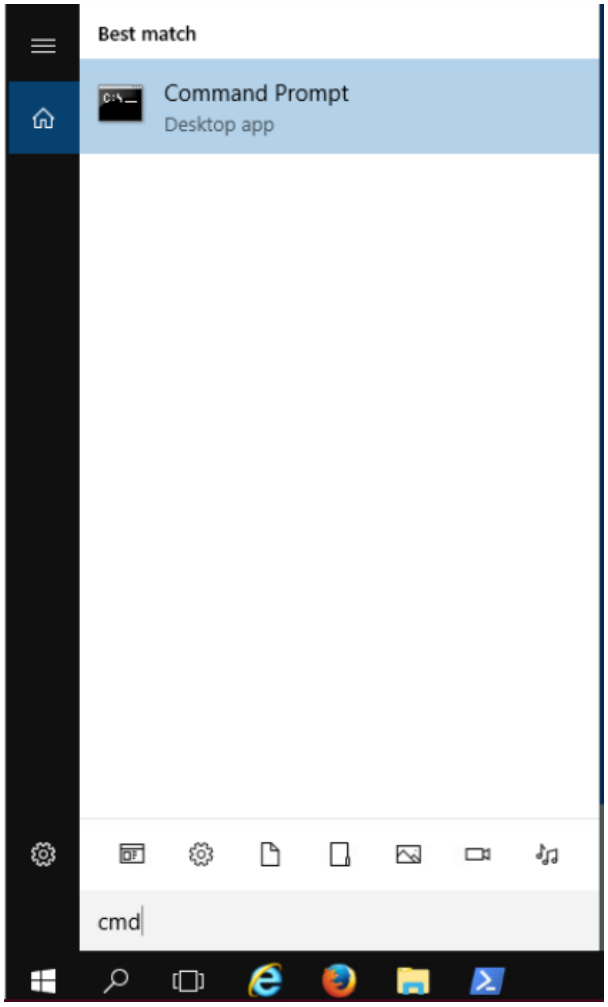
Step 1

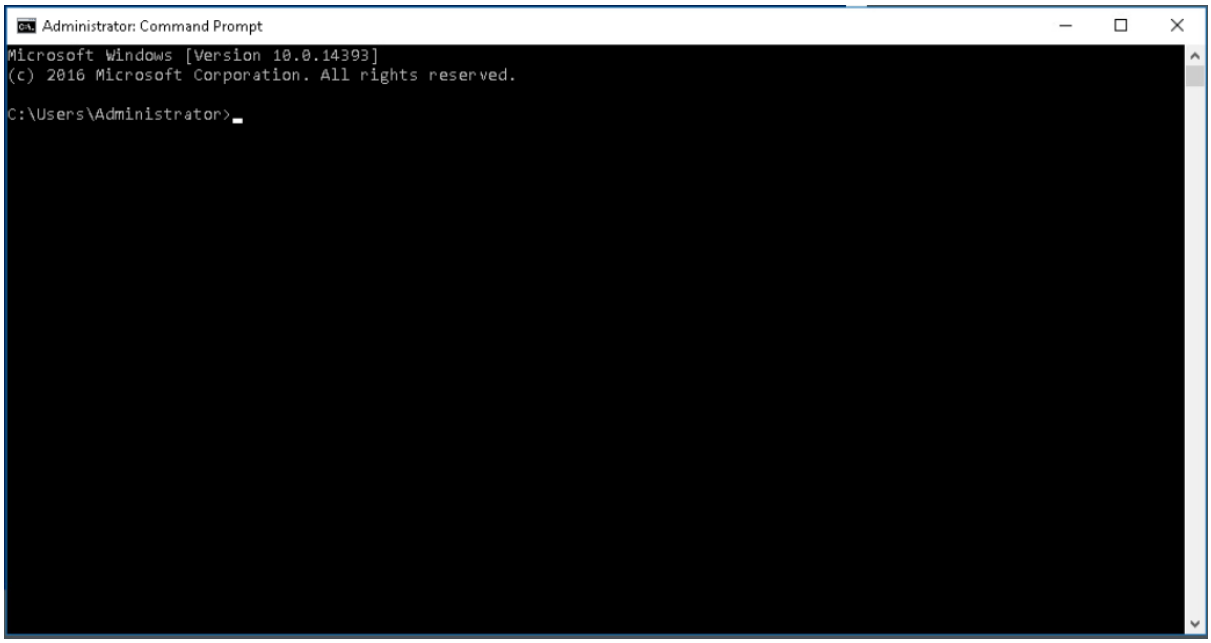


Accessing the VM on netlab


Best match
Windows PowerShell
Desktop app
Windows PowerShell ISE
Desktop app
Apps
Windows PowerShell (x86)
Windows PowerShell ISE (x86)
Settings
Replace Command Prompt with
Windows PowerShell when using Windows
powershell


Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.
PS C:\Users\Administrator>





Opening **Windows PowerShell** and **Command Prompt** via windows search bar

Part 2

```
PS C:\Users\Administrator> dir


    Directory: C:\Users\Administrator


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-r---        11/08/2020     19:50                Contacts
d-r---        11/08/2020     19:50                Desktop
d-r---        11/08/2020     19:50                Documents
d-r---        11/08/2020     19:50                Downloads
d-r---        11/08/2020     19:50                Favorites
d-r---        11/08/2020     19:50                Links
d-r---        11/08/2020     19:50                Music
d-r---        11/08/2020     19:50                Pictures
d-r---        11/08/2020     19:50                Saved Games
d-r---        11/08/2020     19:50                Searches
d-r---        11/08/2020     19:50                Videos


PS C:\Users\Administrator>
```

```
C:\Users\Administrator>dir
 Volume in drive C has no label.
 Volume Serial Number is 224B-75F9

 Directory of C:\Users\Administrator

01/06/2018  23:30    <DIR>          .
01/06/2018  23:30    <DIR>          ..
11/08/2020  19:50    <DIR>          Contacts
11/08/2020  19:50    <DIR>          Desktop
11/08/2020  19:50    <DIR>          Documents
11/08/2020  19:50    <DIR>          Downloads
11/08/2020  19:50    <DIR>          Favorites
11/08/2020  19:50    <DIR>          Links
11/08/2020  19:50    <DIR>          Music
11/08/2020  19:50    <DIR>          Pictures
11/08/2020  19:50    <DIR>          Saved Games
11/08/2020  19:50    <DIR>          Searches
11/08/2020  19:50    <DIR>          Videos
               0 File(s)              0 bytes
              13 Dir(s)  33,589,452,800 bytes free

C:\Users\Administrator>_
```

Using **dir** command on **PowerShell** provides a similar output to the output of **cmd** but with headers. But the **cmd** output contains data like the current directory "**.**" and the previous directory "**..**", alongside the amount of data free on the drive.

```
PS C:\Users\Administrator> ping 192.168.0.12

Pinging 192.168.0.12 with 32 bytes of data:
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.0.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator>
```

```
C:\Users\Administrator>ping 192.168.0.12

Pinging 192.168.0.12 with 32 bytes of data:
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128
Reply from 192.168.0.12: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.0.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>
```

Using the **ping** command provides the same output both on **PowerShell** and **cmd**

Part 3

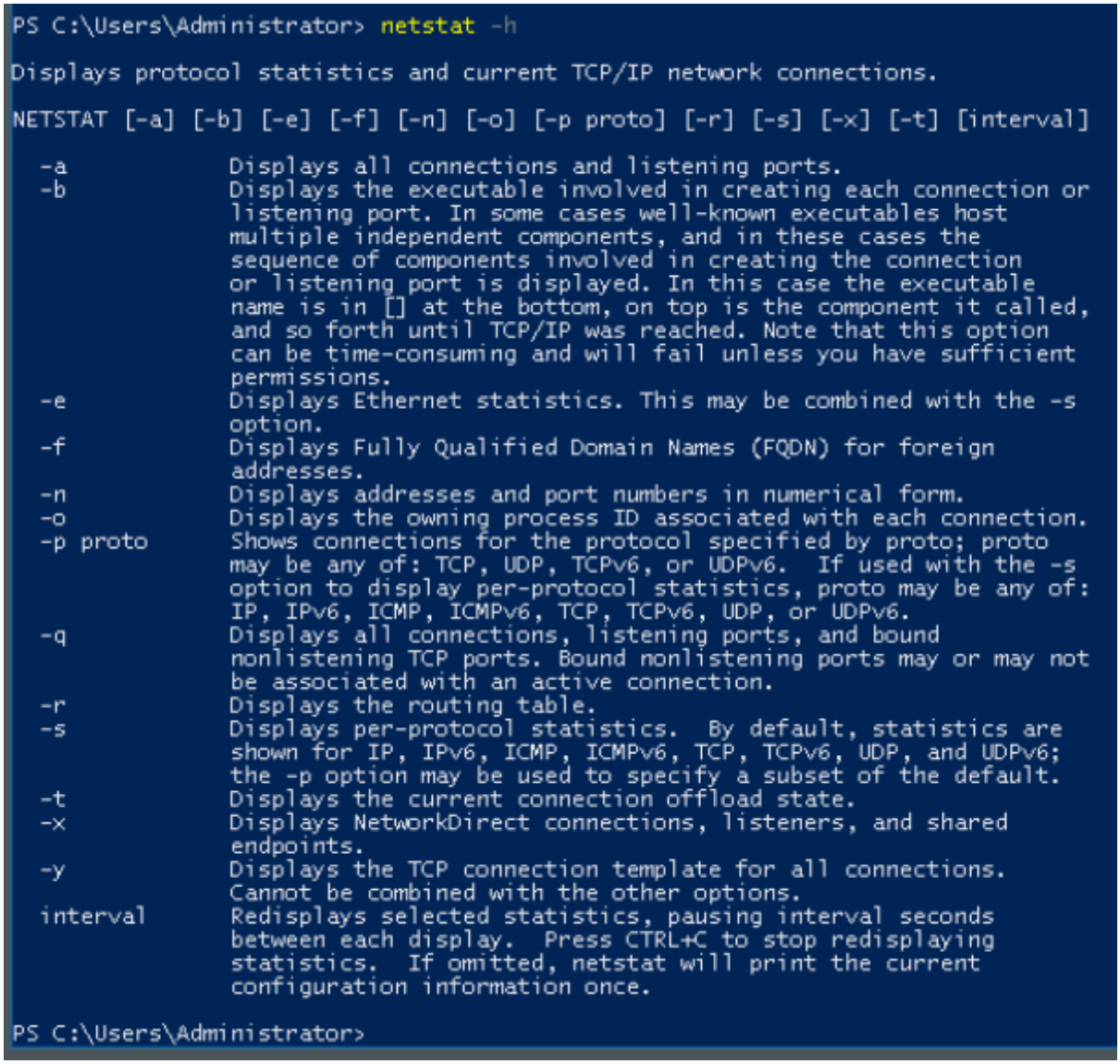
```
PS C:\Users\Administrator> Get-Alias dir

CommandType     Name                                               Version    Source
-----------     ----                                               -------    ------
Alias           dir -> Get-ChildItem

PS C:\Users\Administrator> _
```

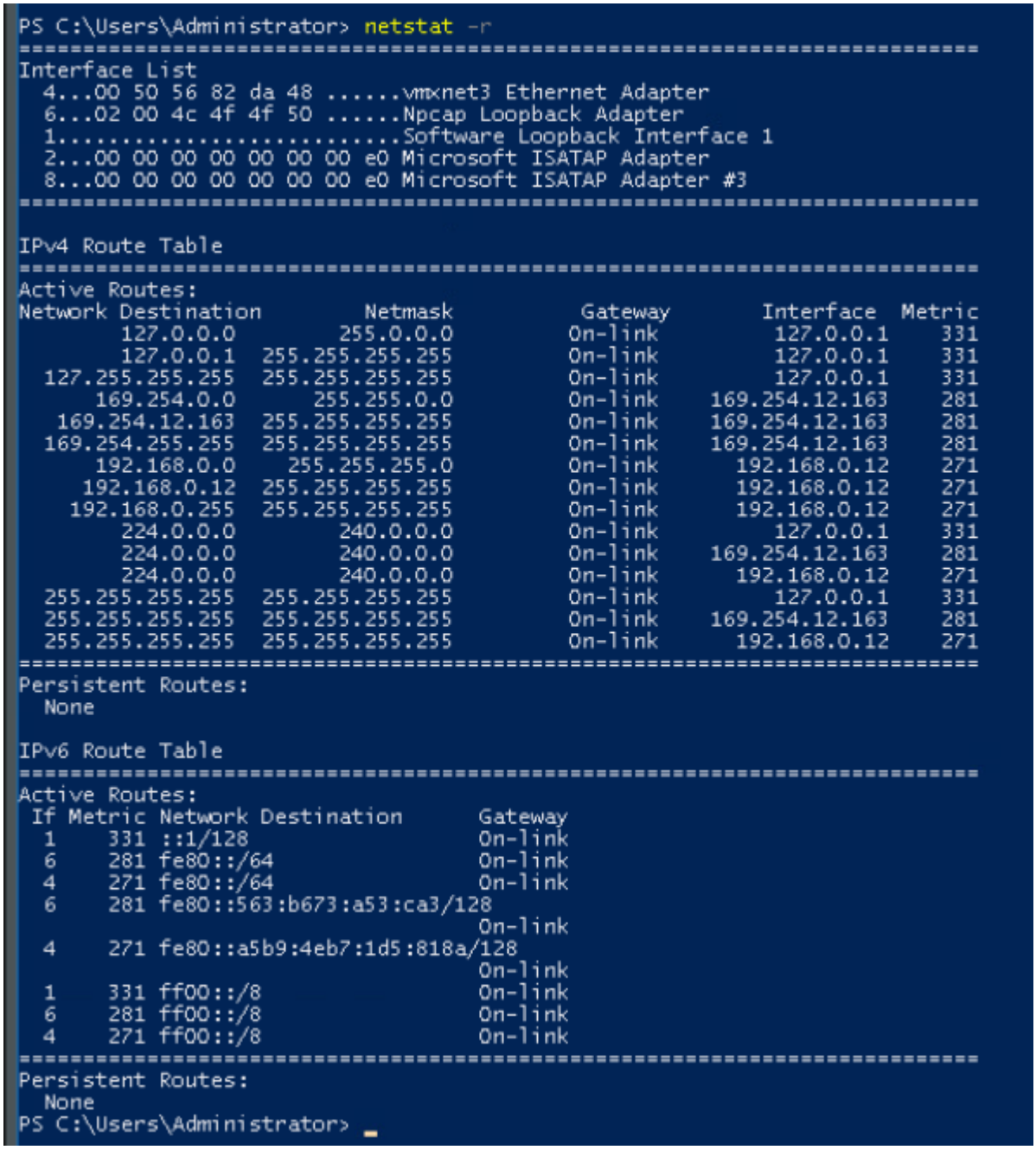
```
PS C:\Users\Administrator> Get-ChildItem


    Directory: C:\Users\Administrator


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-r---        11/08/2020     19:50                Contacts
d-r---        11/08/2020     19:50                Desktop
d-r---        11/08/2020     19:50                Documents
d-r---        11/08/2020     19:50                Downloads
d-r---        11/08/2020     19:50                Favorites
d-r---        11/08/2020     19:50                Links
d-r---        11/08/2020     19:50                Music
d-r---        11/08/2020     19:50                Pictures
d-r---        11/08/2020     19:50                Saved Games
d-r---        11/08/2020     19:50                Searches
d-r---        11/08/2020     19:50                Videos


PS C:\Users\Administrator>
```

The PowerShell command equivalent of **dir** is **Get-ChildItem**

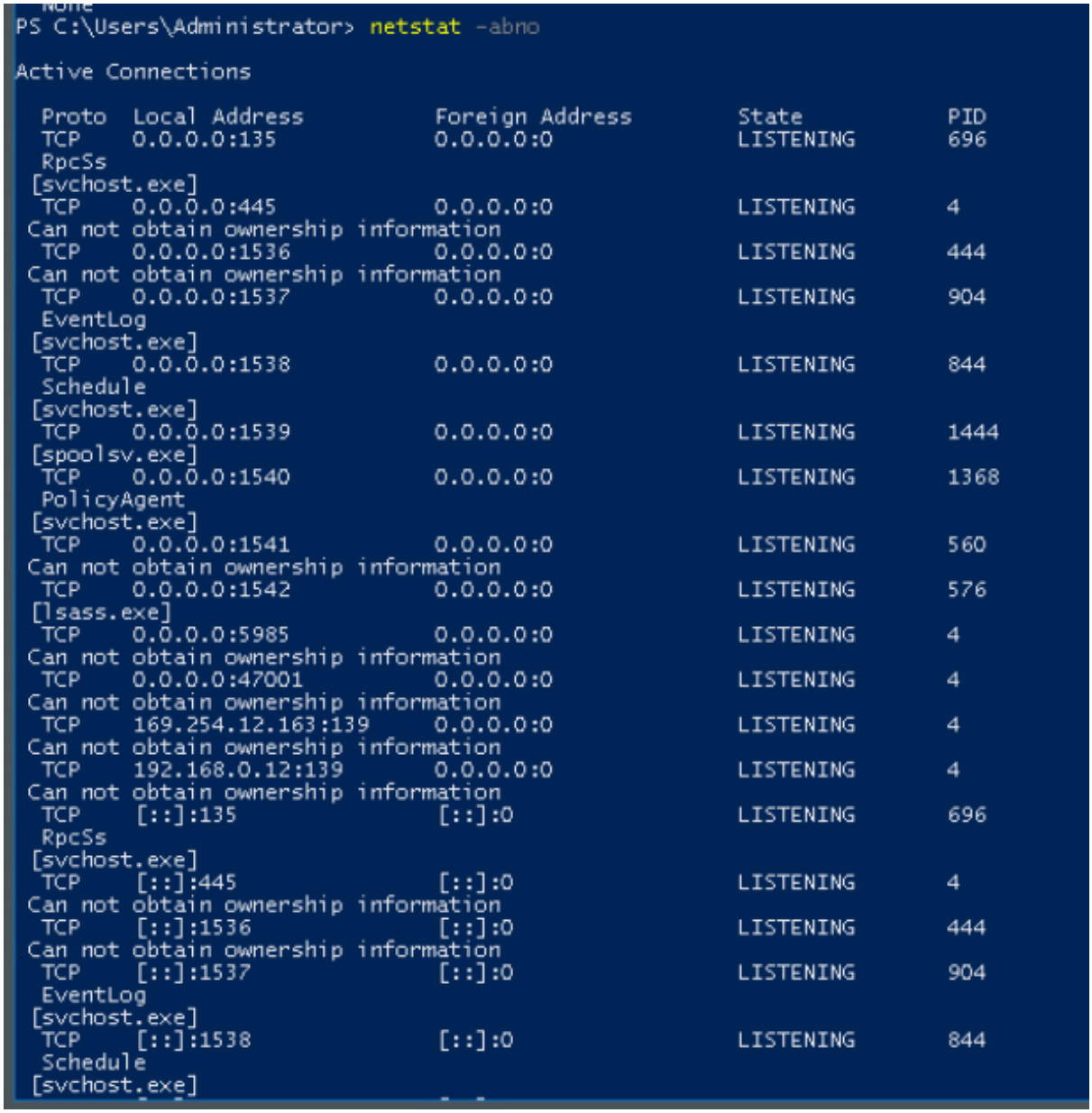
As shown when executing it, it gives the same output as the **dir** command

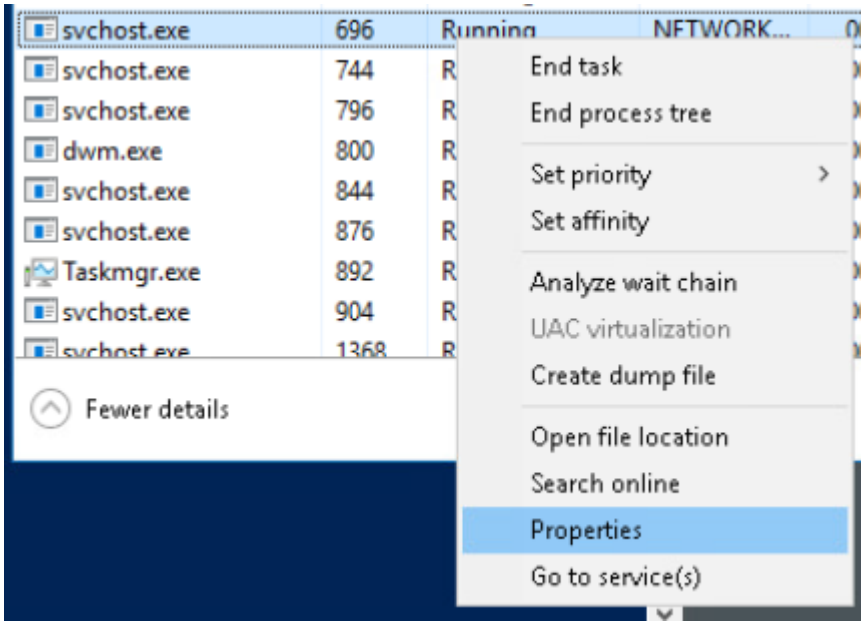
Part 4


PS C:\Users\Administrator> netstat -h
Displays protocol statistics and current TCP/IP network connections.
NETSTAT [-a] [-b] [-e] [-f] [-n] [-o] [-p proto] [-r] [-s] [-x] [-t] [interval]
-a Displays all connections and listening ports.
-b Displays the executable involved in creating each connection or listening port. In some cases well-known executables host multiple independent components, and in these cases the sequence of components involved in creating the connection or listening port is displayed. In this case the executable name is in [] at the bottom, on top is the component it called, and so forth until TCP/IP was reached. Note that this option can be time-consuming and will fail unless you have sufficient permissions.
-e Displays Ethernet statistics. This may be combined with the -s option.
-f Displays Fully Qualified Domain Names (FQDN) for foreign addresses.
-n Displays addresses and port numbers in numerical form.
-o Displays the owning process ID associated with each connection.
-p proto Shows connections for the protocol specified by proto; proto may be any of: TCP, UDP, TCPv6, or UDPv6. If used with the -s option to display per-protocol statistics, proto may be any of: IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, or UDPv6.
-q Displays all connections, listening ports, and bound nonlistening TCP ports. Bound nonlistening ports may or may not be associated with an active connection.
-r Displays the routing table.
-s Displays per-protocol statistics. By default, statistics are shown for IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, and UDPv6; the -p option may be used to specify a subset of the default.
-t Displays the current connection offload state.
-x Displays NetworkDirect connections, listeners, and shared endpoints.
-y Displays the TCP connection template for all connections. Cannot be combined with the other options.
interval Redisplays selected statistics, pausing interval seconds between each display. Press CTRL+C to stop redisplaying statistics. If omitted, netstat will print the current configuration information once.
PS C:\Users\Administrator>

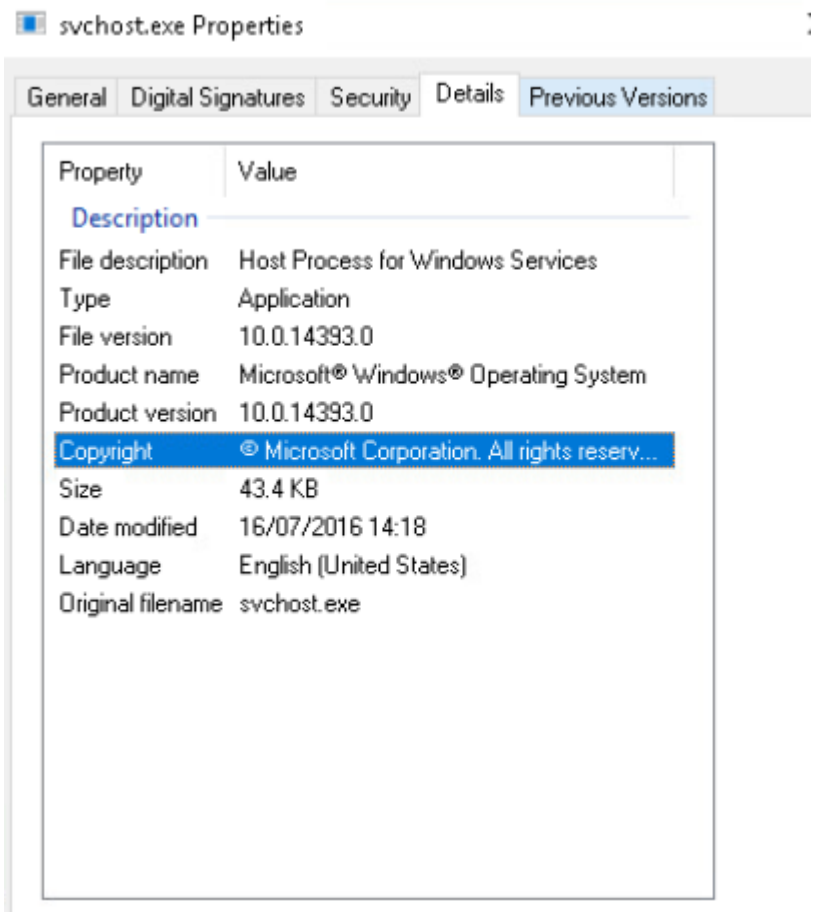


The IPv4 gateway would be 192.168.1.1


PS C:\Users\Administrator> netstat -abno
Active Connections
Proto Local Address Foreign Address State PID
TCP 0.0.0.0:135 0.0.0.0:0 LISTENING 696
RpcSs
[svchost.exe]
TCP 0.0.0.0:445 0.0.0.0:0 LISTENING 4
Can not obtain ownership information
TCP 0.0.0.0:1536 0.0.0.0:0 LISTENING 444
Can not obtain ownership information
TCP 0.0.0.0:1537 0.0.0.0:0 LISTENING 904
EventLog
[svchost.exe]
TCP 0.0.0.0:1538 0.0.0.0:0 LISTENING 844
Schedule
[svchost.exe]
TCP 0.0.0.0:1539 0.0.0.0:0 LISTENING 1444
[spoolsv.exe]
TCP 0.0.0.0:1540 0.0.0.0:0 LISTENING 1368
PolicyAgent
[svchost.exe]
TCP 0.0.0.0:1541 0.0.0.0:0 LISTENING 560
Can not obtain ownership information
TCP 0.0.0.0:1542 0.0.0.0:0 LISTENING 576
[lsass.exe]
TCP 0.0.0.0:5985 0.0.0.0:0 LISTENING 4
Can not obtain ownership information
TCP 0.0.0.0:47001 0.0.0.0:0 LISTENING 4
Can not obtain ownership information
TCP 169.254.12.163:139 0.0.0.0:0 LISTENING 4
Can not obtain ownership information
TCP 192.168.0.12:139 0.0.0.0:0 LISTENING 4
Can not obtain ownership information
TCP [::]:135 [::]:0 LISTENING 696
RpcSs
[svchost.exe]
TCP [::]:445 [::]:0 LISTENING 4
Can not obtain ownership information
TCP [::]:1536 [::]:0 LISTENING 444
Can not obtain ownership information
TCP [::]:1537 [::]:0 LISTENING 904
EventLog
[svchost.exe]
TCP [::]:1538 [::]:0 LISTENING 844
Schedule
[svchost.exe]

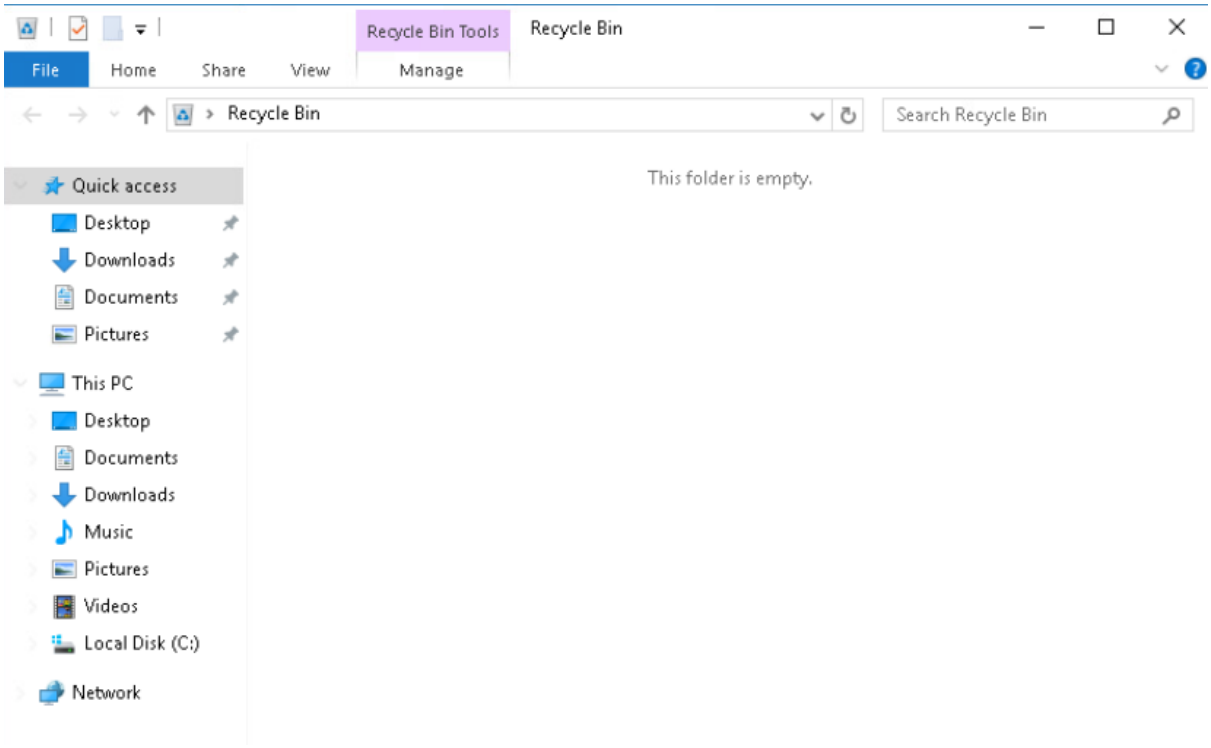


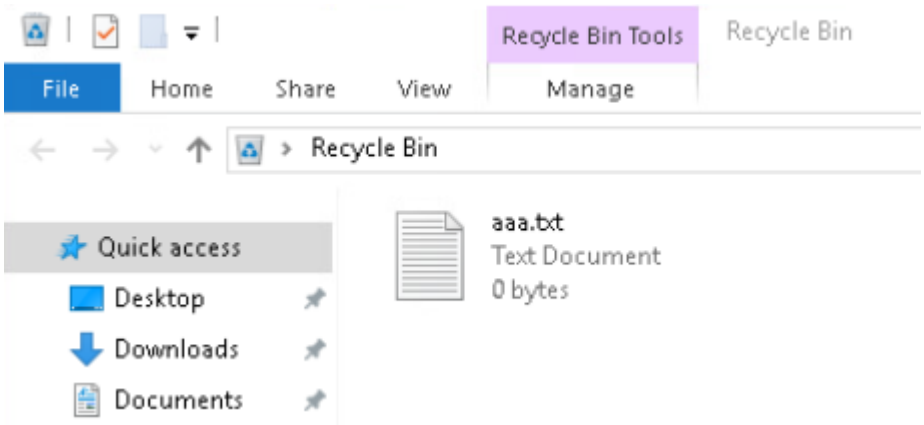
Navigating: "**Task Manager**" > "**Details**" > "**Click on PID**" and finding the PID value of 696 (for this task). Once found, Right click and open **Properties**.



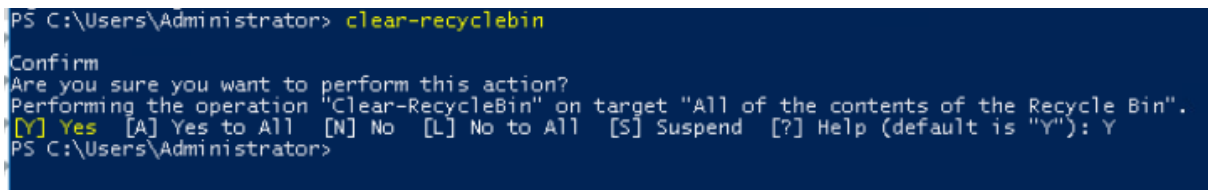
From the "**Details**" tab you can find details like size of the file, the type of file it is (in this case Application) and a description of the file

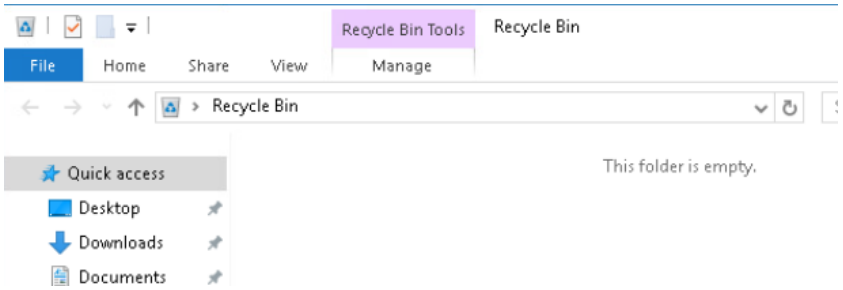
Part 5





Opening the **Recycle Bin** from desktop and creating a dummy text file and dropping it inside the bin

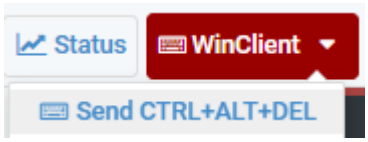




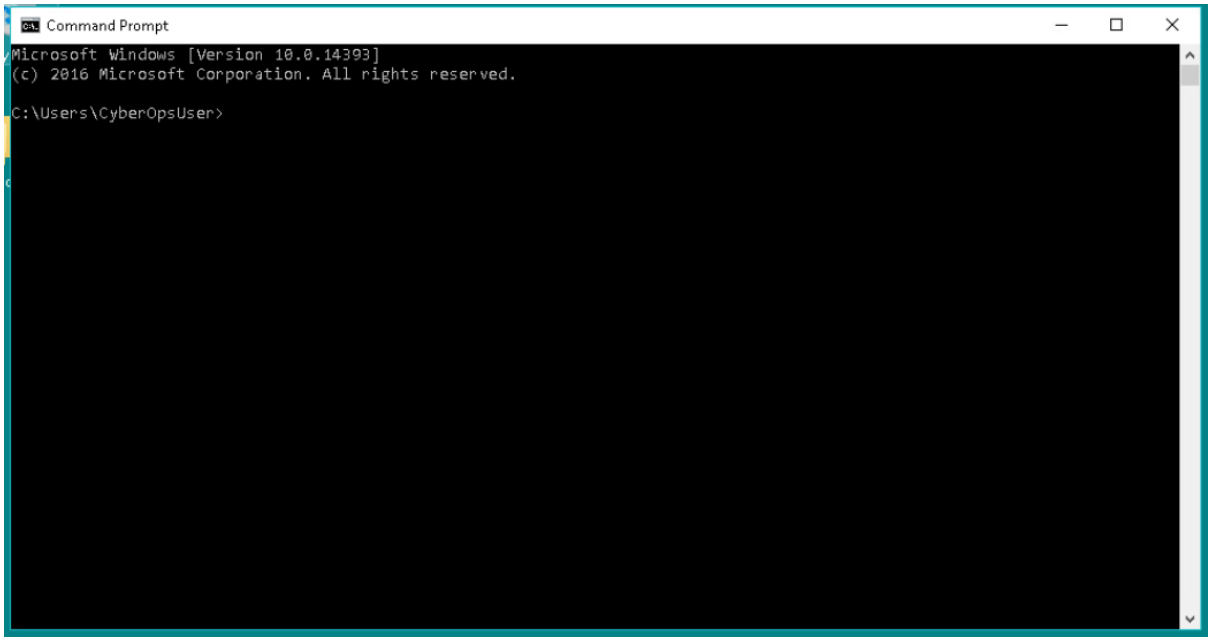
Entering “**clear-recyclebin**” into PowerShell and confirming the execution will clear the bin’s contents.

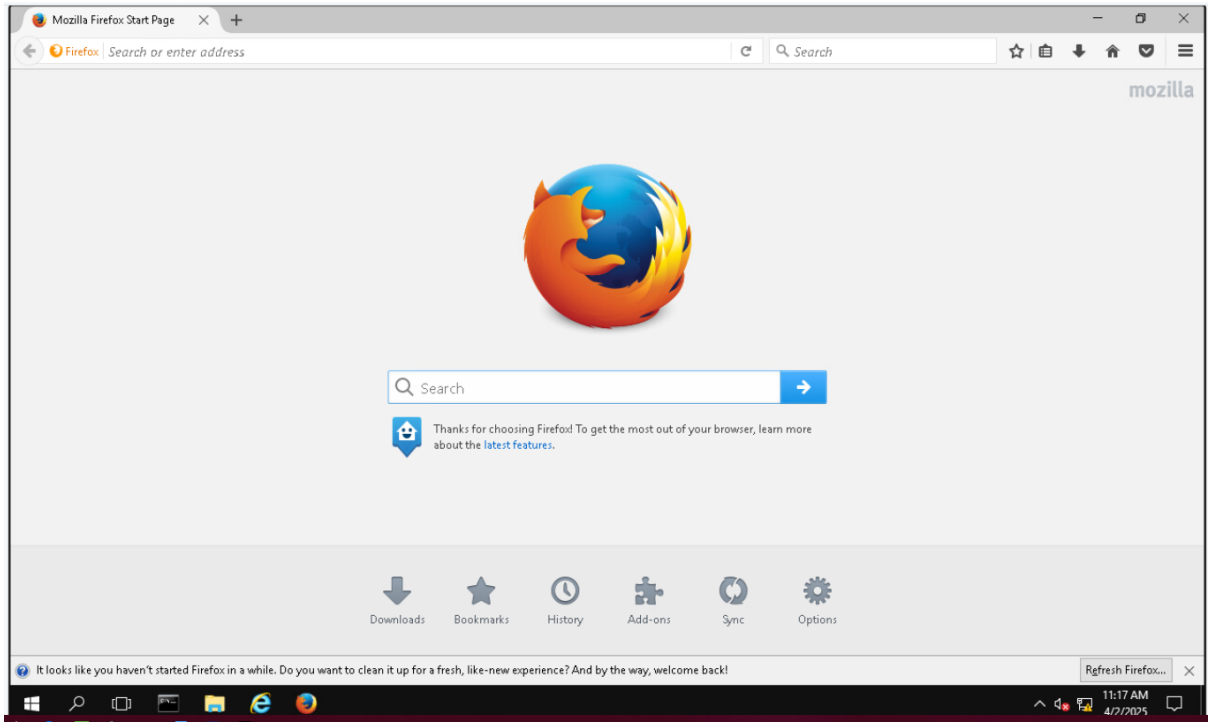
7.3.12 Lab - Windows Task Manager

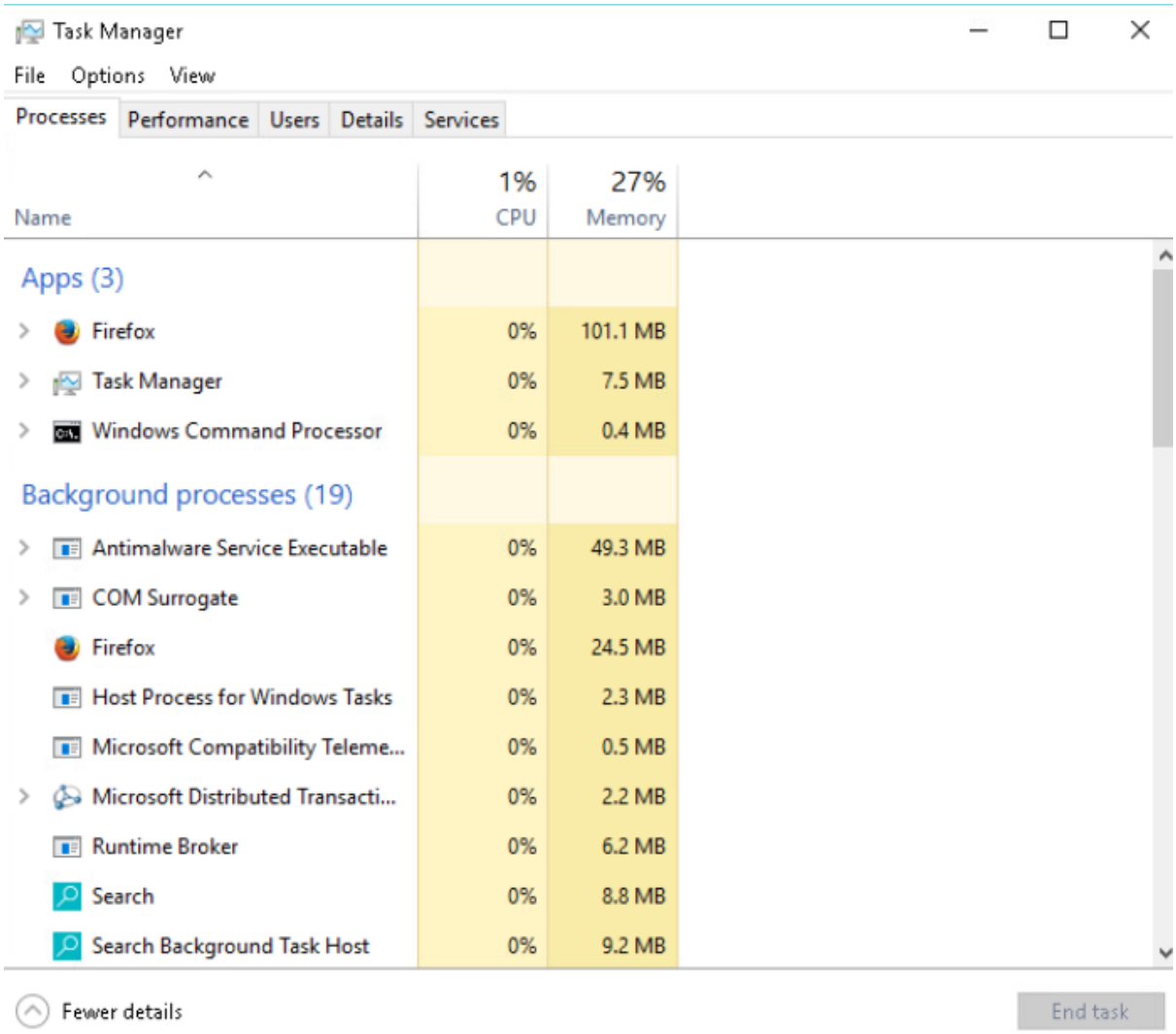
Part 1



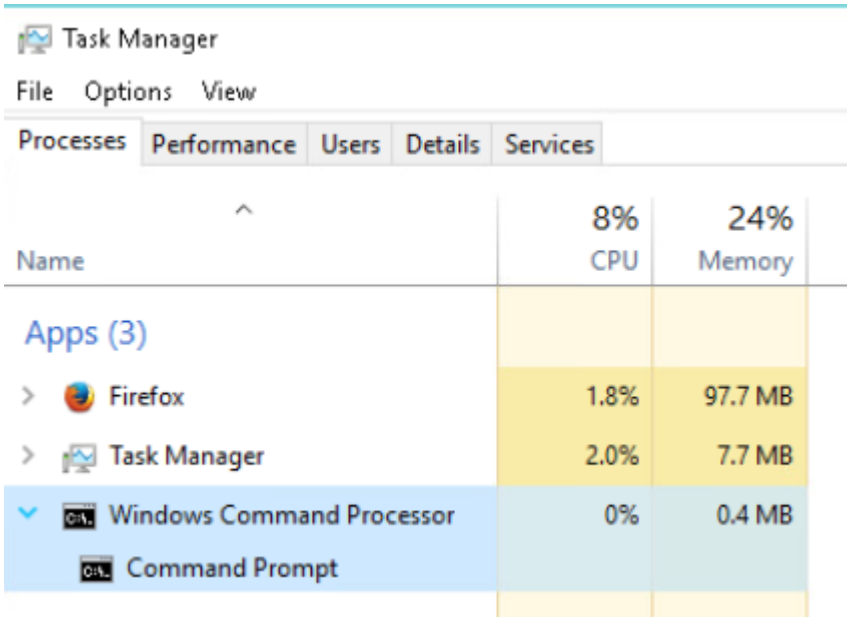
Accessing the VM on netlab



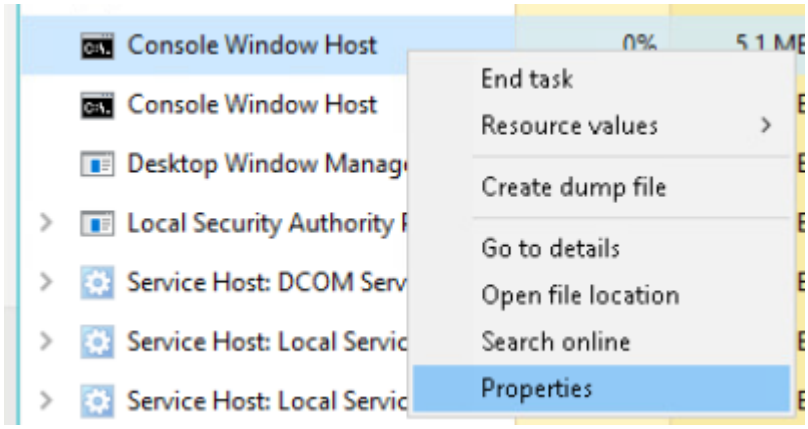

Mozilla Firefox Start Page
Firefox
Search or enter address
Search
mozilla
Search
Thanks for choosing Firefox! To get the most out of your browser, learn more about the latest features.
Downloads
Bookmarks
History
Add-ons
Sync
Options
It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!
Refresh Firefox...
11:17 AM
4/2/2025

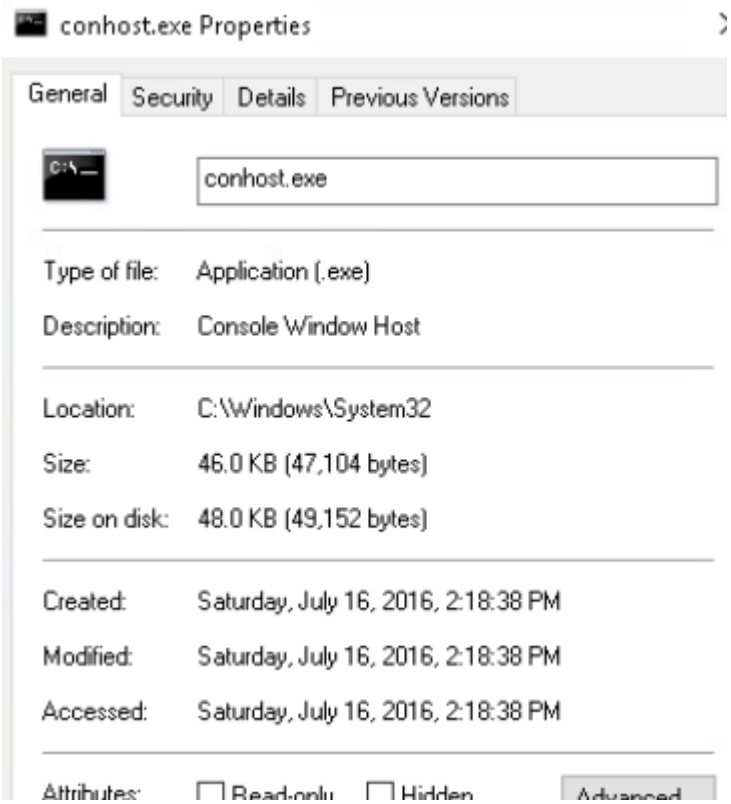


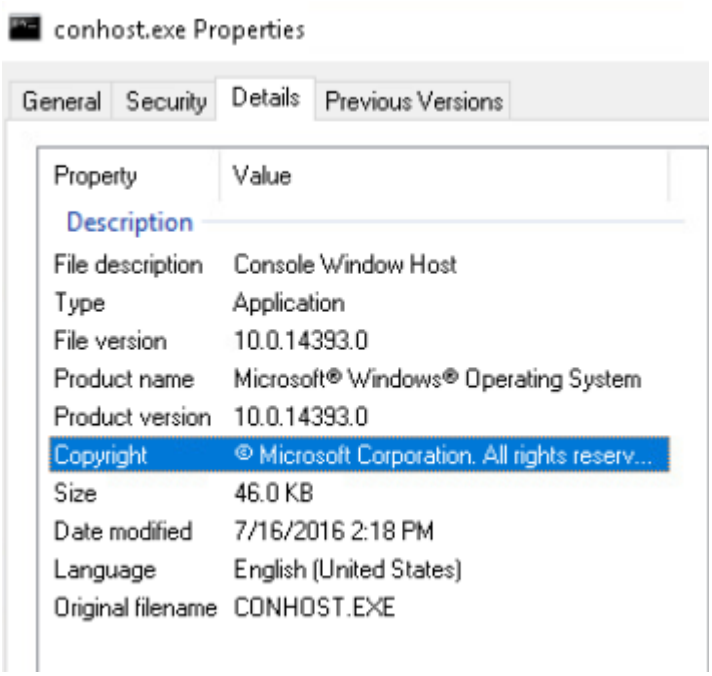
Opening **Firefox**, **Command Prompt** and **Task Manager**



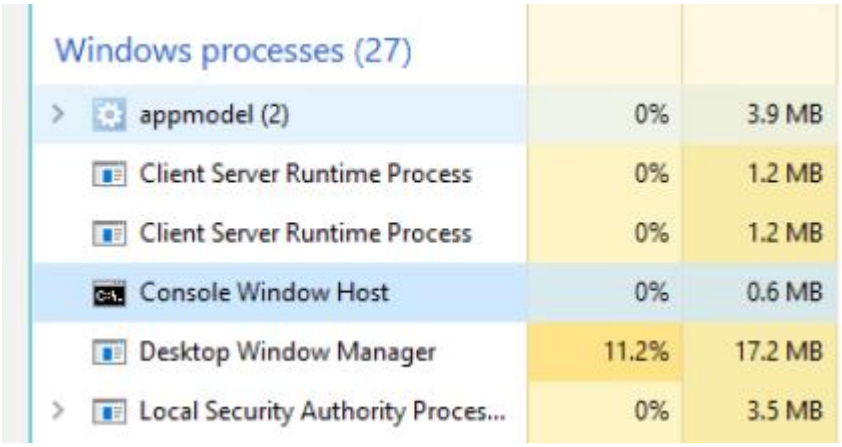
Locating the **Windows Command Processor** and expanding it (which contains **Command Prompt**)



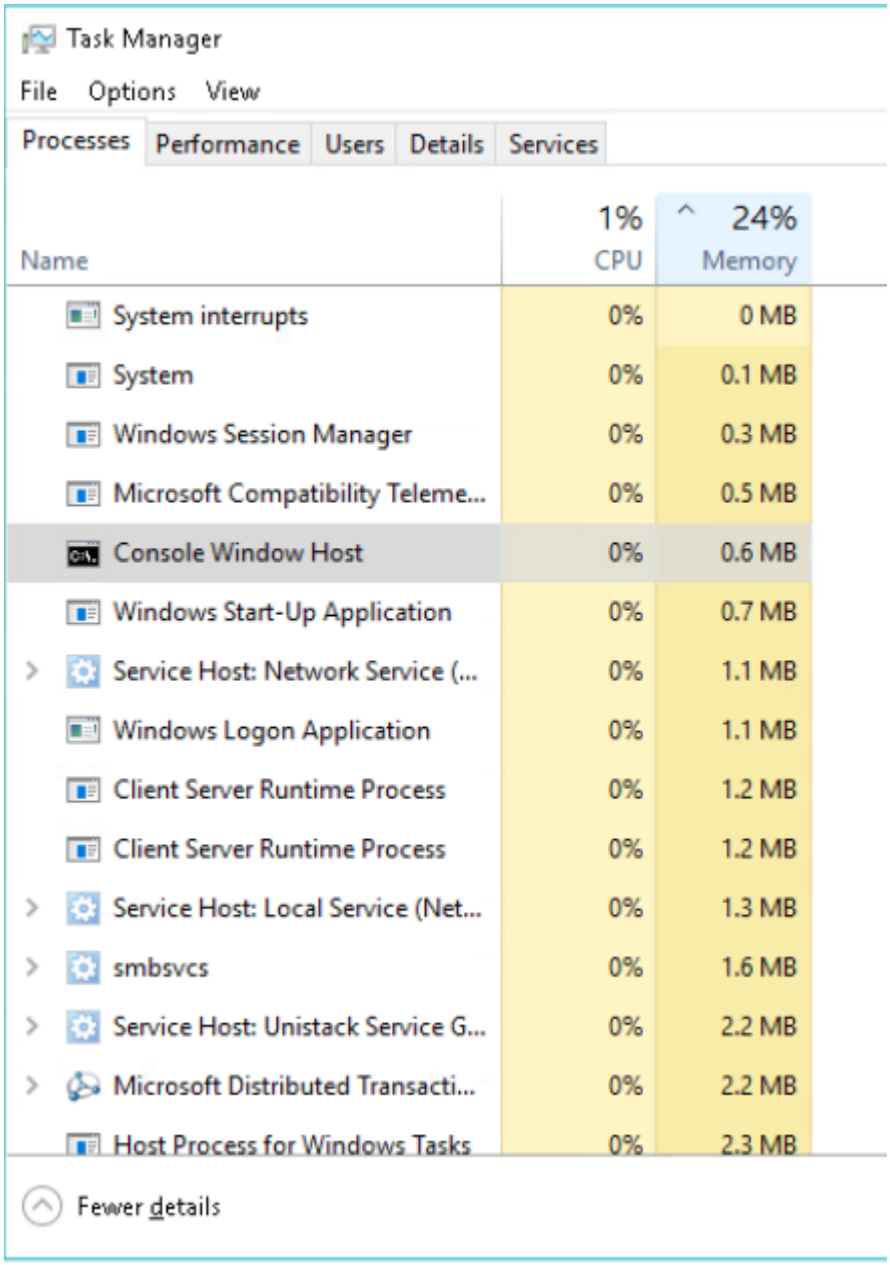




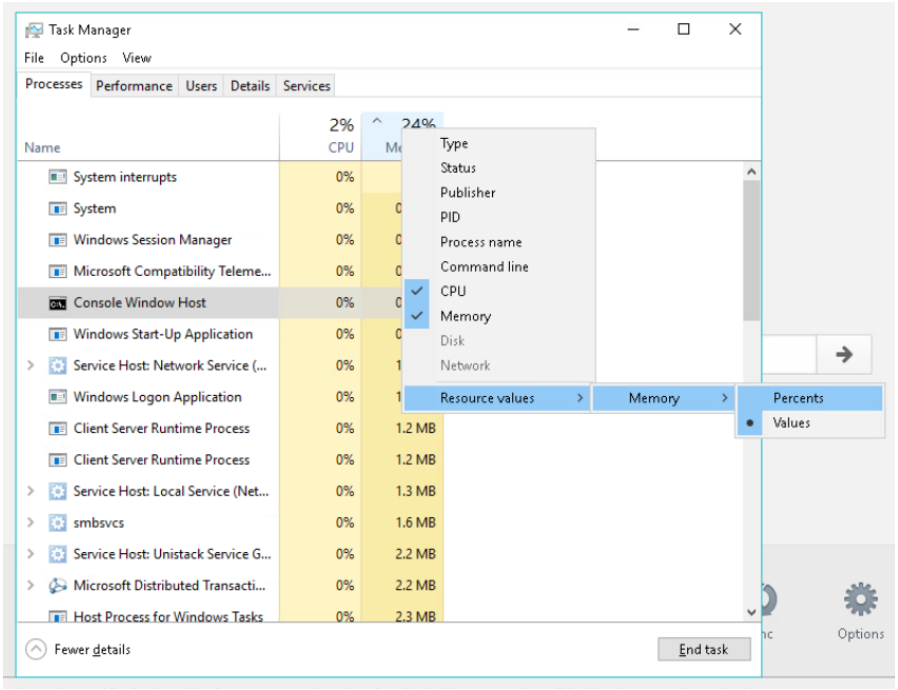
Locating **Console Window Host,** right clicking it and selecting properties will allow us to check the file name (which is found at the top-most entry field) and the location of the file – **C:\Windwos\System32**. The original file name of the file can be found in the **Details** tab.

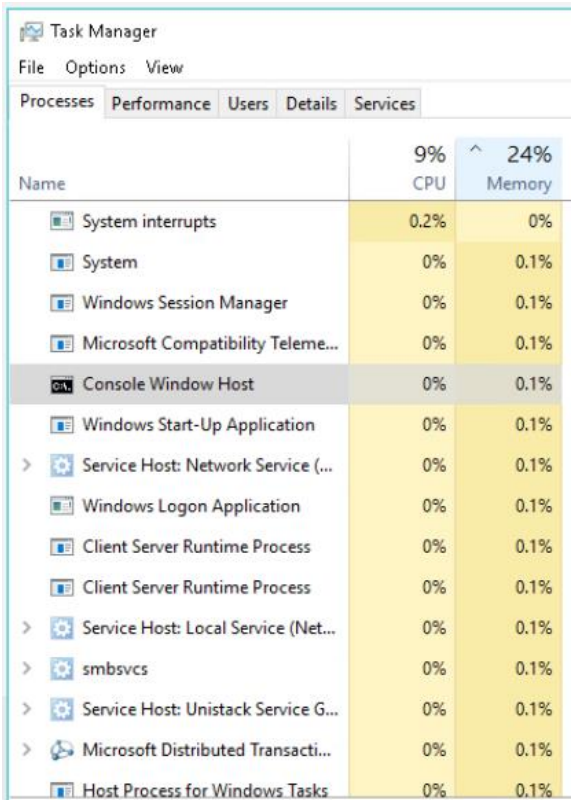


After closing the **Command Prompt** window, it will disappear from task manager and one of the **Console Window Host** processes will disappear, resulting in a slight decrease of CPU and memory usage



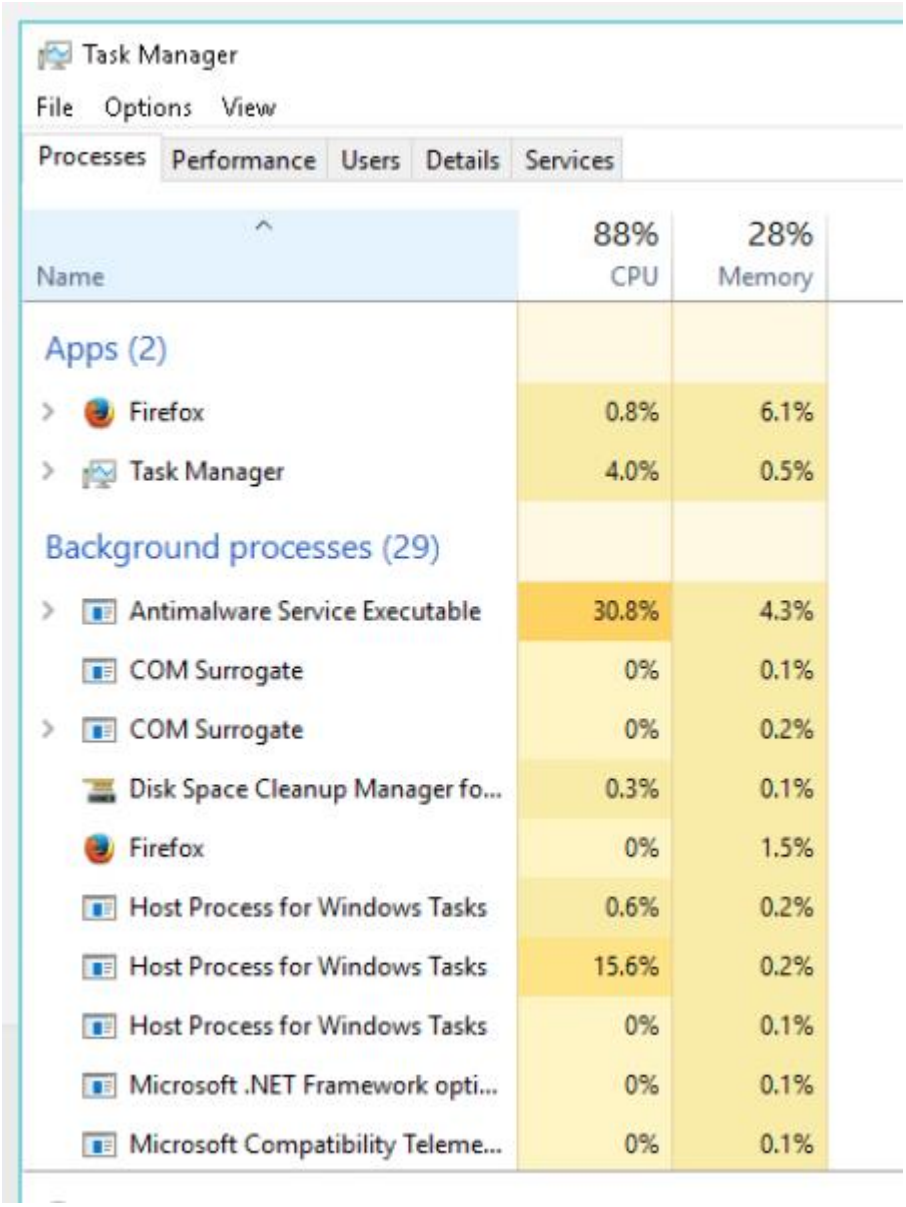
Clicking on the **Memory** heading will sort all the processes in an **Descending** order, after another click on the header will sort all the processes in a **Ascending** order.



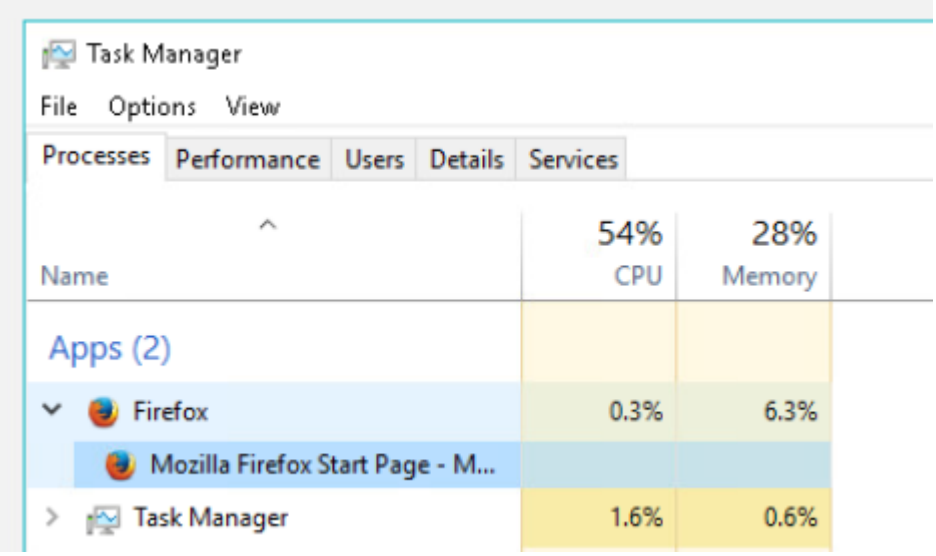


Changing the display of the resource values to **Percents** will show the percentage value of the process out of the total (in this case) memory used (in descending order)

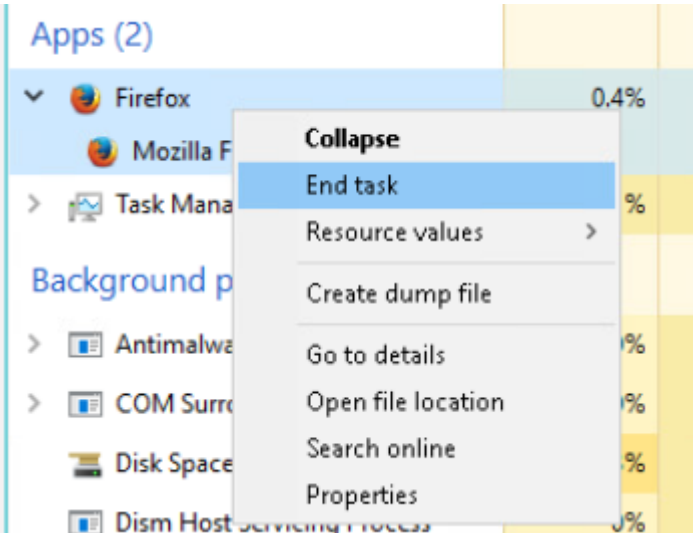
This way the user is able to locate any potentially malicious processes that are using an unusual amount of processing power, be it **CPU**, **Memory** or **Disk Usage**



Clicking on the **Name** header will sort the processes into the 3 categories **Apps**, **Background Processes**, and **Windows Processes –** in descending alphabetical order

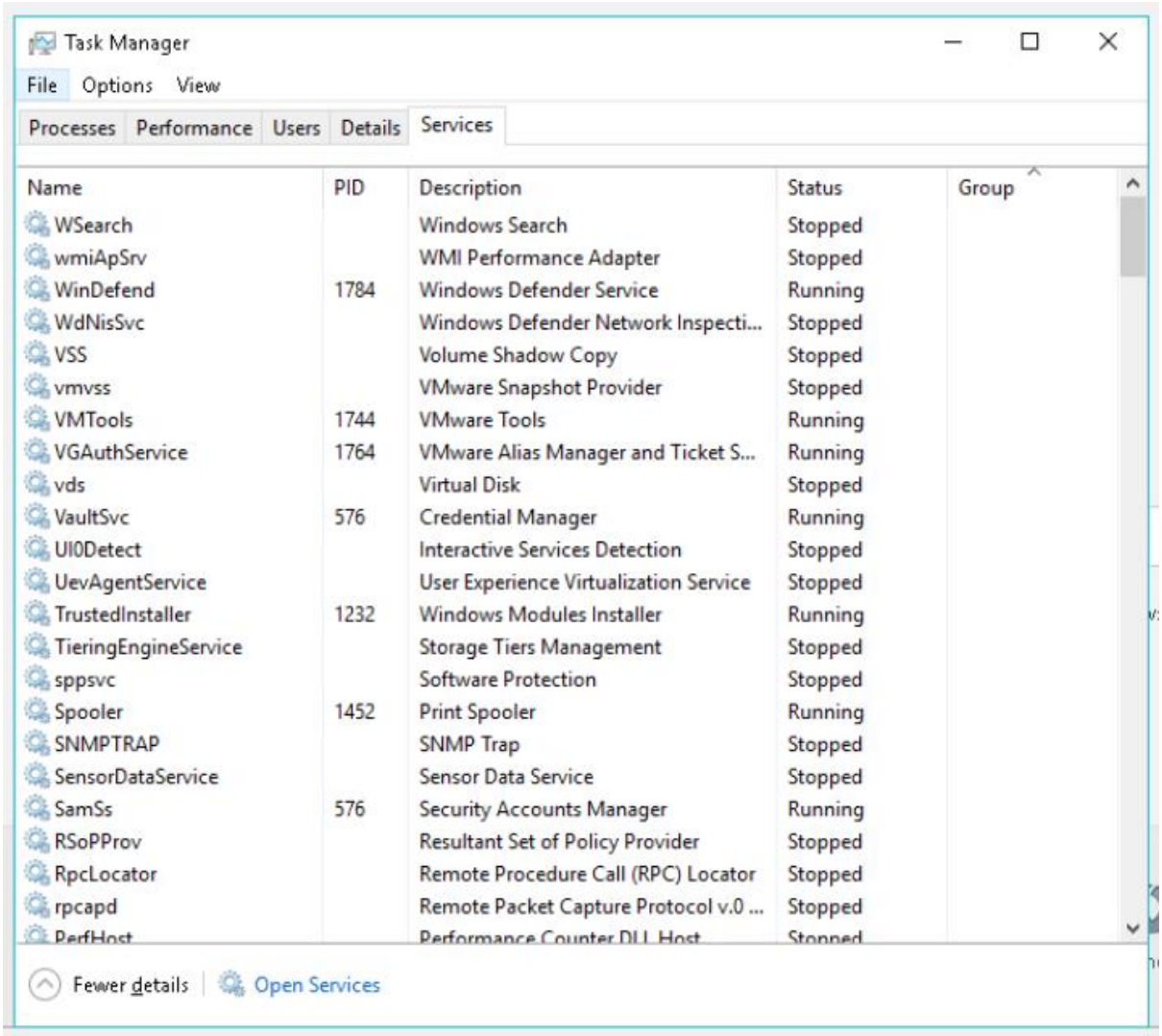


Double clicking the browser will expand the application and show all different processes that the application is using, and allow you to pinpoint which process is using what specific amount of resources



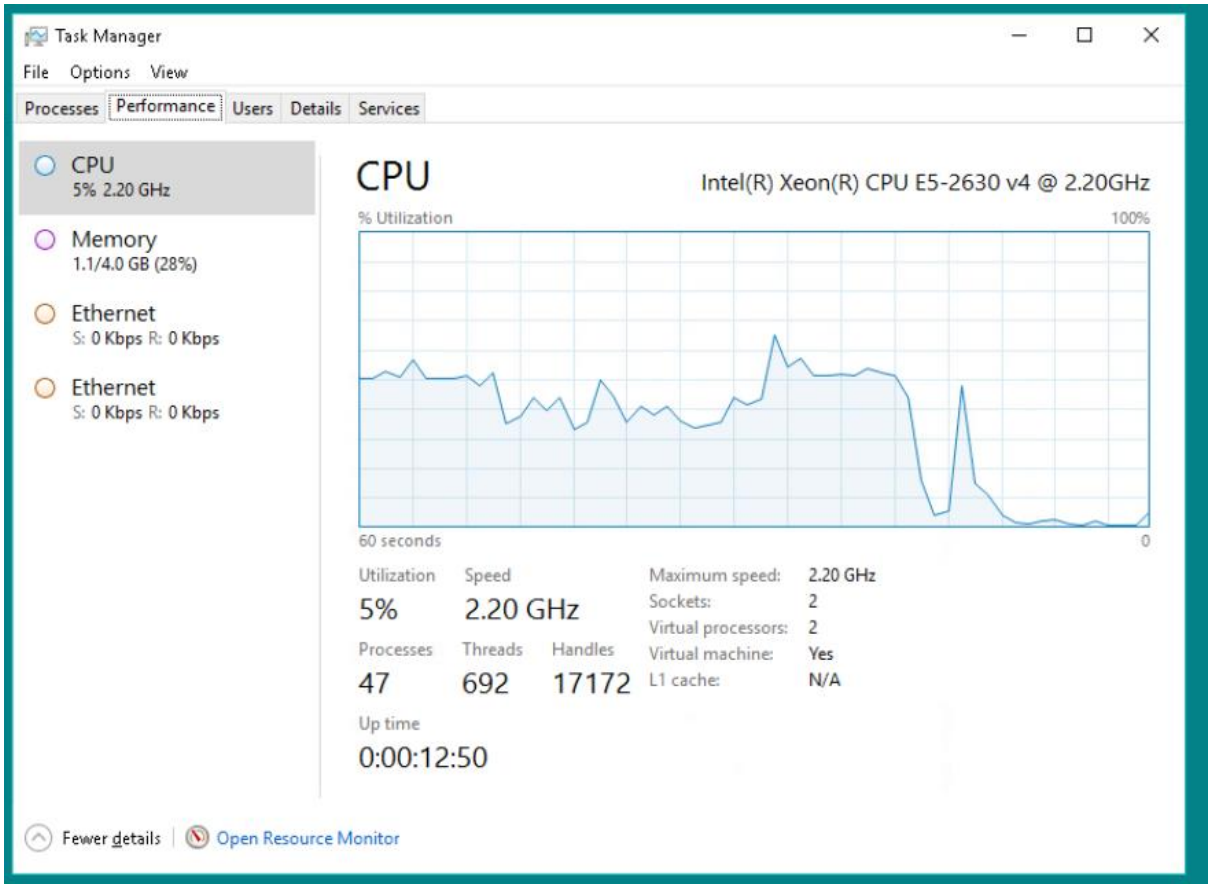
Ending the task on the browser will close the application

Part 2

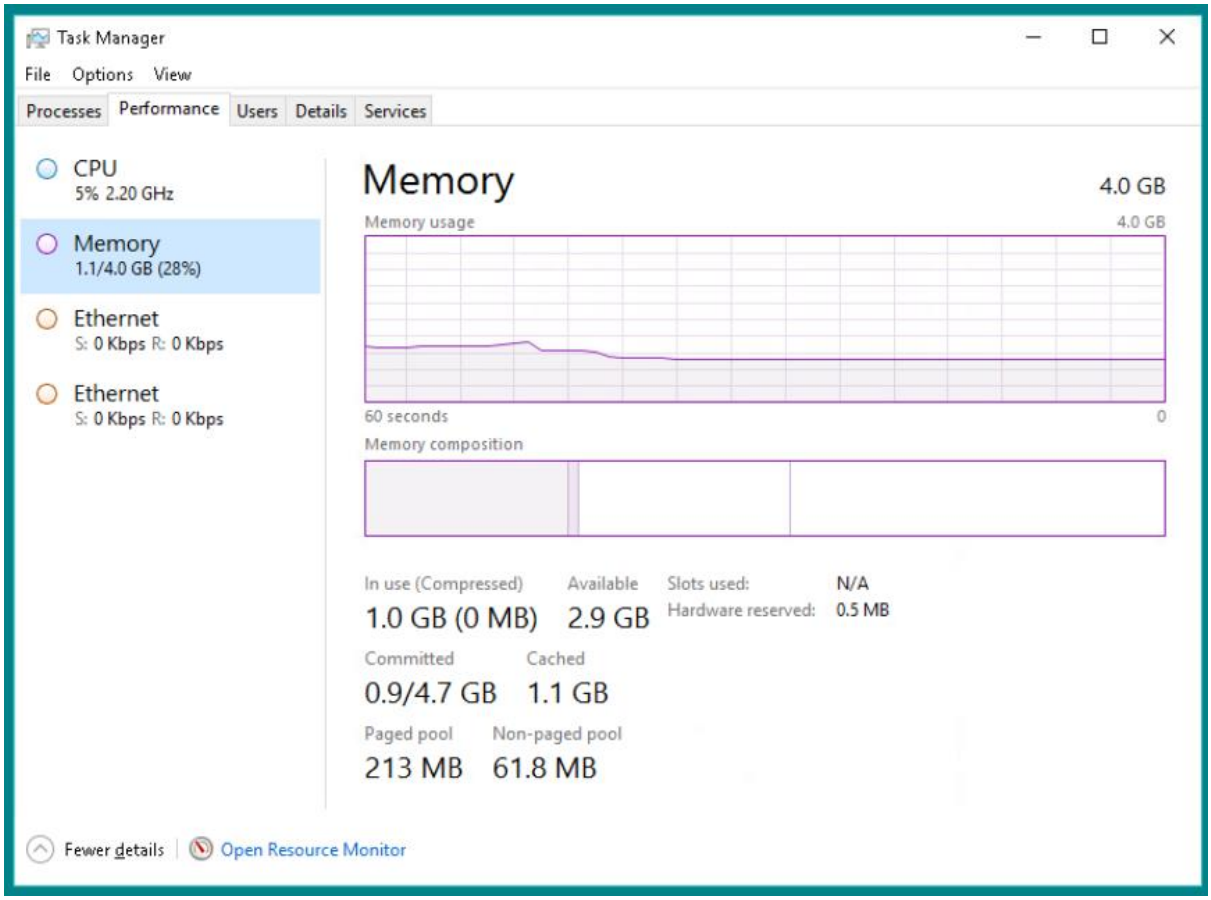


The **Services** tab shows all the windows services that run in the background. Each service has a description that comes with it, a PID (if the service is running) and the status of each service.

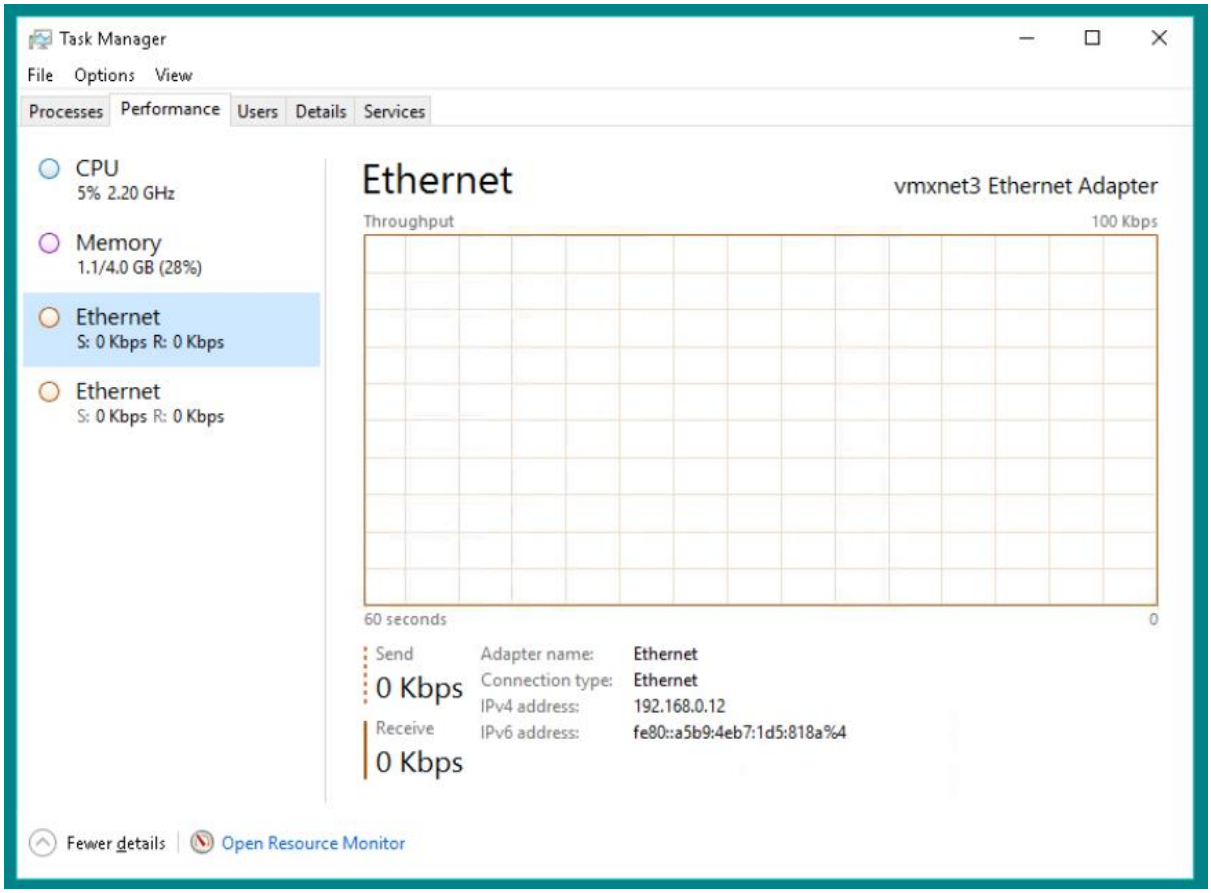
Part 3



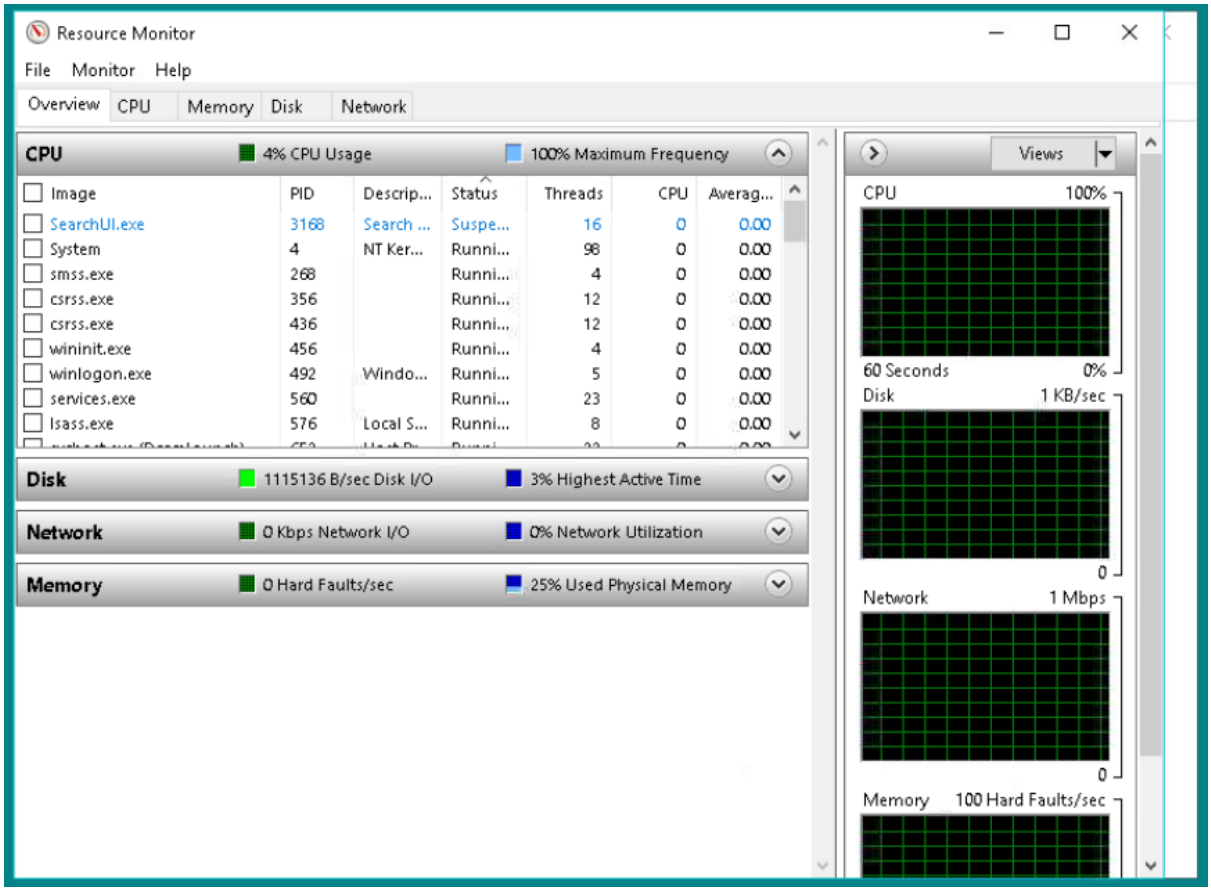
There are currently 692 in use and a total of 104 processes running



There is a total of 4.0 GB physical memory, 2.9 GB available physical memory and 1.0 GB memory in use.



The link speed is shown by the **Receive** section of ethernet section, in this instance its 0kbps as its the VM used on Netlab, and the IPv4 address of the VM is 192.168.0.12

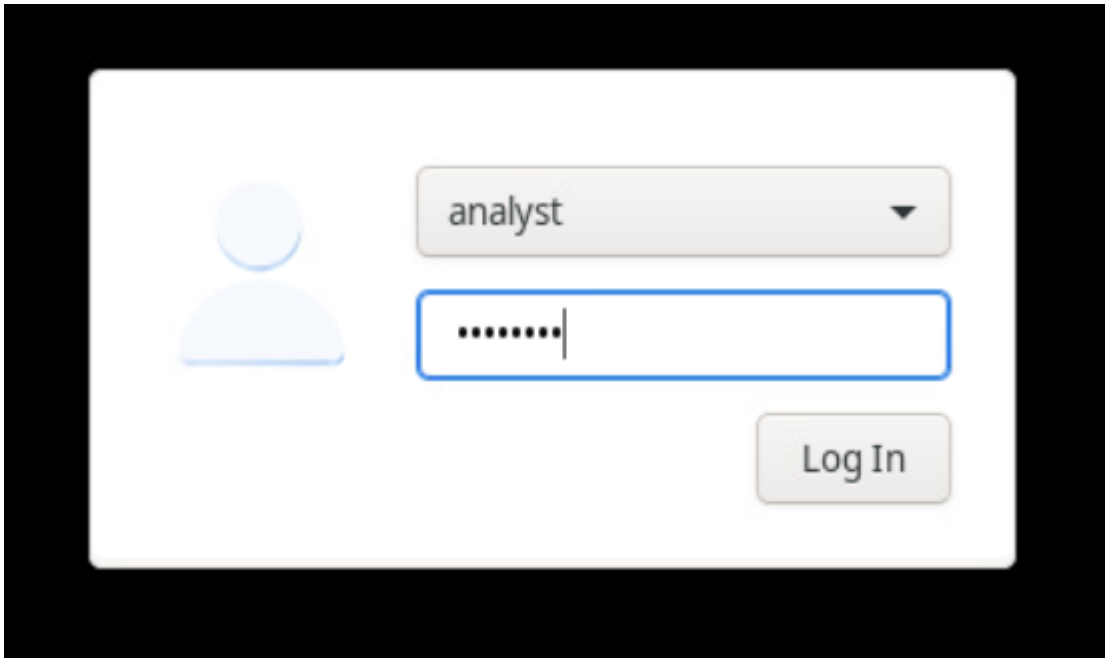

Resource Monitor
File Monitor Help
Overview CPU Memory Disk Network
CPU 4% CPU Usage 100% Maximum Frequency
Image PID Descrip... Status Threads CPU Averag...
SearchUI.exe 3168 Search ... Suspe... 16 0 0.00
System 4 NT Ker... Runni... 98 0 0.00
smss.exe 268 Runni... 4 0 0.00
csrss.exe 356 Runni... 12 0 0.00
csrss.exe 436 Runni... 12 0 0.00
wininit.exe 456 Runni... 4 0 0.00
winlogon.exe 492 Windo... Runni... 5 0 0.00
services.exe 560 Runni... 23 0 0.00
lsass.exe 576 Local S... Runni... 8 0 0.00
Disk 1115136 B/sec Disk I/O 3% Highest Active Time
Network 0 Kbps Network I/O 0% Network Utilization
Memory 0 Hard Faults/sec 25% Used Physical Memory
Views
CPU 100%
60 Seconds 0%
Disk 1 KB/sec
0
Network 1 Mbps
0
Memory 100 Hard Faults/sec

Module 8: Linux Labs

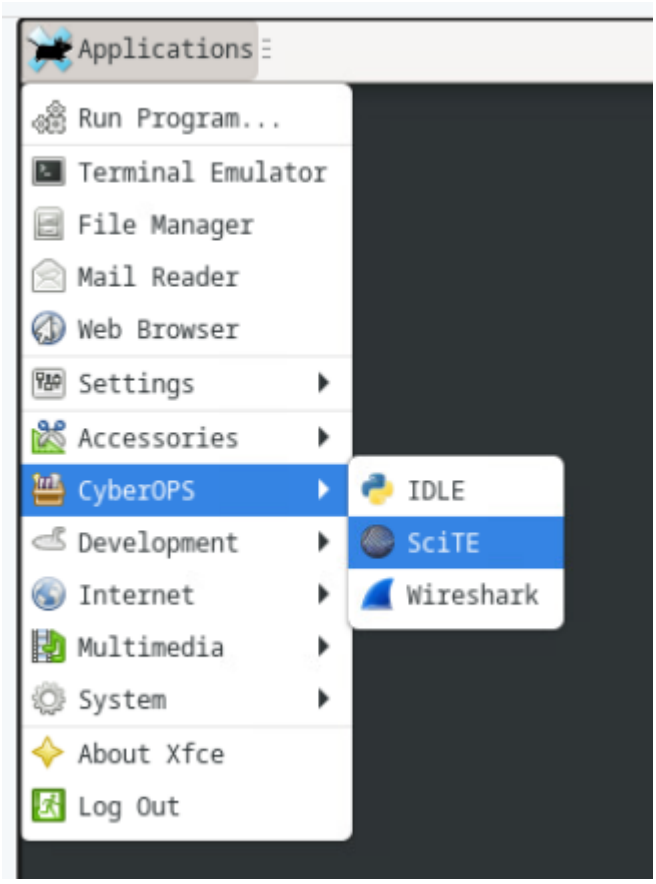
8.2.6 Lab – Working with Text Files in the CLI

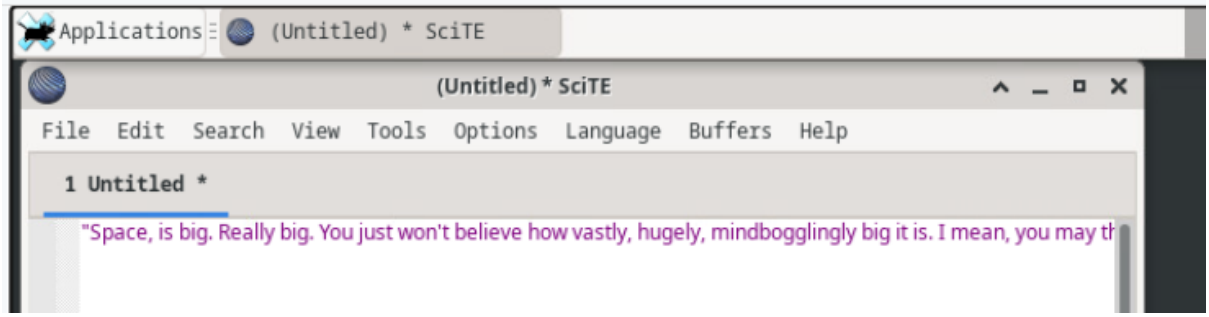
Part 1

Step 1

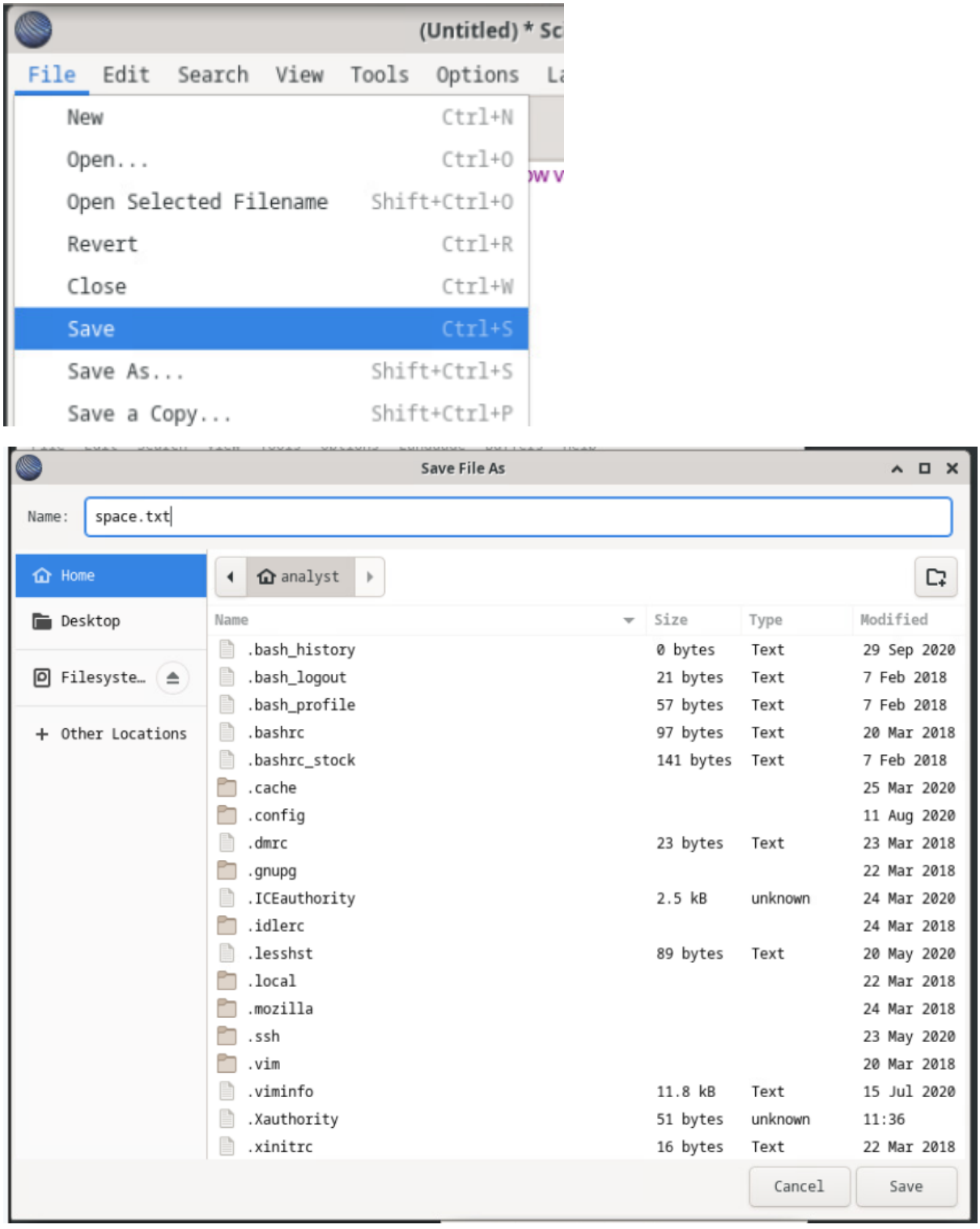


Logging into the VM

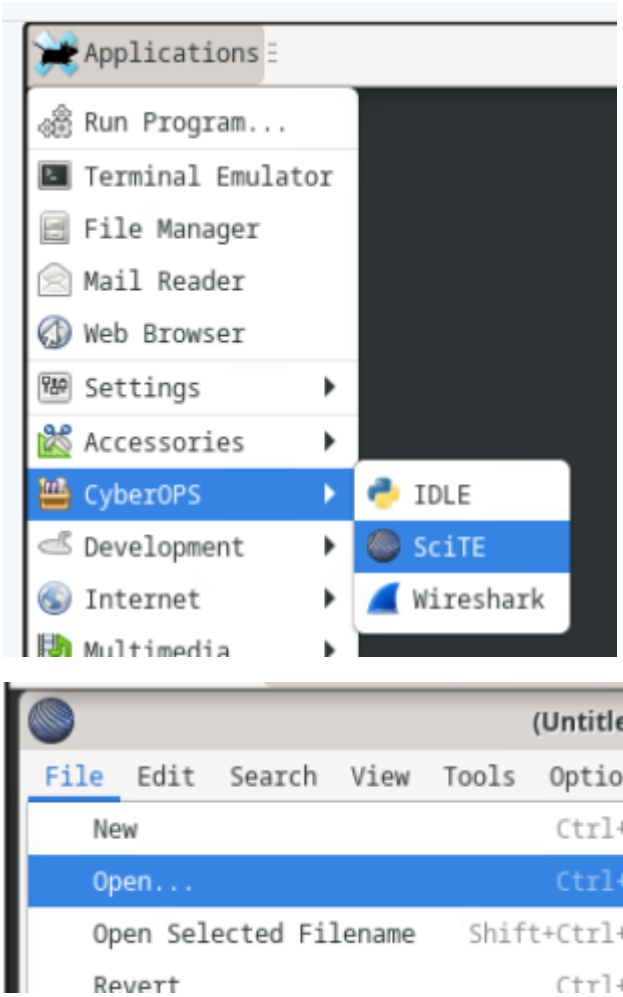




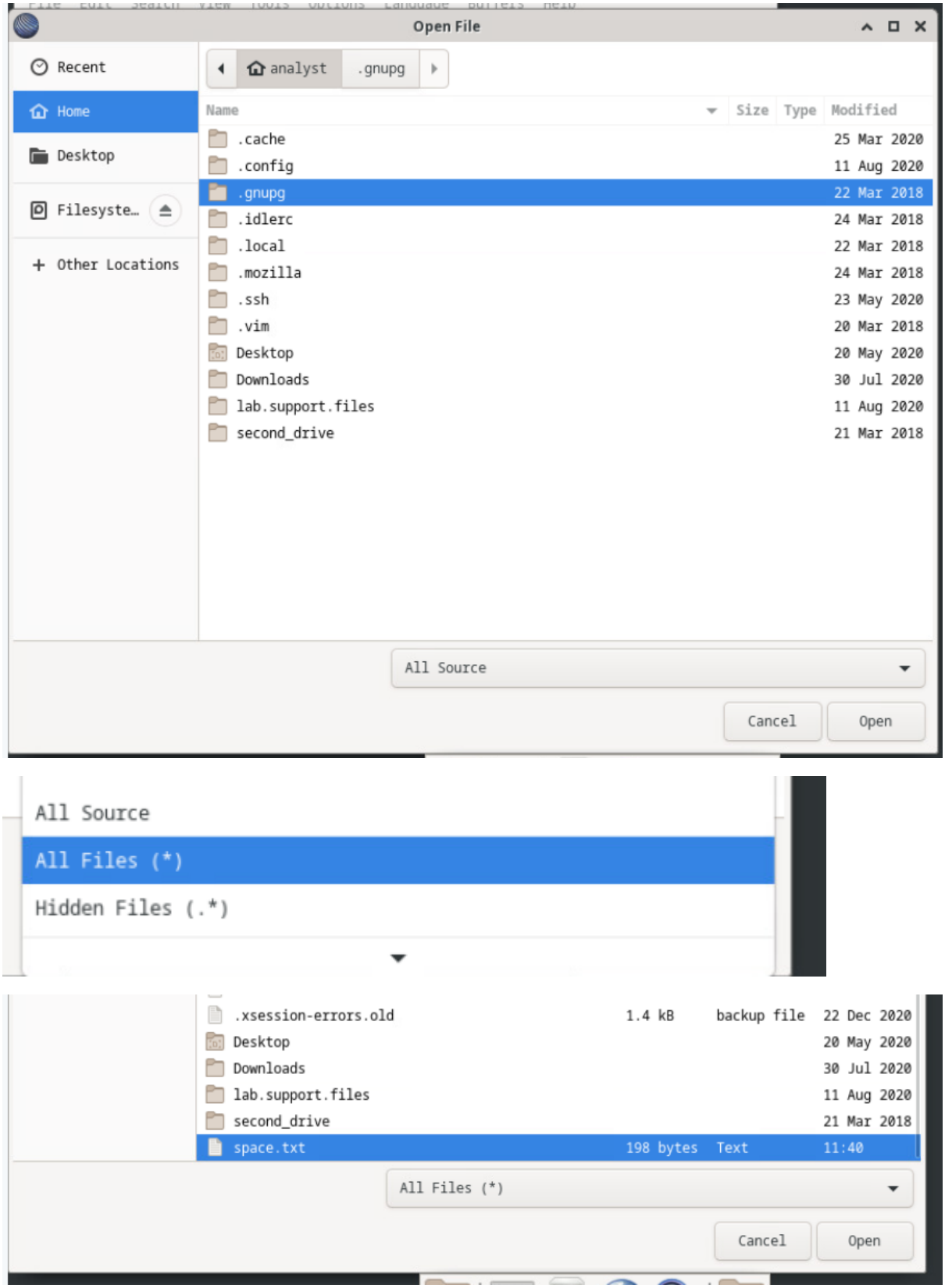
Opening the **SciTE** text editor and entering some text inside it



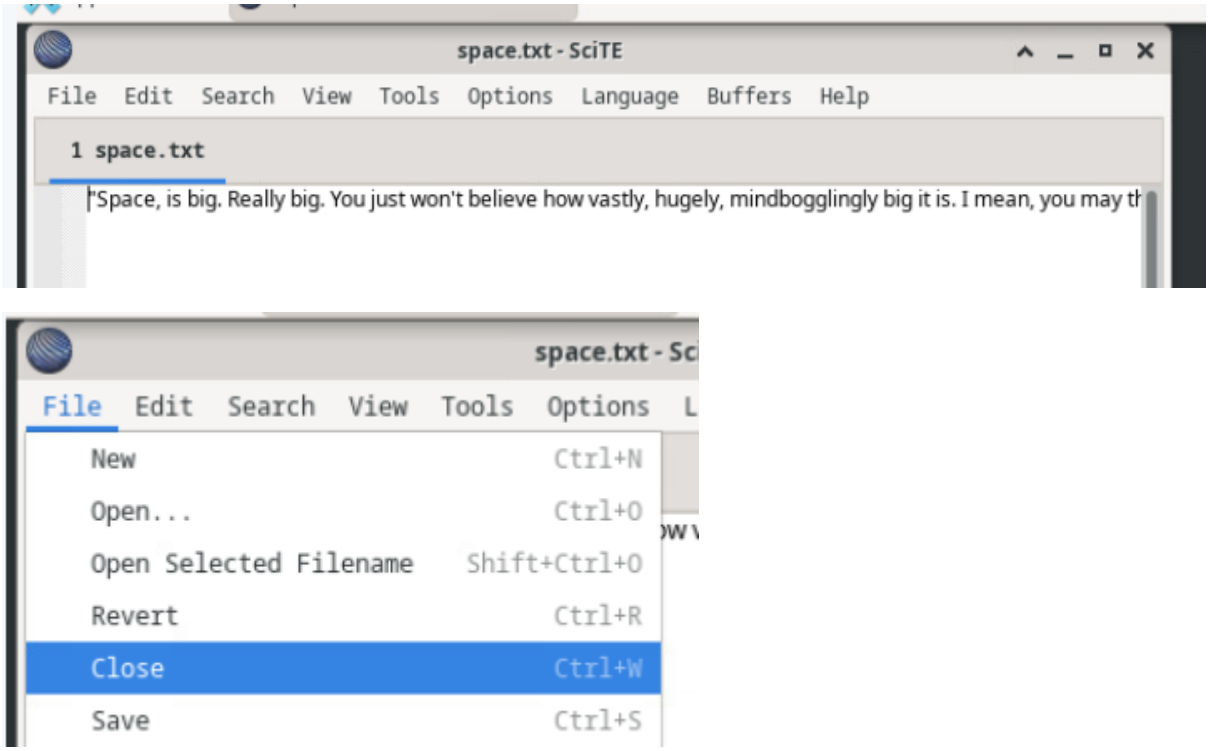
Saving the text file in the **home directory**



Re-opening the text editor and locating the text file

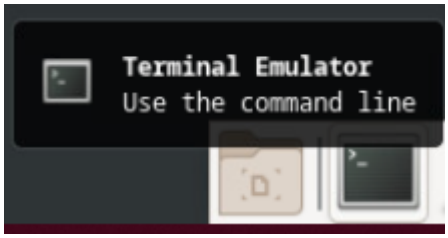


The text file is hidden due to the "**.txt**" file extension not classified as a "known extension" to the editor, as such you need to change visibility to **All Files (*)**



Closing the .txt file

Step 2



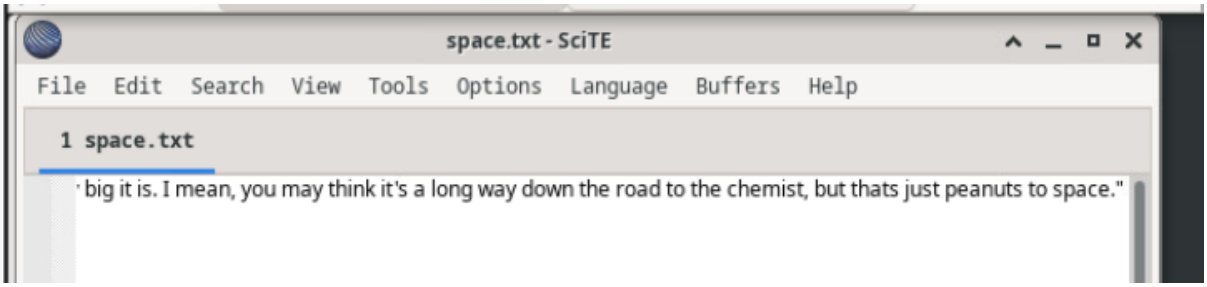
Opening terminal

```
[analyst@secOps ~]$ ls
Desktop  Downloads  lab.support.files  second_drive  space.txt
[analyst@secOps ~]$
```

```
[analyst@secOps ~]$ scite space.txt
```

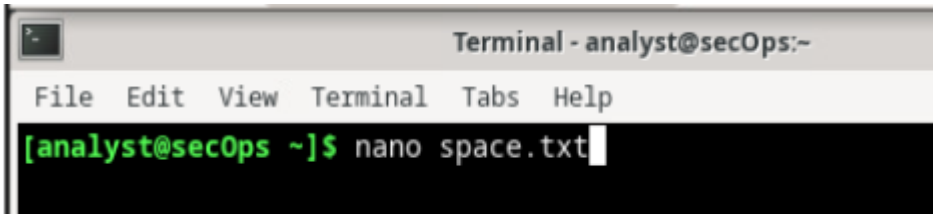
Opening the .txt file and loading it using the command "**scite**"

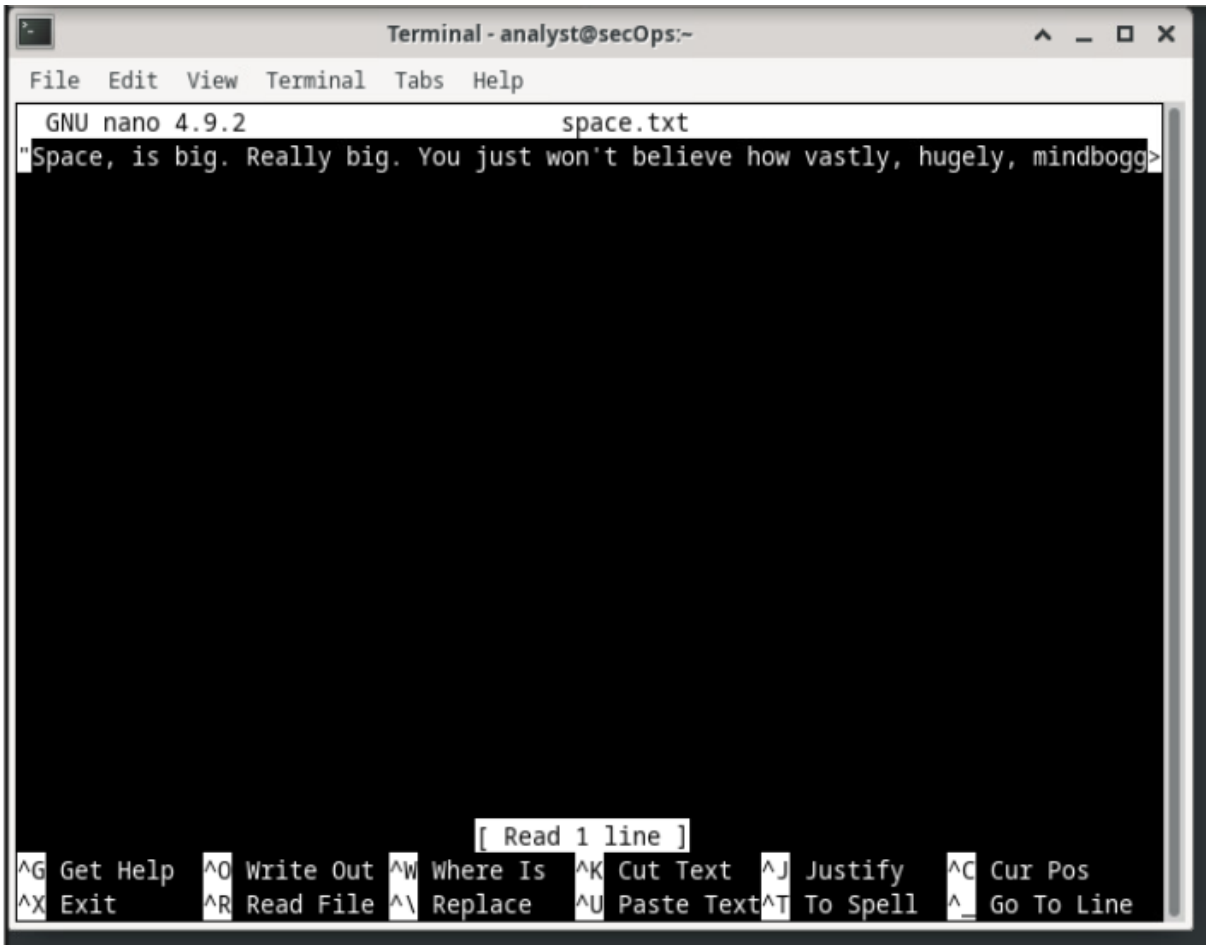
The next prompt doesn't show as the terminal currently has a process running – you need to interrupt the process to be able to enter another prompt



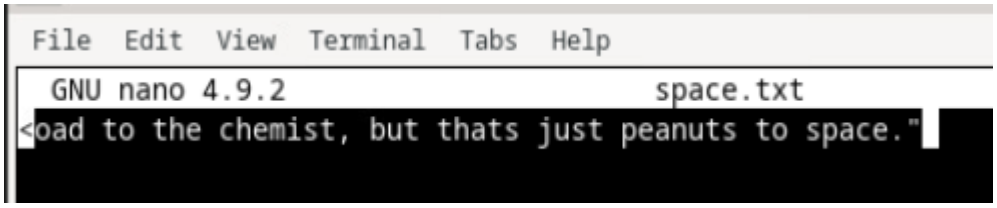

```
[analyst@secOps ~]$ scite space.txt
[analyst@secOps ~]$
```

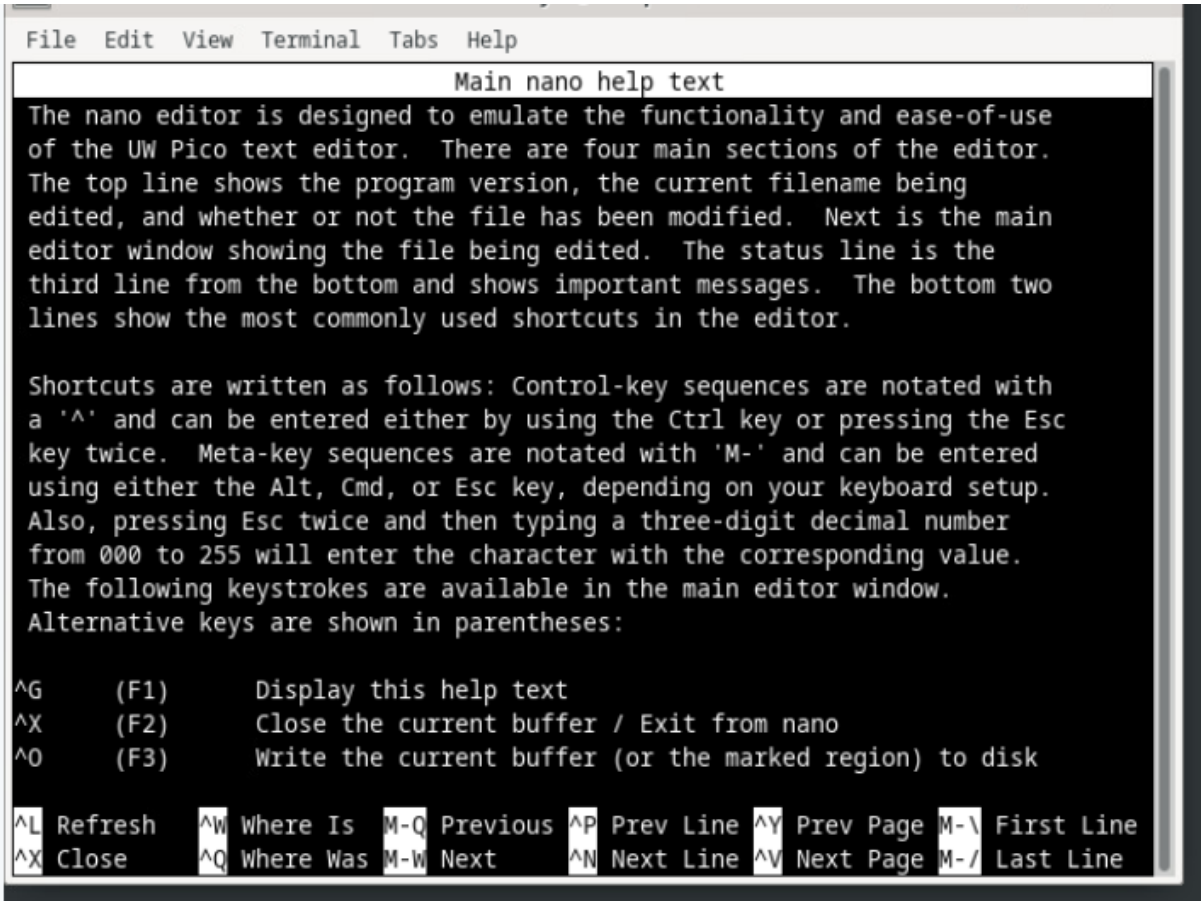
Part 2





Using the command **nano** followed by the **.txt** file, will display the contents of the file. The UI shows that there is more to a line using the **>** symbol found near the end of the border.

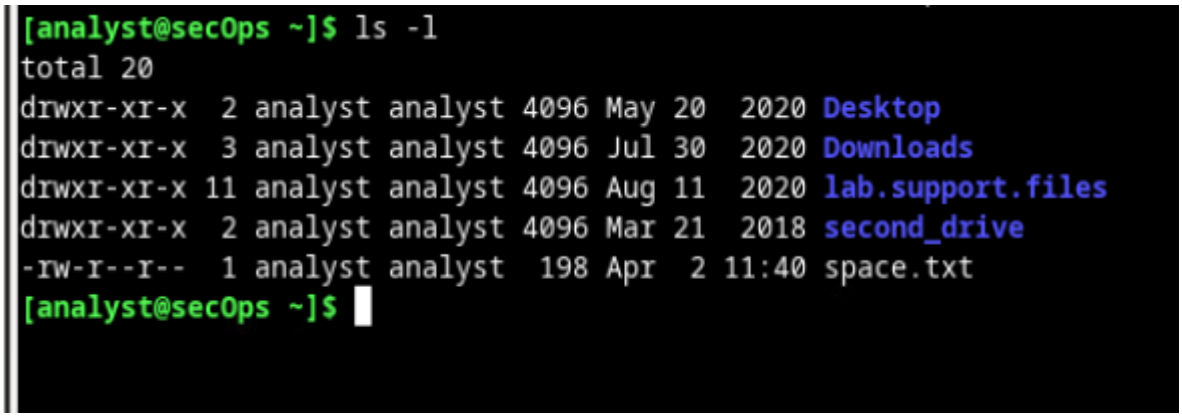




Accessing the help screen of **nano** is done by pressing **CTRL + G**.

Part 3

Step 1



Using the **ls -l** command to list all the files in the home directory

```
[analyst@secOps ~]$ ls -la
total 124
drwx------ 14 analyst analyst  4096 Apr  2 11:40 .
drwxr-xr-x  3 root    root     4096 Mar 20  2018 ..
-rw-------  1 analyst analyst    67 Apr  2 11:49 .bash_history
-rw-r--r--  1 analyst analyst    21 Feb  7  2018 .bash_logout
-rw-r--r--  1 analyst analyst    57 Feb  7  2018 .bash_profile
-rw-r--r--  1 analyst analyst    97 Mar 20  2018 .bashrc
-rw-r--r--  1 analyst analyst   141 Feb  7  2018 .bashrc_stock
drwxr-xr-x  8 analyst analyst  4096 Mar 25  2020 .cache
drwxr-xr-x 10 analyst analyst  4096 Aug 11  2020 .config
drwxr-xr-x  2 analyst analyst  4096 May 20  2020 Desktop
-rw-r--r--  1 analyst analyst    23 Mar 23  2018 .dmrc
drwxr-xr-x  3 analyst analyst  4096 Jul 30  2020 Downloads
drwx------  3 analyst analyst  4096 Mar 22  2018 .gnupg
-rw-------  1 analyst analyst  2520 Mar 24  2020 .ICEauthority
drwxr-xr-x  2 analyst analyst  4096 Mar 24  2018 .idlerc
drwxr-xr-x 11 analyst analyst  4096 Aug 11  2020 lab.support.files
-rw-------  1 analyst analyst    89 May 20  2020 .lesshst
drwxr-xr-x  3 analyst analyst  4096 Mar 22  2018 .local
drwx------  5 analyst analyst  4096 Mar 24  2018 .mozilla
drwxr-xr-x  2 analyst analyst  4096 Mar 21  2018 second_drive
-rw-r--r--  1 analyst analyst   198 Apr  2 11:40 space.txt
drwx------  2 analyst analyst  4096 May 23  2020 .ssh
drwxr-xr-x  3 analyst analyst  4096 Mar 20  2018 .vim
-rw-------  1 analyst analyst 11848 Jul 15  2020 .viminfo
-rw-------  1 analyst analyst    51 Apr  2 11:36 .Xauthority
-rw-r--r--  1 analyst analyst    16 Mar 22  2018 .xinitrc
-rw-r--r--  1 analyst analyst    16 Mar 22  2018 .Xinitrc
-rw-------  1 analyst analyst  1177 Apr  2 11:36 .xsession-errors
-rw-------  1 analyst analyst  1413 Dec 22  2020 .xsession-errors.old
[analyst@secOps ~]$
```

Using the **ls –la** to show all the files in the home directory (including the hidden files)

```
[analyst@secOps ~]$ cat .bashrc
export EDITOR=vim

PS1='\[\e[1;32m\][\u@\h \W]\$\[\e[0m\] '
alias ls="ls --color"
alias vi="vim"
[analyst@secOps ~]$
```

Using the **cat** command to display the contents of the **.bashrc** file

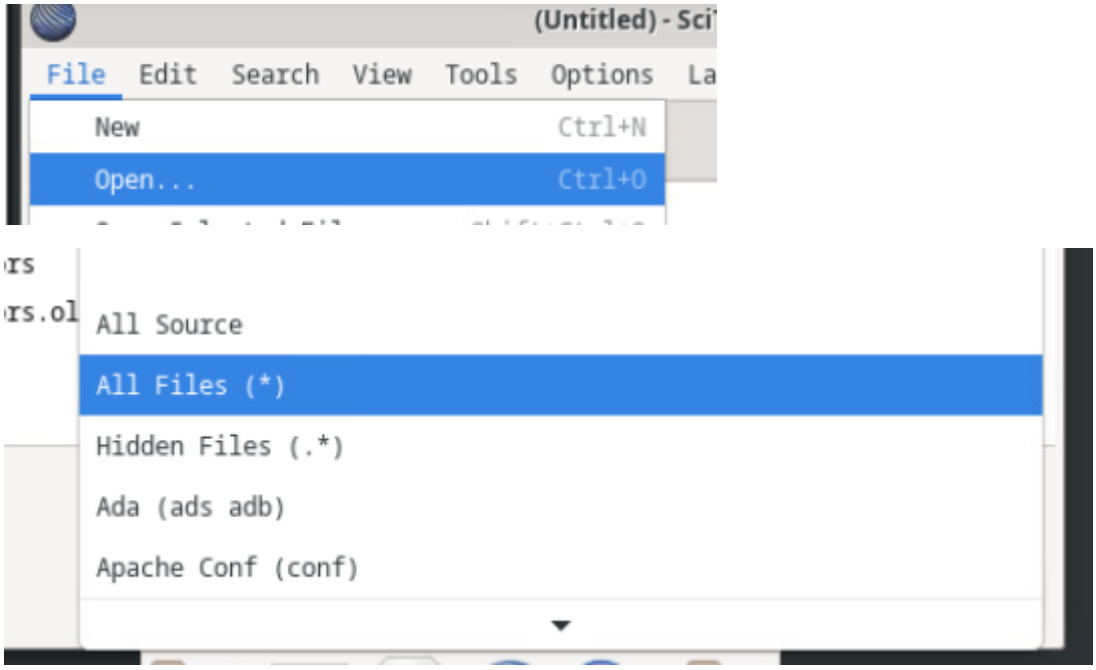
```
[analyst@secOps ~]$ ls /etc
adjtime                 init.d              nanorc              services
apparmor.d              inputrc             netconfig           shadow
arch-release            iproute2            netctl              shadow-
audisp                  iptables            nginx               shells
audit                   issue               nscd.conf           skel
avahi                   kernel              nsswitch.conf       snort
bash.bash_logout        krb5.conf           ntp.conf            ssh
bash.bashrc             ld.so.cache         openldap            ssl
bindresvport.blacklist  ld.so.conf          openvswitch         sudoers
binfmt.d                ld.so.conf.d        os-release          sudoers.d
ca-certificates         libaudit.conf       pacman.conf         sysctl.d
conf.d                  libnl               pacman.conf.pacnew  syslog-ng
crypttab                libva.conf          pacman.d            systemd
dconf                   lightdm             pam.d               tmpfiles.d
default                 locale.conf         passwd              trusted-key.key
depmod.d                locale.gen          passwd-             ts.conf
dhcpcd.conf             locale.gen.pacnew   pcmcia              udev
e2scrub.conf            localtime           pkcs11              UPower
environment             login.defs          polkit-1            vbox
ethertypes              logrotate.conf      profile             vconsole.conf
fonts                   logrotate.d         profile.d           vdpau_wrapper.cfg
```

Using the **ls** command to show the contents of the **/etc** directory

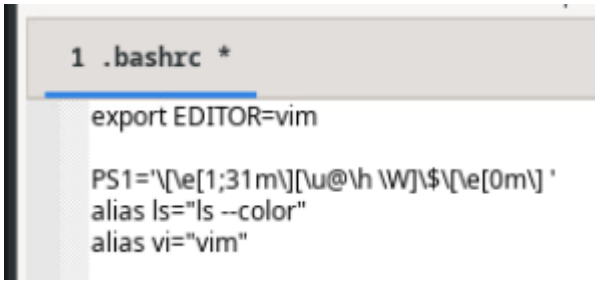
```
[analyst@secOps ~]$ cat /etc/bash.bashrc
#
# /etc/bash.bashrc
#

# If not running interactively, don't do anything
[[ $- != *i* ]] && return

[[ $DISPLAY ]] && shopt -s checkwinsize

PS1='[\u@\h \W]\$ '

case ${TERM} in
  xterm*|rxvt*|Eterm|aterm|kterm|gnome*)
    PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND; }'printf "\033]0;%s@%s:%s\007" "${USER}"
"${HOSTNAME%%.*}" "${PWD/#$HOME/\~}"'

    ;;
  screen*)
    PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND; }'printf "\033_%s@%s:%s\033\\" "${USER}"
"${HOSTNAME%%.*}" "${PWD/#$HOME/\~}"'
    ;;
esac

[ -r /usr/share/bash-completion/bash_completion   ] && . /usr/share/bash-completion/bash_comple
tion
[analyst@secOps ~]$
```

Using the **cat** command to display the **bash.bashrc** file (from the **/etc** directory)

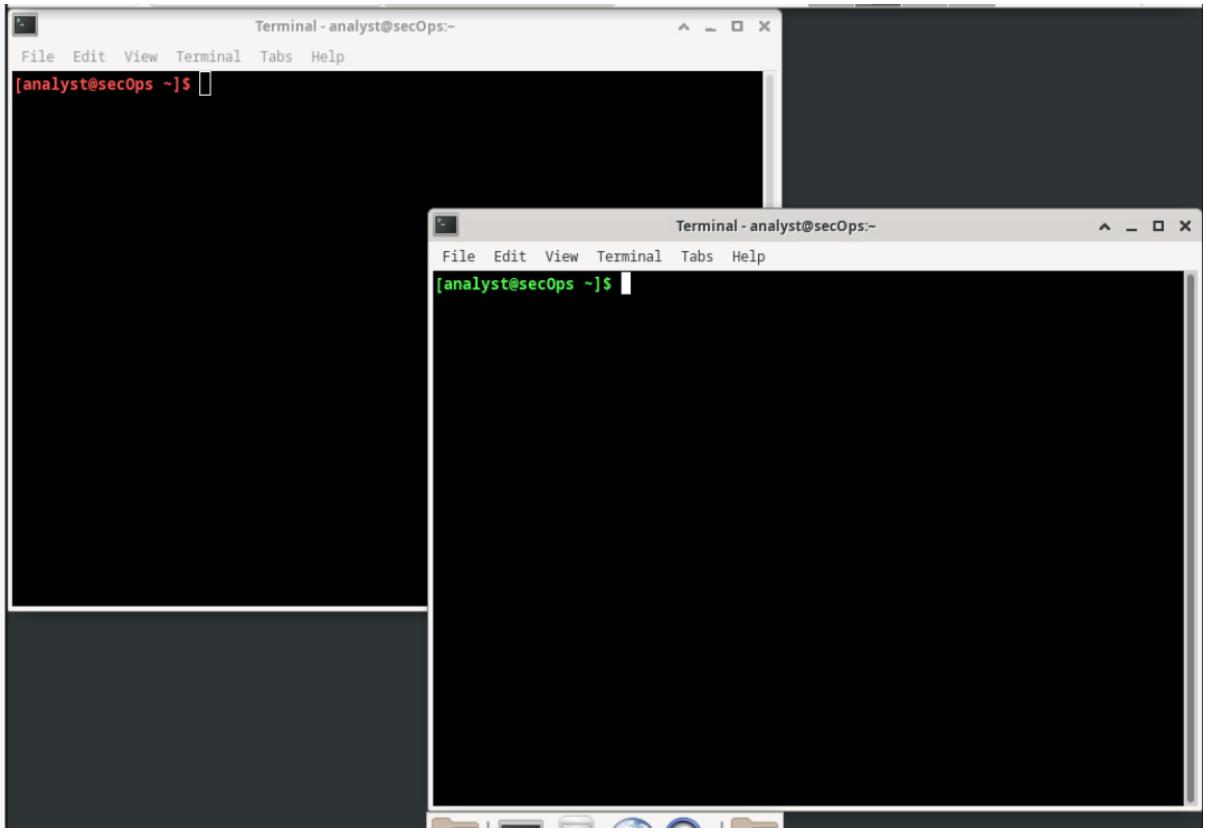
Step 2



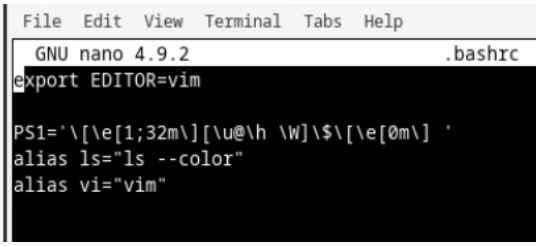
Selecting **All Files (*)** to see the hidden files

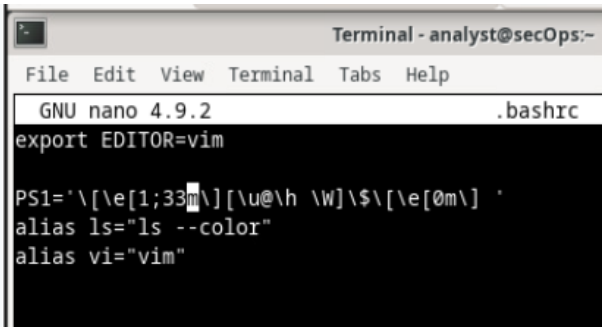


Opening the **.bashrc** file and changing the colour code from 32 (green) to 31 (red)

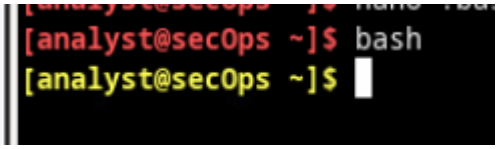


After saving the file the terminal that is already open does not change colour to red, but if a new one is open then the colour of the new terminal is changed to red





Opening the **.bashrc** file using the **nano** command, and changing the colour code to 33 (yellow)



Using the command **bash** will reload the terminal and the colour of the text in the terminal will update.

Step 3

```
[analyst@secOps ~]$ sudo nano -l /etc/nginx/custom_server.conf
[sudo] password for analyst:
```

Using the **sudo nano -l** command to access the configuration file in the **/etc/nginx** directory

```
37
38      server {
39          listen          8080;
40          server_name  localhost;
41
```

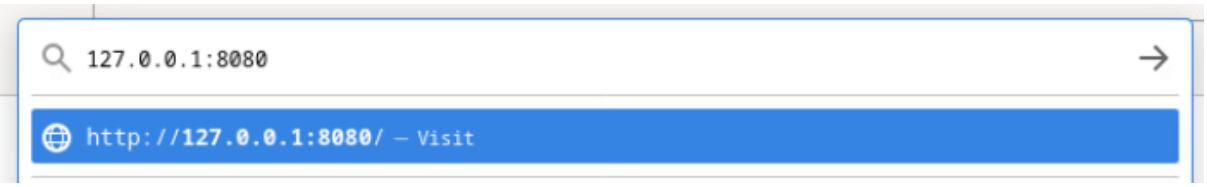
```
46          location / {
47              root   /usr/share/nginx/html/text_ed_lab/;
48              index  index.html index.htm;
49          }
```

```
Save modified buffer?
 Y Yes
 N No            ^C Cancel
```

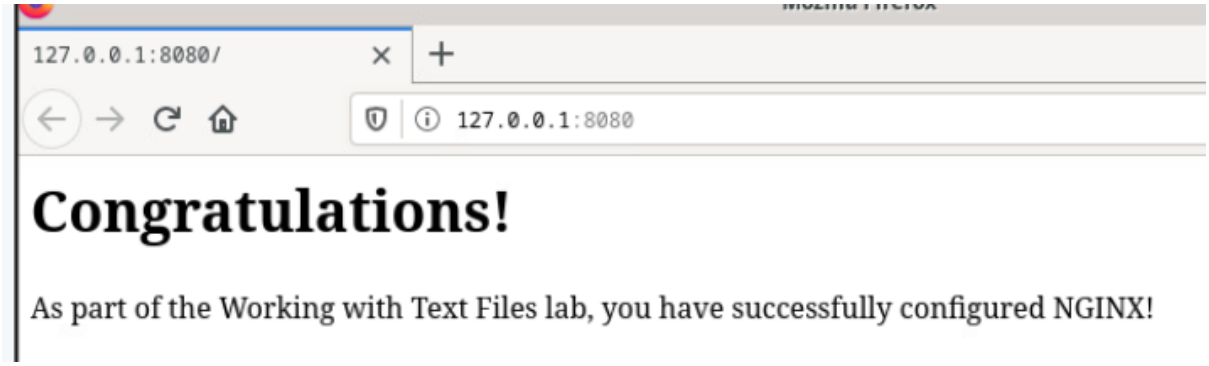
Changing the port number from **81** to **8080** and changing the root path from **/usr/share/nginx/html/** to **/usr/share/nginx/text_ed_lab/**

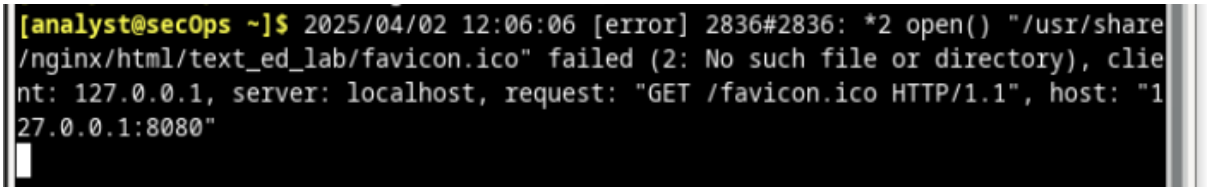
```
[analyst@secOps ~]$ sudo nginx -c custom_server.conf
```

Executing the new configuration

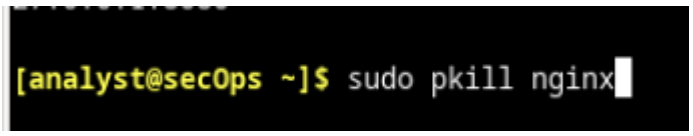


Accessing the website on the new port

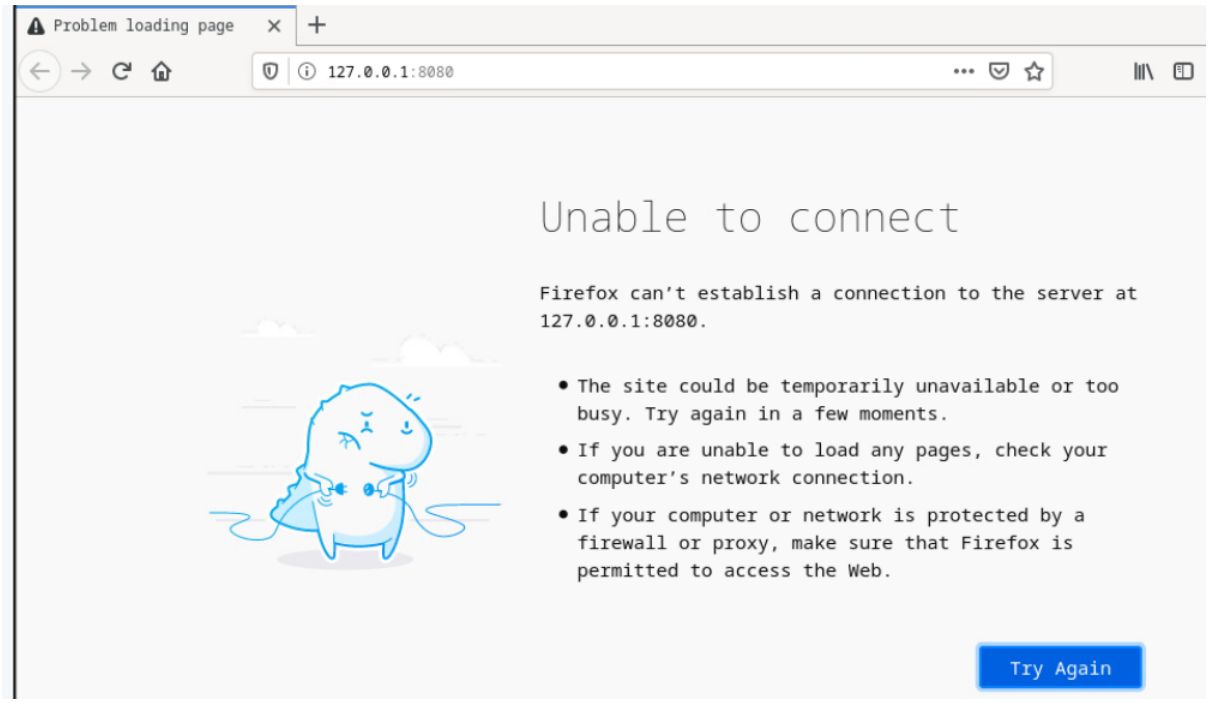




The error message is coming up due to a missing file "**favicon.ico**"



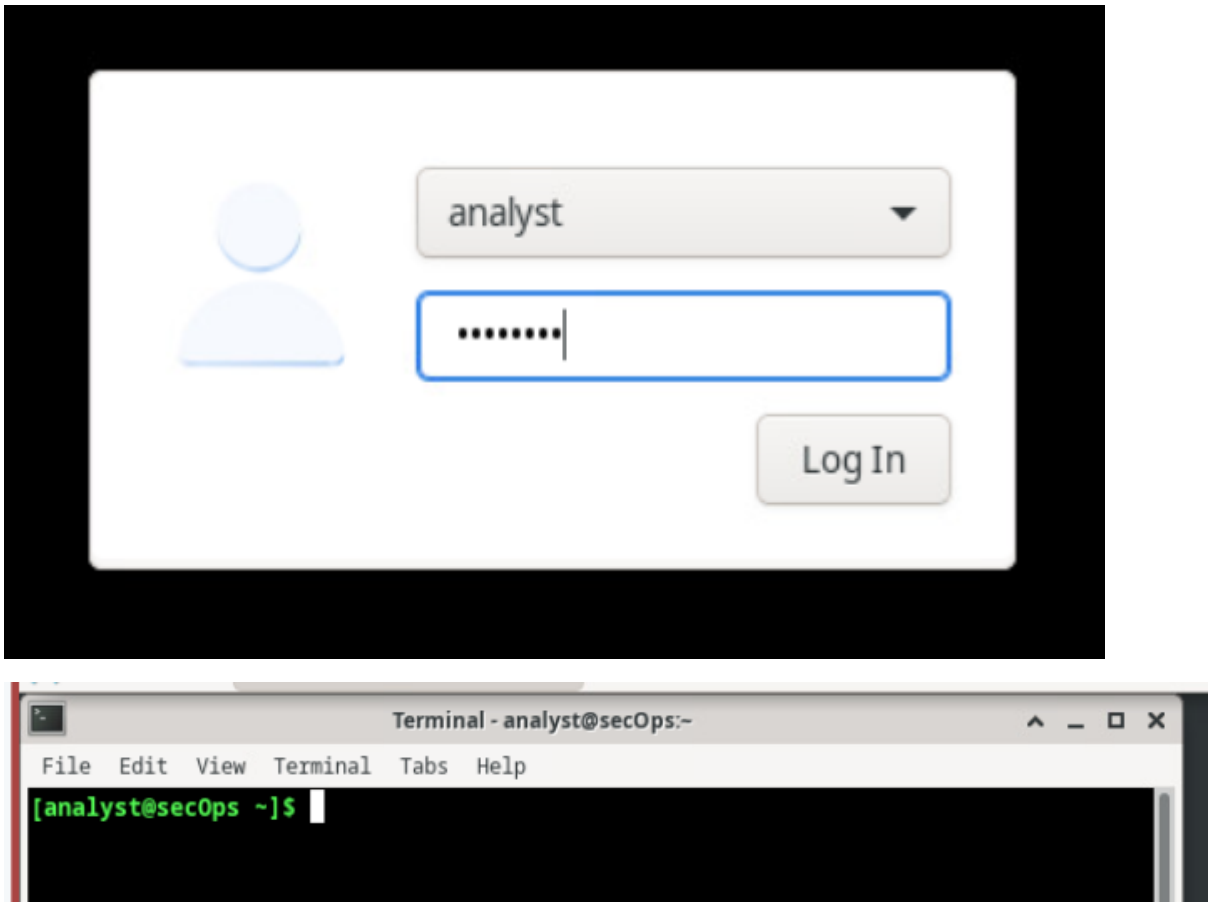
Using the **sudo pkill nginx** will terminate the website



8.2.7 Lab – Getting Familiar with the Linux Shell

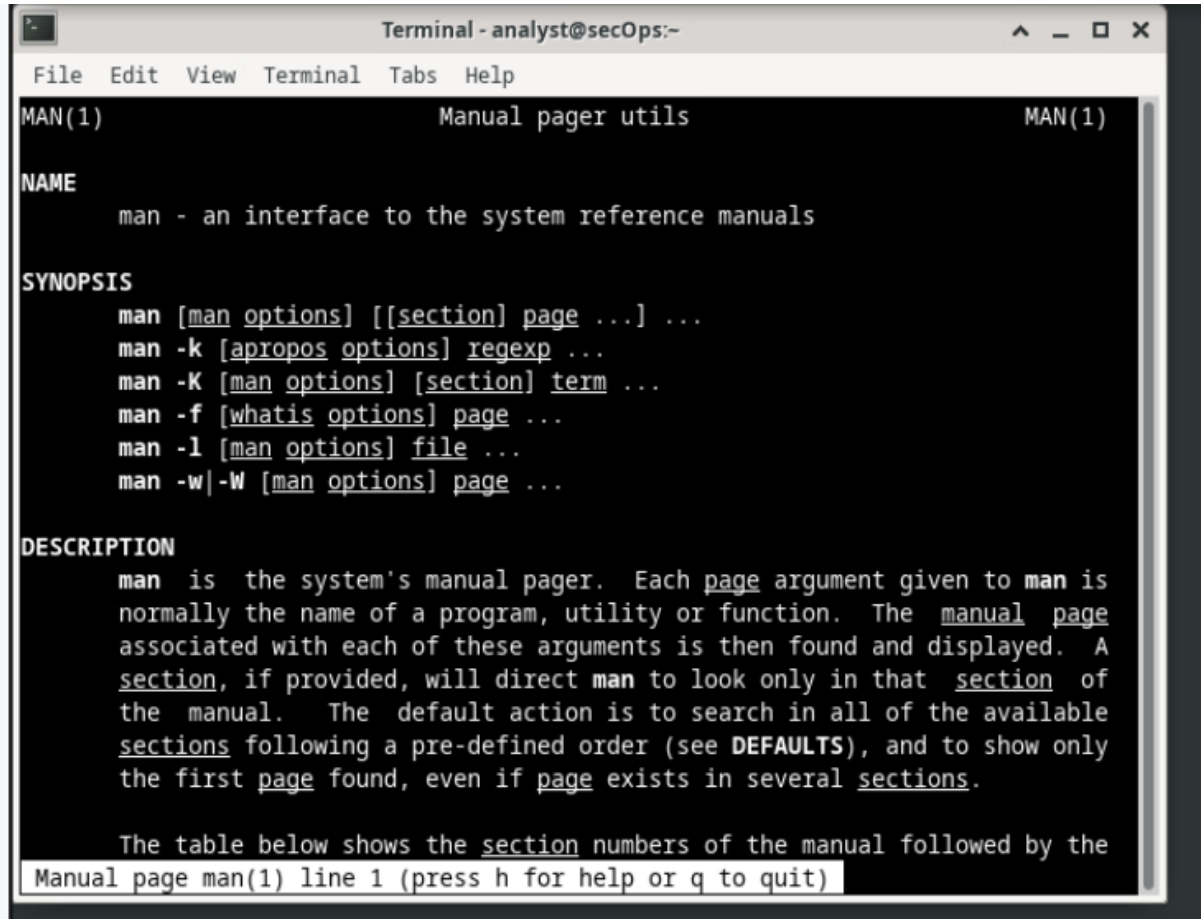
Part 1

Step 1



Logging into the VM on netlab and opening terminal

Step 2

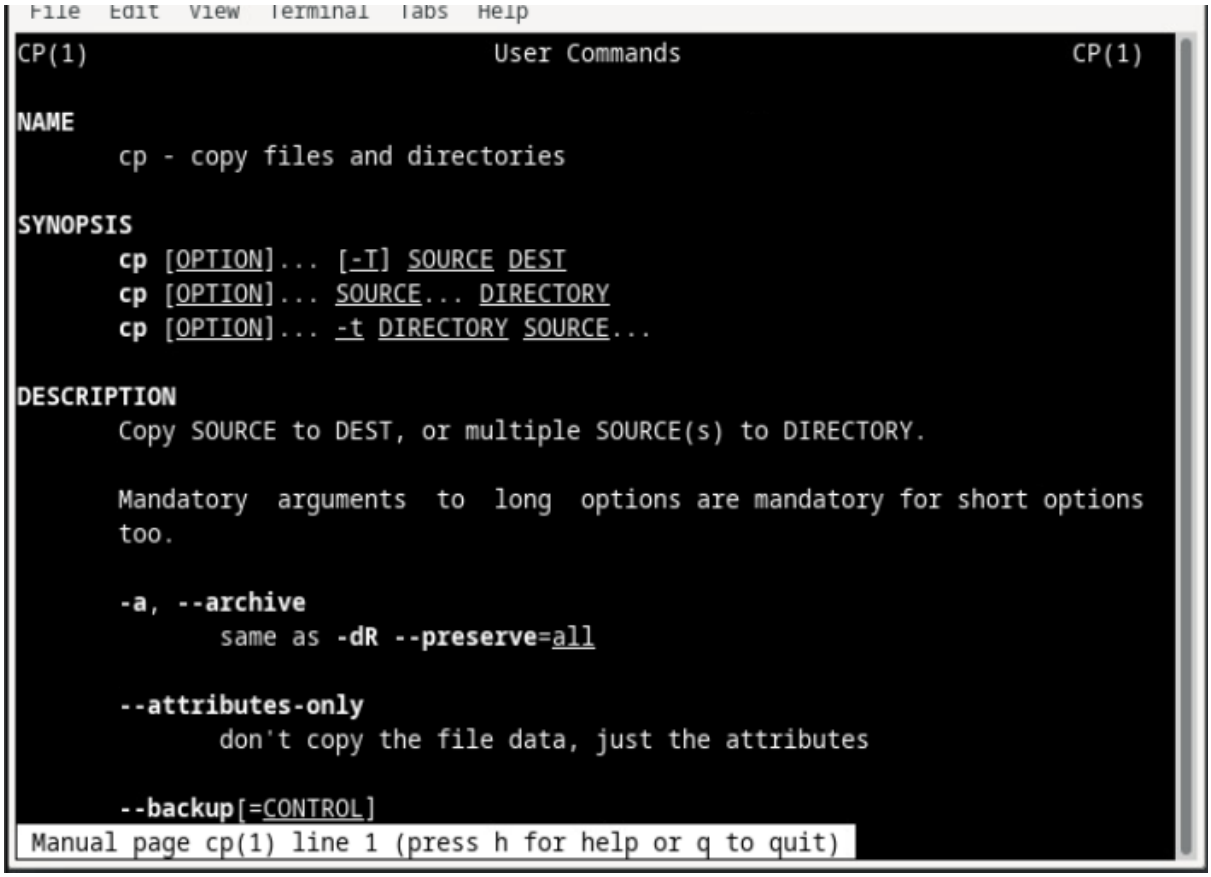


Opening the man (manual) page using **man man** in terminal

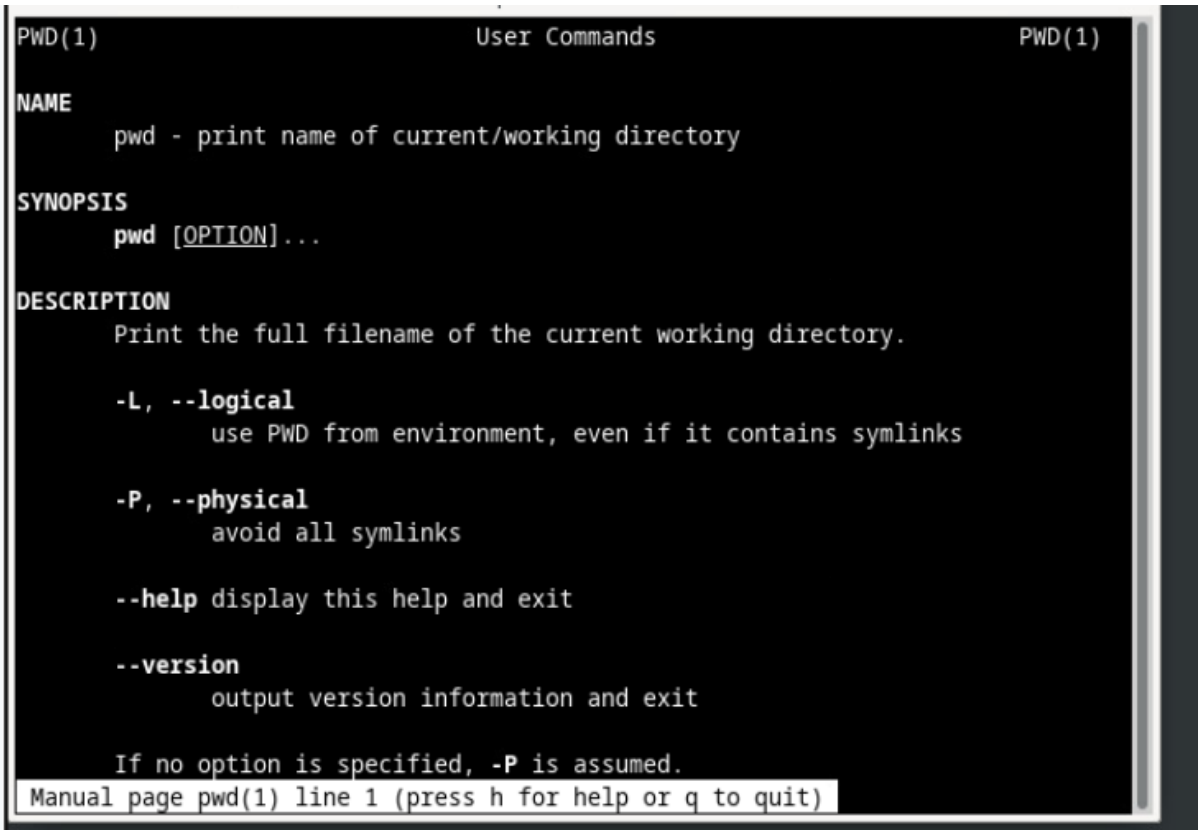
```
The table below shows the section numbers of the manual followed by the
types of pages they contain.

1   Executable programs or shell commands
2   System calls (functions provided by the kernel)
3   Library calls (functions within program libraries)
4   Special files (usually found in /dev)
5   File formats and conventions, e.g. /etc/passwd
6   Games
7   Miscellaneous (including macro packages and conventions), e.g.
    man(7), groff(7)
8   System administration commands (usually only for root)
9   Kernel routines [Non standard]

A manual page consists of several sections.
```

```
Conventional section names include NAME, SYNOPSIS, CONFIGURATION, DE-
SCRIPTION, OPTIONS, EXIT STATUS, RETURN VALUE, ERRORS, ENVIRONMENT,
FILES, VERSIONS, CONFORMING TO, NOTES, BUGS, EXAMPLE, AUTHORS, and
SEE ALSO.
```

Some of the sections on this page are **Library calls**, **Special Files** and **Games**



The **cp** command allows the user to copy files and directories



The command to learn about the **cp** command would be **man cp**

Step 3

```
[analyst@secOps ~]$ pwd
/home/analyst
```

Using the **pwd** command the current directory is **/home/analyst**

```
[analyst@secOps ~]$ cd /home/analyst
[analyst@secOps ~]$ ls -l
total 16
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
[analyst@secOps ~]$
```

Using **ls –l** showing all file sin directory

```
[analyst@secOps ~]$ mkdir cyops_folder1
[analyst@secOps ~]$ mkdir cyops_folder2
[analyst@secOps ~]$ mkdir cyops_folder3
[analyst@secOps ~]$ ls -l
total 28
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder2
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
[analyst@secOps ~]$
```

Creating 3 new folders using the **mkdir** command

```
[analyst@secOps ~]$ cd /home/analyst/cyops_folder3
[analyst@secOps cyops_folder3]$
```

Entering the **cyops_folder3** directory

```
[analyst@secOps cyops_folder3]$ cd ~
[analyst@secOps ~]$
```

Going back to **/home/analyst** directory

```
[analyst@secOps ~]$ mkdir /home/analyst/cyops_folder3/cyops_folder4
[analyst@secOps ~]$ ls -l /home/analyst/cyops_folder3
total 4
drwxr-xr-x 2 analyst analyst 4096 Apr  2 12:31 cyops_folder4
[analyst@secOps ~]$
```

Creating a new folder **cyops_folder4** inside **cyops_folder3**

```
[analyst@secOps ~]$ ls -la /home/analyst/cyops_folder3
total 12
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 .
drwx------ 17 analyst analyst 4096 Apr  2 12:28 ..
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:31 cyops_folder4
[analyst@secOps ~]$
```

```
[analyst@secOps ~]$ cd /home/analyst/cyops_folder3
[analyst@secOps cyops_folder3]$ cd .
[analyst@secOps cyops_folder3]$ cd ..
[analyst@secOps ~]$
```

Changing directory to **cyops_folder3**

Using "**cd .**" doesn't switch the directory, but using "**cd ..**" returns you to the parent directory

```
[analyst@secOps ~]$ cd ..
[analyst@secOps home]$
```

```
[analyst@secOps home]$ cd ..
[analyst@secOps /]$
```

```
[analyst@secOps /]$ cd ..
[analyst@secOps /]$
```

Step 4

```
[analyst@secOps /]$ cd /home/analyst
[analyst@secOps ~]$
```

```
[analyst@secOps ~]$ echo This is a message echoed by the terminal
This is a message echoed by the terminal
[analyst@secOps ~]$
```

Using **echo** command to return a message in terminal

```
[analyst@secOps ~]$ echo This is a message echoed by the terminal > some_text_fi
le.txt
[analyst@secOps ~]$
```

Adding "**> some_text_file.txt**" to the previous line will save the text that was echo'd into a **.txt** file

```
[analyst@secOps ~]$ ls -l
total 32
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
-rw-r--r--  1 analyst analyst   41 Apr  2 12:35 some_text_file.txt
[analyst@secOps ~]$
```

```
[analyst@secOps ~]$ ls -l some_text_file.txt
-rw-r--r-- 1 analyst analyst 41 Apr  2 12:35 some_text_file.txt
[analyst@secOps ~]$
```

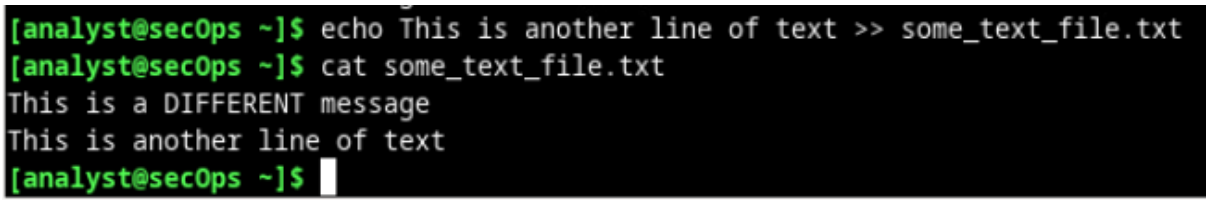
```
[analyst@secOps ~]$ cat some_text_file.txt
This is a message echoed by the terminal
[analyst@secOps ~]$
```

Using **cat** command to return the value of the **.txt** file

```
[analyst@secOps ~]$ echo This is a DIFFERENT message > some_text_file.txt
[analyst@secOps ~]$ cat some_text_file.txt
This is a DIFFERENT message
[analyst@secOps ~]$
```

Changing the content of the file using the same command but different message

Step 5



Same command used but instead of using "**>**", "**>>**" is used – this appends to a current .txt file

Step 6

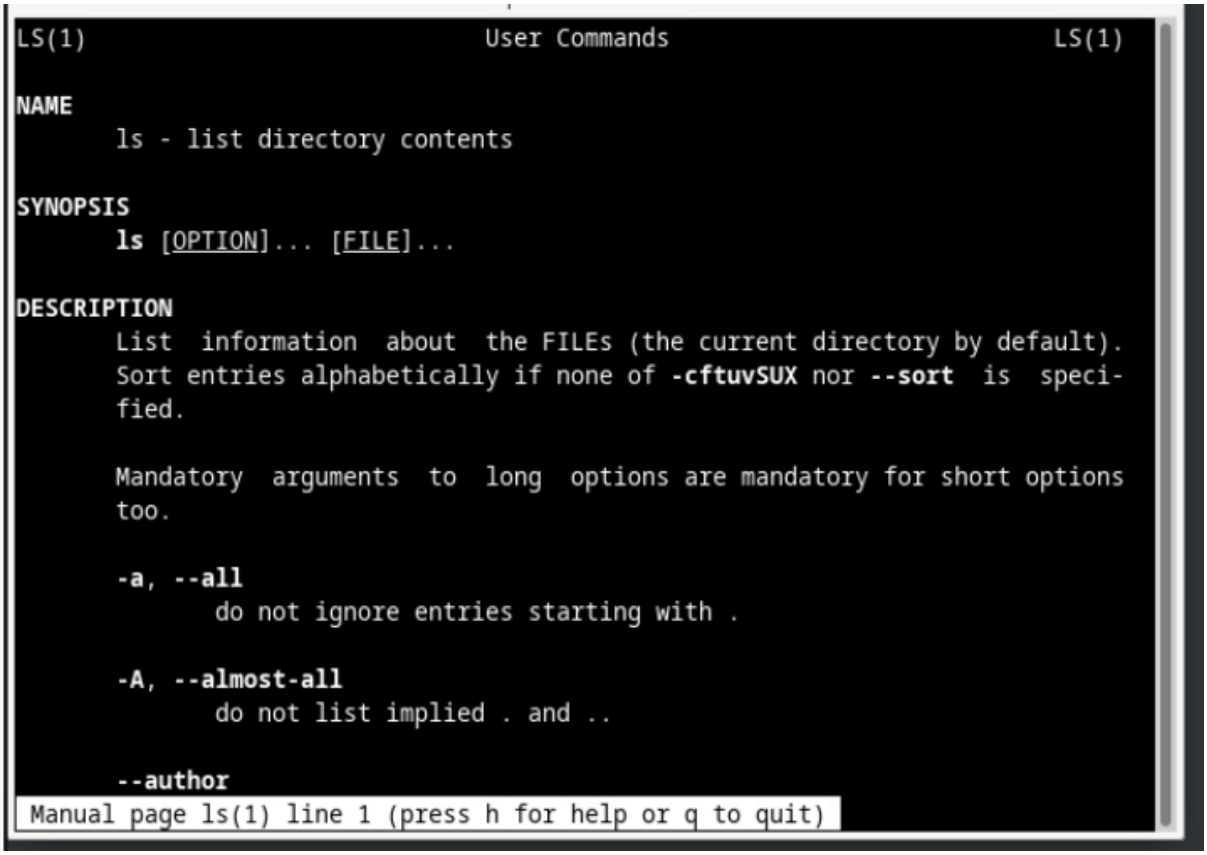
```
[analyst@secOps ~]$ ls -l
total 32
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
-rw-r--r--  1 analyst analyst   57 Apr  2 12:38 some_text_file.txt
[analyst@secOps ~]$
```

Checking home directory

```
[analyst@secOps ~]$ ls -la
total 132
drwx------ 17 analyst analyst  4096 Apr  2 12:35 .
drwxr-xr-x  3 root    root     4096 Mar 20  2018 ..
-rw-------  1 analyst analyst     0 Sep 29  2020 .bash_history
-rw-r--r--  1 analyst analyst    21 Feb  7  2018 .bash_logout
-rw-r--r--  1 analyst analyst    57 Feb  7  2018 .bash_profile
-rw-r--r--  1 analyst analyst    97 Mar 20  2018 .bashrc
-rw-r--r--  1 analyst analyst   141 Feb  7  2018 .bashrc_stock
drwxr-xr-x  8 analyst analyst  4096 Mar 25  2020 .cache
drwxr-xr-x 10 analyst analyst  4096 Aug 11  2020 .config
drwxr-xr-x  2 analyst analyst  4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst  4096 Apr  2 12:28 cyops_folder2
drwxr-xr-x  3 analyst analyst  4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst  4096 May 20  2020 Desktop
-rw-r--r--  1 analyst analyst    23 Mar 23  2018 .dmrc
drwxr-xr-x  3 analyst analyst  4096 Jul 30  2020 Downloads
drwx------  3 analyst analyst  4096 Mar 22  2018 .gnupg
-rw-------  1 analyst analyst  2520 Mar 24  2020 .ICEauthority
drwxr-xr-x  2 analyst analyst  4096 Mar 24  2018 .idlerc
drwxr-xr-x 11 analyst analyst  4096 Aug 11  2020 lab.support.files
-rw-------  1 analyst analyst    89 May 20  2020 .lesshst
```

Checking home directory (with hidden files)

```
[analyst@secOps ~]$ man ls
[analyst@secOps ~]$
```


LS(1) User Commands LS(1)
NAME
ls - list directory contents
SYNOPSIS
ls [OPTION]... [FILE]...
DESCRIPTION
List information about the FILEs (the current directory by default).
Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
fied.
Mandatory arguments to long options are mandatory for short options
too.
-a, --all
do not ignore entries starting with .
-A, --almost-all
do not list implied . and ..
--author
Manual page ls(1) line 1 (press h for help or q to quit)

Part 2

Step 1

```
[analyst@secOps ~]$ cp some_text_file.txt cyops_folder2/
[analyst@secOps ~]$ ls cyops_folder2/
some_text_file.txt
[analyst@secOps ~]$
```

The command **cp | file to copy | directory to paste into**

```
[analyst@secOps ~]$ ls -l
total 32
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:41 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
-rw-r--r--  1 analyst analyst   57 Apr  2 12:38 some_text_file.txt
[analyst@secOps ~]$
```

Checking both **home** directory and **cyops_folder2**

Step 2

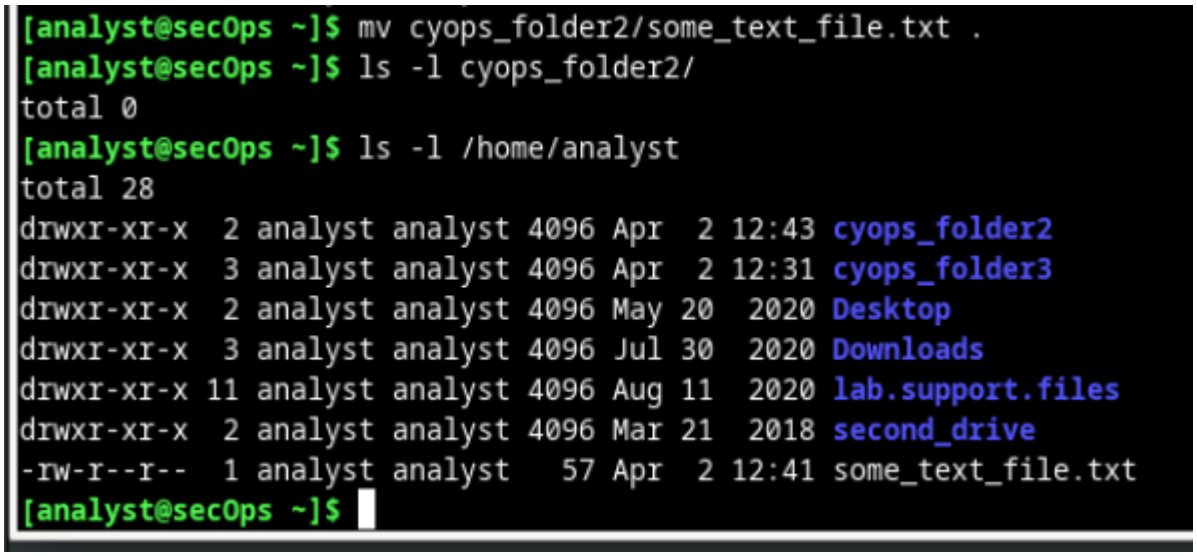
```
[analyst@secOps ~]$ rm some_text_file.txt
[analyst@secOps ~]$ ls -l
total 28
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:28 cyops_folder1
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:41 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
[analyst@secOps ~]$
```

Deleting the **some_text_file.txt** using the **rm** command and checking if the file has been deleted

```
[analyst@secOps ~]$ rm -r cyops_folder1
[analyst@secOps ~]$ ls -l
total 24
drwxr-xr-x  2 analyst analyst 4096 Apr  2 12:41 cyops_folder2
drwxr-xr-x  3 analyst analyst 4096 Apr  2 12:31 cyops_folder3
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
[analyst@secOps ~]$
```

Deleting the folder **cyops_folder1** using the **rm –r** command and checking if it has been deleted

Step 3

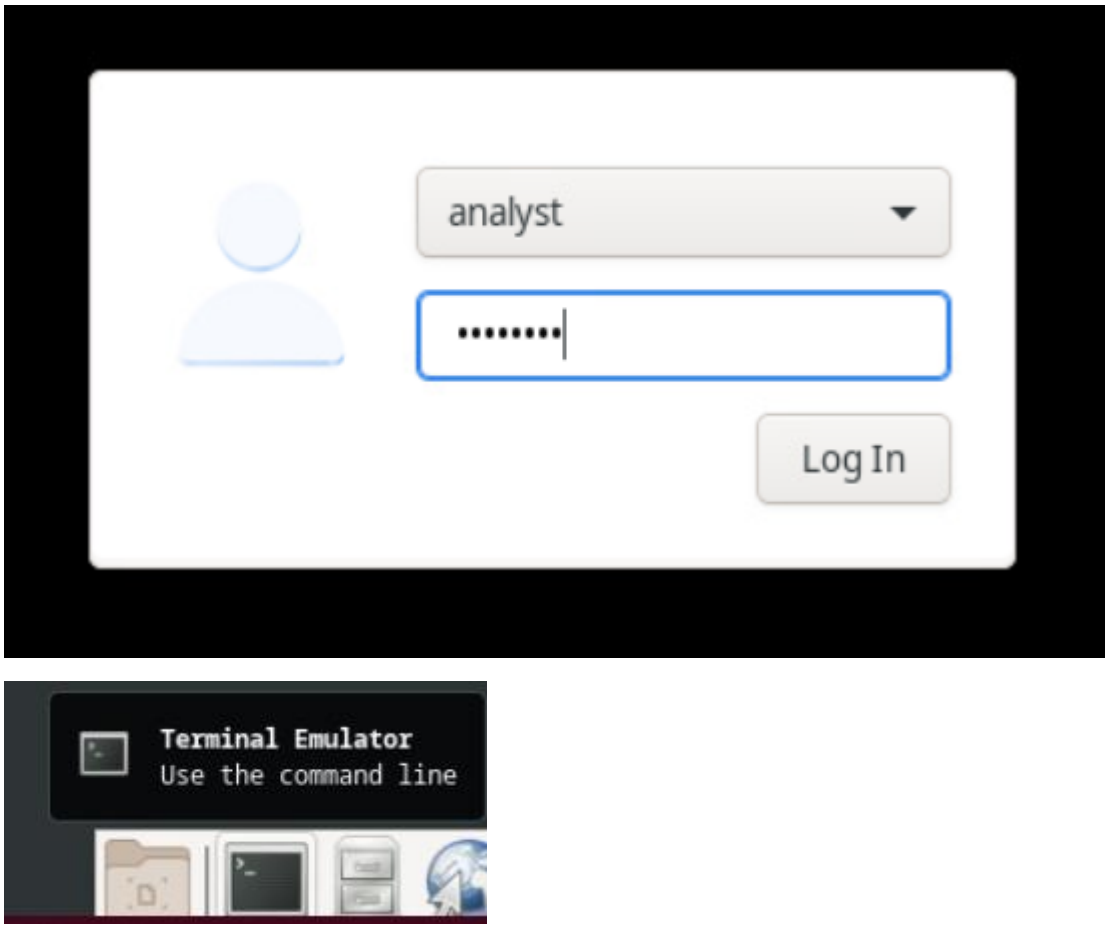


Using the **mv** command followed by the file (with directory) followed by the destination will move the desired file to the location mention (in this case to the **/home/analyst/** directory as that is the directory we are in

8.5.4 Lab - Navigating the Linux Filesystem and Permission Settings

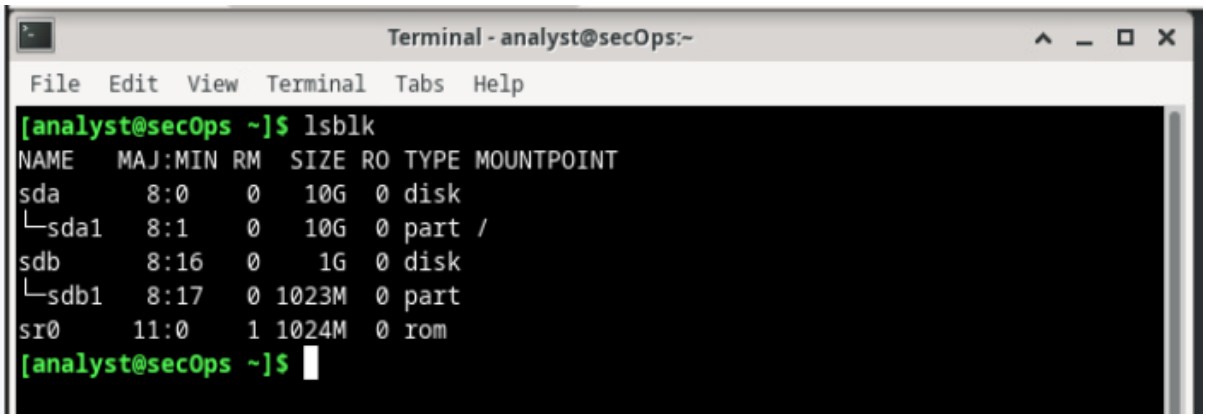
Part 1

Step 1

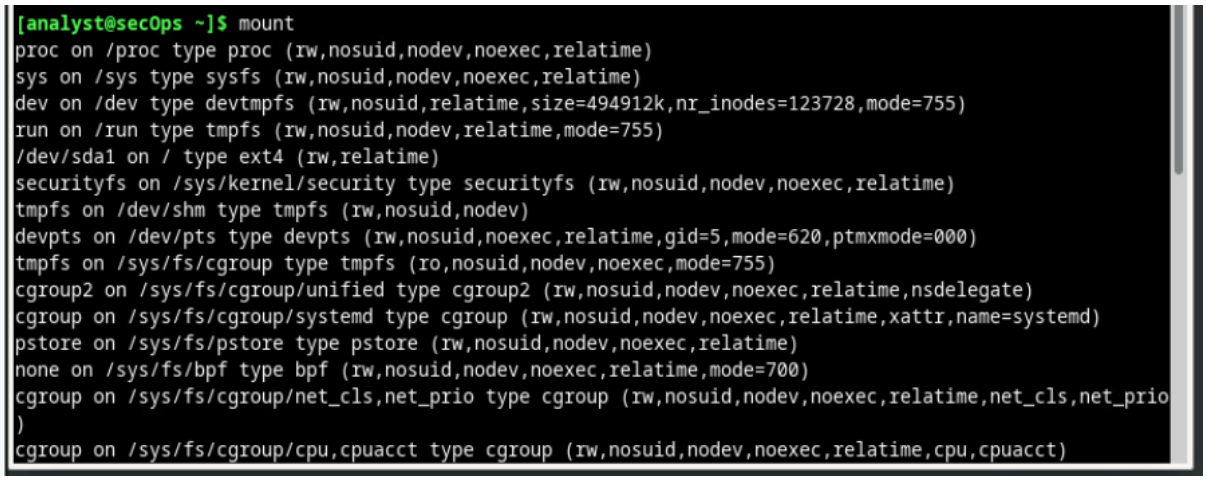


Accessing the VM on netlab and opening terminal

Step 2

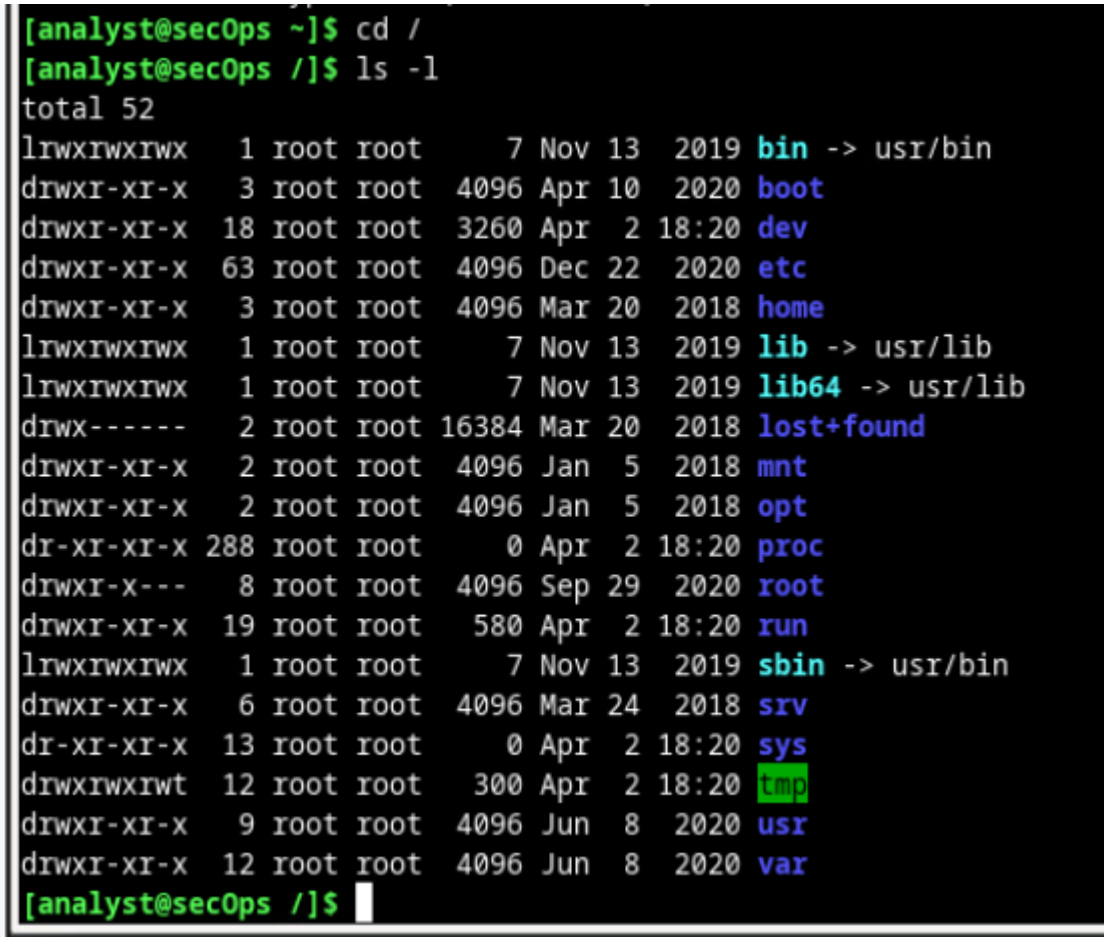


Using the **lsblk** command to check the block devices



Using the **mount** command to check the currently mounted filesystems

Filtering the output of the **mount** command to the root filesystem



Switching the directory to **/home** and checking the visible files & directories on it

Sdb1 isn't visible as it isn't mounted

Step 3

```
[analyst@secOps /]$ cd ~
[analyst@secOps ~]$ ls -l
total 16
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  2 analyst analyst 4096 Mar 21  2018 second_drive
[analyst@secOps ~]$
```

Checking if "**second_drive**" directory is present

```
[analyst@secOps ~]$ ls -l second_drive/
total 0
[analyst@secOps ~]$
```

Checking contents of the directory

```
[analyst@secOps ~]$ sudo mount /dev/sdb1 ~/second_drive/
[sudo] password for analyst:
[analyst@secOps ~]$
```

Mounting **sdb1**

```
[analyst@secOps ~]$ ls -l second_drive/
total 20
drwx------ 2 root    root 16384 Mar 26  2018 lost+found
-rw-rw-r-x 1 analyst root   188 May 19  2020 myFile.txt
[analyst@secOps ~]$
```

Checking if it has been mounted successfully

```
[analyst@secOps ~]$ mount | grep /dev/sd
/dev/sda1 on / type ext4 (rw,relatime)
/dev/sdb1 on /home/analyst/second_drive type ext4 (rw,relatime)
[analyst@secOps ~]$
```

Checking partition details

```
[analyst@secOps ~]$ sudo umount /dev/sdb1
[analyst@secOps ~]$ ls -l second_drive/
total 0
[analyst@secOps ~]$
```

Unmounting **sdb1**

Part 2

Step 1

```
[analyst@secOps ~]$ cd lab.support.files/scripts/
[analyst@secOps scripts]$ ls -l
total 60
-rwxr-xr-x 1 analyst analyst  952 Mar 21  2018 configure_as_dhcp.sh
-rwxr-xr-x 1 analyst analyst 1153 Mar 21  2018 configure_as_static.sh
-rwxr-xr-x 1 analyst analyst 3459 Mar 21  2018 cyberops_extended_topo_no_fw.py
-rwxr-xr-x 1 analyst analyst 4062 Mar 21  2018 cyberops_extended_topo.py
-rwxr-xr-x 1 analyst analyst 3669 Mar 21  2018 cyberops_topo.py
-rw-r--r-- 1 analyst analyst 2871 Mar 21  2018 cyops.mn
-rwxr-xr-x 1 analyst analyst  458 Mar 21  2018 fw_rules
-rwxr-xr-x 1 analyst analyst   70 Mar 21  2018 mal_server_start.sh
drwxr-xr-x 2 analyst analyst 4096 Mar 21  2018 net_configuration_files
-rwxr-xr-x 1 analyst analyst   65 Mar 21  2018 reg_server_start.sh
-rwxr-xr-x 1 analyst analyst  189 Mar 21  2018 start_ELK.sh
-rwxr-xr-x 1 analyst analyst   85 Mar 21  2018 start_miniedit.sh
-rwxr-xr-x 1 analyst analyst   76 Mar 21  2018 start_pox.sh
-rwxr-xr-x 1 analyst analyst  106 Mar 21  2018 start_snort.sh
-rwxr-xr-x 1 analyst analyst   61 Mar 21  2018 start_tftpd.sh
[analyst@secOps scripts]$
```

Changing directory to **lab.support.files/scripts** and checking file permissions

```
[analyst@secOps scripts]$ touch /mnt/myNewFile.txt
touch: cannot touch '/mnt/myNewFile.txt': Permission denied
[analyst@secOps scripts]$
```

Attempting to use the **touch** command

```
[analyst@secOps scripts]$ ls -ld /mnt
drwxr-xr-x 2 root root 4096 Jan  5  2018 /mnt
[analyst@secOps scripts]$
```

Checking the permissions of parent directory

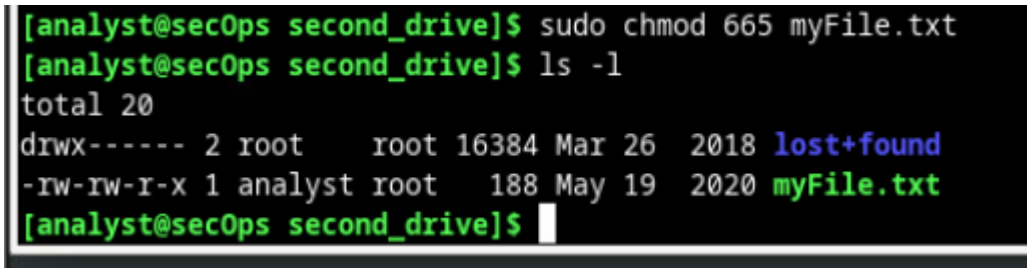
```
[analyst@secOps scripts]$ sudo mount /dev/sdb1 ~/second_drive/
[analyst@secOps scripts]$
```

Changing permissions of a directory using the **chmod** command

```
[analyst@secOps scripts]$ cd ~/second_drive/
[analyst@secOps second_drive]$ ls -l
total 20
drwx------ 2 root    root 16384 Mar 26  2018 lost+found
-rw-rw-r-x 1 analyst root   188 May 19  2020 myFile.txt
[analyst@secOps second_drive]$
```

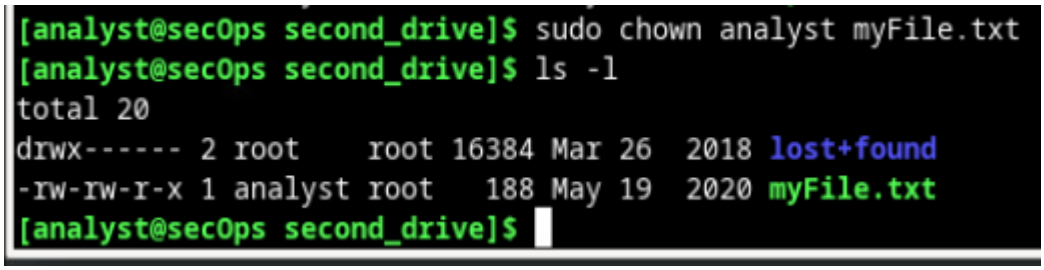
Changing directory to **/~/second_drive**

myFile.txt has read and write permission for owner, read permissions for group owner of the file and read permissions for all other users

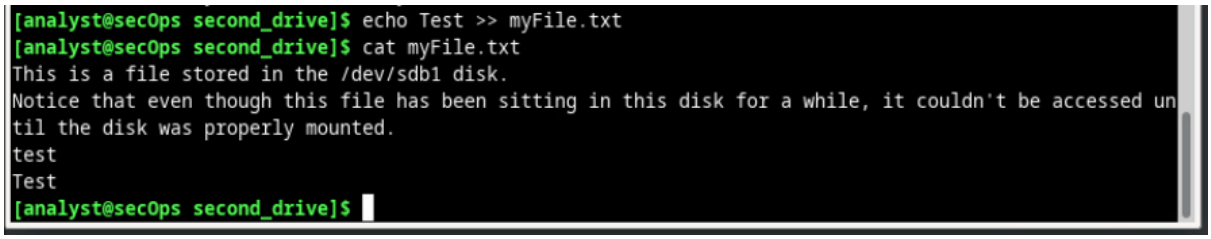


Changing permission of **myFile.txt** so that group owners can read and write

chmod 777 would give –rwxrwxrwx permissions

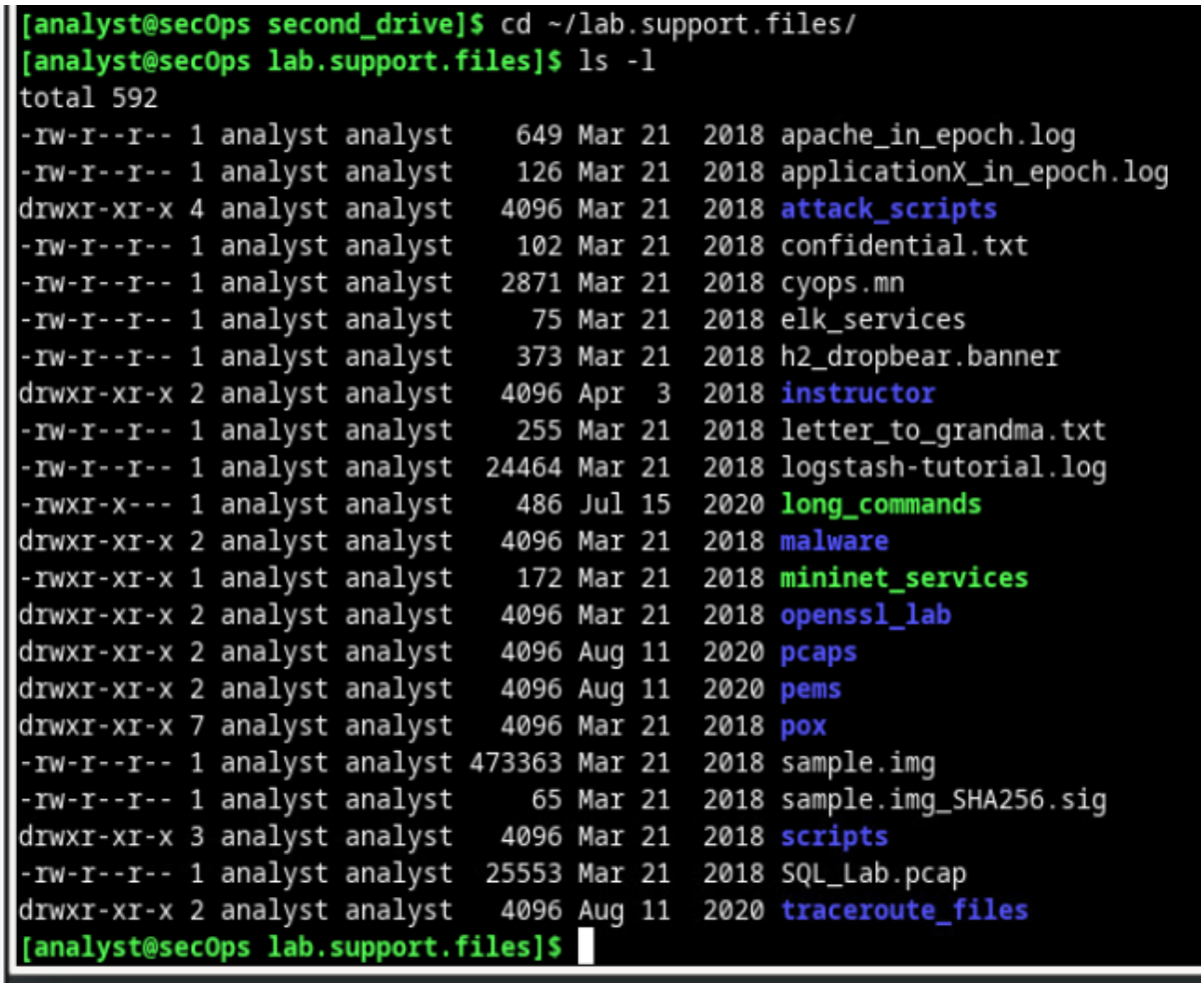


The command **chown** changes ownership of the file



Appendin to the file and the changes have been applied to to permission changes

Step 2



Changing directory and listing all files in the directory

Part 3

Step 1

```
[analyst@secOps lab.support.files]$ cd ~
[analyst@secOps ~]$ ls -l
total 16
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  3 root    root    4096 May  4  2020 second_drive
[analyst@secOps ~]$
```

Changing directory to **/home/analyst** and checking files & directories

```
[analyst@secOps ~]$ ls -l /dev/
total 0
crw-------  1 root root     10, 175 Apr  2 18:20 agpgart
crw-r--r--  1 root root     10, 235 Apr  2 18:20 autofs
drwxr-xr-x  2 root root         140 Apr  2 18:20 block
drwxr-xr-x  2 root root         100 Apr  2 18:20 bsg
crw-------  1 root root     10, 234 Apr  2 18:20 btrfs-control
lrwxrwxrwx  1 root root           3 Apr  2 18:20 cdrom -> sr0
drwxr-xr-x  2 root root        2700 Apr  2 18:20 char
crw-------  1 root root      5,   1 Apr  2 18:20 console
lrwxrwxrwx  1 root root          11 Apr  2 18:20 core -> /proc/kcor
crw-------  1 root root     10,  60 Apr  2 18:20 cpu_dma_latency
crw-------  1 root root     10, 203 Apr  2 18:20 cuse
drwxr-xr-x  6 root root         120 Apr  2 18:20 disk
```

Checking files and directories inside **/dev** directory

```
crw-rw----  1 root disk     10, 237 Apr  2 18:20 loop-control
drwxr-xr-x  2 root root           60 Apr  2 18:20 mapper
crw-r-----  1 root kmem      1,   1 Apr  2 18:20 mem
drwxrwxrwt  2 root root           40 Apr  2 18:20 mqueue
drwxr-xr-x  2 root root           60 Apr  2 18:20 net
crw-rw-rw-  1 root root      1,   3 Apr  2 18:20 null
crw-------  1 root root     10, 144 Apr  2 18:20 nvram
crw-r-----  1 root kmem      1,   4 Apr  2 18:20 port
crw-------  1 root root    108,   0 Apr  2 18:20 ppp
crw-------  1 root root     10,   1 Apr  2 18:20 psaux
crw-rw-rw-  1 root tty       5,   2 Apr  2 18:38 ptmx
drwxr-xr-x  2 root root           0 Apr  2 18:20 pts
crw-rw-rw-  1 root root      1,   8 Apr  2 18:20 random
crw-------  1 root root     10, 242 Apr  2 18:20 rfkill
lrwxrwxrwx  1 root root           4 Apr  2 18:20 rtc -> rtc0
crw-------  1 root root    249,   0 Apr  2 18:20 rtc0
brw-rw----  1 root disk      8,   0 Apr  2 18:20 sda
brw-rw----  1 root disk      8,   1 Apr  2 18:20 sda1
brw-rw----  1 root disk      8,  16 Apr  2 18:20 sdb
brw-rw----  1 root disk      8,  17 Apr  2 18:20 sdb1
drwxrwxrwt  2 root root           40 Apr  2 18:20 shm
crw-------  1 root root     10, 231 Apr  2 18:20 snapshot
drwxr-xr-x  2 root root           80 Apr  2 18:20 snd
brw-rw----+ 1 root optical  11,   0 Apr  2 18:20 sr0
lrwxrwxrwx  1 root root          15 Apr  2 18:20 stderr -> /proc/self/fd/2
lrwxrwxrwx  1 root root          15 Apr  2 18:20 stdin -> /proc/self/fd/0
lrwxrwxrwx  1 root root          15 Apr  2 18:20 stdout -> /proc/self/fd/1
```

```
[analyst@secOps ~]$ echo "symbolic" > file1.txt
[analyst@secOps ~]$ cat file1.txt
symbolic
[analyst@secOps ~]$ echo "hard" > file2.txt
[analyst@secOps ~]$ cat file2.txt
hard
[analyst@secOps ~]$
```

Creating 2 **.txt** files

```
[analyst@secOps ~]$ ln -s file1.txt file1symbolic
[analyst@secOps ~]$ ln file2.txt file2hard
[analyst@secOps ~]$
```

Applying a symbolic link to 1 file and a hard link to the other

```
[analyst@secOps ~]$ ls -l
total 28
drwxr-xr-x  2 analyst analyst 4096 May 20  2020 Desktop
drwxr-xr-x  3 analyst analyst 4096 Jul 30  2020 Downloads
lrwxrwxrwx  1 analyst analyst    9 Apr  2 18:41 file1symbolic -> file1.txt
-rw-r--r--  1 analyst analyst    9 Apr  2 18:40 file1.txt
-rw-r--r--  2 analyst analyst    5 Apr  2 18:40 file2hard
-rw-r--r--  2 analyst analyst    5 Apr  2 18:40 file2.txt
drwxr-xr-x 11 analyst analyst 4096 Aug 11  2020 lab.support.files
drwxr-xr-x  3 root    root    4096 May  4  2020 second_drive
[analyst@secOps ~]$
```

Checking directory to see changes made

```
[analyst@secOps ~]$ mv file1.txt file1new.txt
[analyst@secOps ~]$ mv file2.txt file2new.txt
[analyst@secOps ~]$ cat file1symbolic
cat: file1symbolic: No such file or directory
[analyst@secOps ~]$ cat file2hard
hard
[analyst@secOps ~]$
```

Replacing the names of the files and executing each file to see if the new file names work

Module 18: Cryptography

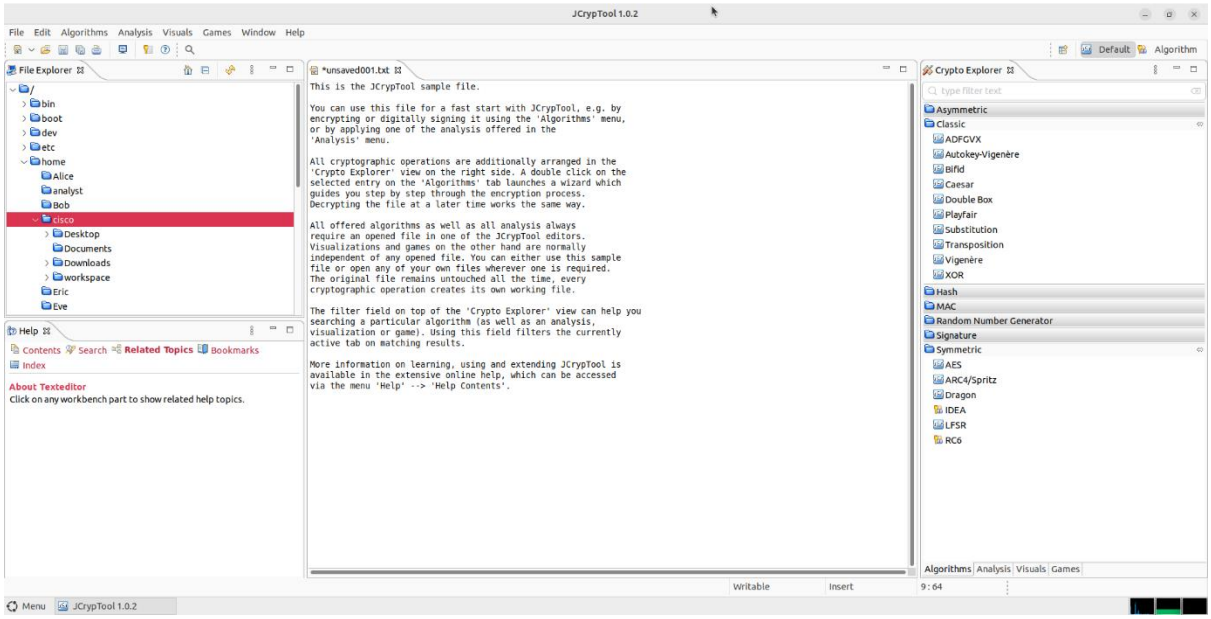
18.1.11 Lab - Use Classic and Modern Encryption Algorithms

Part 1

Step 1

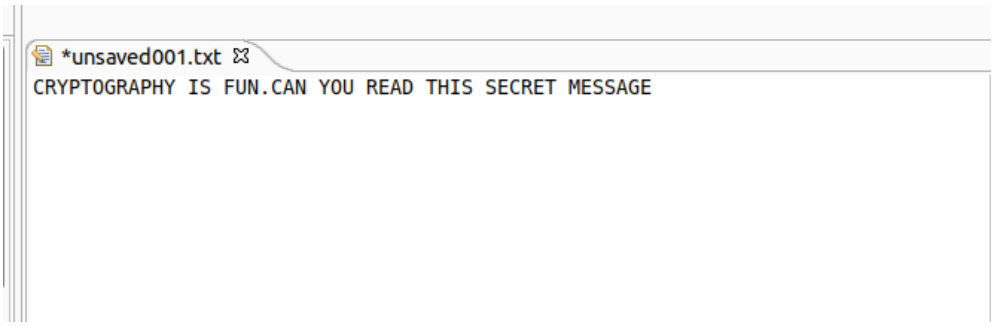
Opening the lab VM

Step 2

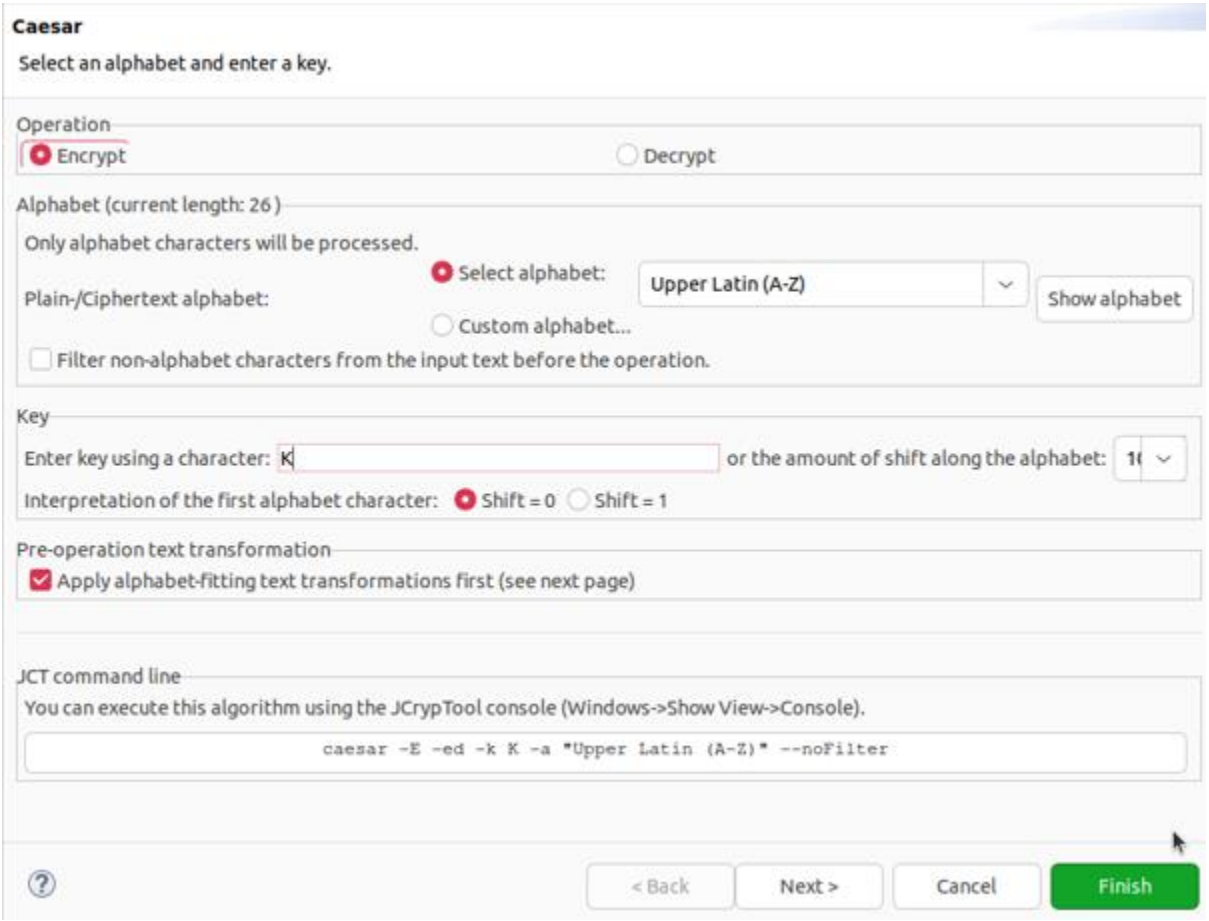


Opening the **JCrypTool**

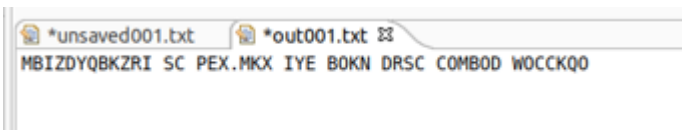
Step 3



Replacing the message inside **unsaved001.txt**

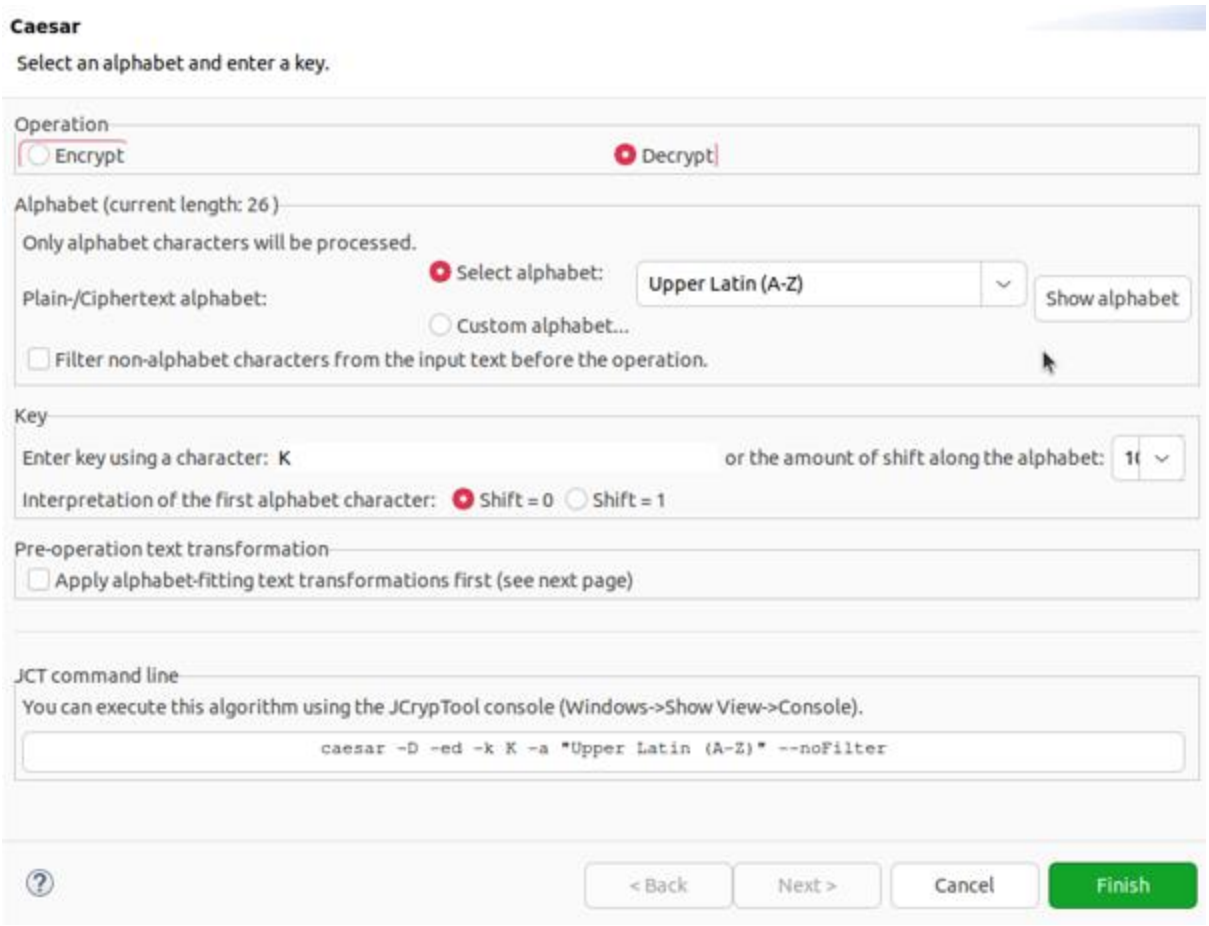


Checking and setting parameters for **Caesar**

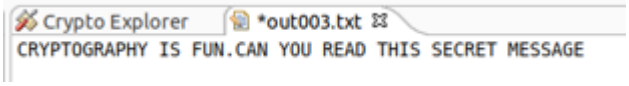


Checking the output on the new **out001.txt** file

Step 4

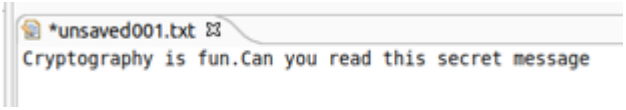


Setting **Caesar** to decrypt the encrypted message

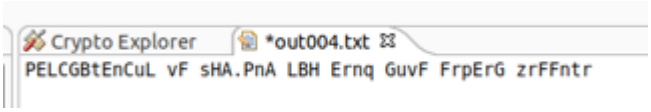


Decrypted message

Step 5



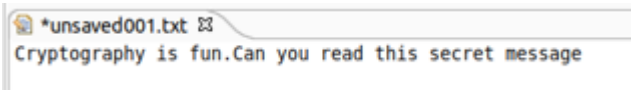
Creating a new file with a new message



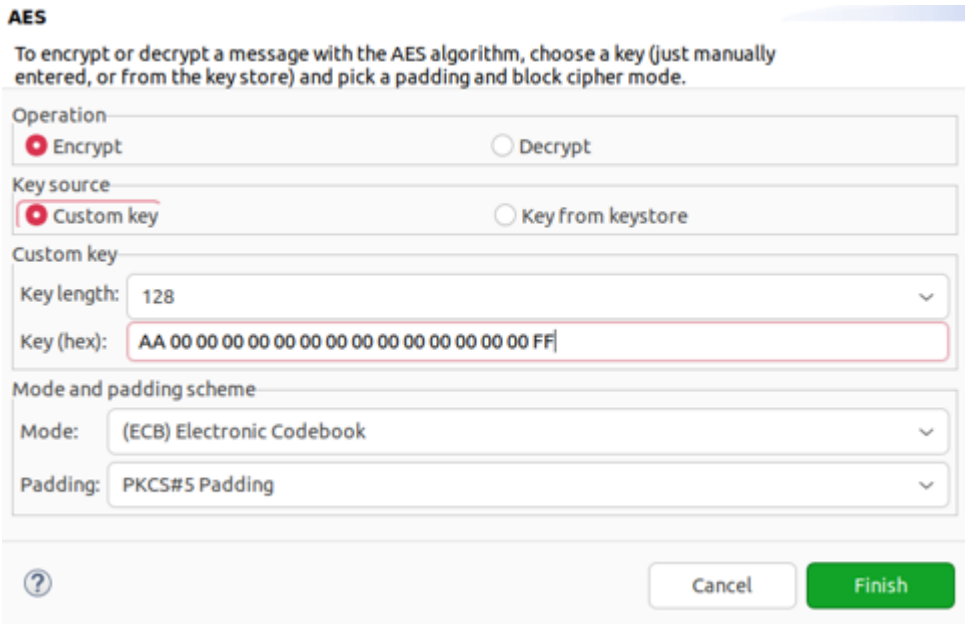
New encrypted message

Part 2

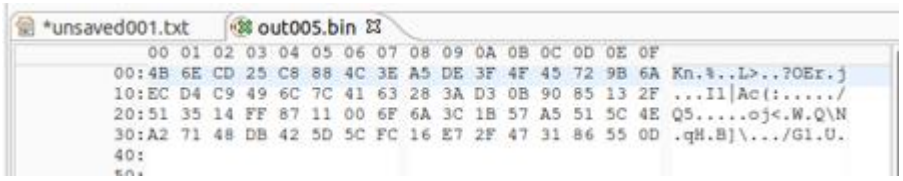
Step 1



Creating a new file with a new message

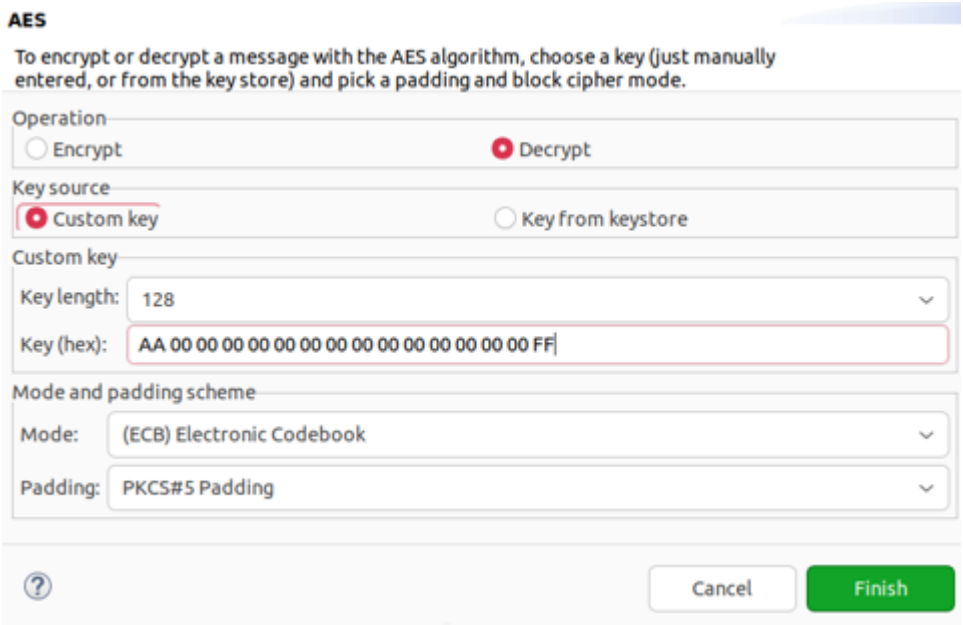


Setting **AES** to encrypt the message



Encrypted message

Step 2

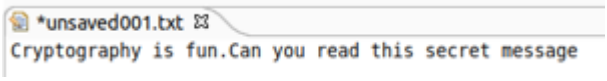


Setting **AES** to decrypt the message

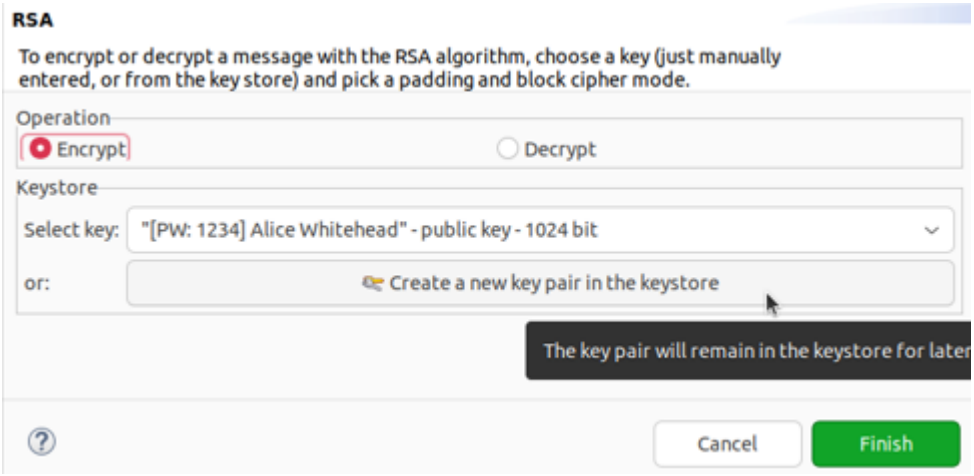
Decrypted message

Part 3

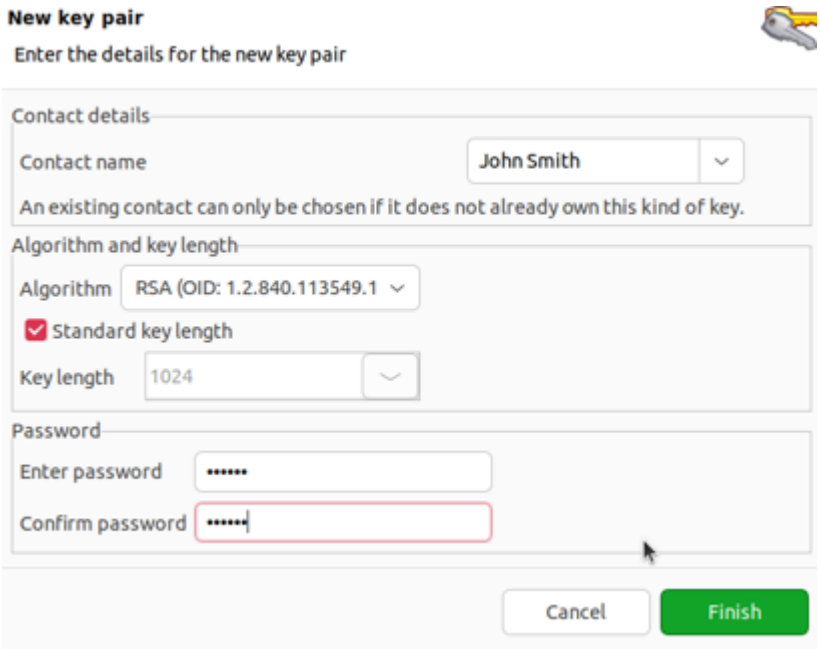
Step 1



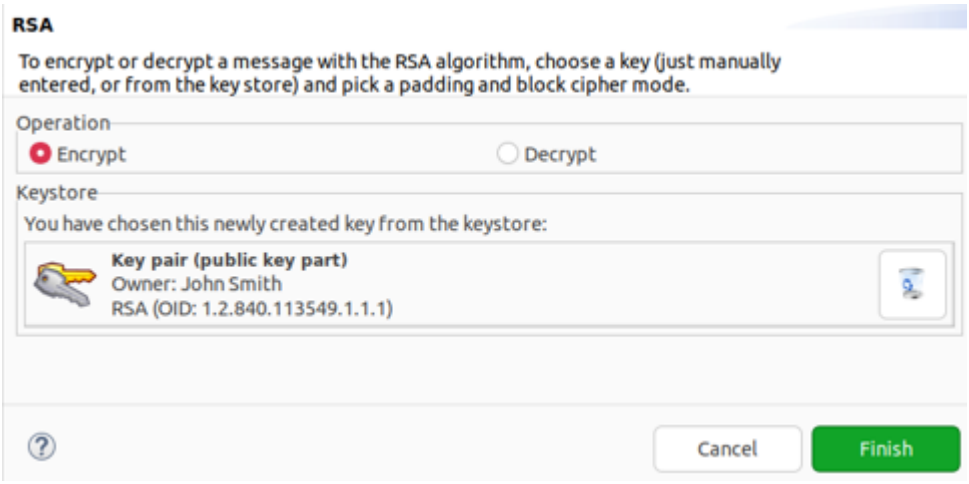
Creating a new file with a new message



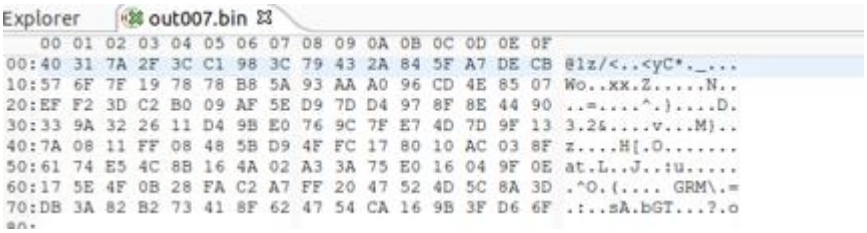
Opening the **RSA** algorithm



Creating a new **key pair**

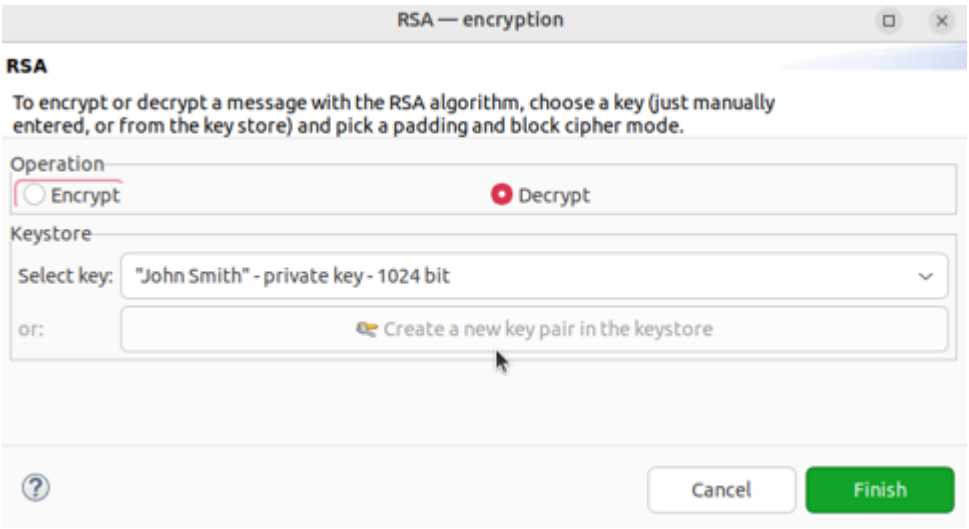


Using the new key pair

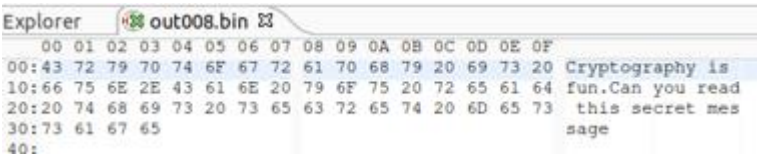


Decrypted message

Step 2



Setting **RSA** to decrypt the message



Decrypted message

18.4.10 Lab – Hashing Things Out

Part 1

```
[analyst@secOps ~]$ cd /home/analyst/lab.support.files/
[analyst@secOps lab.support.files]$
```

Opening a **terminal** and changing directory to **/lab.support.files**

```
[analyst@secOps lab.support.files]$ cat letter_to_grandma.txt
Hi Grandma,
I am writing this letter to thank you for the chocolate chip cookies you sent me
. I got them this morning and I have already eaten half of the box! They are abs
olutely delicious!

I wish you all the best. Love,
Your cookie-eater grandchild.
```

Executing the **.txt** file using the **cat** command to check its contents

```
[analyst@secOps lab.support.files]$ openssl sha256 letter_to_grandma.txt
SHA256(letter_to_grandma.txt)= deff9c9bbece44866796ff6cf21f2612fbb77aa1b2515a900
bafb29be118080b
```

Hashing the **.txt** file using the **sha256** algorithm

```
 GNU nano 4.9.2                  letter_to_grandma.txt                  Modified
Hi Grandpa,
I am writing this letter to thank you for the chocolate chip cookies you sent m

I wish you all the best. Love,
Your cookie-eater grandchild.
```

Altering the contents of the **.txt** file using **nano**

```
[analyst@secOps lab.support.files]$ openssl sha256 letter_to_grandma.txt
SHA256(letter_to_grandma.txt)= 43302c4500b7c4b8e574ba27a59d83267812493c029fd054c
9242f3ac73100bc
```

Hashing the **.txt** file again using the same algorithm, but this time given a different output

```
[analyst@secOps lab.support.files]$ openssl sha512 letter_to_grandma.txt
SHA512(letter_to_grandma.txt)= 7c35db79a06aa30ae0f6de33f2322fd419560ee9af9cedeb6
e251f2f1c4e99e0bbe5d2fc32ce501468891150e3be7e288e3e568450812980c9f8288e3103a1d3
```

Hashing the **.txt** file again using a bit length of 512 (rather than the previous 256)

```
[analyst@secOps lab.support.files]$ sha256sum letter_to_grandma.txt
43302c4500b7c4b8e574ba27a59d83267812493c029fd054c9242f3ac73100bc  letter_to_gran
dma.txt
```

```
[analyst@secOps lab.support.files]$ sha512sum letter_to_grandma.txt
7c35db79a06aa30ae0f6de33f2322fd419560ee9af9cedeb6e251f2f1c4e99e0bbe5d2fc32ce5014
68891150e3be7e288e3e568450812980c9f8288e3103a1d3  letter_to_grandma.txt
```

Generating the hash of both bit lengths using the **sum** appendix to the **shaXXX** command

The hashes are the same as shown in the previous screenshots as the same hash bit-length has been specified as before (per each bit length)

Part 2

```
[analyst@secOps lab.support.files]$ cat sample.img_SHA256.sig
c56c4724c26eb0157963c0d62b76422116be31804a39c82fd44ddf0ca5013e6a
```

Using the **cat** command to display the contents of the **.sig** file

```
[analyst@secOps lab.support.files]$ sha256sum sample.img
c56c4724c26eb0157963c0d62b76422116be31804a39c82fd44ddf0ca5013e6a  sample.img
```

Calculating the hash of the **.img** file

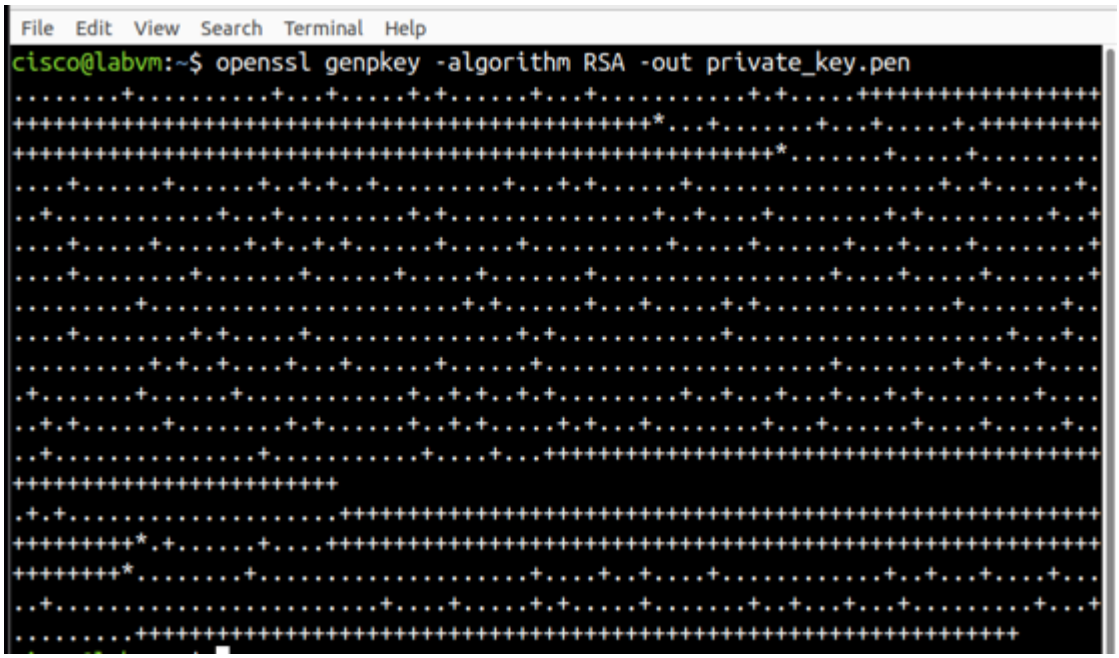
The image was downloaded without errors as the has value of the **.img** is the same as the **.sig** file, resulting in a match

18.5.5 Lab - Generate and Use a Digital Signature

Part 1

Opening the lab VM

Part 2



Generating a private key using the **openssl genkey** command alongside the **RSA** algorithm and outputting it into a **private_key.pen** file

Using the **cat** command to show the contents of the **.pen** file

Part 3

```
cisco@labvm:~$ openssl pkey -in private_key.pen -pubout -out public_key.pen
cisco@labvm:~$ cat public_key.pen
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAzGQCTOWFUd3JGlxZEVjO
lEzijM6fLVncAYlPwJB8KXyYIwwU8KZ/iELZtl6X0rf2Z9jguslSrusvD07gKnZa
w3wE+WBg/cF0SrplkA39XaQaB9hVts7OaZheCSEaXHGIZiWSLh4hg1bqkg1z+rsT
UpF4wGSLh6aAGFoii7pFDcFSygvyzSOrRERZIWgwn2n+jO4S9WADWRZG1e8G+Day
j1HPLmAdelsJCzT1gtln4hv2yeh7q67GjXVlLLnqyEVOJY0FQkMPTOcFPkG+Qh+S
oSTVfFm6HUwr2KFROvWJqVv0nGxpCqAXKBOa1Z8fqbfGMc5dT0gJFEtLWfUM/x4H
3QIDAQAB
-----END PUBLIC KEY-----
```

Generating a public key using the **openssl pkey** command, by using the **private_key.pen** file as the input and outputting a new file called **public_key.pen**

This new file is then executed with the **cat** command

Part 4

```
cisco@labvm:~$ echo Please trasfer 2,000,00 US dollars to Mr. Jester by 6pm toda
y! > contract.txt
cisco@labvm:~$ cat contract.txt
Please trasfer 2,000,00 US dollars to Mr. Jester by 6pm today!
```

Creating a new **.txt** file and using the **cat** command to check its contents

Part 5

```
cisco@labvm:~$ openssl dgst -sha256 -sign private_key.pen -out signature contrac
t.txt
```

Creating a signature for the previously created **.txt** file

```
cisco@labvm:~$ cat signature
&◆k◆◆◆◆Ai◆{◆◆I◆◆)◆◆◆l◆A
                        x◆◆
◆L◆ZG=+<X!)◆◆◆◆ ◆◆
◆◆◆O◆◆◆◆X◆◆◆◆W◆    ◆G◆5◆_,◆◆l◆◆
◆◆{◆YR◆◆◆◆Z)◆8B◆i◆◆◆◆>;◆x◆◆p◆S◆◆◆q 2◆◆◆y◆◆JKV◆9◆S◆åT◆◆◆◆◆P>q◆0◆◆ZrM◆$F◆23Dqf◆D◆◆
7◆G◆◆%f◆J◆)◆"◆y◆◆&◆◆0"◆◆6&|^|◆◆◆◆d◆◆cisco@labvm:~$
```

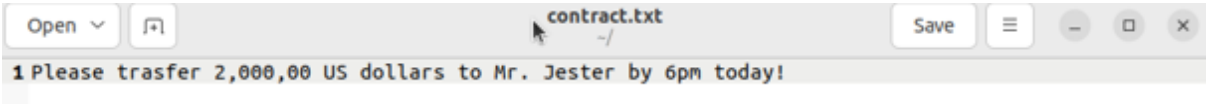
Checking the contents of the signature file using the **cat** command

Part 6

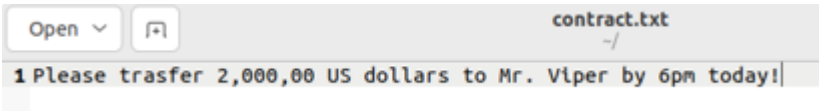
```
cisco@labvm:~$ openssl dgst -sha256  -verify public_key.pen -signature signature
 contract.txt
Verified OK
```

Checking if the file has not been tampered using the **openssl dgst** command, returning a positive validation

Part 7



Opening the **.txt** file using the command **gedit**



Changing the contents of the file

Part 8

```
cisco@labvm:~$ openssl dgst -sha256  -verify public_key.pen -signature signature
 contract.txt
Verification failure
```

Rechecking the validity of the file using the **openssl dgst** command, and this time it returns with a verification failure

Module 19: Technologies and Protocols (ACL)

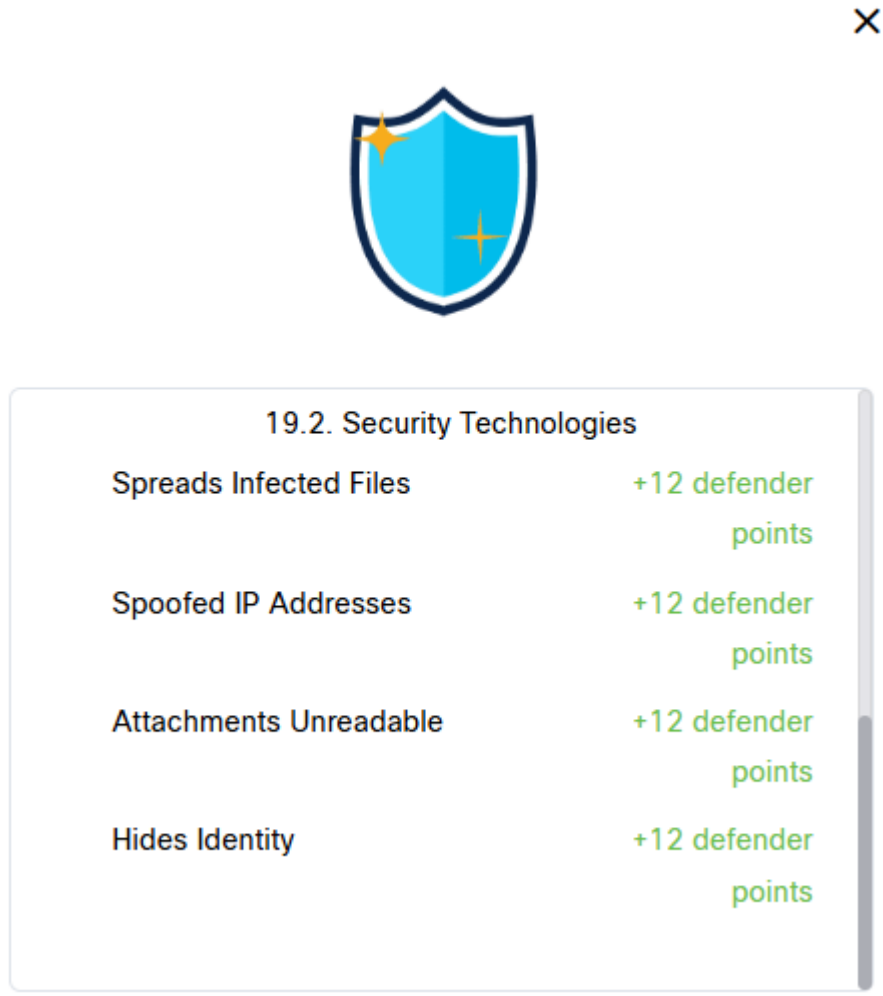
Please complete the **two** multiple choice quizzes available under **19.1 Monitoring Common Protocols** and **19.2 Security Technologies** and provide a screenshot of your completed experience/defender points shield, e.g.:

19.1.7 Check Your Understanding - Identify the Monitored Protocol

19.1. Monitoring Common Protocols

Carry Payloads	+12 defender points
Decoding Captures	+12 defender points
Exfiltrate Data	+12 defender points
Event Correlation	+12 defender points

19.2.6 Check Your Understanding - Identify the Impact of the Technology on Security and Monitoring



Reflective Report (cryptography and access control)

Throught completing these allocated labs and content that has been made available on the cisco website, it has helped strengthen my understanding on different security measures that are available to secure devices, and methods that are potentially used in businesses, such as setting up user accounts or limiting access to specific users.

More specifically the usage of Linux and hashing inside of the terminal was one of my key learning points as one of the tasks to complete was verifying a download. This was done by checking the has value of the downloaded file and comparing it to the **.sig** file (signature) hash value – if the hash output was the same then the download was successful, if not then it failed. Another key task that was completed was changing privileges of files and which user / groups can access it, which is displayed by “-r—r—r—” which would give the owner reading privileges, owner groups reading privileges and other users reading privileges – for a file, if it was a directory then it would start with “d” instead of "-“