Advanced Pen-testing

Comp 357

Bonus Lab :- Evasion

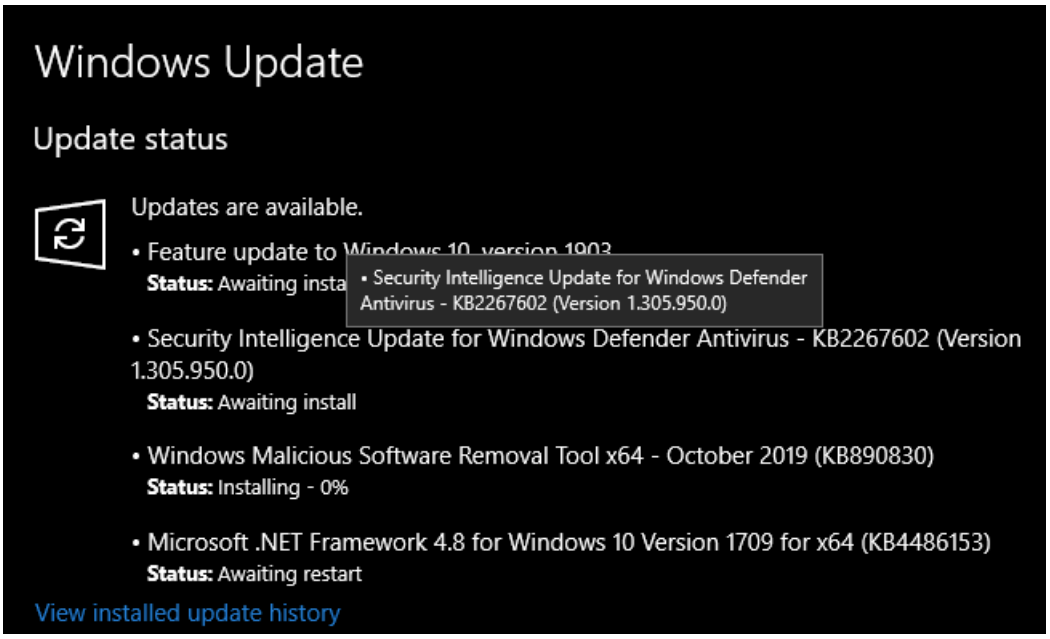
Karan Tank

Introduction

In this lab, we will be using Commando VM platform. Commando Vm is the first of its kind Windows Offensive Distribution System. Commando Vm is specially designed for penetration testers who are looking for a stable and supported Linux testing platform. However, windows is more user friendly and commando vm can allow you access the advanced peneteration testing softwares that are linux based on the windows platform. Commando Vm uses Boxstater, Chocolatey and myget packages to install all of the software and delivers many tools and utilities to support penetration testing.

Step 1

We will install Commando Vm on a new Windows 10 Virtual machine. Before we begin the installation we need to update the Windows Completely, You can do this by going into settings and looking for check updates in the windows 10



Once your Windows 10 Vm is completely up to date, Download the commando Vm zip from this website https://github.com/fireeye/commando-vm

unzip the Zip file in the folder you want to save it.

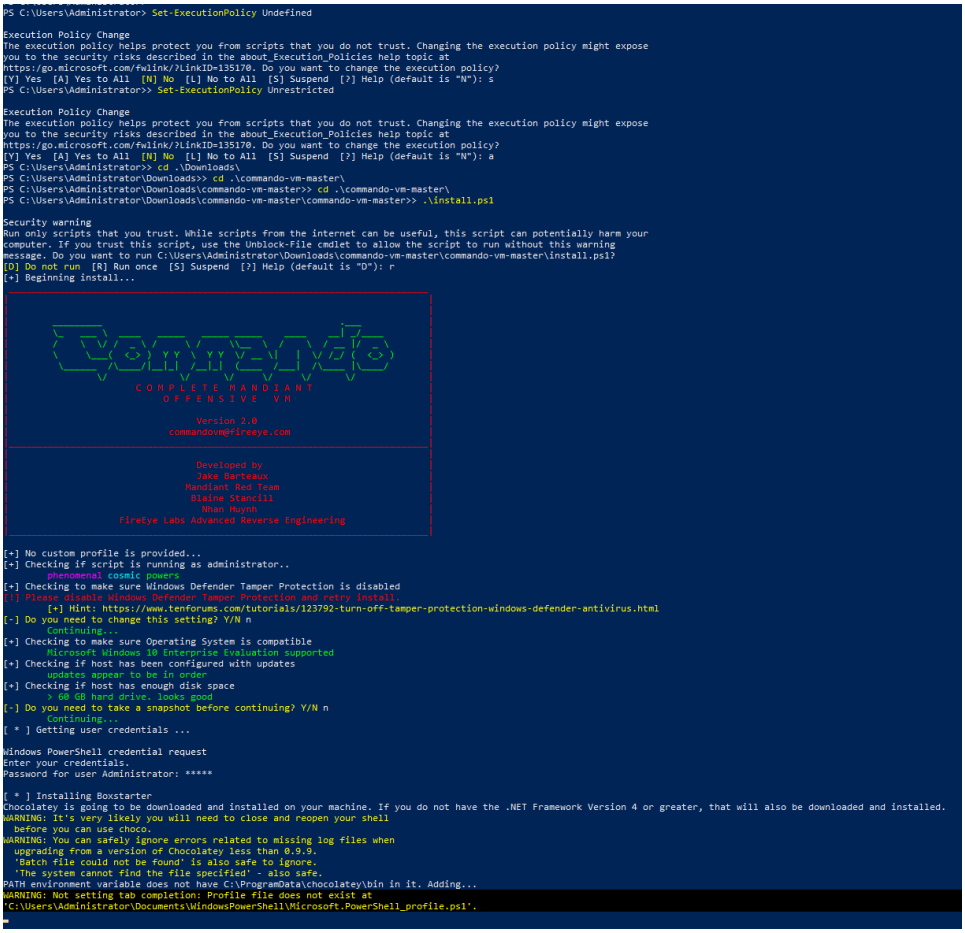
Open Powershell and Run as Administrator.

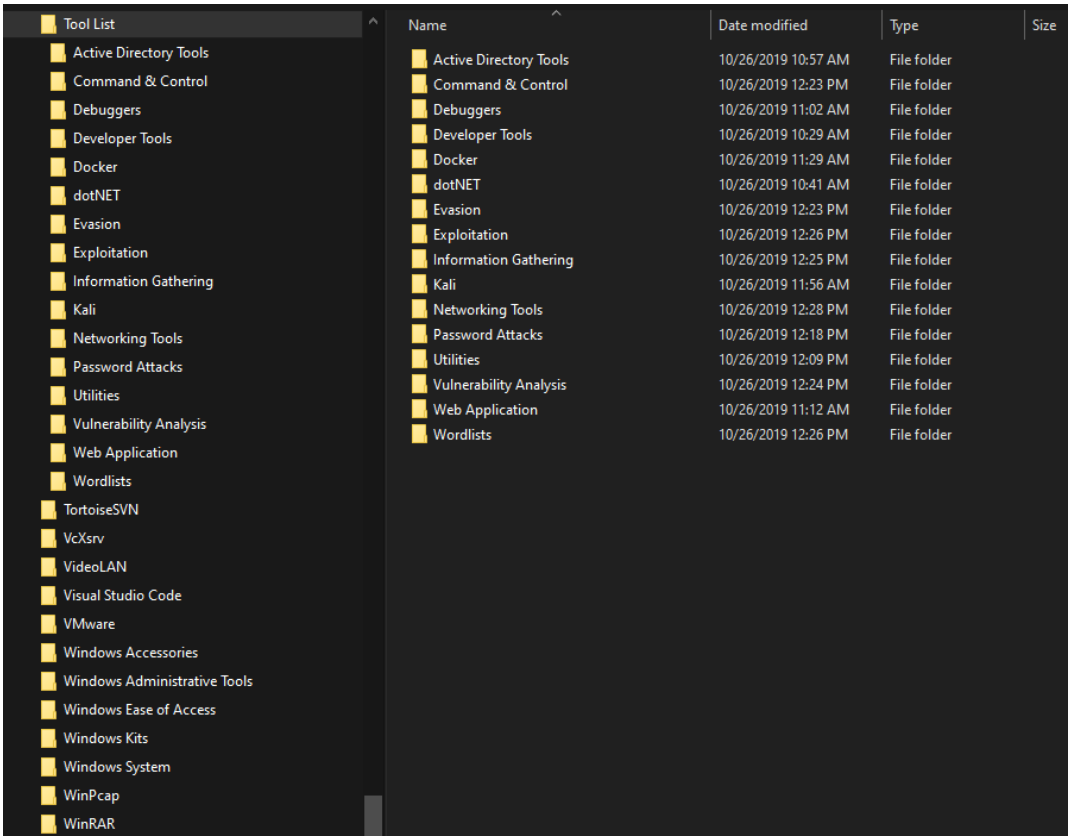
run the command **Set-ExecutionPolicy Unrestricted**

after that, go to the folder you unzipped the commando vm files and run the install.ps1 script.

It will automatically install all the software you need. It is a long process of about 1 to 2 hours and restarts your pc several times.

Once the Commando VM is installed, You can see the new commando vm wallpaper and a tools folder which contains all the tools required for penetration testing.




Tool List
Active Directory Tools
Command & Control
Debuggers
Developer Tools
Docker
dotNET
Evasion
Exploitation
Information Gathering
Kali
Networking Tools
Password Attacks
Utilities
Vulnerability Analysis
Web Application
Wordlists
TortoiseSVN
VcXsrv
VideoLAN
Visual Studio Code
VMware
Windows Accessories
Windows Administrative Tools
Windows Ease of Access
Windows Kits
Windows System
WinPcap
WinRAR
Name
Date modified
Type
Size
Active Directory Tools 10/26/2019 10:57 AM File folder
Command & Control 10/26/2019 12:23 PM File folder
Debuggers 10/26/2019 11:02 AM File folder
Developer Tools 10/26/2019 10:29 AM File folder
Docker 10/26/2019 11:29 AM File folder
dotNET 10/26/2019 10:41 AM File folder
Evasion 10/26/2019 12:23 PM File folder
Exploitation 10/26/2019 12:26 PM File folder
Information Gathering 10/26/2019 12:25 PM File folder
Kali 10/26/2019 11:56 AM File folder
Networking Tools 10/26/2019 12:28 PM File folder
Password Attacks 10/26/2019 12:18 PM File folder
Utilities 10/26/2019 12:09 PM File folder
Vulnerability Analysis 10/26/2019 12:24 PM File folder
Web Application 10/26/2019 11:12 AM File folder
Wordlists 10/26/2019 12:26 PM File folder

Step 2

In the tools folder, there is a sub folder named “Evasion”. It contains many tools. I will be using PSattack for this activity.

Ps attack combines the best projects in the infosec powershelgl community into a self contained custom powershelgl console. It features powerful tab-completion covering commands, parameters and file paths. it does not rely on powershell.exe. It contains over 100 commands for privilege escalation., recon and data Exfilteration.

We will use Psattack command in this. PSattack is a command that allows you to search through the included commands and find the attack you are looking for.

```
C:\ #> get-attack passwords


Module      : PowershellMafia\Invoke-Mimikatz.ps1
Command     : Invoke-Mimikatz
Type        : Passwords
Description : This script leverages Mimikatz 2.0 and
              completely in memory. This allows you t
              mimikatz binary to disk. The script has
              multiple computers.

Module      : PowershellMafia\Invoke-GPPPassword.ps1
Command     : Get-GPPPassword
Type        : Passwords
Description : Retrieves the plaintext password and ot
              Preferences.

Module      : PowershellMafia\PowerUp.ps1
Command     : Get-ApplicationHost
Type        : Escalation
Description : This script will recover encrypted app
              applicationHost.config on the system.

Module      : Nishang\Get-WLAN-Keys.ps1
Command     : Get-WLAN-Keys
Type        : Passwords
Description : Nishang Payload which dumps keys for WL
```

psattack will give you a list of commands and their description what it does. I used the Get-NetProcess command, it will give you a list of all the services and on what computer is it running on if the computer is a member of domain network. It also shows you the process ID and username

```
C:\Tools\PSAttack\x86 #> Get-NetProcess


ComputerName : localhost
ProcessName  : System Idle Process
ProcessID    : 0
Domain       :
User         :

ComputerName : localhost
ProcessName  : System
ProcessID    : 4
Domain       :
User         :

ComputerName : localhost
ProcessName  : Registry
ProcessID    : 136
Domain       : NT AUTHORITY
User         : SYSTEM

ComputerName : localhost
ProcessName  : smss.exe
ProcessID    : 444
Domain       : NT AUTHORITY
User         : SYSTEM

ComputerName : localhost
ProcessName  : csrss.exe
ProcessID    : 528
Domain       : NT AUTHORITY
User         : SYSTEM

ComputerName : localhost
ProcessName  : wininit.exe
ProcessID    : 604
Domain       : NT AUTHORITY
User         : SYSTEM

ComputerName : localhost
ProcessName  : csrss.exe
```

I also used the Invoke-ALLChecks.

```
C:\Tools\PSAttack\x86 #>  Invoke-AllChecks

[*] Running Invoke-AllChecks
[+] Current user already has local administrative privileges!


[*] Checking for unquoted service paths...


[*] Checking service executable and argument permissions...


ServiceName                     : AJRouter
Path                            : C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : GenericAll
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : NT AUTHORITY\LocalService
AbuseFunction                   : Install-ServiceBinary -Name 'AJRouter'
CanRestart                      : True

ServiceName                     : AJRouter
Path                            : C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : {Delete, WriteAttributes, Synchronize, ReadControl...}
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : NT AUTHORITY\LocalService
AbuseFunction                   : Install-ServiceBinary -Name 'AJRouter'
CanRestart                      : True

ServiceName                     : AppIDSvc
Path                            : C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : GenericAll
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : NT Authority\LocalService
AbuseFunction                   : Install-ServiceBinary -Name 'AppIDSvc'
CanRestart                      : True

ServiceName                     : AppIDSvc
Path                            : C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : {Delete, WriteAttributes, Synchronize, ReadControl...}
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : NT Authority\LocalService
AbuseFunction                   : Install-ServiceBinary -Name 'AppIDSvc'
CanRestart                      : True

ServiceName                     : Appinfo
Path                            : C:\Windows\system32\svchost.exe -k netsvcs -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : GenericAll
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : LocalSystem
AbuseFunction                   : Install-ServiceBinary -Name 'Appinfo'
CanRestart                      : True

ServiceName                     : Appinfo
Path                            : C:\Windows\system32\svchost.exe -k netsvcs -p
ModifiableFile                  : C:\Windows\system32
ModifiableFilePermissions       : {Delete, WriteAttributes, Synchronize, ReadControl...}
ModifiableFileIdentityReference : BUILTIN\Administrators
StartName                       : LocalSystem
AbuseFunction                   : Install-ServiceBinary -Name 'Appinfo'
CanRestart                      : True

ServiceName                     : AppMgmt
Path                            : C:\Windows\system32\svchost.exe -k netsvcs -p
```

Figure 5

```
TaskName     : OfficeTelemetryAgentLogOn2016
TaskFilePath : @{ModifiablePath=C:\Program Files\Microsoft Office\root\Office16\msoia.exe; IdentityReference=B
TaskTrigger  : <Triggers xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"><LogonTrigger><Repetiti

TaskName     : Pre-staged app cleanup
TaskFilePath : @{ModifiablePath=C:\Windows\system32; IdentityReference=BUILTIN\Administrators; Permissions=Gen
TaskTrigger  : <Triggers xmlns="http://schemas.microsoft.com/windows/2004/02/mit/task"><LogonTrigger><Delay>PT
```

You can even save the file as the html file and open it into the browser. It shows you everything right from the service their paths, Modifiable permissions, abuse functions and if it can restart.

PowerUp report for 'COMMANDO.Administrator'

User Has Local Admin Privileges!

Unquoted Service Paths

Service File Permissions

ServiceName	Path
AJRouter	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
AJRouter	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
AppIDSvc	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
AppIDSvc	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p
Appinfo	C:\Windows\system32\svchost.exe -k netsvcs -p
Appinfo	C:\Windows\system32\svchost.exe -k netsvcs -p
AppMgmt	C:\Windows\system32\svchost.exe -k netsvcs -p

PowerUp report for 'COMMANDO.Administrator'

User Has Local Admin Privileges!

Unquoted Service Paths

Service File Permissions

ServiceName	Path	ModifiableFile	ModifiableFilePermissions	ModifiableFileIdentityReference	StartName	AbuseFunction	CanRestar
AJRouter	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'AJRouter'	True
AJRouter	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'AJRouter'	True
AppIDSvc	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT Authority\LocalService	Install-ServiceBinary -Name 'AppIDSvc'	True
AppIDSvc	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT Authority\LocalService	Install-ServiceBinary -Name 'AppIDSvc'	True
Appinfo	C:\Windows\system32\svchost.exe -k netsvcs -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'Appinfo'	True
Appinfo	C:\Windows\system32\svchost.exe -k netsvcs -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'Appinfo'	True
AppMgmt	C:\Windows\system32\svchost.exe -k netsvcs -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppMgmt'	True
AppMgmt	C:\Windows\system32\svchost.exe -k netsvcs -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppMgmt'	True
AppReadiness	C:\Windows\System32\svchost.exe -k AppReadiness -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppReadiness'	True
AppReadiness	C:\Windows\System32\svchost.exe -k AppReadiness -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppReadiness'	True
AppXSvc	C:\Windows\system32\svchost.exe -k wsappx -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppXSvc'	True
AppXSvc	C:\Windows\system32\svchost.exe -k wsappx -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AppXSvc'	True
AssignedAccessManagerSvc	C:\Windows\system32\svchost.exe -k AssignedAccessManagerSvc	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AssignedAccessManagerSvc'	True
AssignedAccessManagerSvc	C:\Windows\system32\svchost.exe -k AssignedAccessManagerSvc	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AssignedAccessManagerSvc'	True
AudioEndpointBuilder	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AudioEndpointBuilder'	True
AudioEndpointBuilder	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AudioEndpointBuilder'	True
Audiosrv	C:\Windows\System32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'Audiosrv'	True
Audiosrv	C:\Windows\System32\svchost.exe -k LocalServiceNetworkRestricted -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'Audiosrv'	True
autotimesvc	C:\Windows\system32\svchost.exe -k autoTimeSvc	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'autotimesvc'	True
autotimesvc	C:\Windows\system32\svchost.exe -k autoTimeSvc	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'autotimesvc'	True
AxInstSV	C:\Windows\system32\svchost.exe -k AxInstSVGroup	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AxInstSV'	True
AxInstSV	C:\Windows\system32\svchost.exe -k AxInstSVGroup	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'AxInstSV'	True
BDESVC	C:\Windows\System32\svchost.exe -k netsvcs -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	localSystem	Install-ServiceBinary -Name 'BDESVC'	True
BDESVC	C:\Windows\System32\svchost.exe -k netsvcs -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	localSystem	Install-ServiceBinary -Name 'BDESVC'	True
BFE	C:\Windows\system32\svchost.exe -k LocalServiceNoNetworkFirewall -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BFE'	False
BFE	C:\Windows\system32\svchost.exe -k LocalServiceNoNetworkFirewall -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BFE'	False
BITS	C:\Windows\System32\svchost.exe -k netsvcs -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'BITS'	True
BITS	C:\Windows\System32\svchost.exe -k netsvcs -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'BITS'	True
BrokerInfrastructure	C:\Windows\system32\svchost.exe -k DcomLaunch -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'BrokerInfrastructure'	True
BrokerInfrastructure	C:\Windows\system32\svchost.exe -k DcomLaunch -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'BrokerInfrastructure'	True
BTAGService	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BTAGService'	True
BTAGService	C:\Windows\system32\svchost.exe -k LocalServiceNetworkRestricted	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BTAGService'	True
BthAvctpSvc	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BthAvctpSvc'	True
BthAvctpSvc	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'BthAvctpSvc'	True
bthserv	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'bthserv'	True
bthserv	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'bthserv'	True
camsvc	C:\Windows\system32\svchost.exe -k appmodel -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'camsvc'	True
camsvc	C:\Windows\system32\svchost.exe -k appmodel -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'camsvc'	True
CDPSvc	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'CDPSvc'	True
CDPSvc	C:\Windows\system32\svchost.exe -k LocalService -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'CDPSvc'	True
CertPropSvc	C:\Windows\system32\svchost.exe -k netsvcs	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'CertPropSvc'	True
CertPropSvc	C:\Windows\system32\svchost.exe -k netsvcs	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'CertPropSvc'	True
client_service	C:\Program Files (x86)\VMware\VMware Horizon View Client\ClientService\horizon_client_service.exe" -SCMStartup mfvStartFlag=2	C:\Program Files (x86)\VMware\VMware Horizon View Client\ClientService\horizon_client_service.exe	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'client_service'	True
ClipSVC	C:\Windows\System32\svchost.exe -k wsappx -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'ClipSVC'	True
ClipSVC	C:\Windows\System32\svchost.exe -k wsappx -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'ClipSVC'	True
com.docker.service	"C:\Program Files\Docker\Docker\com.docker.service"	C:\Program Files\Docker\Docker\com.docker.service	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'com.docker.service'	True
CoreMessagingRegistrar	C:\Windows\system32\svchost.exe -k LocalServiceNoNetwork -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'CoreMessagingRegistrar'	True
CoreMessagingRegistrar	C:\Windows\system32\svchost.exe -k LocalServiceNoNetwork -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT AUTHORITY\LocalService	Install-ServiceBinary -Name 'CoreMessagingRegistrar'	True
CryptSvc	C:\Windows\system32\svchost.exe -k NetworkService -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	NT Authority\NetworkService	Install-ServiceBinary -Name 'CryptSvc'	True
CryptSvc	C:\Windows\system32\svchost.exe -k NetworkService -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	NT Authority\NetworkService	Install-ServiceBinary -Name 'CryptSvc'	True
CscService	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\System32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'CscService'	True
CscService	C:\Windows\System32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\System32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'CscService'	True
DcomLaunch	C:\Windows\system32\svchost.exe -k DcomLaunch -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'DcomLaunch'	True
DcomLaunch	C:\Windows\system32\svchost.exe -k DcomLaunch -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'DcomLaunch'	True
defragsvc	C:\Windows\system32\svchost.exe -k defragsvc	C:\Windows\system32	GenericAll	BUILTIN\Administrators	localSystem	Install-ServiceBinary -Name 'defragsvc'	True
defragsvc	C:\Windows\system32\svchost.exe -k defragsvc	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	localSystem	Install-ServiceBinary -Name 'defragsvc'	True
DeviceAssociationService	C:\Windows\system32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'DeviceAssociationService'	True
DeviceAssociationService	C:\Windows\system32\svchost.exe -k LocalSystemNetworkRestricted -p	C:\Windows\system32	System.Object[]	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'DeviceAssociationService'	True
DeviceInstall	C:\Windows\system32\svchost.exe -k DcomLaunch -p	C:\Windows\system32	GenericAll	BUILTIN\Administrators	LocalSystem	Install-ServiceBinary -Name 'DeviceInstall'	True

Step 3

In this step, we will be using OWASP ZSC. It is an open source software in python which lets you generate customized shell codes and converts scripts into obfuscated scripts. Shellcodes are small assembly language which could be used as the payload in software exploitation.

shell codes generated from zsc can be used for penetration testing.

Once you open the zsc and use the command help, it will show you how to work around with the shellcodes.

```
zsc> help

[+] shellcode            generate shellcode
[+] shellcode>generate   to generate shellcode
[+] shellcode>search     search for shellcode in shellstorm
[+] shellcode>download   download shellcodes from shellstorm
[+] shellcode>shell_storm_list  list all shellcodes in shellstorm
[+] obfuscate            generate obfuscate code
[+] back                 Go back one step
[+] clear                clears the screen
[+] help                 show help menu
[+] update               check for update
[+] about                about owasp zsc
[+] restart              restart the software
[+] version              software version
[+] exit/quit            to exit the software
[+] #                    insert comment
[+] zsc -h, --help       basic interface help

zsc>
```

use the command shell code to bet into the shell code interface, then use generate and press tab which will show you all the available OS you can create the shell code for.

```
 ______        __    _____ _____   ______ _____  _____
/ __ \ \      / /\   / ____|  __ \ |___  // ____|/ ____|
| | | | \ \  / /  \ | (___ | |__) |   / /| (___ | |
| | | | |\ \/ / /\ \ \___ \|  ___/   / / \___ \| |
| |__| | \  /\  / ____ \ ____) | |    / /__ ____) | |____
 \____/   \/  \/_/    \_\_____/|_|   /_____|_____/ \_____|

                OWASP ZeroDay Cyber Research Shellcoder

zsc>
zsc> shellcode
zsc/shellcode> generate
zsc/shellcode/generate>
linux_x86       osx_x86         windows_x86     windows_x86_64
```

You can select the different types of encoder. i selected none to keep it simple for this lab.

```
  if len(line) is 13 or len(line) is 12:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:426: SyntaxWarning: "is" with a literal. Did you mean "==
  if len(line) is 13 or len(line) is 12:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:433: SyntaxWarning: "is" with a literal. Did you mean "==
  if len(line) is 9:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:436: SyntaxWarning: "is" with a literal. Did you mean "==
  if len(line) is 10:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:439: SyntaxWarning: "is" with a literal. Did you mean "==
  if len(line) is 15:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:440: SyntaxWarning: "is" with a literal. Did you mean "==
  if _version is 2:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:443: SyntaxWarning: "is" with a literal. Did you mean "==
  if _version is 3:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:448: SyntaxWarning: "is" with a literal. Did you mean "==
  if len(line) is 16:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:449: SyntaxWarning: "is" with a literal. Did you mean "==
  if _version is 2:
C:\Python38\Scripts\zsc\lib\opcoder\linux_x86.py:452: SyntaxWarning: "is" with a literal. Did you mean "==
  if _version is 3:

[+] none
[+] xor_random
[+] xor_yourvalue
[+] add_random
[+] add_yourvalue
[+] sub_random
[+] sub_yourvalue
[+] inc
[+] inc_timesyouwant
[+] dec
[+] dec_timesyouwant
[+] mix_all
```

You can see the output off the shell code by pressing y at the choice. It will output the assembly code.

```
[+] enter encode type
zsc/shellcode/generate/linux_x86/system/encode_type> none

Output assembly code?(y or n)> y

push   $0xb
pop    %eax
cltd
push   %edx
push $0x68732f90
pop %ecx
shr    $0x8,%ecx
push %ecx

push $0x6e69622f

mov    %esp,%esi
push   %edx
push   $0x632d9090
pop    %ecx
shr    $0x10,%ecx
push   %ecx
mov    %esp,%ecx
push   %edx
push   $0x68
push   $0x7361622f
push   $0x6e69622f
mov    %esp,%ebx
push   %edx
push   %edi
push   %esi
push   %ecx
push   %ebx
mov    %esp,%ecx
int    $0x80
```

You can accept the output shell code to screen and see the shell code created.

```
Output shellcode to screen?(y or n)> y
[+] Generated shellcode is:
\x6a\x0b\x58\x99\x52\x68\x90\x2f\x73\x68\x59\xc1\xe9\x08\x51\x68\x2f\x62\x69\x6e\x89\xe6\x52\x68\x90\x90\x2d\x63\x59\xc1\xe9\x10\x51\x89\xe1\x52\x6a\x68\x68\x2f\x62\x61\x73\x68\x2f\x62\x69\x6e\x89\xe3\x52\x57\x56\x51\x53\x89\xe1\xcd\x80

Shellcode output to a .c file?(y or n)> y
Target .c file?> C:\code
[+] File saved as C:\code .
zsc> help
```

You can use this shell code to obfuscate the documents.