

Author: Inj3cti0n P4ck3t

Date: 14/10/10

Nome do Artigo: Criando Scanner para Dectar SMTPs vulneráveis ao Metasploit

Contato: fer_henrick@hotmail.com

Linguagem de Programação: Perl (Practical Extraction and Report Language)

0x01 Introdução

O Metasploit Framework é uma ferramenta utilizada para explorar falhas em diversos tipos de softwares e serviços.

Qualquer usuário pode utilizar o Metasploit Framework para atacar um sistema operacional que executa um determinado serviço. Mas a diferença em usar o Metasploit, está na conhecimento e no sucesso de explorar falhas de segurança.

Uma técnica é atacar o servidor com todos os exploits do Metasploit.

Outra técnica é desenvolver um scanner para identificar serviços vulneráveis aos exploits do Metasploit Framework.

No artigo, escrevo como desenvolver um scanner na linguagem de programação PERL que identifica servidores SMTPs vulneráveis ao software Metasploit Framework.

0x02 Versões de SMTPs vulneráveis aos exploits do Metasploit Framework

O Metasploit possui muitos exploits para explorar falhas em serviços. Mas vamos escrever um scanner na linguagem de programação PERL que identifique o serviços de SMTPs vulneráveis para o sistema operacional Windows.

Os serviços de SMTPs identificados pelo scanner são vulneráveis aos exploits do Metasploit Framework 3.3.

A versões de SMTPs vulneráveis aos exploits do Metasploit Framework e que serão identificadas pelo scanner em PERL estão listadas:

- **YPOPS 0.6 Buffer Overflow**

- **Exploit:** ypops_overflow1

- TABS MailCarrier v2.51 SMTP EHLO Overflow
- Exploit: mailcarrier_smtp_ehlo

0x03 Softwares para testar o Scanner

Metasploit Framework 3.3

Download: <http://www.metasploit.com/releases/framework-3.4.1.exe>

Active Perl (Interpretador Perl para Windows)

Download: <http://downloads.activestate.com/ActivePerl/releases/5.12.2.1202/ActivePerl-5.12.2.1202-MSWin32-x86-293621.msi>

Softwares vulneráveis:

Download: Indisponível, pois não foram encontrados

0x04 Desenvolvendo um Scanner para Dectar SMTPs vulneráveis aos exploits do Metasploit

Se você usa sistema operacional Linux, o caminho dos scripts em Perl será usr/bin/perl, como apresento abaixo.

```
#!/usr/bin/perl
```

Se você fez download do Active Perl e instalou no sistema operacional Windows, o caminho para inserir o script que estamos desenvolvendo será c:\perl\bin.

```
#!C:\Perl\Bin
```

Os Módulos usados no desenvolvimento do scanner.

Vamos utilizar o módulo **IO::Socket** e **IO::Socket::INET** para fazer a conexão com o servidor SMTP.

```
use IO::Socket;  
use IO::Socket::INET;
```

A linha de condição "IF" é executada.

Se o usuário não informou a lista com IPs ou sites para scanear as instruções contidas entre chaves ou no bloco são executadas.

A variável "\$sis" recebe "\$^O" para verificar o sistema operacional: Linux ou Windows.

Se o sistema operacional é Windows o comando de sistema "cls" será executado.

Se o sistema operacional é Linux o comando de sistema "clear" será executado.

O comando "print q { ... }" apresenta as linhas ou o banner de como usar o scanner.

A linha "exit();" finaliza o programa.

```
if (!$ARGV[0]) {  
  
    $sis="$^O";if ($sis eq windows){ $cmd="clear";} else { $cmd="cls"; }  
    system("$cmd");  
    print q {  
  
        Code desenvolvido por: Inj3cti0n P4ck3t  
        e-mail to contact: fer\_henrick@hotmail.com  
        Modo de uso: perl ScanMetasploitSMTP.pl listaDelps_ou_ListaDeSites.txt  
        Nome: Fernando Henrique Mengali de Souza  
  
    };  
  
    exit();  
}
```

A linha abaixo, abre o arquivo de sites ou IPS informado pelo usuário, caso não seja possível abrir o arquivo, a mensagem: "Nao foi possível abrir o arquivo:" é apresentada.

Geralmente, um arquivo não pode ser aberto pelo seguinte:

- O arquivo não está no diretório do script
- O nome do arquivo está errado
- O arquivo pode estar corrompido

```
open( SITE, "< $ARGV[0]" ) or die( "Nao foi possível abrir o arquivo: $" );  
# Abri o arquivo .txt informado pelo usuário
```

Atribuímos ao array "@array" a lista de sites ou IPs para scanear.

A variável "\$numero" recebe o último vetor, pois será usado no laço de interação "for".

Após a variável \$numero receber o último array, na próxima linha temos um "for".

O "for" começará a ser executado:

```
our @array = <SITE>; # array recebe o sites contidos no arquivo  
  
$numero = $#array; # $numero possui o ultimo array  
  
for ($i = 0; $i <= $numero; $i++) { # inicia-se o laço for
```

Vamos usar a variável “\$Url” para armazenar o endereço IP ou do site alvo.
Se o endereço alvo não possui o protocolo HTTP, usamos um “IF” como condição.
Se endereço não possui HTTP, o if inseri HTTP. Exemplo:

Não possui o protocolo HTTP
192.168.0.3

O “IF” verifica o endereço 192.168.0.3, não possui o protocolo HTTP. Então, inseri:
http://192.168.0.3

O endereço IP foi verificado pelo IF, o resultado foi inserir o protocolo HTTP:

```
$Url = "$array[$i]";
```

```
if($Url !~ /http:\V\/) { $Url = "http://$Url"; }
```

Agora, formatamos o endereço alvo usando o código abaixo:

```
$Stop = index($Url,":");
```

```
$Protocolo = substr($Url,0,$Stop);
```

```
$Start = index($Url,"/") + 2;
```

```
$Dominio = substr($Url,$Start);
```

```
$Stop = index($Dominio,"/");
```

```
$Dominio = substr($Dominio,0,$Stop);
```

```
$Start = rindex($Url,"/") + 1;
```

```
$NomeArq = substr($Url,$Start);
```

```
$Compr_Url = length($Url);
```

```
$ponto = "$Dominio"; #A variável $ponto, recebe a $Dominio ou URL formatada.
```

Criamos um simples array para armazenar a porta 25.

Depois criamos um “foreach”, atribuindo a variável “\$porta” o valor 25.

```
our @portas = "25";
```

```
foreach $porta (@portas) {
```

Vamos criar a variável \$sock, que armazenará a conexão com o IP ou site armazenado na posição do array e a porta de conexão: 22.

Na próxima linha, usamos a variável \$sock como expressão para ser avaliada, ou seja, se a conexão for estabelecida com o IP e porta 22 o bloco será executado.

Caso o a expressão não retornar verdadeiro(TRUE), porque não conseguiu conectar ao alvo na porta 22, nenhuma informação é retornada no terminal.

```
$sock = IO::Socket::INET->new("$ponto:$porta");
```

```
if($sock) {
```

Se a conexão retornar TRUE, o bloco do "IF" é executado, portanto a variável \$remote terá uma conexão com o servidor alvo, usando o protocolo TCP e com Timeout 7.

A variável \$line agora possui resposta da conexão, ou seja, "o banner".

Sabemos que a variável \$line possui a resposta do servidor que conectamos, ou seja, temos o banner do servidor que conectamos.

```
$remote = IO::Socket::INET -> new (Proto => "tcp", PeerAddr => $ponto, PeerPort => $porta, Timeout => "7");
```

```
$line = <$remote>;
```

Quando a expressão avaliada pelo comando "IF" resultar em verdadeiro, isto é, quando o operador "=" verificar se o conteúdo armazenado na variável corresponde ao texto ou frases entre aspas. o bloco será executado.

O bloco corresponde ao print do IP ou site com o servidor vulnerável e banner do servidor vulnerável.

Caso, não haja servidor vulnerável aos banners informados não será apresentado nenhuma mensagem.

```
if ($line =~ "ESMTP TABS Mail Server for Windows NT") {
    print "Server: $ponto vulnerável\n";
    print "Exploit: mailcarrier_smtp_ehlo \n\n";
}

if ($line =~ "YahooPOPs! Simple Mail Transfer Service Ready") {
    print "Server: $ponto vulnerável\n";
    print "Exploit: ypops_overflow1 \n\n";
}
}
}
```

0x05 Código Completo do Scanner para Dectar SMTPs vulneráveis aos exploits do Metasploit

```
use IO::Socket;
use IO::Socket::INET;
```

```
if (!$ARGV[0]) {
```

```

$sis="$^O";if ($sis eq windows){ $cmd="clear";} else { $cmd="cls"; }
system("$cmd");

print q { Code desenvolvido por: Inj3cti0n P4ck3t
          e-mail to contact: fer_henrick@hotmail.com
          Modo de uso: perl ScanMetasploitSMTP.pl listaDelps_ou_ListaDeSites.txt
          Nome: Fernando Henrique Mengali de Souza

          };

exit();

}

open( SITE, "< $ARGV[0]" ) or die( "Nao foi possível abrir o arquivo: $!" );
our @array = <SITE>;

$numero = $#array;

for ($i = 0; $i <= $numero; $i++) {

    $Url = "$array[$i]";

    if($Url !~ /http:\\\\\/) { $Url = "http://$Url"; }

    $Stop = index($Url,":");
    $Protocolo = substr($Url,0,$Stop);
    $Start = index($Url,"//") + 2;
    $Dominio = substr($Url,$Start);
    $Stop = index($Dominio,"/");
    $Dominio = substr($Dominio,0,$Stop);
    $Start = rindex($Url,"/") + 1;
    $NomeArq = substr($Url,$Start);
    $Compr_Url = length($Url);

    $ponto = "$Dominio \n";

    our @portas = "25";

    foreach $porta (@portas) {

        $sock = IO::Socket::INET->new("$ponto:$porta");

        if($sock) {

            $remote = IO::Socket::INET -> new (Proto=> "tcp", PeerAddr=> $ponto, PeerPort=> $porta,
            Timeout=> "7");

            $line = <$remote>;

```

```
if ($line =~ "ESMTP TABS Mail Server for Windows NT") {  
    print "Server: $ponto vulnerável\n";  
    print "Exploit: mailcarrier_smtp_ehlo \n\n";  
  
}  
  
if ($line =~ "YahooPOPs! Simple Mail Transfer Service Ready") {  
    print "Server: $ponto vulnerável\n";  
    print "Exploit: ypops_overflow1 \n\n";  
  
        }  
  
    }  
  
}
```

Agradecimentos:

C00l3r - DD3str0y3r - Sh0rtKiller -CODE RED - Archit3ct