****

**Visão**

Com a crescente demanda sobre Tecnologias, percebemos que muitas pessoas apesar de buscarem informações, não possuem fontes que queiram realmente passar o conhecimento da maneira como ela deve ser, livre e com embasamento técnico que permita ser aplicado e utilizado quando necessário, além de serem testados em sua criação, tornando esta informação útil e confiável.

**Missão**

O Laboratório foi criado com a intenção de buscar e disseminar o conhecimento de uma maneira clara e objetiva, de forma gratuita, auxiliando na evolução dos membros e da sociedade na qual estas informações são compartilhadas, buscando o crescimento de todos os envolvidos nesta criação de valores.



Caso você pense que com a leitura dos materiais da How2Security, você irá se tornar um Cracker capaz de invadir sistemas, se você espera encontrar aqui scripts infalíveis para invasão e, a partir deles, sair por aí invadindo computadores, essa não é a leitura indicada. Indicamos, sim a leitura do Código Penal (Lei 2.848/1940), principalmente a Lei Carolina Dickmann (Lei 12.737/2012), nos Artigos 154-A e 154-B.

*154-A Invadir dispositivo informático alheio, conectado ou não à rede de computadores, mediante violação indevida de mecanismo de segurança e com o fim de obter, adulterar ou destruir dados ou informações sem autorização expressa ou tácita do titular do dispositivo ou instalar vulnerabilidades para obter vantagem ilícita:*

*Pena – Detenção, de 3 meses a 1 ano, e multa*

Este material é um conjunto de informações compiladas de documentos e ferramentas do Mundo Underground testadas em ambiente de laboratório na nossa intranet. Desta forma, todo conhecimento aqui condensado é tangível, assim como as orientações das contramedidas.

Dessa forma, esperamos ter sido bem claros que, em momento algum, estamos com a pretensão de ensinar a você como se tornar um invasor. Estaremos sim, mostrando muitas das técnicas utilizadas pelos crackers e, em alguns casos, pelos scripts kiddies, para que você, como administrador de redes, seja capaz de identificá-las em tempo hábil para se defender, antes que alguém com desejos menos nobres ô faça por você.

Assim sendo, todo o conteúdo dessa literatura tem apenas o objetivo didático de informar e preparar os administradores de redes dos novos tempos. Em momento algum nos responsabilizamos pelo mau uso desse conhecimento ou por danos causados em seu equipamento ou de terceiros, assim como também não somos responsáveis pelos códigos e ferramentas aqui citados.

Sandro Melo

Adaptado por Wellington Silva aka Well

**0 – Recon Ativo – Servidor Web**

Neste reconhecimento devemos descobrir quais as tecnologias envolvidas no Servidor Web, onde a aplicação alvo está executando.

Para isso, podemos executar alguns comandos e aplicativos para nos ajudar nessa busca.

**1 – Conhecendo o Servidor Web**

Vamos fazer uma conexão com nosso alvo utilizando o NetCat passar o método HEAD para saber a resposta do servidor.

root@kali-wellx64:/opt/well-tools# nc 192.168.0.110 80

HEAD / HTTP/1.0

HTTP/1.1 200 OK

Date: Fri, 24 Feb 2017 19:44:02 GMT

**Server: Apache/2.2.14 (Ubuntu)**

Vary: Accept-Encoding

Connection: close

Content-Type: text/html;charset=UTF-8

root@kali-wellx64:/opt/well-tools# nc -v www.altoromutual.com 80

Warning: inverse host lookup failed for 65.61.137.117: Unknown host

altoromutual.com [65.61.137.117] 80 (http) open

HEAD / HTTP/1.0

HTTP/1.1 200 OK

Cache-Control: no-cache

Pragma: no-cache

Content-Length: 9550

Content-Type: text/html; charset=utf-8

Expires: -1

**Server: Microsoft-IIS/8.0**

X-AspNet-Version: 2.0.50727

Set-Cookie: ASP.NET\_SessionId=pf5qsx55bu1z5g55cyj3ixyp; path=/; HttpOnly

Set-Cookie: amSessionId=1430351278838; path=/

**X-Powered-By: ASP.NET**

Date: Fri, 24 Feb 2017 20:30:34 GMT

Connection: close

root@kali-wellx64:/opt/well-tools#

A resposta do servidor no primeiro caso foi um Apache/2.2.14 (Ubuntu), onde podemos deduzir que é um servidor Apache na versão 2.2.14 instalado em um sistema operacional Linux Ubuntu.

Na segunda resposta obtivemos que o servidor é um Microsoft IIS versão 8.0 utilizando ASP.NET e provavelmente instalado em um Windows Server 2008/2012.

Também podemos utilizar o cURL da seguinte forma:

root@kali-wellx64:/opt/well-tools# curl -v -X HEAD http://192.168.0.110

Warning: Setting custom HTTP method to HEAD with -X/--request may not work the

Warning: way you want. Consider using -I/--head instead.

\* Rebuilt URL to: http://192.168.0.110/

\* Trying 192.168.0.110...

\* TCP\_NODELAY set

\* Connected to 192.168.0.110 (192.168.0.110) port 80 (#0)

> HEAD / HTTP/1.1

> Host: 192.168.0.110

> User-Agent: curl/7.52.1

> Accept: \*/\*

>

< HTTP/1.1 200 OK

< Date: Fri, 24 Feb 2017 20:28:18 GMT

**< Server: Apache/2.2.14 (Ubuntu)**

< Vary: Accept-Encoding

< Content-Type: text/html;charset=UTF-8

\* no chunk, no close, no size. Assume close to signal end

<

\* Curl\_http\_done: called premature == 0

\* Closing connection 0

root@kali-wellx64:/opt/well-tools# curl -v -X HEAD http://www.altoromutual.com

Warning: Setting custom HTTP method to HEAD with -X/--request may not work the

Warning: way you want. Consider using -I/--head instead.

\* Rebuilt URL to: http://www.altoromutual.com/

\* Trying 65.61.137.117...

\* TCP\_NODELAY set

\* Connected to www.altoromutual.com (65.61.137.117) port 80 (#0)

> HEAD / HTTP/1.1

> Host: www.altoromutual.com

> User-Agent: curl/7.52.1

> Accept: \*/\*

>

< HTTP/1.1 200 OK

< Cache-Control: no-cache

< Pragma: no-cache

< Content-Length: 9550

< Content-Type: text/html; charset=utf-8

< Expires: -1

**< Server: Microsoft-IIS/8.0**

< X-AspNet-Version: 2.0.50727

< Set-Cookie: ASP.NET\_SessionId=qg1s1eejnks0ph55oy2gs52b; path=/; HttpOnly

< Set-Cookie: amSessionId=142911278352; path=/

**< X-Powered-By: ASP.NET**

< Date: Fri, 24 Feb 2017 20:29:01 GMT

<

\* Curl\_http\_done: called premature == 0

\* Closing connection 0

root@kali-wellx64:/opt/well-tools#

A resposta foi a mesma do comando anterior, apenas utilizamos uma ferramenta diferente.

**2 – Conhecendo os Métodos Suportados Pelo Servidor Web**

Agora vamos descobrir quais os métodos HTTP estão habilitados no servidor Web, para isso, devemos passar o método OPTIONS.

root@kali-wellx64:/opt/well-tools# nc -v 192.168.0.110 80

192.168.0.110: inverse host lookup failed: Unknown host

(UNKNOWN) [192.168.0.110] 80 (http) open

OPTIONS / HTTP/1.0

HTTP/1.1 200 OK

Date: Fri, 24 Feb 2017 20:43:12 GMT

Server: Apache/2.2.14 (Ubuntu)

**Allow: GET,HEAD,POST,OPTIONS**

Content-Length: 0

Connection: close

Content-Type: httpd/unix-directory

root@kali-wellx64:/opt/well-tools# curl -v -X OPTIONS http://www.altoromutual.com

\* Rebuilt URL to: http://www.altoromutual.com/

\* Trying 65.61.137.117...

\* TCP\_NODELAY set

\* Connected to www.altoromutual.com (65.61.137.117) port 80 (#0)

> OPTIONS / HTTP/1.1

> Host: www.altoromutual.com

> User-Agent: curl/7.52.1

> Accept: \*/\*

>

< HTTP/1.1 200 OK

**< Allow: OPTIONS, TRACE, GET, HEAD, POST**

< Server: Microsoft-IIS/8.0

< Public: OPTIONS, TRACE, GET, HEAD, POST

< X-Powered-By: ASP.NET

< Date: Fri, 24 Feb 2017 20:40:59 GMT

< Content-Length: 0

<

\* Curl\_http\_done: called premature == 0

\* Connection #0 to host www.altoromutual.com left intact

root@kali-wellx64:/opt/well-tools#

Como podemos ver no Apache estão habilitados os métodos GET, POST, HEAD e OPTIONS, já no IIS estão habilitados GET, POST, HEAD, OPTIONS e TRACE.

Só para demostrar outras duas ferramentas muito utilizadas que é o NMAP e o Metasploit, iremos utilizar ambas para capturar essas informações.

Primeiro iremos utilizar o NMAP com o script **http-methods.nse**. Todos os script que podemos utilizar com o NMAP estão no seguinte diretório: ***/usr/share/nmap/scripts/***.

Para uma documentação completa da utilização dos scripts basta consultar a documentação no próprio site do NMAP (https://nmap.org/nsedoc/scripts/).

root@kali-wellx64:/opt/well-tools# nmap --script=http-methods.nse 192.168.0.110 -n -p 80

Starting Nmap 7.40 ( https://nmap.org ) at 2017-02-24 17:53 BRT

Nmap scan report for 192.168.0.110

Host is up (0.00073s latency).

PORT STATE SERVICE

**80/tcp open http**

**| http-methods:**

**|\_ Supported Methods: GET HEAD POST OPTIONS**

MAC Address: 00:0C:29:73:06:CC (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.44 seconds

root@kali-wellx64:/opt/well-tools# nmap --script=http-methods.nse www.altoromutual.com -n -p 80

Starting Nmap 7.40 ( https://nmap.org ) at 2017-02-24 17:54 BRT

Nmap scan report for www.altoromutual.com (65.61.137.117)

Host is up (1.5s latency).

PORT STATE SERVICE

**80/tcp open http**

**| http-methods:**

**| Supported Methods: OPTIONS TRACE GET HEAD POST**

**|\_ Potentially risky methods: TRACE**

Nmap done: 1 IP address (1 host up) scanned in 8.98 seconds

root@kali-wellx64:/opt/well-tools# nmap --script=http-methods.nse --script-args http-methods.retest=1 www.altoromutual.com -n -p 80

Starting Nmap 7.40 ( https://nmap.org ) at 2017-02-24 17:55 BRT

Nmap scan report for www.altoromutual.com (65.61.137.117)

Host is up (0.19s latency).

PORT STATE SERVICE

**80/tcp open http**

**| http-methods:**

**| Supported Methods: OPTIONS TRACE GET HEAD POST**

**| Potentially risky methods: TRACE**

**| Status Lines:**

**| OPTIONS: HTTP/1.1 200 OK**

**| GET: HTTP/1.1 200 OK**

**| TRACE: HTTP/1.1 501 Not Implemented**

**| POST: HTTP/1.1 200 OK**

**|\_ HEAD: HTTP/1.1 200 OK**

Nmap done: 1 IP address (1 host up) scanned in 4.03 seconds

root@kali-wellx64:/opt/well-tools#

Agora vamos utilizar o Metasploit com seus módulos auxiliares.

root@kali-wellx64:/opt/well-tools# msfconsole

msf > use auxiliary/scanner/http/options

msf auxiliary(options) > show options

Module options (auxiliary/scanner/http/options):

Name Current Setting Required Description

---- --------------- -------- -----------

Proxies no A proxy chain of format type:host:port[,[...]

**RHOSTS yes** The target address range or CIDR identifier

**RPORT 80 yes** The target port (TCP)

SSL false no Negotiate SSL/TLS for outgoing connections

THREADS 1 yes The number of concurrent threads

VHOST no HTTP server virtual host

msf auxiliary(options) > set RHOSTS 192.168.0.110

RHOSTS => 192.168.0.110

msf auxiliary(options) > set RPORT 80

RPORT => 80

msf auxiliary(options) > run

[\*] **192.168.0.110 allows GET,HEAD,POST,OPTIONS methods**

[\*] Scanned 1 of 1 hosts (100% complete)

[\*] Auxiliary module execution completed

msf auxiliary(options) > set RHOSTS www.altoromutual.com

RHOSTS => www.altoromutual.com

msf auxiliary(options) > run

[\*] **65.61.137.117 allows OPTIONS, TRACE, GET, HEAD, POST methods**

[\*] **65.61.137.117:80 - TRACE method allowed.**

[\*] Scanned 1 of 1 hosts (100% complete)

[\*] Auxiliary module execution completed

msf auxiliary(options) >

Como podemos ver todas as ferramentas executaram e fizeram seu trabalho de forma coesa.

Em muitos casos, os atacantes serão capazes de ver muito mais informações e reunir dados de inteligência sobre um alvo.

**3 – Referências Bibliográficas**

**[1]** MUNIZ, JOSEPH and LAKHANI, AAMIR – Web Penetration Testing with Kali Linux, 2013, Birmingham – Mumbai, Packet Publishing

**[2]** GRAVES, KIMBERLY – CEH – Official Ethical Hacker Review Guide, Indianapolis - Indiana, 2007, Wiley Publishing Inc.

**[3]** STUTTARD, DAFYDD and PINTO, MARCUS – The Web Application Hacker’s Handbook, Indianapolis - Indiana, 2º Edition, Wiley Publishing Inc.

**[4]** SCAMBRAY, JOEL, McCLURE, STUART and KURTZ, GEORGE – Hackers Expostos Segredos e Soluções para a Segurança de Redes, São Paulo, 2001, MAKRON Books Ltda.

**[5]** LONG, JOHNNY - GOOGLE HACKING - Para Teste de Invasão [tradução Sérgio Pereira Couto] 2005, DIGERATI (BOOKS)

**[6]** MELO, Sandro – Estudo de Técnicas para Exploração de Vulnerabilidades em Redes TCP/IP, 2º Ed, Rio de Janeiro, 2006, Editora Alta Books Ltda.

**[7]** AHARONI, Mati; Offensive Security Lab Exercises - Curso de Pen-Test oferecido pela Offensive Security, 2007

**[8]** SCAMBRAY, Joel; LIU, Vicent; SIMA, Caleb - Hacking Exposed Web Application: Web Application Security Secrets And Solutions – 3ª Edition, New York – 2011, McGraw-Hill

**[9]** DHANJANI, Nitesh; RIOS, Billy; HARDIN, Brett - Hacking A Próxima Geração – Rio de Janeiro – 2011, Alta Books