**ESQUEMA DE PLANO DE AULA PRÁTICA**

**TAREFA 5** "Ferramentas de Aquisição de Informações"

Tópico no Moodle: Scanner de Vulnerabilidades para Servidores Web  
 (Nikto)

Data:   
  
Tempo de Aula: 1 hora-aula  
  
Grupo: <Aluno\_1> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

<Aluno\_2> \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ASSUNTO**

Nikto é um scanner de servidor web Open Source (lincença GPL) que realiza testes abrangentes contra servidores para vários itens, incluindo mais de 6.500 arquivos potencialmente perigosos/CGIs, verificações de versões desatualizadas de mais de 1.250 servidores, e os problemas específicos de versão sobre mais de 270 servidores.

Nikto também verifica a existência de itens de configuração do servidor, tais como a presença de vários arquivos de índice, opções de servidor HTTP, e vai tentar identificar servidores web e software instalados. Itens de varredura e plugins são atualizados com frequência e podem ser atualizados automaticamente.

Nikto não foi concebido como uma ferramenta excessivamente furtiva. Ele irá testar um servidor web no menor tempo possível, e é bastante óbvio em arquivos de log. No entanto, há suporte para métodos anti-IDS do *LibWhisker* no caso de você querer experimentá-lo (ou testar o seu sistema IDS).

Nem todo teste é um problema de segurança, embora a maioria seja. Existem alguns itens que são apenas verificações de tipos de informação, que olham para as coisas que não podem ter uma falha de segurança, mas o webmaster ou engenheiro de segurança pode não saber se estão presentes no servidor. Estes itens são normalmente marcados apropriadamente com informação impressa. Há também algumas verificações para os itens desconhecidos que tem sido vistos como escaneados nos arquivos de log.

**Nikto é baseada em PERL**

Nikto é implementada em PERL (**Practical Extraction and Report Language).   
PERL** foiprojetada por **Larry Wall** como uma ferramenta para escrever programas no UNIX e está continuamente sendo mantidaa por ele.

Perl oferece o melhor de vários mundos. Por exemplo: Perl tem o poder da flexibilidade de uma linguagem de programação de alto nível, como C. Na verdade, como você poderá ver, muitas das características da linguagem são emprestados a partir de C.

Como linguagens de script shell, Perl não requer um compilador especial e vinculador de transformar os programas que você escrever em código de trabalho. Em vez disso, tudo que você tem a fazer é escrever o programa e dizer Perl para executá-lo. Isso significa que Perl é ideal para a produção de soluções rápidas para pequenos problemas de programação, ou para a criação de protótipos para testar possíveis soluções para problemas maiores.

Perl fornece todas as características das linguagens de script sed e awk, além de recursos não encontrados em qualquer uma destas duas línguas. Perl também suporta um tradutor sed-de-Perl e um tradutor awk-to-Perl.

Em suma, Perl é tão poderoso como C, mas tão conveniente como awk, sed e scripts shell.

**BIBLIOGRAFIA BÁSICA E INSTALAÇÕES**

Instalando PERL no Ubuntu:

<http://exploreurtalents.blogspot.com.br/2010/04/how-to-install-perl-on-ubuntu.html>  
  
1. Abre um terminal em $ (Ctrl Alt+t)

2. Instalando PERL

$sudo apt-get install perl

3. Instalando um editor PERL  
  
$sudo apt-get install scite

4. Link Importante:   
  
http://hackertarget.com/nikto-tutorial/

<http://cirt.net/nikto2/>

5. Instalando Nikto no Ubuntu:  
<http://www.vivaolinux.com.br/artigo/Checando-vulnerabilidades-com-o-Nikto>

Estando PERL pré-instalado no Ubuntu:

$sudo wget -c <http://www.cirt.net/nikto/nikto-current.tar.gz>

Após baixar o arquivo .tar.gz é preciso descompactar.   
$sudo tar -xzpf nikto-current.tar.gz

Será gerado um diretório com o nome   
nikto-x.y.z, onde "x" é a versão e "y.z" é o release mais atual do programa Nikto.   
Neste exemplo, "nikto-2.1.5" corresponde a versão mais atual.   
  
No diretório raiz, verifique que o arquivo nikto-2.1.5 está lá.

Você desempacotar com uma ferramenta gerenciadora de archive (tar).  
$sudo tar zxvf nikto-2.1.5.tar.gz ou

$sudo cd nikto-2.1.5

$sudo perl nikto.pl

Segue os resultados do funcionamento da instalação:

– \*\*\*\*\* SSL support not available (see docs for SSL install) \*\*\*\*\*  
- Nikto v2.1.5  
—————————————————————————  
+ ERROR: No host specified

-config+ Use this config file  
-Cgidirs+ scan these CGI dirs: ‘none’, ‘all’, or values like “/cgi/ /cgi-a/”  
-dbcheck check database and other key files for syntax errors  
-Display+ Turn on/off display outputs  
-evasion+ ids evasion technique  
-Format+ save file (-o) format  
-host+ target host  
-Help Extended help information  
-id+ Host authentication to use, format is id:pass or id:pass:realm  
-list-plugins List all available plugins  
-mutate+ Guess additional file names  
-mutate-options+ Provide extra information for mutations  
-output+ Write output to this file  
-nocache Disables the URI cache  
-nossl Disables using SSL  
-no404 Disables 404 checks  
-port+ Port to use (default 80)  
-Plugins+ List of plugins to run (default: ALL)  
-root+ Prepend root value to all requests, format is /directory  
-ssl Force ssl mode on port  
-Single Single request mode  
-timeout+ Timeout (default 2 seconds)  
-Tuning+ Scan tuning  
-update Update databases and plugins from CIRT.net  
-vhost+ Virtual host (for Host header)  
-Version Print plugin and database versions  
+ requires a value

Note: This is the short help output. Use -H for full help.

------------------------------------------------------------------------------------------------------------------------------  
 **OBJETIVOS DA AULA**

Conhecer um analisador de vulnerabilidades para Servidores Web, quanto a seus resultados básicos:

1. Examinando um site Web.
2. Mostrando os resultados entre outras informações.

**SELEÇÃO DE CONTEÚDO – A PRÁTICA**

Um conteúdo mínimo, consistindo de um roteiro de aula, com Nikto, visando o objetivo acima, pode ser colocada como:

**MOSTRE SEU EXPERIMENTO (Deposite a tarefa no Moodle).**

Abra terminal (Ctrl Alt+t)

$sudo su (passwd = 01031978)

#cd /nikto-2.1.5 (vai para o diretório criado para nikto na instalação)

nikto-2.1.5# ls –l (verifique que nikto.pl esta neste diretorio)

nikto-2.1.5# chmod 777 nikto.pl (dá permissão de execução)

nikto-2.1.5#./nikto.pl –update (atualiza a base de dados de plug-ins)

nikto-2.1.5#./nikto.pl –h <url-web-site> / <IP> -o /nikto-2.1.5/<arquivo-resultados>.txt (examina um web-site, fornecendo resultados em um arquivo .txt)

1. Informe o nome de sua tarefa. (Analisando o Site <nome-site>)
2. O escopo de sua tarefa (Vulnerabilidades em Servidor Web)
3. O seu alvo (indique o endereço URL do site web ou diretamente pelo seu IP)   
   Agora basta avaliar as opções de uso de Nikto e criar o relatório com as vulnerabilidades encontradas no seu sistema.
4. Indique o software analisado. (Versão)
5. Se existir, qual o SSL instalado no servidor Web analisado. Indique sua versão.
6. Exemplifique uma vulnerabilidade encontrada. Se você verificou algum web-site Apache, a versão atual deste é Apache 2.2.22. Normalmente, Nikto indica sites desatualizados.
7. Indique o que fazer para mitigar tal(is) vulnerabilidade(s).
8. Postar no Moodle, na data da aula de realização da tarefa.

**CONCLUSÃO**

Você não precisa de um OpenVAS ou um Nessus quando tem um Nikto !

**CONCLUSÃO CARACTERÍSTICAS DO NIKTO**

Aqui estão algumas das principais características do Nikto. Consulte a documentação em <http://cirt.net/nikto2/> para obter uma lista completa de recursos e como usá-los.

Suporte SSL (Unix com OpenSSL ou talvez do Windows com ActiveState de Perl / NetSSL).

Suporte a proxy HTTP completo.

Verifica os componentes desatualizados do servidor.

Salvar relatórios em texto puro, XML, HTML, CSV ou NBE.

Templates para personalizar facilmente relatórios.

Faz varredura em várias portas de um servidor, ou vários servidores através de arquivo de entrada (incluindo a saída nmap).

Técnicas de codificação IDS do *libwhisker*.

Facilmente atualizado via linha de comando.

Identifica software instalado via cabeçalhos, favicons (???) e arquivos.

Autenticação de host com o básico e NTLM.

Descoberta de subdomínio (???).

Enumeração de nome de usuário no Apache e *cgiwrap.*

Técnicas de mutação para "pescar" conteúdos em servidores Web.

Ajuste de varreduras para incluir ou excluir classes inteiras de verificações de vulnerabilidades.

Adivinha credenciais para reinos de autorização (incluindo muitos id/pw padrão combos).

Descoberta de autorização lida em qualquer diretório, e não apenas no diretório raiz.

Maior redução de falsos positivos através de vários métodos: cabeçalhos, conteúdo da página e hashing de conteúdo.

Relatórios de cabeçalhos (headers) "incomuns".

Estado interativo, pausa e alterações nas configurações de verbosidade.

Salvar pedido/resposta completo para testes positivos.

Repetição salva de pedidos positivos.

Máximo tempo de execução por alvo.

Auto-pausa em uma hora especificada.

Verifica locais comuns de "parking".

Log para Metasploit.

Documentação completa.