Segurança de Redes
Prática – Wireshark e nmap – Sniffer de rede e Scanner de portas

Entregar um relatório com as principais observações, análises e resultados obtidos durante a prática de laboratório nos experimentos do Wireshark (A – 1 a 3) e Nmap (B – 1 e 2).

A. Sniffer de redes – Wireshark

1. Capturando dados de uma sessão FTP

Vamos capturar uma sessão FTP (o FTP é uma aplicação que usa o protocolo de comunicação TCP para transferência de arquivos).

1. Ative a captura de pacotes (Menu *Capture→Start*). Escolha a interface de rede.
2. Pressione OK. Observe que agora existe uma janela de captura ativada.
3. Abra uma janela de terminal no Linux.
4. Conecte-se via FTP com algum site. Exemplo: **ftp ftp3.usa.openbsd.org**
   Atenção: Se o servidor FTP indicado no exemplo acima estiver fora do ar (não responde), tente um dos seguintes: **download.intel.com**, **ftp-linux.cc.gatech.edu**.
5. Quando for solicitado o login, digite **anonymous**
6. Na senha, digite: eu@provedor.com.br (o seu endereço de email, ou qualquer outro)
7. Observe que a senha não aparece na tela.
8. Agora você pode digitar comandos do FTP. Por exemplo, para ver os arquivos na máquina remota, digite do comando **ls**
9. Para sair digite **quit**
10. Pare a captura de pacotes (na janela de captura).
11. No Wireshark, filtre a visualização de pacotes usando o seguinte critério: apenas pacotes FTP do host IP, onde IP é o número IP do seu host. Ou seja, supondo que seu IP é 10.0.4.132, deve digitar o seguinte filtro: **ftp and ip.addr==10.0.4.132**

2. Visualizando a senha da sessão FTP

Quando digitamos a senha (ex.: eu@provedor.com.br) para a sessão FTP do exercício anterior, ela foi enviada em um pacote TCP para o servidor FTP. Como estávamos capturando pacotes durante a sessão, ela vai poder ser observada.

1. Faça um filtro para mostrar apenas os pacotes FTP do seu host (exemplo: **ftp and ip.addr==10.0.4.132**).
2. Clique duas vezes no primeiro pacote mostrado, para que ele fique marcado.
3. No menu, escolha a opção *Tools→Follow TCP Stream. Dependendo da versão do Wireshark você pode clicar com o botão direito do mouse → Follow → TCP Stream*
4. Aparece uma janela mostrando todos os dados (em modo ASCII) que foram trafegados entre seu computador (cliente) e o servidor durante a sessão FTP. Dados mostrados em **azul** foram enviados pelo seu host e em **vermelho** foram

recebidos. Não se preocupe em entender o formato dos dados, pois eles são entendidos pela aplicação FTP.

5. Procure dentro do texto mostrado as linhas contendo as palavras "**USER anonymous**" e "**PASS eu@provedor.com.br**".

   Suponha que você estivesse acessando o site FTP da sua empresa e tivesse sido autenticado (feito o login) com seu nome e senha reais. Você acredita que seria difícil alguém capturar sua senha usando um "sniffer" do tipo Wireshark na sua rede local?

3. Capturando outras senhas

Cada vez mais os provedores e sites estão usando criptografia para a transferência do nome do usuário e senha. No passado, provedores de e-mail não usavam criptografia. Tente achar algum site na Internet em que a senha trafegue em claro.

B. Scanner de portas - Nmap

Preparação: Este exercício usará a versão do nmap para o sistema operacional **Linux** (embora exista também nmap para Windows). Para realizar o exercício será necessária a utilização de 2 máquinas. Em uma máquina será usado o nmap e na outra o Wireshark (para Linux ou Windows). Verifique o endereço IP das máquinas A e B.

Visão geral do nmap

O nmap localiza e identifica todas as portas TCP e UDP disponíveis em um host. Também conhecido com um "port scanner", ele tenta determinar qual o serviço que está "escutando" cada porta e é capaz de identificar o tipo de sistema operacional rodando.

O nmap é visto como uma ferramenta de segurança, usada para descobrir "brechas" em sistemas, ajudando na tarefa de monitoração da rede e identificação de serviços rodando em servidores. Entretanto, o nmap é também uma das ferramentas preferidas de atacantes para fazer reconhecimento na sua rede.

Existem várias maneiras de obter ajuda com a utilização do nmap, listadas a seguir:

- Listagem de opções: digitando o comando **nmap** (sem parâmetros) aparece uma listagem das opções disponíveis
- Manual do Unix/Linux: digitando o comando **man nmap** aparece o manual do nmap em formato do Unix
- Página do nmap: na página do nmap (nmap.org) existem muitas informações úteis, inclusive uma página com documentação (http://nmap.org/docs.html) que possui inclusive um guia de referência em português (http://nmap.org/man/pt-br).

1. **Explorando o nmap**

Execute os seguintes de comandos nmap e verifique o resultado:

Observação: os endereços IP estão representados na faixa 10.0.1.x. Modifique esses endereços para representar a realidade da sua rede ou de outros hosts.

Comando	**Descrição**
nmap –sT 127.0.0.1	Mostra as portas TCP abertas no host 127.0.0.1 (sua própria máquina). Scan TCP é o default.
nmap –sT 10.0.1.123 10.0.1.114	Mostra as portas TCP abertas nos hosts indicados
nmap –sV 127.0.0.1	Mostra portas abertas e tenta determinar a informação sobre o serviço/versão
nmap –sA 127.0.0.1	Mostra o sistema operacional e informações sobre os serviços que estão rodando nas portas abertas
nmap –sU 10.0.1.123 10.0.1.114	Mostra as portas UDP abertas nos hosts indicados
nmap –p 80 10.0.1.109-144	Vasculha portas 80 (HTTP) nos hosts de 10.206.83.109 a 10.206.83.144
nmap –p 21-23,80 10.0.1.109	Vasculha portas 21, 22, 23 e 80 no host 10.206.83.109
nmap –p 161 –sU 10.0.1.109-144	Vasculha portas 161 (SNMP) nos hosts de 10.206.83.109 a 10.206.83.144. Opção–sU indica para fazer scan com UDP (SNMP usa UDP)
nmap –O 10.0.1.200	Procura identificar o sistema operacional do host indicado (a opção é a letra ‘O’ maiúscula e não o número ‘0’)

Faça um scan na rede para procurar hosts com servidor web ativado (porta 80), SSH (porta 22), TELNET (porta 23) e FTP (porta 21).

2. **Capturando pacotes**

   Na máquina **A** execute o Wireshark e capture apenas o tráfego direcionado a ela (utilize os filtros do wireshark). Na máquina **B** execute os comandos do nmap

listados abaixo e aguarde o fim da operação. Salve o resultado obtido na máquina **A** e na máquina **B**, individualmente para cada um dos comandos.

nmap -sT IP_HOST_A

nmap -sS IP_HOST_A

nmap -sF IP_HOST_A

nmap -sX IP_HOST_A

nmap -sN IP_HOST_A

nmap -sU IP_HOST_A

Após a coleta dos dados analise o tráfego capturado pelo Wireshark na máquina **A** e tente identificar o padrão utilizado pelo Nmap. Descreva os pacotes capturados (flags ativas, por exemplo) e faça uma comparação entre cada uma das opções utilizadas no Nmap.

Abaixo são apresentadas as descrições de cada opção utilizada nos testes acima pelo nmap:

-sT TCP connect() scan: *This is the most basic form of TCP scanning. The connect() system call provided by your operating system is used to open a connection to every interesting port on the machine. If the port is listening, connect() will succeed, otherwise the port isn't reachable.*

-sS TCP SYN scan: *This technique is often referred to as "half-open" scanning, because you don't open a full TCP connection. You send a SYN packet, as if you are going to open a real connection and you wait for a response. A SYN|ACK indicates the port is listening. A RST is indicative of a non-listener. If a SYN|ACK is received, a RST is immediately sent to tear down the connection (actually our OS kernel does this for us). The primary advantage to this scanning technique is that fewer sites will log it.*

-sN; -sF; -sX (TCP NULL, FIN, and Xmas scans): *These three scan types (even more are possible with the --scanflags option described in the next section) exploit a subtle loophole in the TCP RFC to differentiate between open and closed ports. Page 65 of RFC 793 says that "if the [destination] port state is CLOSED an incoming segment not containing a RST causes a RST to be sent in response." Then the next page discusses packets sent to open ports without the SYN, RST, or ACK bits set, stating that: "you are unlikely to get here, but if you do, drop the segment, and return."*

-sX (Xmas Tree): *faz uma "árvore de natal" ou seja, liga todos os flags (FIN, URG, and PSH)*

-sN: *faz um Null Scan – ou seja, desliga todos os flags. Isso serve para distinguir tipos de sistemas operacionais, pois cada um se comporta de maneira diferente quando recebem pacotes inesperados.*ego direcionado apenas para ela. Na máquina B execute o Nmap com o comando abaixo e aguarde o fim da operação:

-sU: *UDP scans: This method is used to determine which UDP (User Datagram Protocol, RFC 768) ports are open on a host. The technique is to send 0 byte udp packets to each port on the target machine. If we receive an ICMP port unreachable message, then the port is closed. Otherwise we assume it is open.*