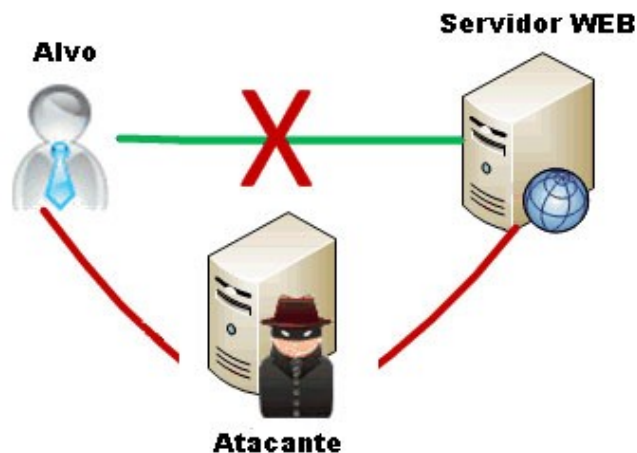


HARDENING

O *hardening* é uma técnica usada para mapear ameaças, mitigação(reduzir) os riscos . É um processo utilizado em diversos níveis de recursos em servidores para proporcionar segurança. O foco é a infraestrutura e seu objetivo é tornar um sistema mais seguro para enfrentar tentativas de ataque e de invasões. O *hardening* pode ser utilizada em qualquer sistema operacional, com isso medidas e ações tem o objetivo de fortalecer a segurança e proteger os sistemas contra futuras invasões. A seguir implementei a técnica de *hardening* em dois sistemas diferentes, um sistema *GNU/LINUX* e outro sistema *WINDOWS*.

O que é *MITM*?

Qualquer usuário conectado a uma rede está exposto a vários tipos de ataques. Entre um dos ataques, que pode-se considerar gravíssimo está o *Man-in-the-middle* . O objetivo do ataque é interceptar dados que trafegam na rede, sem que o alvo perceba. Quando disse que o ataque é considerado gravíssimo é que além do alvo não saber que está sendo atacado o atacante pode capturar dados sensíveis do alvo. Talvez essa seja uma forma simples, porém para perceber um pouco como é o ataque *MITM*. Pelo tamanho da gravidade que seja tal ataque, a técnica de defesa e ataque é simples de fazer. Com isso evitando roubo de dados, entre outras informações sensíveis do usuário. Durante o projeto vamos saber como ATACAR, como também aprenderemos a DEFENDER, com exemplos de ferramentas. Abaixo uma figura que demonstra o ataque.



Na figura mostra a comunicação sendo interceptada pelo atacante.

Evitando ataques *MITM* no *GNU/LINUX*.

Para evitar ataques *Man-in-the-middle* no *GNU/LINUX*, vou instalar uma ferramenta chamada *arpon*.

No meu caso, para fazer a instalação e configuração da ferramenta vou usar a distribuição *BackBox*, uma distribuição baseada em *Ubuntu*, com o foco em *pentest*.

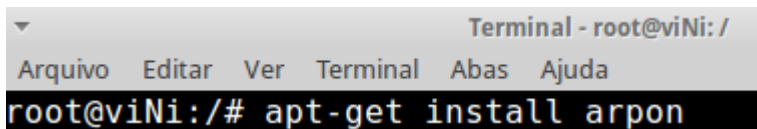
Antes de fazer todo o processo para proteger o sistema *GNU/LINUX*, vou dar uma explicação simples, e clara sobre tal ferramenta.

Arpon é uma ferramenta *open-source*, que faz o *ARP* seguro. Podendo então evitar ataques como, *Man-in-the-middle*, *DHCP spoofing*, *DNS spoofing*, *WEB spoofing*, sequestro de sessão *SSL*, etc. A ferramenta funciona monitorando a tabela *ARP*, ele vai gerar e bloquear alterações na tabela.

Depois de uma explicação básica, e clara sobre a ferramenta, agora vou começar a instalar e configurar.

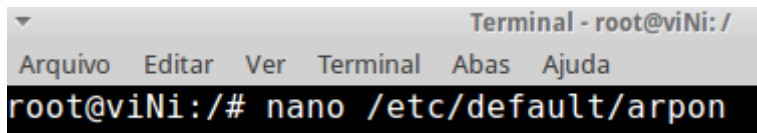
Let's go:D

Primeiramente vou fazer a instalação da ferramenta.

A terminal window titled "Terminal - root@viNi: /" with a menu bar containing "Arquivo", "Editar", "Ver", "Terminal", "Abas", and "Ajuda". The command "apt-get install arpon" is entered at the prompt "root@viNi:/#".

```
Terminal - root@viNi: /
Arquivo  Editar  Ver    Terminal  Abas  Ajuda
root@viNi:/# apt-get install arpon
```

O arquivo de configuração do *arpon* fica em:

A terminal window titled "Terminal - root@viNi: /" with a menu bar containing "Arquivo", "Editar", "Ver", "Terminal", "Abas", and "Ajuda". The command "nano /etc/default/arpon" is entered at the prompt "root@viNi:/#".

```
Terminal - root@viNi: /
Arquivo  Editar  Ver    Terminal  Abas  Ajuda
root@viNi:/# nano /etc/default/arpon
```

No *arpon* é possível fazer a configuração de duas formas. A primeira forma de fazer tal configuração é pelo *SARPI*. Nesse método é necessário que você insira cada endereço *IP* da rede, juntamente o endereço *MAC*. Esse processo é estático.

A segunda opção é pelo *DARPI*. Nesse método o *arpon*, faz a associação de cada endereço *IP* da rede, juntamente com o endereço *MAC*. Esse processo é dinâmico.

Independente do modo de configuração, se houver qualquer alteração na tabela *ARP*, o *arpon* vai exibir essa alteração como alerta no arquivo de *log*, ele não fará alteração na tabela *ARP*, ele apenas vai bloquear o ataque *MITM*.

Vou fazer a configuração usando o *SARPI* (configuração estática),

Entendendo a configuração:

```
Terminal - root@vini: /
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
GNU nano 2.2.6  Arquivo: /etc/default/arpon

# Defaults for arpon initscript
# sourced by /etc/init.d/arpon
# installed at /etc/default/arpon by the maintainer scripts

# You must choose between static ARP inspection (SARPI) and
# dynamic ARP inspection (DARPI)
#
# For SARPI uncomment the following line (please edit also /e$
DAEMON_OPTS="-q -f /var/log/arpon/arpon.log -g -s"

# For DARPI uncomment the following line
# DAEMON_OPTS="-q -f /var/log/arpon/arpon.log -g -d"

# Modify to RUN="yes" when you are ready
RUN="yes"
```

Como vou fazer uma configuração estática(*SARPI*), descomentei a linha 10.

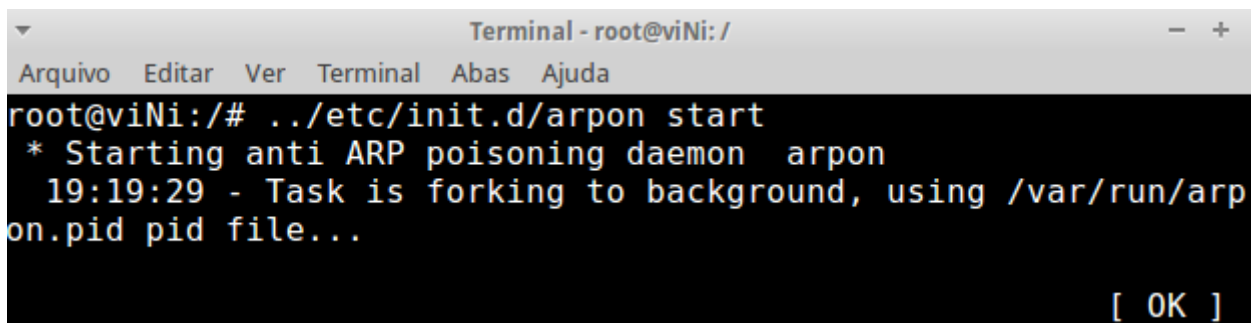
Na linha 16, mudei de *no* para *yes*.

No arquivo */etc/arpon.sarpi*, vamos definir estaticamente os endereços *IP*'s juntamente com o endereço *MAC*. Apenas de teste vou inserir o *IP/MAC* de uma máquina e o endereço *IP/MAC* do roteador. Porém o correto é inserir o *IP/MAC* de todas as máquinas que estão conectadas na rede toda.

```
Terminal - vini@vini: /etc
Arquivo  Editar  Ver  Terminal  Abas  Ajuda
GNU nano 2.2.6  Arquivo: arpon.sarpi

# Example of arpon.sarpi
#
#      # Gw
#      #192.168.1.1      00:25:53:29:f6:69
#
#      # Spyro virtual
#      #172.16.159.1     0:50:56:c0:0:8
#
# Gateway
192.168.6.1      F4:EC:38:F3:FD:C0
# Debian 8.2
192.168.6.110   08:00:27:92:3a:e9
```

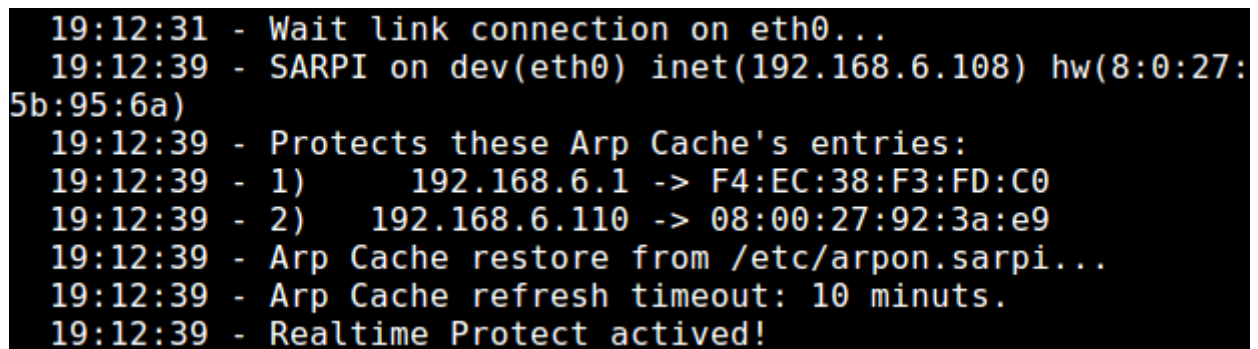
Feito as configurações, agora vamos iniciar o serviço do *arpon*.

A terminal window titled "Terminal - root@viNi: /" with a menu bar containing "Arquivo", "Editar", "Ver", "Terminal", "Abas", and "Ajuda". The terminal shows the command `root@viNi:/# ../etc/init.d/arpon start` being executed. The output is: `* Starting anti ARP poisoning daemon arpon`, `19:19:29 - Task is forking to background, using /var/run/arpon.pid pid file...`, and a prompt `[OK]` at the bottom right.

```
Terminal - root@viNi: /
Arquivo  Editar  Ver   Terminal  Abas  Ajuda
root@viNi:/# ../etc/init.d/arpon start
* Starting anti ARP poisoning daemon arpon
19:19:29 - Task is forking to background, using /var/run/arpon.pid pid file...
[ OK ]
```

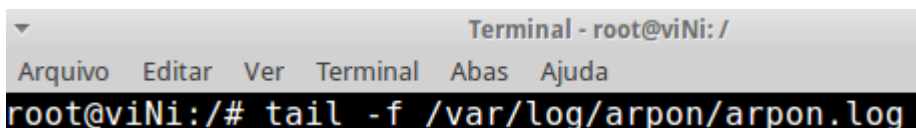
Tudo "ok":D

Como mostrado na imagem abaixo mostra, a proteção de tempo real está ativo. (*Realtime Protect activated*).

A terminal window showing the output of the arpon service. It displays the time 19:12:31 for waiting a link connection on eth0. At 19:12:39, it shows SARPI configuration on dev(eth0) with IP 192.168.6.108 and MAC 8:0:27:5b:95:6a. It then lists the ARP cache entries being protected: 1) 192.168.6.1 -> F4:EC:38:F3:FD:C0 and 2) 192.168.6.110 -> 08:00:27:92:3a:e9. It also shows the ARP cache restore from /etc/arpon.sarpi... and a refresh timeout of 10 minutes. Finally, it confirms "Realtime Protect activated!" at 19:12:39.

```
19:12:31 - Wait link connection on eth0...
19:12:39 - SARPI on dev(eth0) inet(192.168.6.108) hw(8:0:27:5b:95:6a)
19:12:39 - Protects these Arp Cache's entries:
19:12:39 - 1)      192.168.6.1 -> F4:EC:38:F3:FD:C0
19:12:39 - 2)      192.168.6.110 -> 08:00:27:92:3a:e9
19:12:39 - Arp Cache restore from /etc/arpon.sarpi...
19:12:39 - Arp Cache refresh timeout: 10 minuts.
19:12:39 - Realtime Protect activated!
```

Caso há alguma tentativa de mudança na tabela *ARP* seja realizado(como um ataque MITM por exemplo). Conferir os *log's* em tempo real:

A terminal window titled "Terminal - root@viNi: /" with a menu bar containing "Arquivo", "Editar", "Ver", "Terminal", "Abas", and "Ajuda". The terminal shows the command `root@viNi:/# tail -f /var/log/arpon/arpon.log` being entered.

```
Terminal - root@viNi: /
Arquivo  Editar  Ver   Terminal  Abas  Ajuda
root@viNi:/# tail -f /var/log/arpon/arpon.log
```

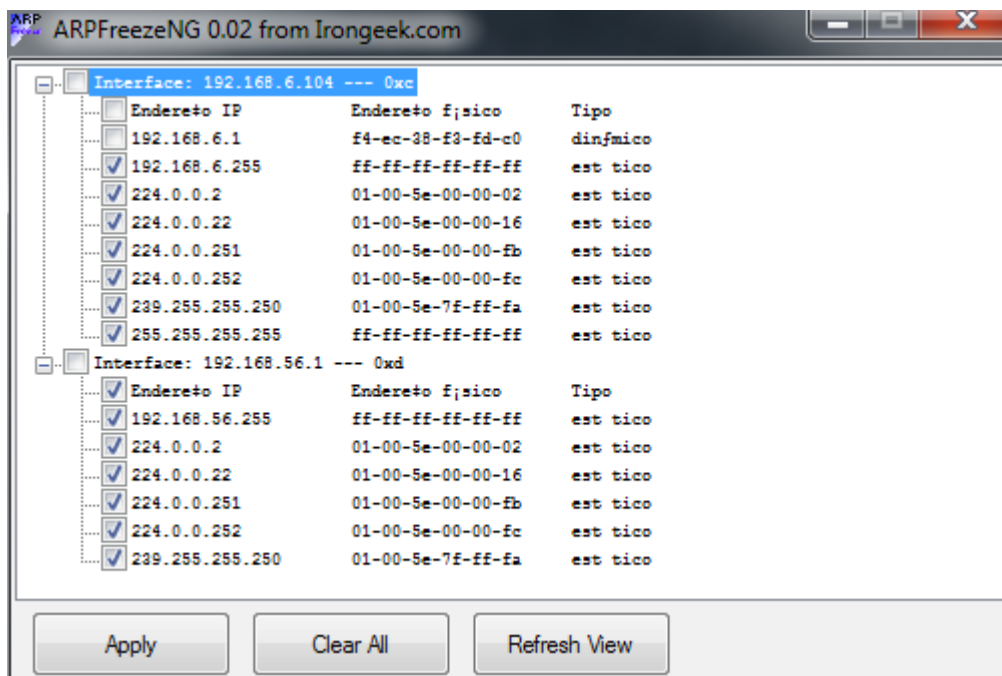
Evitando ataques *MITM* no *WINDOWS*.

Para proteger contra ataques *MITM* no *Windows*, vou usar uma ferramenta chamada *ARPFreezeNG*, esta versão está disponível para *Windows 7*, *Windows Vista*. Para a versão do *Windows XP*, tem o *ARPFreeze*.

A ferramenta é simples de usar, basta fazer o *download*, descompactar a mesma, executar, ela não necessita de instalação no sistema.

O objetivo da ferramenta é proteger o sistema contra ataques de envenenamento *ARP*.

ARPFreeze é uma ferramenta para prevenção. *ARPFreeze / ARPFreezeNG* permite configuração estática tabelas *ARP* para que outros atacantes não façam um ataque de envenenamento de *ARP* contra o servidor. *Windows* tem ferramentas internas para fazer isso (o comando *arp* e *netsh*), mas estes não são fáceis ou automatizados.



Obs: Todos os testes, tanto o ataque, quanto a defesa, foram feitos em laboratórios isolados. Não houve roubo de nenhuma informação.

printf(" Só sabe se defender, quem sabe atacar :D . ")

Referências

Livro introdução ao Pentest – Daniel Moreno

www.vivaolinux.com.br