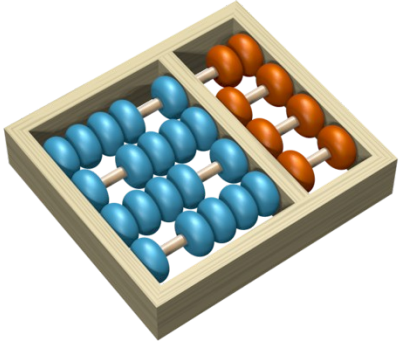
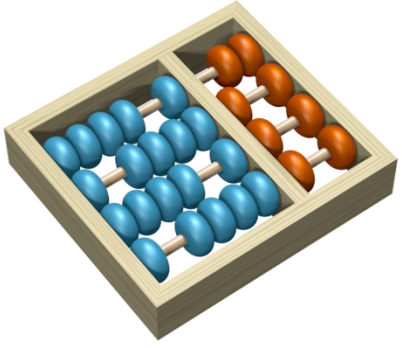
Aluno: Fernando Henrique Santorsula
E-mail: f208918@g.unicamp.br

Professora: Islene Calciolari Garcia
Disciplina: MO806



Apresentação FINAL do projeto:

SACK Panic



NETFLIX

Resumo das apresentações:

1º FASE – Apresentação do tema do projeto: SACK Panic

2º FASE – Apresentação intermediária: Abordagem real sobre o SACK Panic

3º FASE – Apresentação final: Simulação prática do bug SACK Panic com DDoS

Objetivo

Demonstrar de forma prática e dinâmica, a falha de segurança do kernel, quando é utilizado um ataque de negação de serviço, mais conhecido como: DDoS (Distributed Denial of Service), que deverá inundar uma determinada interface de rede sem perder a conexão, derrubando um determinado site que será hospedado localmente ou na Internet, nesta apresentação a simulação será “local” de forma “virtualizada”.

Simulação prática do SACK Panic

Para efetuar o procedimento será preciso um conjunto de softwares para demonstrar a apresentação de forma prática, os softwares são:

Simulação prática do SACK Panic

- Sistemas Operacionais: (Debian 8 e Ubuntu 18.04 LTS)
- htop: (Visualizador de processos e status de máquina)
- ping: (Aplicativo de teste de conectividade ICMP)
- hping: (Gerador de pacotes em massa “flood”)
- speedometer: (Analisador de tráfego)
- Virtualizador: (VirtualBox – Versão 6)
- Wireshark: (Analisador de pacotes)

Obs: Todos os softwares são Open Source.

Simulação do SACK Panic

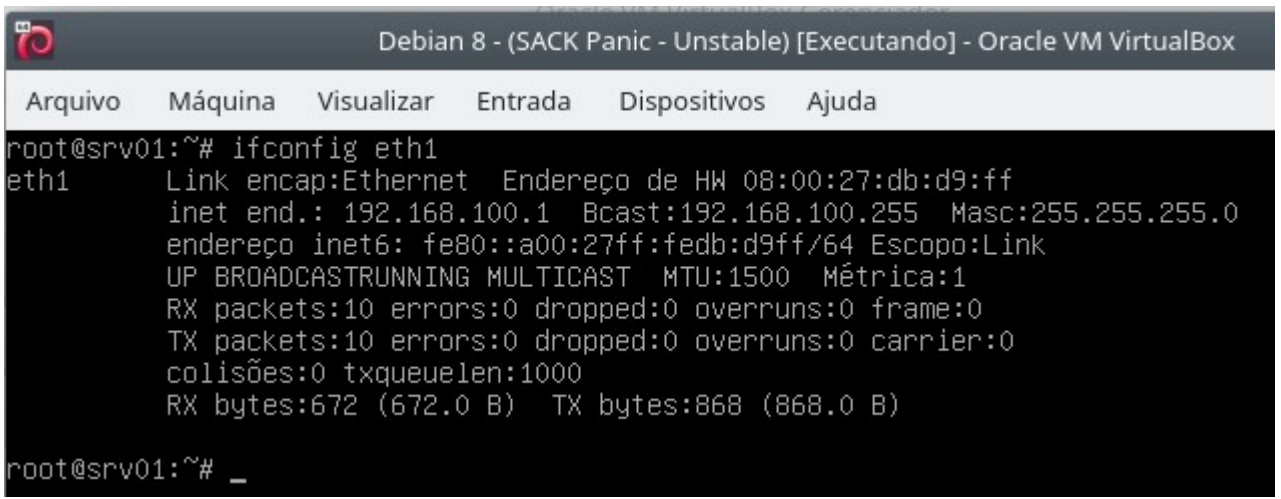
1º ETAPA – Verificar endereçamento e comunicação, entre as máquinas, cliente e servidor, sendo que a máquina cliente será a máquina do “atacante”, os comandos executados são:

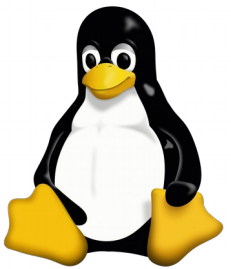
ifconfig / ping

Obs. O Ubuntu 18.04, não possui suporte ao software “ifconfig”, é necessário instalar o pacote net-tools: **sudo apt update && sudo apt install net-tools -y**

NETFLIX

Máquina servidor (Debian 8)

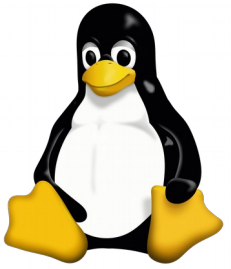




Comando: ifconfig eth1

Máquina cliente (Ubuntu 18.04 LTS)

```
root@vm01: /home/santorsula
Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
root@vm01:/home/santorsula# ifconfig enp0s8
enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.100.2  netmask 255.255.255.0  broadcast 192.168.100.255
        inet6 fe80::a00:27ff:fe1c:1daf  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:1c:1d:af  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 84  bytes 8694 (8.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```



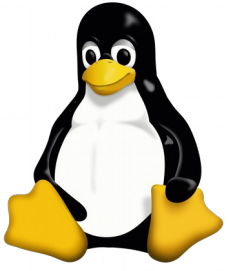
Comando: ifconfig enp0s8

NETFLIX

Máquina servidor (Debian 8)

```
root@srv01:~# ping -c 4 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data.
64 bytes from 192.168.100.2: icmp_seq=1 ttl=64 time=0.949 ms
64 bytes from 192.168.100.2: icmp_seq=2 ttl=64 time=0.715 ms
64 bytes from 192.168.100.2: icmp_seq=3 ttl=64 time=0.586 ms
64 bytes from 192.168.100.2: icmp_seq=4 ttl=64 time=0.718 ms

--- 192.168.100.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 2999ms
rtt min/avg/max/mdev = 0.586/0.742/0.949/0.130 ms
```



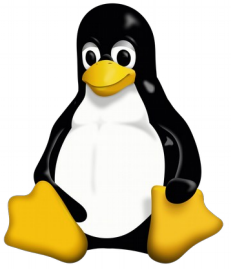
Comando: ping -c 4 192.168.100.2

NETFLIX

Máquina cliente (Ubuntu 18.04 LTS)

```
                                                root@vm01: /home/santorsula
 Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
root@vm01:/home/santorsula# ping -c 4 192.168.100.1
PING 192.168.100.1 (192.168.100.1) 56(84) bytes of data.
64 bytes from 192.168.100.1: icmp_seq=1 ttl=64 time=0.354 ms
64 bytes from 192.168.100.1: icmp_seq=2 ttl=64 time=0.734 ms
64 bytes from 192.168.100.1: icmp_seq=3 ttl=64 time=0.736 ms
64 bytes from 192.168.100.1: icmp_seq=4 ttl=64 time=0.704 ms

--- 192.168.100.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3058ms
rtt min/avg/max/mdev = 0.354/0.632/0.736/0.161 ms
```

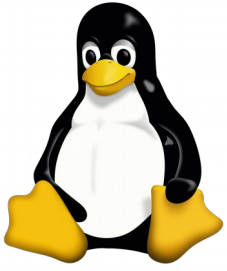


Comando: ping -c 4 192.168.100.1

Simulação do SACK Panic

1º ETAPA – Verificando tráfego da interface eth1 da máquina Servidor, utilizando o software “speedometer”, comando executado:

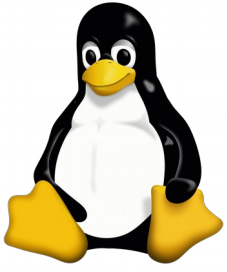
sudo speedometer -t eth1 -r eth1



Simulação do SACK Panic

1º ETAPA – Acesso à um site dinâmico "interno", rodando os serviços de Webserver e um SGBD: (Apache & MySQL), URL de acesso:

http://192.168.100.1/site



Website dinâmico, utilizando o CMS (Wordpress), utilizando a tecnologia LAMP:
Linux, **A**pache, **M**ySQL, **P**HP

Simulação do SACK Panic

1º ETAPA – Efetuar a 1º análise de pacotes com Wireshark, em seguida selecionar a interface **enp0s8** para captura de pacotes.

Simulação do SACK Panic

2º ETAPA – Efetuando o ataque “DDoS” através da máquina cliente, verificando a transparência do ataque, através do protocolo ICMP, utilizando o software “ping”, comandos executados:

ping 192.168.100.1
hping3 --syn --flood -p 80 192.168.100.1

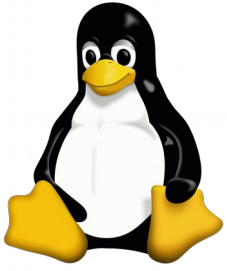
Descrição de uso do software hping

hping3	Software de geração de pacotes “personalizados”
--syn	Sincronização dos pacotes que serão fragmentados
--flood	Faz o processo de inundação de pacotes
-p	Informa que uma determinada porta a ser utilizada no ataque
80	Porta de destino do ataque DDoS
192.168.100.1	Endereço de rede que será utilizado como vitima do ataque

Simulação do SACK Panic

3º ETAPA – Verificando tráfego da interface eth1 da máquina Servidor, utilizando o software “speedometer”, comando executado:

sudo speedometer -t eth1 -r eth1



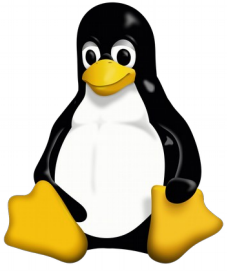
Simulação do SACK Panic

3º ETAPA – Efetuar a 2º análise de pacotes com Wireshark, em seguida selecionar a interface **enp0s8** para captura de pacotes.

Simulação do SACK Panic

3º ETAPA – Acesso à um site dinâmico "interno", rodando os serviços de Webserver e um SGBD: (Apache & MySQL), URL de acesso:

http://192.168.100.1/site



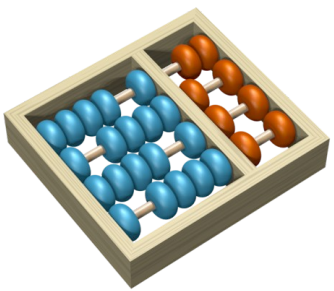
NETFLIX

Simulação do SACK Panic

3º ETAPA – Abordagem final sobre os testes realizados de forma prática (htop e speedometer), por fim parar o testes e verificar status: (rede e cpu), acessar o site novamente.

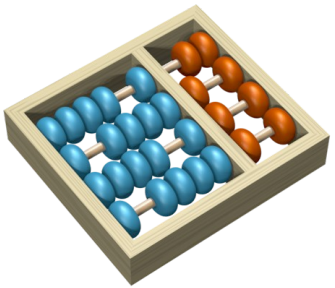
Observação:

A máquina poderá travar, devido a inundação de pacotes na interface enp0s8, podendo utilizar 100% de CPU.



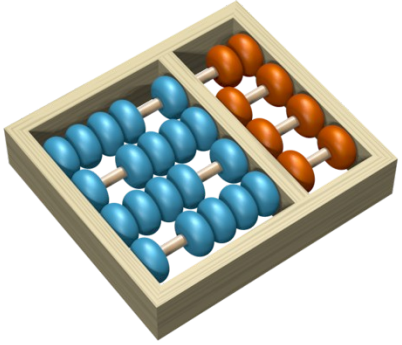
Conclusão

Utilizando um ataque da família DDos, a interface de rede do servidor é inundada com centenas ou milhares de pacotes, ocorrendo a “negação de serviço”, tornando o servidor e seus serviços indisponível, devido a falha de kernel denominada de SACK Panic.



NETFLIX

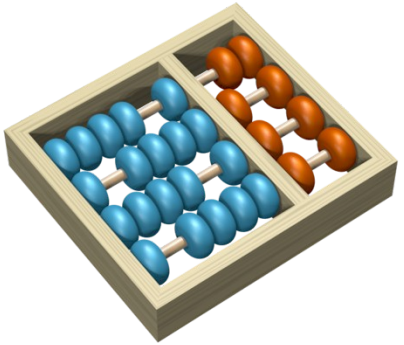
Dúvidas ?



References:

https://unaaldia.hispasec.com/2019/06/sack-panic-dos-a-traves-de-tcp-ip-en-el-kernel-linux.html
https://www.openwall.com/lists/oss-security/2019/06/17/5
https://github.com/Netflix/security-bulletins/blob/master/advisories/third-party/2019-001.md
https://kb.pulsesecure.net/articles/Pulse_Security_Advisories/SA44193
https://kc.mcafee.com/corporate/index?page=content&id=SB10287
https://psirt.global.sonicwall.com/vuln-detail/SNWLID-2019-0007
https://seclists.org/bugtraq/2019/Jul/30
https://security.netapp.com/advisory/ntap-20190625-0001/
https://support.f5.com/csp/article/K26618426
https://wiki.ubuntu.com/SecurityTeam/KnowledgeBase/SACKPanic
https://www.kb.cert.org/vuls/id/905115
https://www.synology.com/security/advisory/Synology_SA_19_28
https://www.sentinelone.com/blog/grab-our-free-tool-linux-sack-panic/
https://www.ukfast.co.uk/blog/2019/07/02/tcp-sack-panic-linux-kernel-vulnerability/
https://github.com/Sentinel-One/sack-cve-fixer
https://allelesecurity.com.br/asa-2019-00366/

http://packetstormsecurity.com/files/153346/Kernel-Live-Patch-Security-Notice-LSN-0052-1.html
http://www.openwall.com/lists/oss-security/2019/06/28/2
http://www.openwall.com/lists/oss-security/2019/07/06/3
http://www.openwall.com/lists/oss-security/2019/07/06/4
http://www.vmware.com/security/advisories/VMSA-2019-0010.html
https://isc.sans.edu/forums/diary/What+You+Need+To+Know+About+TCP+SACK+Panic/25046/
https://access.redhat.com/security/vulnerabilities/tcpsack
https://access.redhat.com/articles/4329821
https://unit42.paloaltonetworks.com/tcp-sack-panics-linux-servers/
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-11477
http://www.slackware.com/security/viewer.php?l=slackware-security&y=2019&m=slackware-security.1029019
https://git.kernel.org/pub/scm/linux/kernel/git/davem/net.git/commit/?id=f070ef2ac66716357066b683fb0baf55f8191a2e
https://linux.die.net/man/8/hping3
https://www.virtualbox.org/wiki/Documentation



Obrigado!

Aluno: Fernando Henrique Santorsula
E-mail: f208918@g.unicamp.br
Disciplina: MO806