Ingenius. Revista de Ciencia y Tecnología
ISSN: 1390-650X
revistaingenius@ups.edu.ec
Universidad Politécnica Salesiana
Ecuador

Zapata Molina, Lina Patricia
Evaluación y mitigación de ataques reales a redes IP utilizando tecnologías de virtualización de libre distribución
Ingenius. Revista de Ciencia y Tecnología, núm. 8, julio-diciembre, 2012, pp. 11-19
Universidad Politécnica Salesiana
Cuenca, Ecuador

UNIVERSIDAD POLITÉCNICA SALESIANA · ECUADOR
INGENIUS
Revista de Ciencia y Tecnología

Evaluación y mitigación de ataques reales a redes IP utilizando tecnologías de virtualización de libre distribución

Lina Patricia Zapata Molina[1,*]

Resumen

Los ataques a redes IP pueden colapsar la continuidad de los servicios de las empresas afectando su imagen y causando graves pérdidas económicas. La presente investigación se centra en la evaluación de diversos ataques reales de redes IP utilizando plataformas de virtualización con el fin de establecer mecanismos de seguridad para mitigarlos. Para llevarlo a cabo, se diseñaron e implementaron varias topologías de experimentación usando entornos virtuales de red, dentro de las cuales se probaron el escaneo de puertos, fuerza bruta, suplantación de identidad y denegación de servicios, tanto en una red de área local como en una extendida. Para cada topología, se utilizó diferente software libre tanto para producir el ataque como para obtener el flujo de tráfico, evaluándose las consecuencias del ataque. Para contrarrestar dichos ataques, se desarrolló un demonio en Shell script que es capaz de detectar, controlar y mitigar los ataques mencionados de manera programable y constante. Los resultados muestran la funcionalidad de esta investigación que reduce las amenazas y vulnerabilidades de las redes en producción.

Palabras clave: ataques de seguridad, tecnologías de virtualización.

Abstract

IP networks attacks can collapse the continuity of business services affecting its image and causing economic losses. This research focuses on the evaluation of several IP networking real attacks using virtualization platforms to provide security mechanisms to mitigate them. To carry out this work, we designed and implemented several experimentation topologies using virtual network environments, within which were tested port scans, brute force, spoofing and denial of services, both on a local area network as wide area network. For each topology, different free open source software was used both to produce the attack and to obtain the traffic flow, evaluating the consequences of these attacks. To deal with such attacks, we developed a demon program that is able to prevent, detect and mitigate these attacks mentioned. The results show the functionality of this research that reduces threats and vulnerabilities in production networks.

Keywords: security attacks, virtualization technology.

[1,*] *Máster en Redes de Información y Conectividad, Ingeniera en Sistemas, Analista de Sistemas. Docente de la carrera de Ingeniería de Sistemas, Universidad Politécnica Salesiana, sede Quito. Autor para correspondencia ✉: lzapata@ups.edu.ec*

Recibido: 07 - Noviembre - 2012, *Aprobado tras revisión: 15 - Noviembre - 2012*

Forma sugerida de citación: Zapata Molina, L. (2012). "Evaluación y mitigación de ataques reales a redes IP utilizando tecnologías de virtualización de libre distribución". *INGENIUS*. N.°8, (Julio/Diciembre). pp 11-19. ISSN: 1390-650X

1. Introducción

Las redes teleinformáticas están expuestas a ataques e intrusiones que pueden dejar inoperativos los recursos y causar pérdidas de la imagen, productividad, credibilidad y competitividad, provocando perjuicios económicos que podrían comprometer la continuidad del negocio [1]. Esta incertidumbre sigue agravándose, pues continúan apareciendo diversas amenazas, vulnerabilidades y tipos de ataques que implican: hurto, modificación, espionaje, interrupción, falsificación, denegación de servicios, etc., perjudicando directamente a los negocios que son altamente dependientes de sus sistemas y redes de información [2].

Para prevenir y contrarrestar una amplia gama de amenazas a las redes, es necesario conocer sus vulnerabilidades e identificar diversos tipos de ataques. Para manejar esta situación se propone crear un ambiente de red controlado con los componentes necesarios que detecten ataques maliciosos, para analizarlos y contrarrestarlos. Una primera alternativa sería mediante equipos reales, sin embargo, esto encarecería la solución y pondría en riesgo la red en producción. Otra alternativa sería utilizar máquinas virtuales, con las cuales es posible reducir costos de inversión de hardware, costos de mantenimiento, costo y tiempo de experimentación y sobre todo reduciría el riesgo del colapso de la red en producción [3].

En este contexto, la comunidad científica ha mostrado un creciente interés por investigar e implementar soluciones para disminuir los ataques a redes aprovechando las tecnologías de virtualización. De acuerdo con la guía de Seguridad para Tecnologías de Virtualización, del Instituto Nacional de Estándares y Tecnología (NIST) la virtualización podría reducir el impacto de esta explotación [4]. Bajo este precepto, el trabajo propuesto en [5], formula la implementación de un laboratorio colaborativo de seguridad utilizando máquinas virtuales.

En [6], se propone la integración de las tecnologías de virtualización para la instrucción de seguridad en redes implementando un laboratorio remoto de detección de intrusiones. Otros investigadores [7], [8], han utilizado el concepto de Honeynet basada en máquinas virtuales, como una herramienta de seguridad cuyo propósito es el estudio de las técnicas y motivaciones de los atacantes al romper los sistemas de seguridad. En este mismo ámbito [9], [10], [11], han utilizado las plataformas de virtualización para recuperación de desastres y mitigación de ataques reales a redes IP.

El objetivo del presente trabajo fue diseñar e implementar una plataforma de experimentación para evaluar ataques reales de redes IP utilizando plataformas de virtualización de libre distribución e implementar mecanismos de control y mitigación para contrarrestarlos. Para llevarlo a cabo, se diseñaron y se pusieron en funcionamiento los diferentes escenarios de experimentación utilizado VMware Player y VirtualBox. Luego se aplicaron diversos tipos de ataques a cada escenario creado. Posteriormente, se evaluó el impacto que provocan los diversos ataques analizando la información de las trazas. Finalmente se proponen mecanismos de mitigación de cada uno de estos ataques. Todo esto utilizando diversas herramientas de código abierto y de libre distribución.

2. Estado del arte

La investigación planteada en este proyecto cobra importancia debido a que permitirá analizar un problema actual y real, existente en las redes de información, mediante la utilización de la tecnología de virtualización, que es una tecnología disruptiva para la presente década.

Este proyecto se desarrollará con plataformas de virtualización de libre distribución, a fin de emular diversos ataques a redes IP.

Los principales beneficios fueron:

1. Desde el punto de vista investigativo, el diseño y configuración de plataformas virtuales para experimentación, con las mismas funcionalidades de una red real, con el consiguiente ahorro en gastos en equipos y dispositivos de red.

2. El análisis y evaluación de las vulnerabilidades en la red permitió identificar algunos tipos de ataques a la red y se identificó medidas de seguridad preventivas para contrarrestarlos.

3. Tecnológicamente el presente estudio de ambientes virtuales servirá como base para otras emulaciones o pruebas de temas relacionados con redes. Además, se busca contribuir, de esta manera, para investigaciones futuras.

4. Con los resultados de esta investigación se pretende contribuir al conocimiento.

3. Configuración e implementación de la plataforma de experimentación

3.1 Escenario virtual de una red IP con tecnología de virtualización

La virtualización puede verse como un particionado de un servidor físico de manera que pueda albergar

distintos servidores dedicados (o privados) virtuales que ejecutan de manera independiente su propio sistema operativo y dentro de él los servicios que quieran ofrecer [12].

Un escenario virtual de red puede ser definido como un conjunto de equipos virtuales (tanto sistemas finales como elementos de red (enrutadores y conmutadores) conectados entre sí en una determina topología, cuyo entorno deberá ser percibido como si fuera real [13].

Para implementar los escenarios virtuales en esta investigación, se ha elegido VMware Player [14] 3.0 y VirtualBox [15], herramientas de libre distribución basadas en tecnología de virtualización completa que permiten la creación de máquinas virtuales X86 de 32 y 64 bits, y que son muy utilizadas en la industria [13]. La Figura 1, busca expresar un escenario de red LAN/WAN sometida a ataques IP desde la Intranet o del Internet.

Dos plataformas virtuales son creadas, en dos servidores diferentes, funcionan sobre un computador Pentium Intel CoreDuo, con RAM de 4 GB, partición Ext3 de 120 GB en disco duro y sistema operativo GNU/LinuxUbuntu 9.04 -i386. Estos servidores constituyen máquinas anfitrionas, donde cada una alberga a seis máquinas virtuales (MV) que cumplen funciones diferentes y específicas (Figura 1).

3.2 Ataques reales de redes IP sobre un escenario virtual de red IP

Los ataques elegidos para la realización del presente trabajo, por ser los más comunes, fueron rastreo de sistemas, ataque a contraseñas, suplantación de identidad y denegación de servicios.

3.2.1 Rastreo de sistemas (escaneo de puertos)

Este ataque consiste en el envío de una serie de señales (paquetes) hacia una máquina víctima que responde reenviando paquetes, que el atacante decodifica y traduce a fin de conseguir información sobre: direcciones IP activas, puertos TCP y UDP activos y; reconocimiento del tipo de sistema operativo del equipo como elemento de una red, entre las más importantes. La herramienta utilizada para realizar el escaneo de puertos es Nmap (Network Mapper) [16].

3.2.2 Ataques a contraseñas

El objetivo de este ataque es ingresar al sistema de la víctima, a través de la red, con credenciales (nombre de usuario y contraseña) y haciendo uso de una conexión remota (ssh, telnet, etc). Para ello genera el diccionario (hash) de todas las posibles combinaciones y las compara con el patrón (hash) que permita el acceso [17], [18].

Las formas de operación de las aplicaciones:

- Ataque con diccionario: Se apoya en un fichero que contiene un gran número de palabras que son comparadas con la contraseña encriptada dentro del fichero shadow.
- Ataque por fuerza bruta: Este método se basa en la formación de palabras mediante combinación de caracteres hasta encontrar una que coincida con la contraseña protectora.

Para la generación de ataques a contraseña se emplearon dos herramientas: John The Ripper [19] y Medusa que opera sobre el modo con diccionario.

3.2.3 Ataque de suplantación de identidad (Spoofing)

Un ataque *Spoofing* consiste en aplicar técnicas de suplantación de identidad generalmente con usos maliciosos o de investigación [20]. El objetivo de este ataque es alcanzar la confianza de su víctima haciéndose pasar por otra máquina.

Para el IP Spoofing se utilizó la herramienta Hping a fin de lograr suplantar la IP de la máquina atacante por otra distinta.

En el caso del ARP Spoofing se hizo uso de la herramienta Némesis [21], la misma que permite modificar las direcciones MAC de los equipos de una red. Este tipo de ataque genera también ataques de denegación de servicio (DoS).

3.2.4 Ataque de denegación de servicios

Son ataques que provocan que un servicio, equipo o recurso sea inaccesible para usuarios legítimos. Para esto se envía mensajes TCP de petición de conexión por parte del cliente, pero sin enviar su confirmación lo cual provoca colapsos en equipos y consumo de recursos en forma desproporcionada, muchas veces la dirección de origen es falsificada [21].

Para la generación de este ataque se hizo uso del programa Nemesis y Hpingen el equipo atacante con S.O. Windows y Linux respectivamente.

3.3 Mecanismos para contrarestar los ataques definidos

3.3.1 BashScript como contramedida ante un ataque de fuerza bruta

El sistema de logs (registro) de GNU/Linux es un mecanismo estándar que se encarga de recoger los

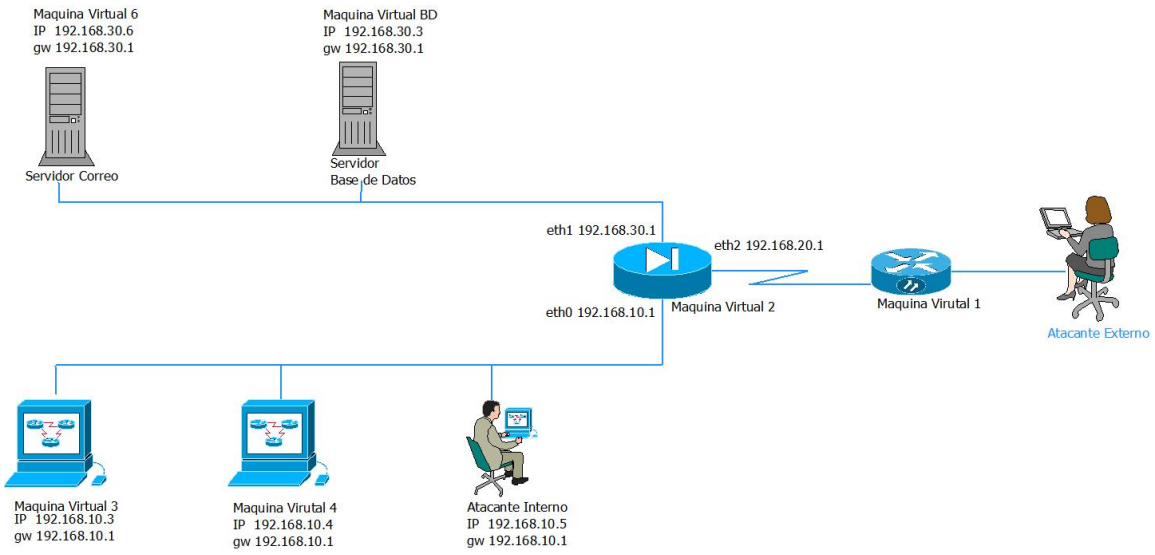

Figura 1. Diseño de la topología de pruebas

mensajes generados por los programas, aplicaciones y demonios [22].

El Bash Script propuesto como contramedida ante intentos de conexión no solicitados (ataque y fuerza bruta) monitoriza el fichero `auth.log` cada segundo, y filtra los intentos fallidos, que superado cierto límite, envía la IP atacante al fichero `/etc/hosts.deny`, a fin de denegar la conexión hacia y desde el `host` víctima.

Adicionalmente, este script genera un fichero con los IP que se van registrando en `hosts.deny` y envía un e-mail de notificación al administrador de la red sobre la IP que acaba de ser negada la conexión junto a la IP de su víctima.

En la Figura 2, se describe el diagrama de secuencia sobre la lógica de funcionamiento del script `bloqueo.sh`, creado e implementado como contramedida ante intentos de conexión no solicitados.

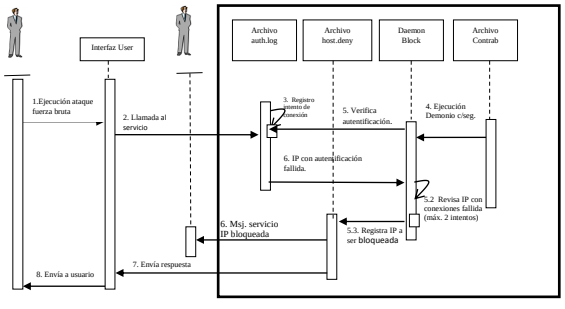

Figura 2. Diagrama de secuencias del script de mitigación a un ataque de fuerza bruta.

Cabe mencionar que para la ejecución de script `bloque.sh` cada segundo, se hizo uso del cron. El cron de Ubuntu revisa cada minuto la tabla de tareas crontab (`/etc/crontab`) en búsqueda de tareas que se deban cumplir. Crontab es un simple archivo de texto que guarda una lista de comandos a ejecutar en un tiempo especificado por el usuario [23].

3.3.2 Script que modifica la configuración del firewall en Ubuntu

La configuración del *firewall* consiste en filtrar el tráfico TCP/UDP/ICMP/IP y decidir qué paquete pasa, se modifica, se convierte o se descarta; todo esto se logra haciendo uso de iptables, que son cadenas formadas por agrupación de reglas encargadas de decir qué destino tiene un paquete.

La lógica de funcionamiento optada para el cortafuego se describe brevemente a continuación:

- Filtrar el acceso al propio *firewall* permitiendo las conexiones que se crea oportunas.
- Filtrar en la cadena *forward* aquellas conexiones permitidas desde la redLan.
- Enmascarar la red local y habilitar el *forwarding*.
- Implementar el mecanismo que le cambie por una dirección válida. Esto se lo hace con el comando `masquerade`.

4. Resultados experimentales

4.1 Rastreo de sistemas (escaneo de puertos)

En la Figura 3, tenemos el tiempo que se demora en hacer un escaneo de puertos el equipo atacante a

un equipo víctima, la moda (valor que más se repite) es de 1,75 s, adicionalmente tenemos que el tiempo promedio en realizar un escaneo de puertos es de $1,67$ s. Al calcular la desviación estándar $0,546$ s. y la desviación típica de la varianza con la Ecuación 1, tenemos que la diferencia es pequeña lo que demuestra que las medidas tomadas son certeras.

$$\partial = \sqrt{\frac{\sum_i^n (X_i - \hat{x})^2}{N-1}} \quad (1)$$

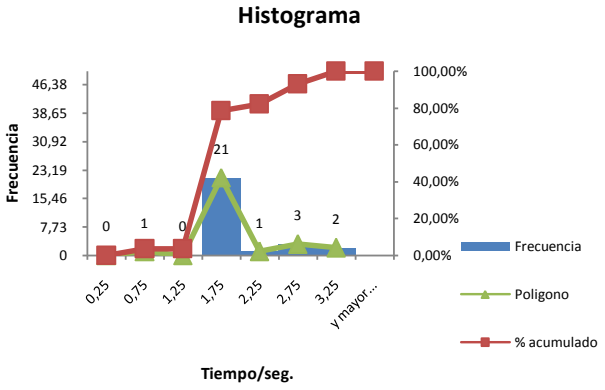

Figura 3. Tiempo en segundos que demora un ataque de rastreo de sistemas. Eje horizontal: tiempo en segundos para realizar un escaneo de puertos. Eje vertical: cantidad de ataques. En rojo: frecuencia acumulada. En verde: frecuencia de la clase. En azul: número de ataques.

En la Figura 4, se observa las muestras tomadas referente a la cantidad de paquetes en kb/s que viaja por la red. El valor más alto de recurrencias es de 42 kb/s, valor relativamente bajo que no afecta al rendimiento de la red tomando en cuenta que las redes tienen anchos de banda de cientos y hasta miles de Mb/s (Megabyte/segundo).

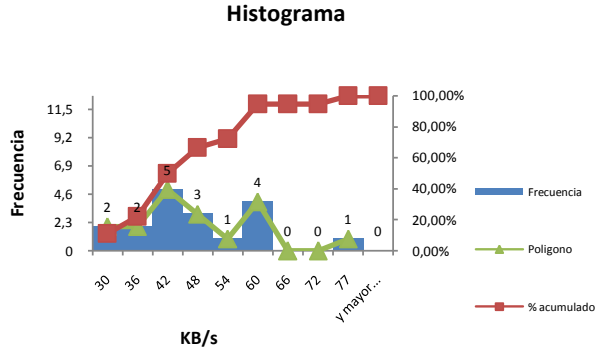

Figura 4. Recurso de red que ocupa un ataque de rastreo de sistemas. Eje horizontal: ancho de banda ocupado por un escaneo de puertos. Eje vertical: cantidad de ataques. En rojo: frecuencia acumulada. En verde: frecuencia de la clase. En azul: número de ataques.

4.2 Ataque a contraseñas (fuerza bruta)

En la Figura 5, se observa el tiempo más frecuente que se demora, un ataque de fuerza bruta, en descifrar una clave o contraseña. Las formas de combinación de caracteres (alfanuméricos) para las contraseñas tienen el siguiente tamaño: pequeño (2-3 caracteres), mediano (5-6 caracteres) y largo (más de 6 caracteres). Siendo la forma pequeña la más concurrente por su número de aciertos alcanzados.

Cabe mencionar que en el proceso de descifrar contraseñas, a través de la herramienta Medusa, se empleó un archivo (diccionario) de contraseñas con 3600 palabras (aprox.), estas palabras junto al nombre del usuario del equipo víctima son empleadas por el equipo atacante para intentar hacer una conexión remota con el equipo víctima.

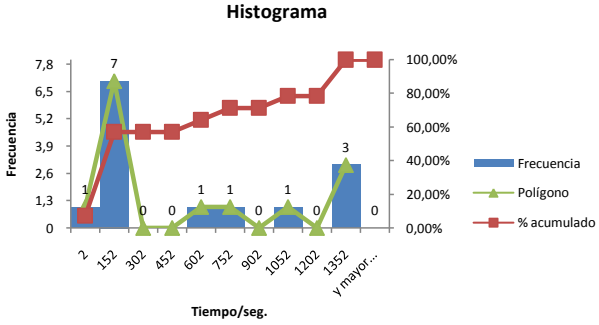

Figura 5. Tiempo consumido al realizar un ataque de fuerza bruta. Eje horizontal: tiempo en segundos en realizar un ataque de fuerza bruta. Eje vertical: cantidad de ataques. En rojo: frecuencia acumulada. En verde: frecuencia de la clase. En azul: número de ataques.

4.3 Ataque de suplantación de identidad

La Figura 6, demuestra la diferencia existente, en cuanto al número de paquetes transmitidos, entre el atacante A y su víctima B y viceversa, al momento de producir un ataque de denegación de servicio.

También se puede observar la diferencia bastante significativa al transmitir de A hacia B en relación de B hacia A, esto se debe a que A no recibe el mensaje de respuesta de B, ya que estos mensajes son enviados a la IP por la que A se hace pasar.

4.4 Ataque de denegación de servicios

Los resultados obtenidos durante la ejecución de un ataque ARP Spoofing, fueron que todos los `host` detectados quedaron sin servicio web, correo electrónico e internet.

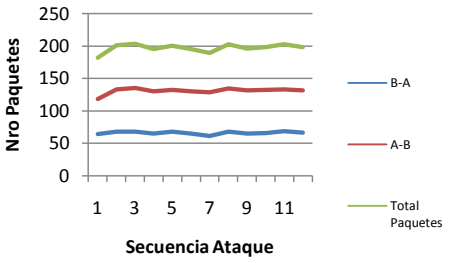

Figura 6. Paquetes transmitidos por un ataque de fuerza bruta. Eje horizontal: secuencia de ataques de suplantación de identidad. Eje vertical: cantidad de paquetes transmitidos durante el ataque. En verde: total de paquetes trasmitidos por el canal de transmisión. En azul: paquetes trasmitidos desde el equipo A (atacante) hacia su víctima B. En rojo: paquetes trasmitidos desde el equipo B (víctima) hacia su atacante A.

En la Figura 7, se puede observar la cantidad de bytes que se transmite entre el atacante A y su víctima B, siendo el equipo B el que transmite pocos bytes, en relación a la gran cantidad de bytes que trasmite su atacante A, esto se debe a que B queda sin servicio por el ataque generado y no puede trasmitir dado que no encuentra al ruteador.

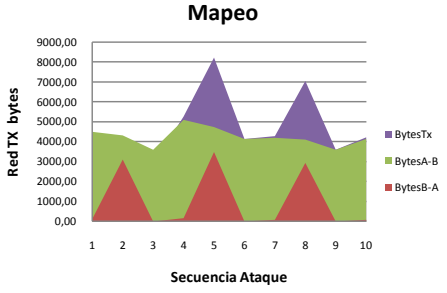

Figura 7. Recurso de red consumido al realizar un ataque DoS. Eje horizontal: secuencia de ataques de suplantación de identidad. Eje vertical: ancho de banda ocupado por el ataque de denegación de servicio. En azul: total de bytes trasmitidos entre A y B. En verde: cantidad de bytes trasmitidos desde el equipo A (atacante) hacia su víctima B. En rojo: cantidad de bytes trasmitidos desde el equipo B (vćctima) hacia su atacante A.

5. Resultado de los ataques implementando mecanismos de mitigación

5.1 Rastreo de sistemas o escaneo de puertos

Ejecución del ataque:

```
#Nmap -sF 192.168.30.2 -p 21
```

Los resultados obtenidos se muestran en la Figura 8.

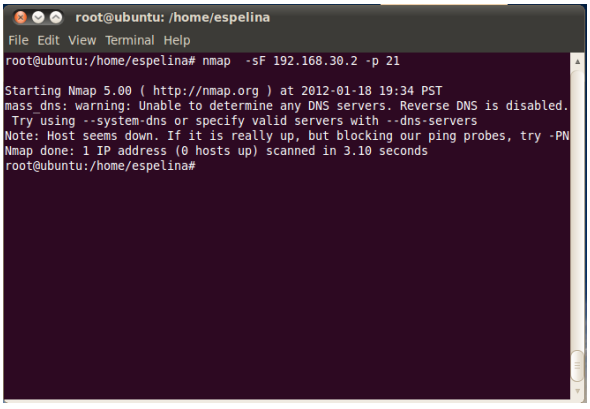

Figura 8. Resultado del ataque de un escaneo de puertos.

Como se observa, la ausencia de respuesta ante un paquete FIN por parte del equipo `192.168.30.2` es notoria. Se puede interpretar este resultado como una acción de bloqueos de paquetes FIN por parte del *firewall* implementado como mecanismo de mitigación.

5.2 Ataque de fuerza bruta

Ejecución del ataque

```
#medusa -h 192.168.10.1 -u coralia -P
        passwords.txt - M ssh
```

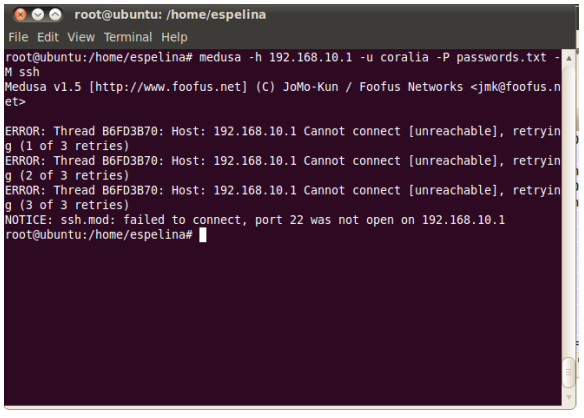

Figura 9. Resultado del ataque de fuerza bruta con Medusa

Como se observa en la Figura 9, el intento de conexión a la dirección IP `192.168.10.1` no se logra. Se puede interpretar esta información como un bloqueo efectuado por el script `BloqueoF.sh` al equipo atacante que tiene como fin descifrar la contraseña del equipo víctima con dirección IP `192.168.10.1` a través del puerto 22.

5.3 Ataque de suplantación de identidad (Spoofing)

Existen cinco clases de direcciones que se deben negar en la interfaz externa, a fin de evitar un ataque *Spoofing* y son: La propia dirección IP, direcciones IP de multidifusión de clase D, direcciones reservadas clase E, direcciones de difusión mal formadas como por ejemplo la dirección `0.0.0.0` que es de difusión especial.

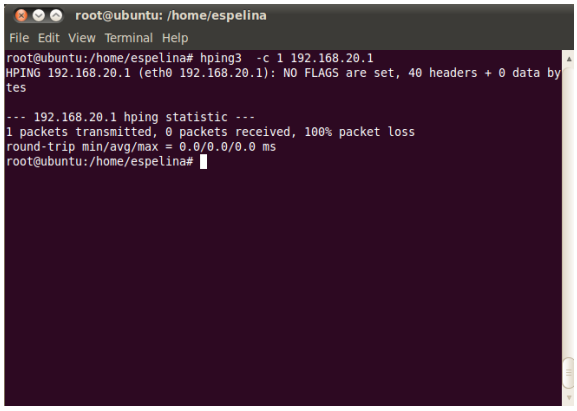

Figura 10. Resultados del efecto producir un ataque *Spoofing*.

Como se observa en la Figura 10, el intento de conexión a la IP `192.168.20.1` no se logra. Se puede interpretar este resultado como una acción del *firewall* al bloquear el acceso al equipo atacante debido a que los paquetes legales nunca proceden de dicho equipo.

5.4 Ataque de Denegación de Servicios (DoS)

Ejecución del ataque

```
#Hping3 -a 192.168.10.1 192.168.100.10
```

En la Figura 11 se puede ver los resultados obtenidos, a través de la herramienta Wireshark, en el intento de un equipo atacante por realizar una inyección de ARP constante, en donde la dirección MAC del *router* es una inexistente. Para el caso en que no estuviese implementado el *firewall*, como mecanismo de mitigación de este ataque, en tan solo unos segundos se habría inundado la red con cientos de paquetes transmitidos de forma ininterrumpida. Por lo contrario se observa claramente una transmisión de paquetes TCP nula, lo que refleja que el ataque no se ejecuta.

6. Conclusiones

La utilización de máquinas virtuales para la realización de este proyecto fue un punto clave, ya que se

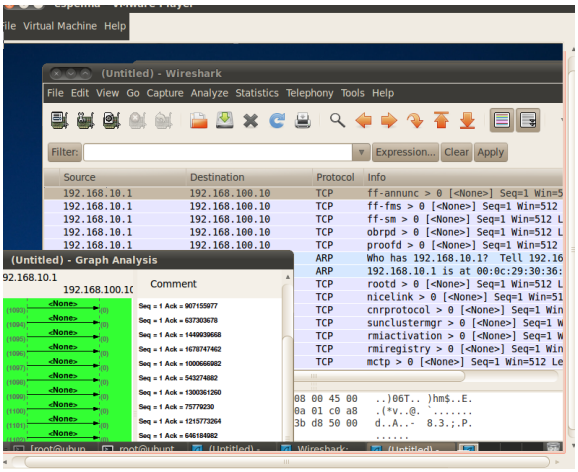

Figura 11. Resultados del efecto producido por un ataque DoS.

pudo trabajar con varios equipos virtuales para establecer los diferentes escenarios de pruebas de ataques y mitigación a los mismos, sobre un mismo computador anfitrión; también se han podido establecer dos escenarios de pruebas utilizando distintos computadores anfitriones y máquinas virtuales invitadas. Estas opciones permitieron que pruebas de un laboratorio de computación sea más sencilla.

En la creación de los escenarios propuestos, algunas MV (máquinas virtuales) creadas tienen sistema operativo Windows XP y otras Linux (Ubuntu), a fin de dar una apariencia real a la topología propuesta, y que a pesar de haber encontrado herramientas de uso gratuito para Windows que permitieron ejecutar ataques a la red LAN, Linux no deja de ser uno de los sistemas operativos más destacados debido a su gran versatilidad y funcionalidad frente a otras soluciones.

Las plataformas de virtualización empleadas en el presente trabajo, VirtualBox y VMWare, permitieron implementar por completo la topología de pruebas propuesta en la Figura 1. Sin embargo, la herramienta de VMWare proporciona un entorno más amigable desde su instalación, configuración, creación y administración de las MVs.

Entre las herramientas de código abierto para enrutamiento basado en software más destacado se encuentra el paquete Quagga, debido a que su configuración en consola es muy parecida al realizado en los enrutadores comerciales como Cisco. Además, por ser una herramienta de código abierto y de libre distribución constituye una alternativa de bajo costo y de buen funcionamiento ante escenarios de enrutamiento para el aprendizaje.

En las prácticas desarrolladas se pudo analizar algunas de las actividades previas realizadas por los

atacantes de redes TCP/IP para conseguir sus objetivos, como por ejemplo: obtención de información de un sistema, descubrimiento de usuarios y exploración de puertos. Adicionalmente se realizó un estudio detallado de algunos ataques concretos contra redes TCP/IP como son los ataques de escaneo de puertos, fuerza bruta, suplantación de identidad y denegación de servicios, tanto en una red de área local como en una extendida.

Para contrarrestar los ataques concretos a redes TCP/IP estudiados en el presente proyecto, se desarrolló un demonio en Shell script que detectó, controló y mitigó los ataques mencionados de manera automática y constante. Los resultados redujeron considerablemente las amenazas y vulnerabilidades de los ataques en redes en producción.

En la evaluación de resultados obtenidos durante las prácticas realizadas sobre los ataques a redes TCP/IP, donde se consideró los mismos parámetros de medición y evaluación para las dos plataformas de experimentación virtualizadas, no existiendo diferencias significativas en los resultados obtenidos.

El *firewall* construido con IPTables es una poderosa herramienta para filtrado, denegación o aceptación de paquetes, a través de una correcta configuración de filtrado de paquetes a nivel de *kernel*, siendo para ello necesario y muy importante conocer la estructura de éstos y la manera en la que son transmitidos.

Referencias

[1] M. Krause and H. Tipton, *Handbook of information security management*, 5th ed. Auerbach Publications, 1998.

[2] S. Garfinkel and G. Spafford, *Web security, privacy & commerce*, 2nd ed. O'Reilly Media, Incorporated, 2001.

[3] W. Fuertes, J. de Vergara, and F. Meneses, "Educational platform using virtualization technologies: Teaching-learning applications and research uses cases," in *Proc. II ACE Seminar: Knowledge Construction in Online Collaborative Communities*, Albuquerque, NM - USA, October, 2009.

[4] K. Scarfone, S. M, and P. Hoffman, *Guide to Security for Full Virtualization Technologies.* DIANE Publishing, 2010, Recommendations of the National Institute of Standards and Technology, Gaithersburg, MD.

[5] J. Keller and R. Naues, "A collaborative virtual computer security lab," in *Second IEEE International Conference on e-Science and Grid Computing e-Science'06.* California, EEUU: IEEE, 2006, p. 126.

[6] P. Li and T. Mohammed, "Integration of virtualization technology into network security laboratory," in *38th Annual Procedings Frontiers in Education Conference, FIE.* Saratoga, New York: IEEE, October, 2008, pp. S2A–7.

[7] F. Abbasi and R. Harris, "Experiences with a generation iii virtual honeynet," in *Telecommunication Networks and Applications Conference (ATNAC).* Canberra, Australia: IEEE, May, 2009, pp. 1–6.

[8] F. Galán and D. Fernández, "Use of VNUML in virtual honeynets deployment," in *IX Reunión Española sobre Criptología y Seguridad de la Información (RECSI)*, Barcelona, Spain, September, 2006.

[9] E. Damiani, F. Frati, and D. Rebeccani, "The open source virtual lab: a case study," in *Procedings of the Workshop on Free and Open Source Learning Environments and Tools, FOSLET*, Italy, 2006, pp. 5–12.

[10] Co-innovation lab Tokyo. Disaster recovery solution using virtualization technology. White paper. [Online]. Available: http://www.cisco.com/en/US/prod/collateral/ps4159/ps6409/ps5990/N037_COIL_en.pdf

[11] P. Ferrie, "Attacks on more virtual machine emulators," *Symantec Technology Exchange*, 2008.

[12] F. Galán, D. Fernández, W. Fuertes, M. Gómez, and J. López de Vergara, "Scenario-based virtual network infrastructure management in research and educational testbeds with VNUML," *Annals of Telecommunications*, vol. 64, no. 5, pp. 305–323, 2009.

[13] W. Fuertes and J. López de Vergara M, "An emulation of VoD services using virtual network environments," in *Procedings of the GI/ITG Workshop on Overlay and Network Virtualization NVWS'09*, vol. 17, Kassel-Germany, March, 2009.

[14] VMware home page. [Online]. Available: http://www.vmware.com

[15] VirtualBox home page. [Online]. Available: http://www.virtualbox.org

[16] C. Lee, C. Roedel, and E. Silenok, "Detection and characterization of port scan attacks," *Univeristy of California, Department of Computer Science and Engineering*, 2003. [Online]. Available: http://cseweb.ucsd.edu/users/clbailey/PortScans.pdf

[17] Hacking. VII Ataques por fuerza bruta. [Online]. Available: http://jbercero.com/index.php?option=com_content&view=article&id=71:hacking-vii-ataques-por-fuerza-bruta&catid=40:hacking-tecnicas-y-contramedidas&Itemid=66

[18] Laboratorios. Hacking, técnicas y contramedidas, ataques por fuerza bruta (BruteForce) III. [Online]. Available: http://labs.dragonjar.org/laboratorios-hacking-tecnicas-fuerza-bruta-brute-force-iii

[19] Jhon the Ripper 1.7.6. [Online]. Available: www.openwall.com/jhon/

[20] F. Callegati, W. Cerroni, and M. Ramilli, "Man-in-the-Middle attack to the HTTPS protocol," *Security & Privacy*, vol. 7, no. 1, pp. 78–81, 2009.

[21] Nemesis. Última comprobación. [Online]. Available: http://nemesis.sourceforge.net/

[22] S/a. [Online]. Available: http://www.estrellateyarde.org/so/logs-en-linux

[23] S/n. [Online]. Available: http://usemoslinux.blogspot.com/2010/11/cron-crontab-explicados.html

[24] J. Li, N. Li, X. Wang, and T. Yu, "Denial of service attacks and defenses in decentralized trust management," *International Journal of Information Security*, vol. 8, no. 2, pp. 89–101, 2009.

[25] J. Matthews, W. Hu, M. Hapuarachchi, T. Deshane, D. Dimatos, G. Hamilton, M. McCabe, and J. Owens, "Quantifying the performance isolation properties of virtualization systems," in *Proceedings of the 2007 Workshop on Experimental Computer Science*. San Diego, CA: ACM, June, 2007, p. 6.