Evaluación y Mitigación de Ataques Reales a Redes IP utilizando Tecnologías de Virtualización de Libre Distribución

W. Fuertes, P. Zapata, L. Ayala y M. Mejía
Dirección de Postgrado, Escuela Politécnica del Ejército, Sangolquí - Ecuador
wfuertesd@espe.edu.ec, lzapata@ups.edu.ec, flacoleas@hormail.com, mike_m78@hotmail.com

RESUMEN: 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. La presente investigación se enfoca en la evaluación de diversos ataques reales de redes IP utilizando plataformas de virtualización *VMware Player 3.01 y Virtual Box 3.2,* con el fin de establecer mecanismos de seguridad para mitigarlos. Para llevarlo a cabo, se diseñó e implementó varias topologías de experimentación utilizando entornos de red virtuales, dentro de las cuales se probaron el escaneo de puertos, fuerza bruta, y suplantación de identidad, 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 el tiempo que se tarda y las consecuencias en el rendimiento de la red causadas por ataque. Para contrarrestar dichos ataques, se desarrolló un programa en Shell script que sea 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 seguridades de las redes de información.

Palabras clave: Ataques de seguridad, evaluación, mitigación, tecnologías de virtualización

ABSTRACT: IP networks Attacks can collapse the continuity of business services affecting its image and causing important economic losses. This research focuses on the evaluation of several IP networking real attacks using virtualization platforms *VMware Player 3.01 y Virtual Box 3.2,* 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 and spoofing, 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 networks security.

Keywords: security attacks, virtualization technology

1. INTRODUCCION

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 seguridades de las redes, es necesario conocer las vulnerabilidades de las empresas 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 en investigar e implementar soluciones para disminuir los ataques de seguridad a la redes aprovechado las tecnologías de virtualización. El trabajo propuesto por Keller y Naues [4.], formula la implementación de un laboratorio colaborativo de seguridad utilizando máquinas virtuales. Li y Mohammed [5.], proponen la integración de las tecnologías de virtualización para la instrucción de seguridad en redes implementando un laboratorio remoto de detección de intrusiones. Otros investigadores [6.][7.], 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 [8.][9.][10.], han utilizado las plataformas de virtualización para recuperación de desastres y mitigación de ataques reales a redes IP.

El presente trabajo tiene como objetivo 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 diseñó e implementó diferentes escenarios de experimentación utilizado VMware Player y VirtualBox. Luego se aplicó diversos tipos los 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.

Como principal contribución de esta investigación se han evaluado diversos ataques utilizando como plataforma de experimentación las tecnologías de virtualización, contrarrestando dichos ataques con programas en Shell script que corren como procesos en segundo plano.

El resto del artículo ha sido organizado de la siguiente manera: La sección 2 presenta el marco teórico que fundamenta esta investigación. En la sección 3 se describe el entorno en el que se desarrollaron los ataques, la configuración de la topología de pruebas y los diversos tipos de ataques evaluados. La sección 4 analiza, evalúa y discute los resultados. En la sección 5, se resume los trabajos relacionados. Finalmente en la sección 6 se establecen las conclusiones sobre la base de los resultados obtenidos y se delimita el trabajo futuro.

2. FUNDAMENTACION

2.1 Virtualización y Escenarios Virtuales de Red como plataforma de experimentación

La *Virtualización* es la forma de particionamiento lógico de un equipo físico en diversas máquinas virtuales, para compartir recursos de hardware, como CPU, memoria, disco duro y dispositivos de entrada y salida [11.]. Esta tecnología permite la ejecución de múltiples máquinas virtuales y sus aplicaciones simultáneamente, siendo una gran alternativa para la implementación de escenarios virtuales de red que permiten la reproducción de la funcionalidad de redes reales, facilitando la evaluación de múltiples ambientes de experimentación y validación de software [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 desplegada sobre uno o múltiples equipos físicos, que emula un sistema equivalente y cuyo entorno deberá ser percibido como si fuera real [13.].

Para implementar los escenarios virtuales para esta investigación, se ha elegido *VMware Player y VirtualBox* que son plataformas 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 [14.]. En el caso de *VMware Player*, porque es capaz de crear y desplegar máquinas virtuales, de tal forma que múltiples sistemas operativos pueden ejecutarse sin modificación y al mismo tiempo. VMware Player funciona bajo Microsoft Windows, Linux, NetWare y Solaris [15.]. En el caso de *VirtualBox*, porque es un software que dispone de una interfaz gráfica denominada Virtual Box Manage, la misma que permite crear máquinas virtuales, definiendo sus características virtuales de memoria, disco, teclado, mouse y CDROM, así como la respectiva configuración de red [16.].

2.2 Tipos de ataques a redes IP evaluados

A través de los años se han desarrollado formas cada vez más sofisticadas de ataque a redes IP. Entre los más comunes y que han sido evaluados a lo largo de esta investigación se pueden describir los siguientes:

Escaneo de Puertos, que consiste en el envío de una serie de señales (paquetes), que llegan a la máquina atacada, y ésta responde reenviando otra determinada cantidad de paquetes, que el escaneador decodificará y traducirá. Dicha información consta esencialmente del número IP de la máquina atacada y datos sobre el o los puertos que se encuentran en ese momento abiertos. La aplicación por excelencia para realizar exploración de puertos es *Nmap (Network Mapper)*[17.]

Fuerza Bruta, que es la forma de recuperar una clave probando todas las combinaciones posibles hasta encontrar aquella que permite el acceso. Consiste en generar el diccionario (hash) de todas las posibles combinaciones y compararlas con el patrón (hash) que permita el acceso [18.]. Técnicamente, el término Hash se refiere a una función o método para generar claves que representen de manera casi unívoca a un documento, registro, archivo, etc., [19.]. Una manera eficiente de realizar ataque de fuerza bruta es mediante el uso de diccionarios de contraseñas. Los ataques tradicionales más conocidos de fuerza bruta son *Jhon the Ripper [20.] e Hydra*.

Suplantación de Identidad (Spoofing), que consiste en aplicar técnicas de suplantación de identidad generalmente con usos maliciosos o de investigación [21.]. Existen diferentes tipos como el IP spoofing, ARP spoofing, DNS spoofing, Web spoofing o e-mail spoofing. Para efecto del presente estudio nos hemos enfocado al ARP spoofing. ARP Spoofing hace referencia a la construcción de tramas de solicitud y respuesta ARP falseadas, de forma que en una red local se puede forzar a una determinada máquina a que envíe los paquetes a un host atacante en lugar de hacerlo a su destino legítimo.

Los métodos de ataque descritos serán evaluados en los escenarios virtuales de red cuya topología será descrita en la siguiente sección.

3. CONFIGURACION DEL EXPERIMENTO

3.1. Diseño y configuración del escenario

Se ha diseñado una topología de prueba tanto con VMware Player como con VirtualBox, aplicando las mismas condiciones y parámetros de configuración para ambas plataformas, tomado como modelos aquellos escenarios de uso más común en pequeña y mediana organización. A continuación, se ha implementado dicha topología donde los equipos involucrados, ya sean virtuales o físicos, comparten un mismo espacio de direcciones IP. La Fig. 1 representa el caso real en el cual una red LAN/WAN es sometida a ataques IP y los atacantes son usuarios de la Intranet o del Internet (véase Fig.1).

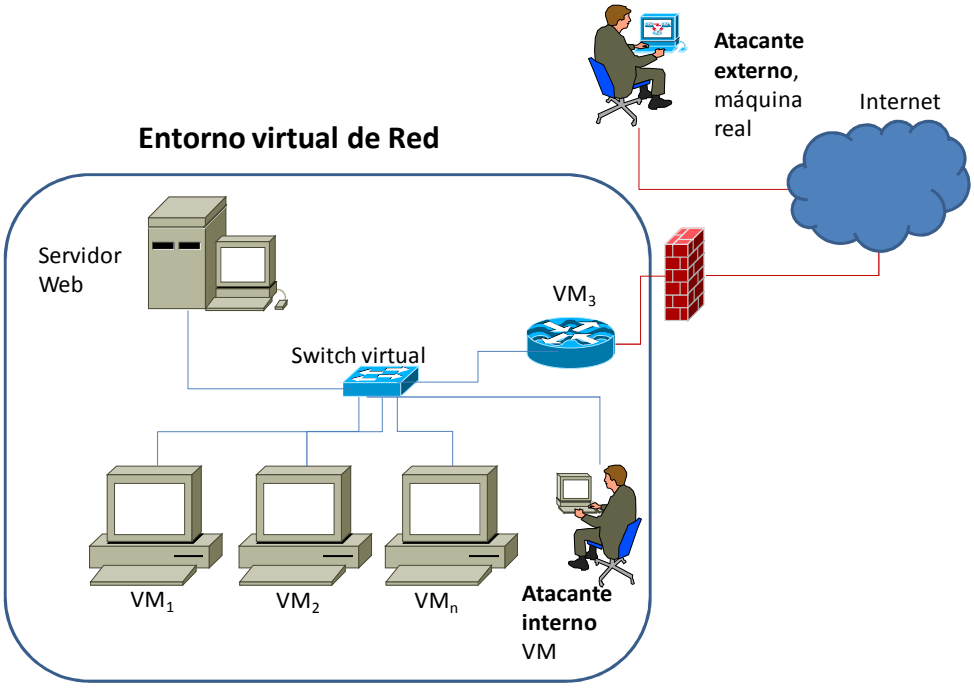

Figura 1. Diseño de la topología de prueba

3.2. Implementación del escenario

Todas las pruebas se desarrollaron sobre Linux Ubuntu Server -i386, en un computador Pentium Intel core duo, RAM de 4GB y una partición Ext3 de 120 GB. En todas las VMs se instaló el mismo sistema de ficheros y el mismo kernel.

El procedimiento utilizado para implementar el experimento consistió en los siguientes pasos: instalación de VMware Player y VirtualBox, creación de máquinas virtuales en cada herramienta de virtualización, direccionamiento IP, configuración de servicios Web y Ssh, creación y aplicación de algoritmos para arranque automático en shell script del escenario, sincronización de reloj con NTP (Network Time Protocol), configuración del ataque y aplicación del algoritmo o aplicación de software para análisis de tráfico.

Para la captura de tráfico de los dos experimentos que se exponen a continuación, se utilizó Tcpdump [22.]. Tcpdump es una herramienta en línea de comandos cuya utilidad principal es analizar el tráfico que circula por la red.

3.3. Implementación de los ataques

Para la implementación de cada uno de los ataques, fue necesario instalar algunas herramientas de libre distribución que han permitido generar los diversos ataques y que también han facilitado la captura de tráfico, tanto para Linux como para Windows. La Tabla 1 describe el tipo de ataque y las diversas herramientas utilizadas en esta plataforma de experimentación.

TABLA 1. RESUMEN DE HERRAMIENTAS UTILIZADAS PARA LA EJECUCION DE LOS ATAQUES.

Nro. Ataque	*Descripción*	*Sistema Operativo*	*Software para el ataque*	*Software para obtener el Flujo de tráfico*
1	Escaneo de Puertos	Ubuntu	Nmap	Tcpdump
		Windows	Zenmap	Ettercap
2	Fuerza Bruta	Ubuntu	Medusa Linux_hash_password.py	
		Windows	John the Ripper	
3	Suplantación de Identidad	Ubuntu	Nemesis	Ettercap
		Windows	Cain & Abel	Cain&Abel

3.3.1 Escaneo de Puertos

Para la generación de este ataque se utilizó *Nmap [23.]*, el mismo que con sus opciones específicas, fue capaz de detectar equipos conectados, puertos abiertos, servicios y aplicaciones en ejecución, el tipo de sistema operativo, el firewall, entre los principales.

Tanto para el caso del *atacante interno*, como externo se probaron algunas opciones de *nmap*, obteniendo una diferencia poco significativa referente a la velocidad de escaneo. Los tipos de scaneo seleccionados fueron: el *ACK scan* que permite identificar de forma precisa cuándo un puerto se encuentra en estado silencioso; y *TCP connect()* que intenta establecer una conexión con cada uno de los puertos del host a escanear, es muy rápido y no se necesita privilegios de root para poder efectuar el escaneo.

3.3.2 Ataque de fuerza bruta

Para la generación de este ataque se aplicó el tradicional *Jhon the Ripper*, el mismo que tiene como objetivo medir el nivel de seguridad de las contraseñas asignadas, utilizando la topología de experimentación basada en plataformas de virtualización descritas en el apartado 3.1. Para aplicarlo, se requiere de un fichero de contraseñas cifradas que se genera para el caso de Ubuntu con el comando *unshadow* y para Windows a través del programa *pwdump*. Una vez generado el fichero, el programa *Jhon the Ripper* empieza a trabajar automáticamente y va mostrando por pantallas las contraseñas que haya ido descifrando sobre dicho archivo.

3.3.3 Ataque de Suplantación de Identidad

Para la generación de este ataque se utilizó *Némesis* [24.], que es una herramienta mediante consola de comandos, con la cual se puede crear diferentes tipos de ataques de suplantación de identidad denominados ARP-spoofing. Este ataque consiste en la inyección de diferentes tipos de paquetes (ARP, TCP, UDP, ICMP) y enviarlos por la red. Se trata de la construcción de tramas de solicitud y respuesta ARP modificadas con el objetivo de falsear la tabla ARP de una máquina víctima y forzarla a que envíe los paquetes a un host atacante en lugar de hacerlo a su destino legítimo. ARP es el protocolo encargado de traducir direcciones IP a direcciones MAC para que la comunicación pueda establecerse.

4 EVALUACION DE RESULTADOS

4.1 Escaneo de puertos

En relación al ataque de ***escaneo de puertos***, las Tabla 2 y 3, muestran el tiempo que se tarda un atacante interno y externo en hacer un escaneo de puertos con *Nmap* a un equipo víctima en los escenarios descritos en el apartado 3.1, tanto con VMware, como con VirtualBox con Ubuntu o Windows como sistemas operativos hospedados. .

TABLA 2. Comparación de Resultados al ejecutar el escaneo de un atacante interno.

INTERNO	VIRTUALBOX			VMWARE		
MAQ1	53,39	62,30	45,16	56,91	45,22	52,92
VM1(ub)	67,13	66,11	69,94	69,68	85,64	52,44
VM2(win)	86,14	46,73	78,13	68,63	68,78	56,19

TABLA 3. Comparación de Resultados al ejecutar el escaneo de un atacante externo.

EXTERNO	VIRTUALBOX			VMWARE		
MAQ1	201,42	201,64	201,51	201,59	201,64	201,86
VM1(ub)	202,53	202,78	202,42	202,20	202,39	202,15
VM2(win)	402,44	402,46	402,28	59,58	66,48	56,94

Con el fin de determinar el tiempo máximo que se demora un equipo atacante en hacer un escaneo de puertos a sus equipos víctimas, se realizaron varios ataques a equipos tanto reales como virtuales. La Fig. 2 muestra el histograma y la frecuencia acumulada, en donde se puede apreciar que la mayoría de quipos toma un promedio de 72 segundos en realizar un escaneo *TCP connect()* que es el tipo de escaneo más eficiente.

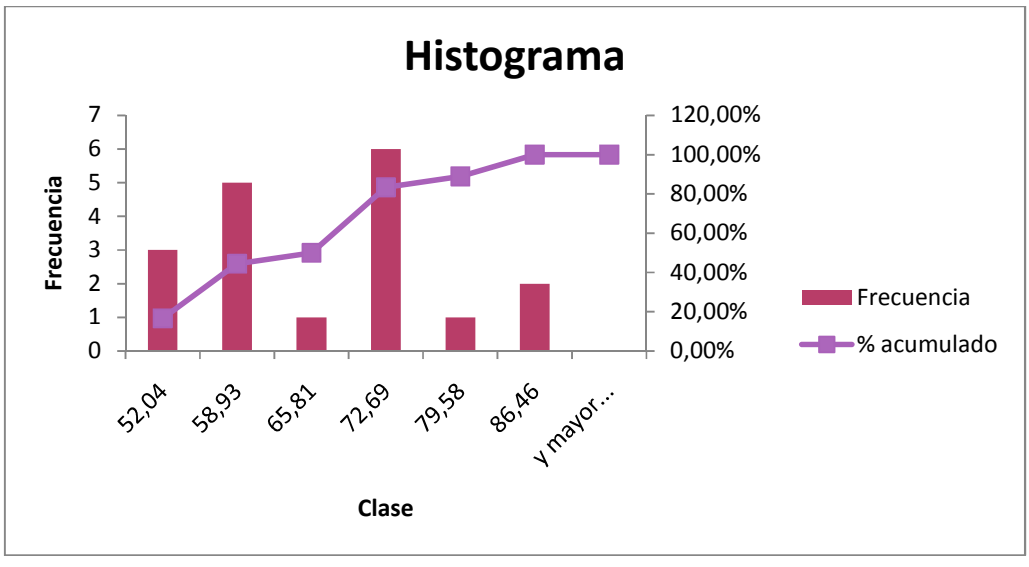

Figura 2. Histograma y % acumulado del escaneo de un atacante externo TCP Connect.

4.2 Fuerza Bruta

En relación al ataque de fuerza bruta, la Fig. 3, muestra el tiempo en segundos que se toma un equipo en descifrar su contraseña utilizando John the Ripper, con sistema operativo Windows. La medición se efectuó sobre claves de cuatro, seis y ocho caracteres de longitud y con combinaciones entre minúsculas, mayúsculas y números, obteniéndose como resultado que mientras más caracteres y combinaciones de letras y números tiene una clave, más tiempo tomará el programa en descifrarla. Cabe mencionar que las pruebas se efectuaron en 5 equipos sobre los cuales se aplicaron las mis mismas claves no habiendo diferencia en el tiempo que tomaron en descifrar cada clave. Adicionalmente se puede concluir que el tiempo que se demora en descifrar una clave también depende de las letras con que se inicie la misma, es decir si una clave está formada por las primeras letras del alfabeto le tomará menos tiempo que una que comience con las últimas letras del alfabeto.

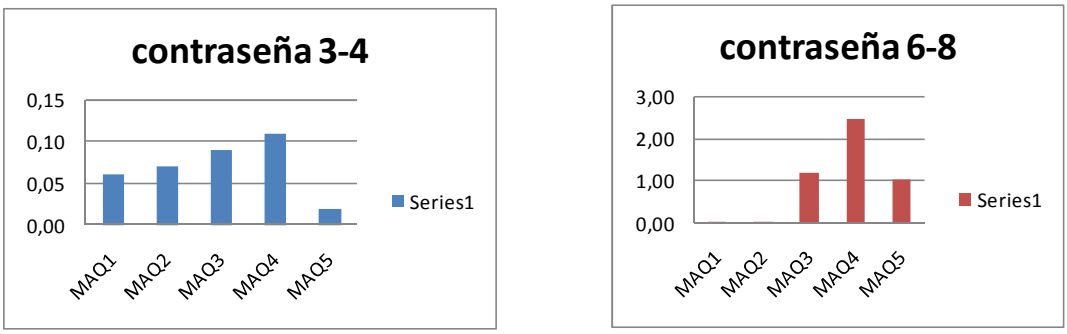

Figura 3. Tiempo que utiliza un equipo en descifrar una clave con John the Ripper. **3.a** contraseña de 3 a cuatro caracteres, **3.b** contraseña de 6 a 8 caracteres.

4.3 Ataque de suplantación de Identidad

En la Fig. 4 se observa los resultados frente a un ataque ARP-Spoofing utilizando Némesis. Con el fin de determinar el consumo de ancho de banda ante este tipo de ataque, se realizaron varias pruebas donde los equipos víctimas fueron máquinas virtuales tanto de Virtual Box como VMWare. Se tomaron 35 muestras en 60 segundos. Como se puede apreciar los equipos víctimas ocupan un ancho de bando de 963 kbps a los 60 segundos.

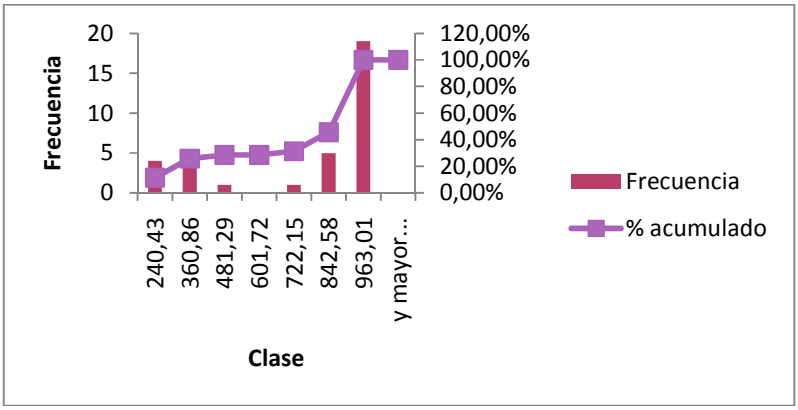

Figura 4. Función de distribución acumulada del consumo de Ancho de Banda frente a un ataque de suplantación de identidad.

4.4 Evaluación del Algoritmo de detección, control y mitigación

Cómo alternativa de solución para detectar, controlar y mitigar los ataques descritos en este artículo, se desarrolló un demonio o administrador regular de procesos en segundo plano que ejecuta comandos programados en Shell scripts a intervalos regulares, basado en la configuración del archivo ***crontab***. Este demonio automatiza los mecanismos de mitigación para controlar los cuatro tipos de ataques evaluados. Luego se procede a registrar la ejecución del monitoreo de manera constante (parametrizable) en el cron ejecutando crontab –e, registrando y definiendo cronogramas (cada minuto por ejemplo).

Para el caso del ataque de escaneo de puertos, suplantación de identidad y denegación de servicios citados en la sección 4, ha sido implementado un firewall de Linux. En concreto es un algoritmo que ante la evidencia de un paquete ICMP, ARP-, *Nmap*, ejecuta un Shell script que activa por consola comandos *Iptables*, que modifican la configuración del firewall de Linux. El algoritmo se basó tanto en *reglas* para filtrar paquetes y decidir si dejarlo pasar o no; así como *cadenas*, que se ejecutan de arriba a abajo hasta que se cumpla una de ellas, que son políticas para decidir qué hacer con los paquetes que no coincidieron con ninguna de las reglas. Una vez que se cumple las condiciones y se activa, se cierra la conexión a dicha IP y por lo tanto se detiene el ataque.

Para el caso del ataque de fuerza bruta, ha sido implementado un mecanismo de autenticación mediante un Shell script que realiza una revisión sobre el archivo de logs de validación de autentificaciones (auth.log). Este busca autentificaciones inválidas y cuando supera el límite definido de fallas (parametrizable) accede al archivo de denegación de hosts para registrar la IP de la máquina que está intentando realizar los accesos fallidos. Una vez que se cumple las condiciones y se activa, se cierra la conexión a dicha IP y por lo tanto se detiene el ataque. La Fig. 5 muestra el diagrama de secuencias del Proceso de mitigación cuando se trata del ataque de fuerza bruta.

Cabe señalar que al aplicarse este demonio, y al repetir los ataques descritos en la sección 4 en la topología de experimentación utilizando tecnologías de virtualización, los resultados muestran la funcionalidad de esta investigación que reduce las amenazas y vulnerabilidades de las redes en producción.

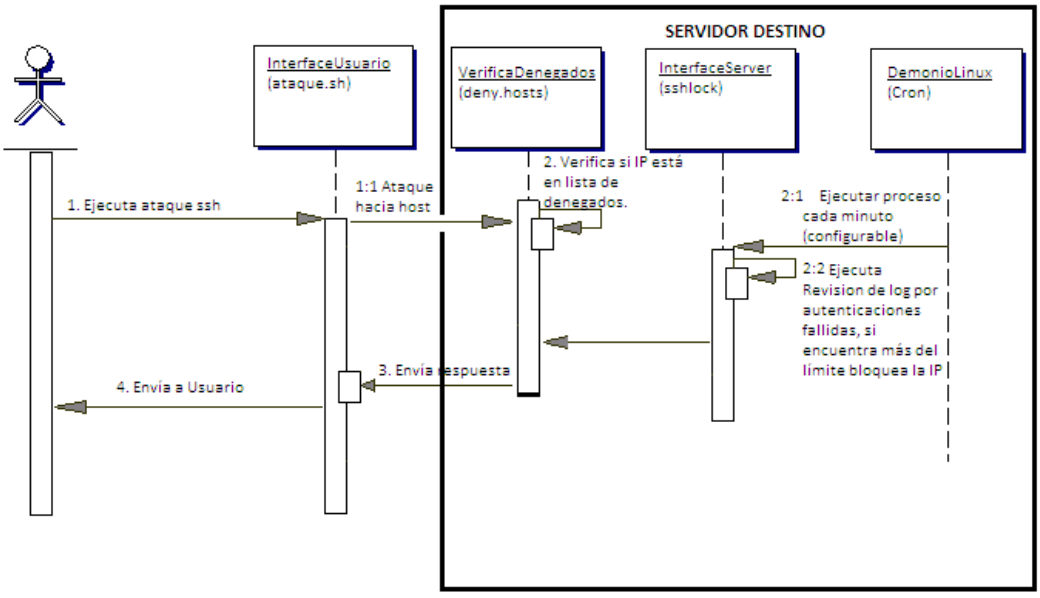

Figura 5. Diagrama de secuencias del algoritmo en shell script para contrarrestar el ataque de fuerza bruta.

5 TRABAJOS RELACIONADOS.

Aunque exista una diversidad de trabajos relacionados, en esta sección se han incluido los más relevantes, que se han encontrado durante la investigación:

En lo que se refiere al ámbito educativo, el trabajo desarrollado por Keller y Naues [4.], expone la implementación de un laboratorio colaborativo de seguridad utilizando máquinas virtuales. Esta investigación permite realizar las tareas de administración de seguridad mediante un Shell remoto, además cuenta con otra interfaz Web que permite saber los resultados de su práctica de laboratorio, y tareas pendientes. En este mismo ámbito, Li y Mohammed [5.], proponen la integración de las tecnologías de virtualización para la instrucción de seguridad en redes implementando un laboratorio remoto de detección de intrusiones. Adicionalmente, [6.][7.] 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 contexto, El trabajo propuesto por Damiani [8.], describe un laboratorio virtual basado en tecnología de código abierto, utilizando la plataforma Xen, que tiene como objetivo la configuración de un firewall para proteger el servidor de ataques externos mediante Iptables. Todos estos trabajos han sido utilizados como insumos en esta investigación.

En relación a soluciones de recuperación de desastres mediante virtualización, el trabajo propuesto por [9.], demuestra que el uso de esta tecnología como una opción, debido a que minimizan el uso de servidores y liberan a los administradores del hecho de tener el mismo ambiente de hardware que los servidores en operación, representando una mayor flexibilidad y costos mucho menores de mantenimiento y administración.

En un contexto más cercano al nuestro, el trabajo propuesto por Ferrie [10.], utilizó código malicioso y ataques de denegación de servicio contra máquinas virtuales VMware, VirtualPC, Paralles e Hydra. Sin embargo en este estudio solo se recomiendan pero no se han desarrollado soluciones tangibles. Comparando este trabajo con el nuestro existen dos diferencias fundamentales, la primera hemos realizado la evaluación de diversos ataques de redes y hemos desarrollado e implementado un demonio que permita detectar, controlar y mitigar los ataques evaluados.

6 CONCLUSIONES Y TRABAJO FUTURO

La presente investigación se enfocó en la evaluación de diversos ataques reales de redes IP utilizando plataformas de virtualización. Durante esta investigación, se diseñó e implementó varias topologías de experimentación basadas en entornos virtuales de red. Los tipos de ataques evaluados por ser tradicionales fueron escaneo de puertos, fuerza bruta y suplantación de identidad, tanto en una red de área local como en una red de área extendida. Para cada topología, se utilizó diferente software libre tanto para producir el ataque como para obtener el flujo de tráfico, evaluándose las vulnerabilidades de la red virtual. Para contrarrestar dichos ataques, se desarrolló un set de programas en Shell script que detectó, controló y mitigó los ataques mencionados de manera programable y constante. Los resultados redujeron las amenazas y vulnerabilidades de los ataques en redes en experimentación.

Como trabajo futuro se planea evaluar ataques distribuidos de denegación de servicio, utilizando otros mecanismos de mitigación como la encriptación, sistemas de detección de intrusos y VPNs en un entorno de red virtualizado.

Referencias Bibliográficas

[1.] H. Tipton, M. Krause, "Information Security Management Handbook", Auerbach Publications. Fifth Edition. ISBN: 08493-1997-8

[2.] S. Garfinkel with Gene Spafford Web Security, Privacy & Comerse.. O'Really. Second Edition. ISBN 0-596000-456

[3.] W. Fuertes, J. E. Lopez de Vergara, F. Meneses, "Educational Platform using Virtualization Technologies: Teaching-Learning Applications and Research Uses Cases", In proceedings of II ACE Seminar: Knowledge Construction in Online Collaborative Communities, Albuquerque, NM - USA, October 2009.

[4.] J. Keller, R. Naues, "A Collaborative Virtual Computer Security Lab," e-science, In Proc. Second IEEE International Conference on e-Science and Grid Computing, pp. 126, CA, USA, Dec. 2006

[5.] P. Li, T. Mohammed, "Integration of Virtualization Technology into Network Security Laboratory", In Proc. 38th ASEE/IEEE Frontiers in Education Conference, Saratoga, NY, October, 2008.

[6.] F. Abbasi, R. Harris, "Experiences with a Generation III virtual Honeynet", In Proccedings of the Telecommunication Networks and Applications Conference (ATNAC), 2009 Australasian, Canberra, ACT , ISBN: 978-1-4244-7323-6. May 2009.

[7.] Fermín Galán, David Fernández, "Use of VNUML in Virtual Honeynets Deployment", IX Reunión Española sobre Criptología y Seguridad de la Información (RECSI), Barcelona (Spain), pp. 600-615, September 2006. ISBN: 84-9788-502-3.

[8.] E. Damiani, F. Frati, D. Rebeccani, "The open source virtual lab : a case study". In proceedings of the workshop on free and open source learning environments and tools, hosted by: FOSLET 2006; pp. 5-12, Italy nel 2006.

[9.] Co-innovation lab Tokyo, "Disaster Recovery Solution Using Virtualization Technology", White paper, http://www.cisco.com/en/US/prod/collateral/ps4159/ps6409/ps5990/N037_COIL_en.pdf.

[10.] P. Ferrie, Attacks on Virtual Machine Emulators, Symantec White Paper, 2008.

[11.] F. Galán, D. Fernández, W. Fuertes, M. Gómez and J. E. López de Vergara, "Scenario-based virtual network infrastructure management in research and educationnal testbeds with VNUML,", Annals of Telecommunications, vol. 64(5), pp. 305-323, May 2009.

[12.] Matthews, J., Hapuarachi, W., Deshane, Hu, M. T., Quantifying the Performance Isolation Properties of Virtualization Systems. In Proc. of Workshop on Experimental computer science ExpCS'07, 13–14 June, 2007, San Diego, CA.

[13.] W. Fuertes and J. E. López de Vergara, "An emulation of VoD services using virtual network environments,". In Proc. GI/ITG Workshop on Overlay and Network Virtualization NVWS'09, Kassel-Germany, March 2009.

[14.] W. Fuertes and J. E. López de Vergara, "A quantitative comparison of virtual network environments based on performance measurements,", in Proceedings of the 14th HP Software University Association Workshop, Garching, Munich, Germany, 8-11 July 2007.

[15.] VMware home page, [Online:] http://www.vmware.com

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

[17.] C. Lee, C. Roedel, E. Silenock, "Detection and Characterization of Port Scan Attacks", [Online:] "http://cseWeb.ucsd.edu/users/clbailey/PortScans.pdf

[18.] Hacking: VII Ataques por Fuerza Bruta. [Online:]: 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

[19.] Laboratorios: Hacking, Técnicas y contramedidas, Ataques por fuerza bruta (Brute Force) III. [Online:] http://labs.dragonjar.org/laboratorios-hacking-tecnicas- fuerza-bruta-brute-force-iii

[20.] Jhon the Ripper 1.7.6., [Online:] www.openwall.com/jhon/

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

[22.] Jacobson, V., Leres, C., and McCanne, S. Tcpdump. Available at anonymous@ftp.ee.lbl.gov

[23.] Nmap, www.nmap.org. Ultima comprobación Octubre de 2010.

[24.] Nemesis, http://nemesis.sourceforge.net/. Ultima comprobación, 20 de octubre de 2010.

[25.] Ethercap, http://ettercap.sourceforge.net/. Ultima comprobación, 21 de octubre de 2010