Seguridad de la información, Generación y Mitigación de un Ataque de Denegación de Servicios

Hector Avalos[a], Estevan Gómez[a]

[a] Instituto de Informática y Computación, Universidad Tecnológica Equinoccial, Calle Rumipamba s/n, entre Burgeois y Av. Atahualpa, Quito, Ecuador
havalos@ute.edu.ec, exergomez@ute.edu.ec

Resumen. El presente artículo contiene una breve introducción al tema de seguridad de la información, enfocado a los ataques de denegación de servicios que se pueden realizar sobre las redes de datos. Se describe el experimento realizado en un ataque Low Orbit Ion Cannon del tipo SYN Flood Attack a un servicio web de GLPI sobre un servidor Ubuntu, también se presenta el procedimiento para la mitigación del ataque mediante la implementación y configuración de la herramienta Iptables (área de memoria donde todas las aplicaciones, en modo de usuario, pueden ser intercambiadas hacia memoria virtual cuando sea necesario) a través de la cual los administradores crean reglas para cada filtrado de paquetes y módulos de Traducción de dirección de red (NAT). Iptables es la herramienta estándar de todas las distribuciones modernas de GNU/Linux. Iptables en un servidor Centos. Se realizó la experimentación en un servidor de la Universidad Tecnológica Equinoccial que se preparó con este propósito. La experimentación contribuye a mostrar cómo la denegación de servicios puede afectar a la continuidad del negocio reflejadas en no poder brindar un servicio normal, generando incomodidad, posibles pérdidas de tiempo, económicas, reputación.

Palabras Clave: Ataque, Denegación de servicios (DoS), Denegación de servicios distribuido (DDoS), Low Orbit Ion Cannon (LOIC), Iptables, Traducción de direcciones de red (NAT).

1 Introducción

Un ataque de Denegación de servicios también conocido como ataques DoS (Denial of Services) tiene como principal objetivo atacar un grupo o red de computadoras causando que los servicios sean inaccesibles para los usuarios que acceden de una forma legítima a los mismos, dicho ataque normalmente ocasiona la pérdida total de conectividad a la red, por el aumento excesivo del consumo del ancho de banda de la víctima o también una sobrecarga de los recursos de los sistemas informáticos.

Un ataque se manifiesta a través de la saturación intencional de los puertos con un flujo constante de información, sobrecargando los recursos de los servidores y la capacidad de responder a las peticiones realizadas por los usuarios originales, dicha denegación es muy utilizada por los hackers para vulnerar la disponibilidad y utilidad de los recursos ofrecidos sobre todo de entidades gubernamentales.

Existe también el denominado ataque distribuido de denegación de servicios o DDoS por sus siglas en inglés (Distributed Denial of Service) el cual genera un gran flujo de

información desde distintos puntos de conexión, este es un ataque más poderoso ya que el daño colateral que genera puede ser mucho más grave que el ataque en sí.

Existen varios métodos mediante los cuales se pueden realizar denegaciones a los servicios, en la tabla 1 se aprecia algunos métodos de denegación [1]:

Tabla 1. Métodos de Ataques

Método	**Descripción**
Spoofed	Envío de paquetes con una dirección de origen falsificada
Malformed	Envío de paquetes con bits o flags encendidos en forma anormal
Floods	Envío de paquetes conformados de manera legítima en gran cantidad
Null	Envío de paquetes sin contenido
Protocol	Envío de paquetes con protocolos ilegítimos
Fragmented	Envío de paquetes fragmentados los cuales nunca serán completados
Brute Force	Envío de paquetes que exceden el umbral definido de 'flow rates'

Los ataques de denegación de servicio suelen ser de dos tipos:

Alto número de conexiones/agotamiento ancho de banda existente: En este tipo los administradores deciden utilizar una región de ordenadores para lanzar peticiones contra un único objetivo, la meta principal es consumir el ancho de banda disponible para el servicio o los recursos de todo el equipo que existan en el camino: routers, switches, firewalls, balanceadores, servidores, etc. [2]

Consumo de la capacidad de procesamiento de recursos: Este tipo de ataque es más común y difícil de detectar ya que su objetivo es atacar únicamente aplicaciones específicas como pueden ser protocolo (HTTP, DNS, VOIP, etc) mediante paquetes que tardan mucho en procesarse y descartarse. En este caso con un número menor de ataques se logra que se bloquee el servicio [2].

El impacto que una denegación haga sobre un negocio es muy grave ya que puede haber afectaciones tanto en lo económico, como en la productividad, multas, reputación, imagen, etc.

1.1 SYN Flood Attack

Este tipo de ataque envía paquetes tratados a la máquina activando el bit SYN en la conexión RCP y alterando la ip origen mediante técnicas de Spoofing, la víctima responde con un SYN/ACK (SA flags) considerando que se trata de una conexión legítima y espera por un ACK (A flag) por parte del cliente [3].

Al tratarse de direcciones falsas, la respuesta nunca llegará y la secuencia no llega a completarse ocasionando que la víctima se sature de conexiones bloqueando de esta manera el acceso a conexiones genuinas.

En la siguiente figura, se aprecia una secuencia normal del "saludo de las tres vías" [4]

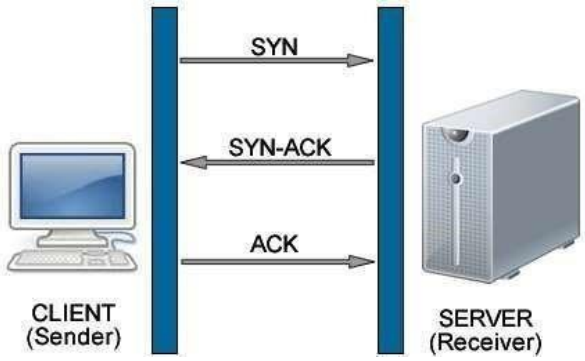


Figura 1: Interacción normal Cliente-Servidor

En tanto que, modificando los headers, la conexión se realizará del siguiente modo:

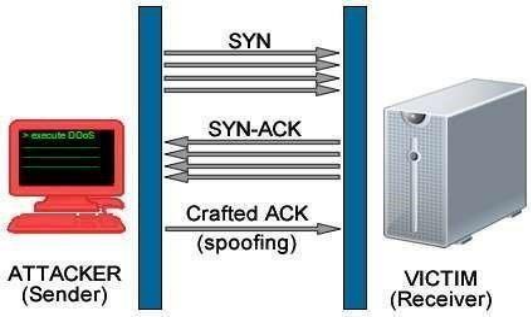


Figura 2: Interacción cliente servidor cambio de Headers [1]

[1] http://www.dragonjar.org/wp-content/uploads/2012/09/three-way-handshake-ATTACK.jpg

Este es sin duda uno de los ataques más conocidos por su simplicidad y efectividad, además es la técnica principal utilizada por el grupo Anonymous que entre sus herramientas principales están LOIC (Low Orbit Ion Cannon) y HOIC (High Orbit Ion Cannon) [5]

La utilización de equipos tipo IDMS (Intelligent DDoS Mitigation System) situados en la frontera antes de cualquier equipo como primera defensa, incluso antes de firewall permitirán detectar y bloquear los ataques de una forma inteligente utilizando diferentes métodos de protección y añadiendo una capa más de protección.

2. Metodología.

Sistemas Operativos: Para ejecutar el ataque de denegación de servicios se trabajó sobre el sistema operativo UBUNTU en su versión No. 12 y para recibir y mitigar el mismo se usó el sistema operativo CENTOS.

Sistema de ataque de denegación de servicios: Para realizar el ataque se utilizó la herramienta LOIQ que sirve para generar un ataque de denegación de servicios a través del envío de paquetes a uno o varios puertos de comunicación con el fin de saturar tanto el CPU como la memoria evitando de esta menar que otros usuarios tenga acceso a un determinado servicio.

Herramientas: Para la práctica se utilizó Tomcat 5 y Mysql sobre los cuales se instaló un aplicativo para el registro de inventarios y control de incidentes denominado GLPI con lo cual se contó con un ambiente de producción real orientado a la gestión de incidentes tecnológicos dentro de una central de emergencias.

Sistema de monitoreo: Para realizar el seguimiento del consumo de recursos del servidor CENTOS, se instaló el control de monitoreo, herramienta que permite conocer el estado de la memoria, CPU y de los puertos TCP/IP documentando de esta manera el consumo actual e histórico del servidor GLPI.

2.1 Proceso Practico

Para realizar el ataque se realizó una simulación de un servicio web, en este caso la instalación de un servidor en apache con una aplicación conocida como GLPI que es un software libre que facilita la administración de recursos informáticos.

GLPI es una aplicación basada en la administración Web que permite registrar y administrar el inventario tanto de hardware como de software de una empresa, facilitando las labores de los equipos técnicos gracias a su diseño, el software GLPI incluye también una mesa de ayuda para el registro y atención de las solicitudes de soporte técnico, con posibilidades de notificación por correo electrónico a usuarios y

al personal de soporte documentando de esta manera el inicio, el seguimiento y el cierre de la solicitud.

Las principales funcionalidades de GLPI están articuladas sobre dos ejes:

1. El inventario preciso de todos los recursos informáticos, y el software existente, cuyas características se almacenan en bases de datos.
2. La administración e historiales de las diferentes labores de mantenimiento y procedimientos relacionados, llevados a cabo sobre esos recursos informáticos.

Para esta simulación se instalará en un ambiente virtual el GLPI sobre el sistema operativo LINUX CENTOS, para ello realizamos los siguientes pasos: Bajar el archivo de la siguiente dirección:

```
https://forge.indepnet.net/attachments/download/597/glpi0.72.4.
ta
r.gz.
Descomprimir en la carpeta

  /var/www/html/glpi
```

Permisos de escritura (765) a las carpetas dentro DE:

```
  dir /glpi /files /config
```

Permisos de escritura a las siguientes carpetas dentro de /files _dumps, _cache, _log,

```
  _sessions, _cron, _uploads vim

  /etc/php.ini
```

En la parte de memory_limit colocar 64MB

Se requiere las extensiones json para php, para revisar si se encuentra instalada se usó el siguiente comando:

```
  php -i | grep json
```

Obteniendo el siguiente resultado:

```
  /etc/php.d/json.ini,  Json json support => enabled json version

  => 1.2.1
```

Agregar la librería json de la siguiente manera: Actualizar `php yum install php-devel php-pear gcc` –y luego de las configuraciones anteriores abrimos el explorador y digitamos en la parte superior http://srvute.uio.ute.edu.ec, cabe recalcar que se realizó configuraciones adicionales como inclusión con DNS en el servidor de dominio y configuraciones adicionales en el software GLPI para obtener la siguiente pantalla. \

Figura 3: Instalación de GLPI

2.2 Ataque de denegación de servicios a través de LOIQ.

Para realizar el ataque de denegación de servicios se utilizó la herramienta LOIQ (Low Orbit Ion Cannon) esta herramienta puede ser ejecutada en diferentes sistemas operativos Windows, Mac o Linux en sus distintas distribuciones; LOIQ fue desarrollado en un ambiente C++ con librerías libre Qt4 la cual permite ser compilado en cualquier plataforma.

Para realizar el ataque en primera instancia se instaló LOIQ a través de los siguientes comandos:

Se descargó de la ruta especificada a continuación el fichero LOIQ

Comando: wget http://ftp.desdelinux.net/loiq-0.3.1a.tar.bz2

Posterior a esto, se descomprimió el archivo e instaló la aplicación.

Comando: `bzip2 -dc loiq-0.3.1a.tar.bz2 | tar -xv && mv loiq0.3.1a .loiq-0.3.1a && sudo ln -s $HOME/.loiq-0.3.1a/loiq /usr/local/bin/}`

Finalmente se digitó en la terminal de comando la palabra “loiq” (sin las comillas) para ejecutar la aplicación.

El software permite ingresar la siguiente información:

Figura 4: LOIQ

IP o el Dominio; es la dirección IP o el nombre del dominio al cual se desea atacar en este caso se realizara un ataque al servidor del HELP DESK cuya ruta es http://srvute.uio.ute.edu.ec

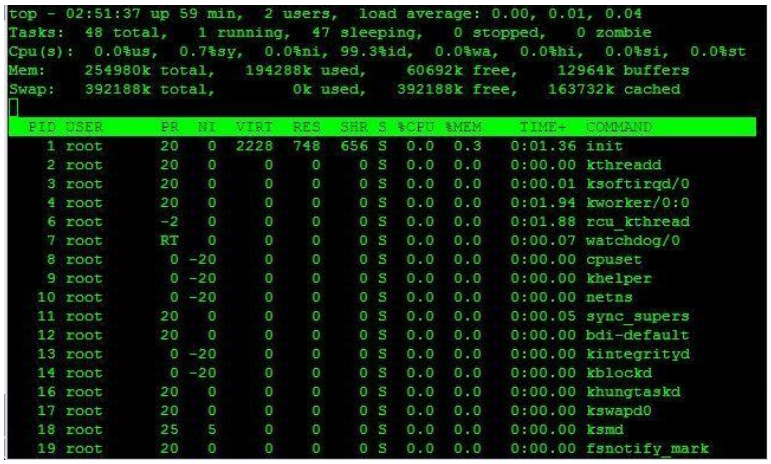
Luego de ello se modifica las opciones del ataque, como número de puerto, en este caso en particular se configuró el puerto 80 para afectar los servicios HTTP y la tasa de transferencia de paquetes por segundo faster, speed, slower; en este caso speed para obtener una curva gráfica del ataque comprensible para el análisis.

2.3 Ejecución del Ataque

Para realizar el ataque se configuró la IP y/o el nombre del dominio srvute.uio.ute.edu.ec/index.php y se activó el botón Lock On, luego de ello se eligió el tipo de ataque en este caso HTTP puerto 80 y la tasa de transferencia threads, después se hizo clic sobre el botón ubicado en la parte superior derecha que dice IMMAH CHARGIN MAH LAZOR lo cual activó el ataque.

Figura 5: LOIQ ataque Dos

Además el porcentaje de consumo del CPU, memoria con sus respectivos IDs.



```
top - 02:51:37 up 59 min,  2 users,  load average: 0.00, 0.01, 0.04
Tasks:  48 total,   1 running,  47 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.0%us,  0.7%sy,  0.0%ni, 99.3%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:    254980k total,   194288k used,    60692k free,    12964k buffers
Swap:   392188k total,        0k used,   392188k free,   163732k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
    1 root      20   0  2228  748  656 S  0.0  0.3   0:01.36 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.01 ksoftirqd/0
    4 root      20   0     0    0    0 S  0.0  0.0   0:01.94 kworker/0:0
    6 root      -2   0     0    0    0 S  0.0  0.0   0:01.88 rcu_kthread
    7 root      RT   0     0    0    0 S  0.0  0.0   0:00.07 watchdog/0
    8 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 cpuset
    9 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
   10 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   11 root      20   0     0    0    0 S  0.0  0.0   0:00.05 sync_supers
   12 root      20   0     0    0    0 S  0.0  0.0   0:00.00 bdi-default
   13 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kintegrityd
   14 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kblockd
   16 root      20   0     0    0    0 S  0.0  0.0   0:00.00 khungtaskd
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kswapd0
   18 root      25   5     0    0    0 S  0.0  0.0   0:00.00 ksmd
   19 root      20   0     0    0    0 S  0.0  0.0   0:00.00 fsnotify_mark
```

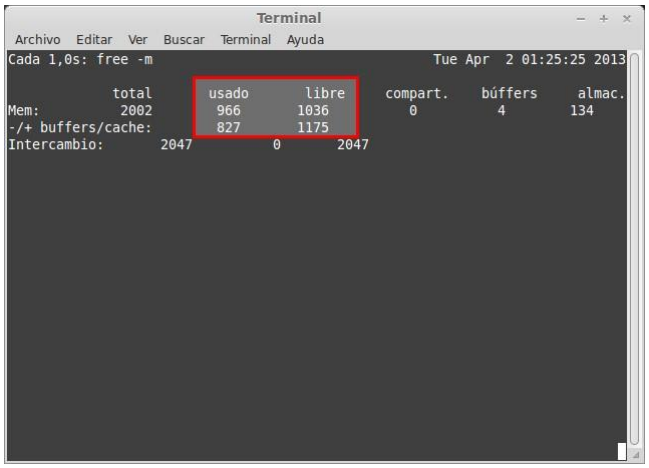
Figura 6: Comando TOP

A los pocos segundos el servidor consume el 100% del procesador y la memoria RAM.
Con el comando TOP se puede obtener en tiempo real un listado de los procesos que se están ejecutando en el sistema,

Como se puede apreciar, en la figura anterior, el LOIQ realiza varias peticiones al puerto 80 del httpd con la más alta prioridad para que sea atendido de forma inmediata,

lo cual obliga a que el uso del CPU y la memoria sean destinados para el despacho de los paquetes, saturando de esta manera los equipos y haciendo que la página se caiga.

Esta información se corroboro con el comando free, mismo que muestra la información relativa al uso de la memoria.



`**Figura 7**: Comando Free, estado de la memoria

Para conocer gráficamente el comportamiento del servidor al momento del ataque utilizamos la herramienta llamada monitor del sistema la cual tiene las siguientes funcionalidades:

- Sistema.- En este se observa el núcleo, el Hardware, y el estado del sistema. En este último está la cantidad de espacio libre que tiene el disco duro o partición de Ubuntu.
- Procesos.- Sirve para ver los servicios o aplicaciones en ejecución en primer y segundo plano.
- Recursos Contiene los siguientes parámetros:

Histórico del CPU: Es un gráfico con el historial o recursos consumidos hace 60 segundos.
Histórico de memoria e intercambio: Se observa la cantidad de memoria RAM que se ha consumido, la cantidad de memoria que se está consumiendo y la cantidad de memoria de intercambio.

Histórico de Red: Se indica la cantidad de KB que el equipo ha enviado y ha recibido

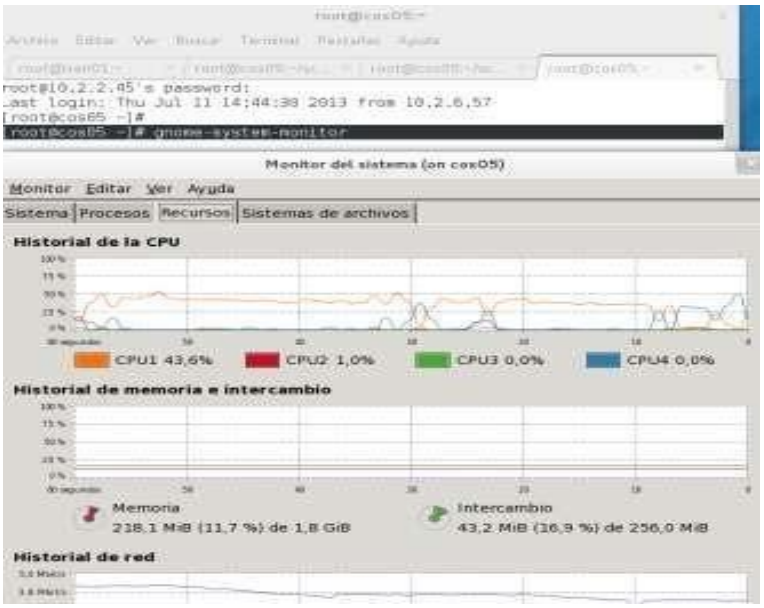


Figura 8 : Sistema de monitoreo, estado actual antes del ataque

En la siguiente imagen se observa el consumo de memoria, el CPU y el tráfico de red se encuentra normal.

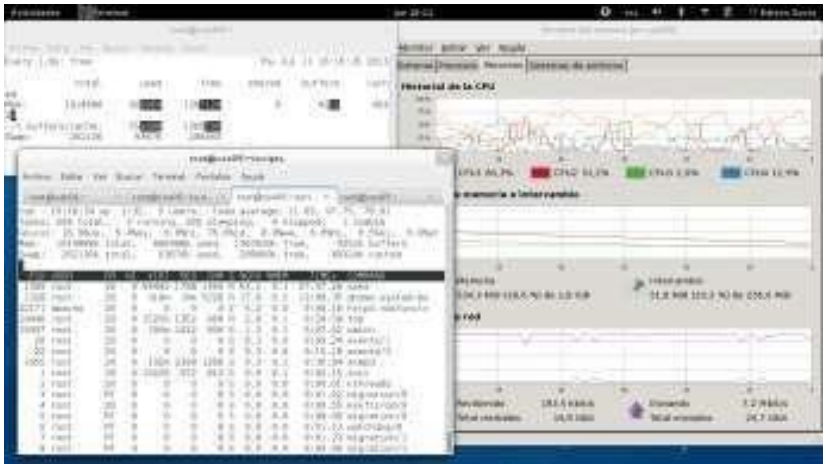


Figura 9: Monitoreo ataque

Histórico de memoria e intercambio: Se observa la cantidad de memoria RAM que se ha consumido.

Figura 10 : Ataque LOIQ al puerto 80

Estado Luego del Ataque:

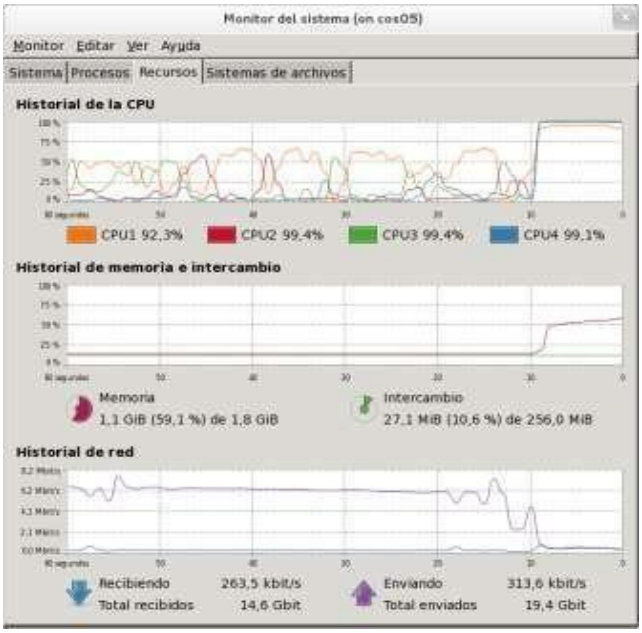


Figura 11 : Estado del Servidor luego del Ataque

En la figura anterior se aprecia que, luego del ataque con el software LOI Q al puerto 80, los valores de consumo de memoria como del CPU se incrementaron.

Monitoreo después de un Minuto:

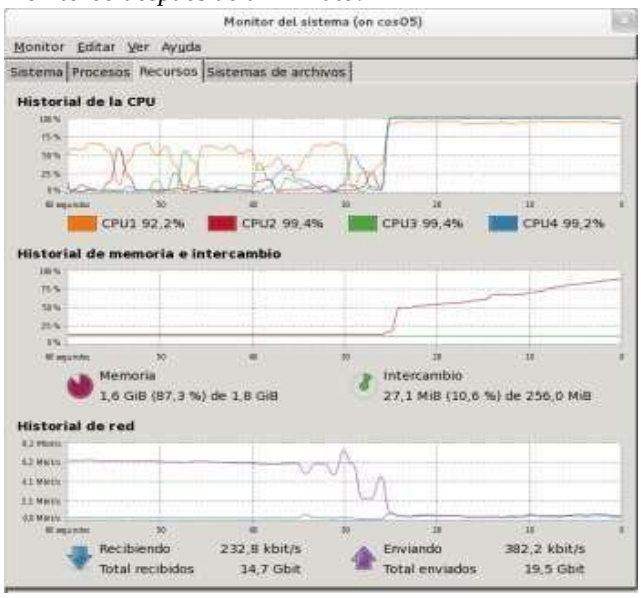


Figura 12: Estado del servidor luego de un minuto

Se observa que después de 60 segundos de haber iniciado el ataque, el consumo del CPU fue de 92% y de la memora un 98%, causando que el equipo no pueda recibir más peticiones colapsando la página.

Figura 13: Página sin respuesta a **srvute.uio.ute.edu.ec**

2.4 Mitigación del Ataque

El Kernel de Linux posee un sistema de filtrado de paquetes llamando Netfilter o Iptables encargado de filtrar todo el tráfico que llega a las interfaces de red de una PC o servidor por medio de un grupo de reglas.

Iptables:

FILTER.- Se encarga del filtrado de paquetes. Las cadenas son: `INPUT OUTPUT y FORWARD` [6]
NAT (Network Address Translation).- Se encarga del enmascaramiento de IP, se usa para redireccionar puertos o cambiar las IPs de origen y destino. Las cadenas serán: PREROUTING y POSTROUTING [6]
MANGLE.- Permite la modificación de paquetes como ToS (Type of Service), TTL (Time to live) o mark. [6]

RAW.- Se usa para configurar excepciones en el seguimiento de paquetes en combinación con la acción o target NOTRACK.
Trabaja sobre la cadena PREROUTING Y OUTPUT y su única acción es Notrack. [6] **Parámetros:**
Toda regla iptables está definida por una serie de parámetros tales como: [6]

- i Interfaz de entrada (eth0, eth1, eth2, etc.)
-o Interfaz de salida (eth0, eth1, eth2,...)

-sport Puerto de origen

-dport Puerto destino

-p El protocolo del paquete a comprobar, tcp, udp, icmp o all.

- j Esto especifica el objetivo de la cadena de reglas, o sea una acción Acciones:

Existen varias acciones entre las cuales se destacan las siguientes: [6]

ACCEPT: Paquete aceptado.

REJECT: Paquete rechazado con notificación mediante ICMP

DROP: Paquete rechazado.

MASQUERADE: Enmascaramiento de la dirección IP origen de forma dinámica. Esta acción es sólo válida en la tabla NAT en la cadena postrouting.
DNAT: Enmascaramiento de la dirección destino, muy conveniente para re-enrutar paquetes
SNAT: Enmascaramiento de la IP origen.

Para instalar el Iptables se siguió los siguientes pasos:

```
sudo apt-get install -test iptables
```

Esto activó las configuraciones por defecto, posterior a ello se deben configurar las reglas como se explica a continuación:

1. Permisos de ejecución:

```
sudo touch iptables chmod 755
/etc/init.d/iptables
```

2. Se creó un script para la ejecución de la mitigación llamado rules.sh con el siguiente contenido.

```
iptables -F iptables -X iptables -A INPUT -p TCP
-m state --state NEW --dport
80 -m recent --set iptables -A INPUT -p TCP -m state -state NEW
-
-dport 80 -m iptables - A INPUT - p TCP - m state --
state NEW -- dport
80 - j ACCEPT
hitcount 15 -j DROP
```

El script realiza el monitoreo del puerto 80 cada 10 segundos revisando el estado de los paquetes recibidos y un muestreo de cómo se está tratando, si existe varias peticiones de un mismo servicio con el mismo direccionamiento lo que realiza es bloquear directamente a la IP o pool de ips que están atacando a dicho puerto.

Para realizar la mitigación del ataque se ejecutó el archivo rules.sh, después de ejecutar el script se visualizó (figura 16) que el comportamiento tanto de la memoria como del
CPU se estabilizó, de igual forma el sistema LOIQ recibió falla (failed) en los paquetes enviados, esto se observa en la figura 14.

Figura 14: Paquetes fallidos (failed) luego de la mitigación

Monitoreo luego de la Mitigación del Ataque

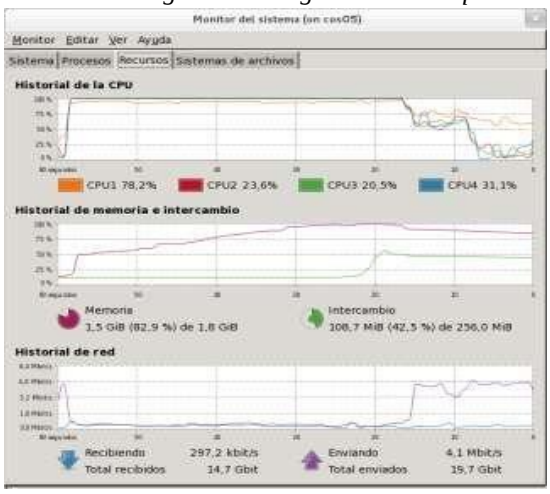


Figura 15: Monitoreo luego de la mitigación del ataque

Estabilización de los Servicios

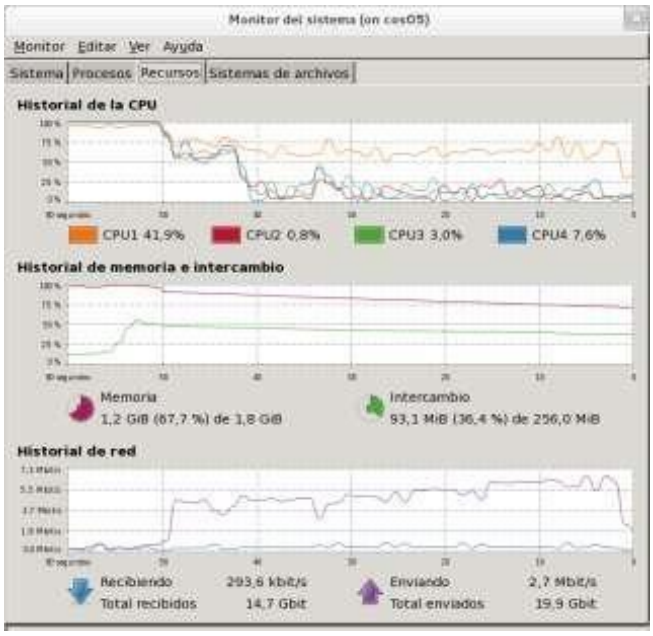


Figura 16: Estabilización de los servicios

Estabilización de Servicios

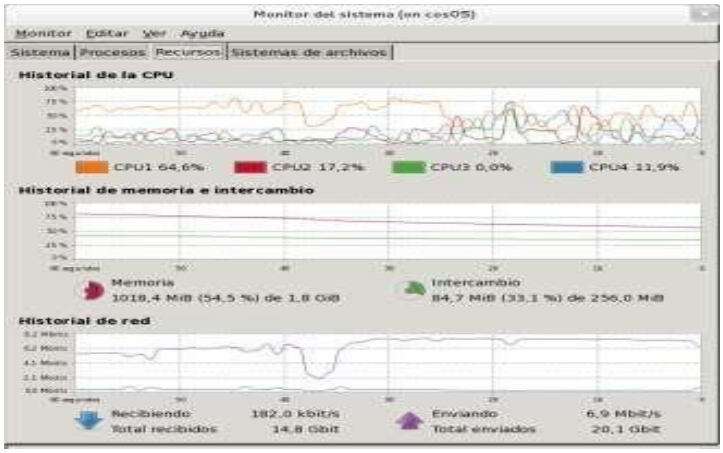


Figura 17: Estabilización total de los servicios luego de la mitigación

Acceso al Servicio luego de la Mitigación

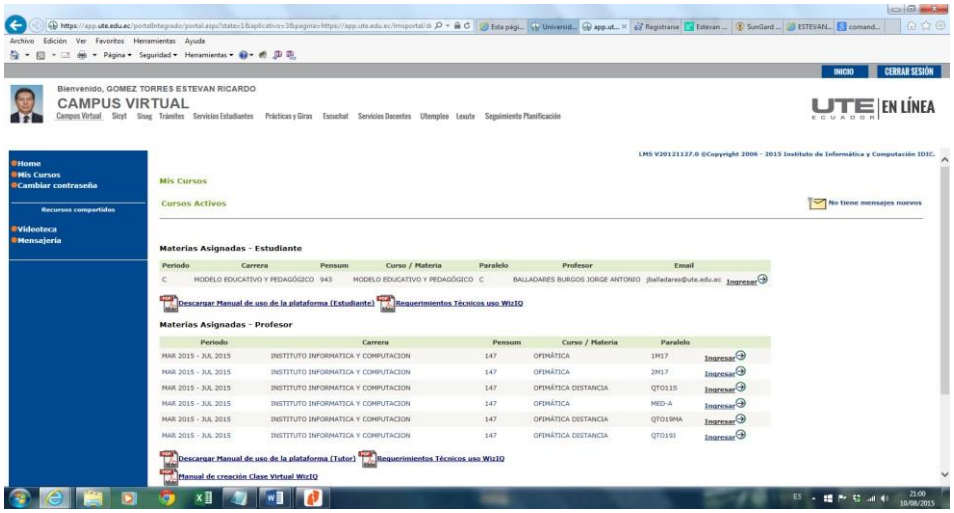


Figura 18: Acceso al servicio GLPI luego de la mitigación

3. Evaluación de resultados y discusión

Dentro de los resultados obtenidos con el Script de mitigación, mediante las herramientas gráficas propias del CENTOS se pudo observar el comportamiento en tiempo real del servidor tanto antes, durante y después del ataque, logrando así obtener una gráfica que describe todo el proceso para un mejor análisis

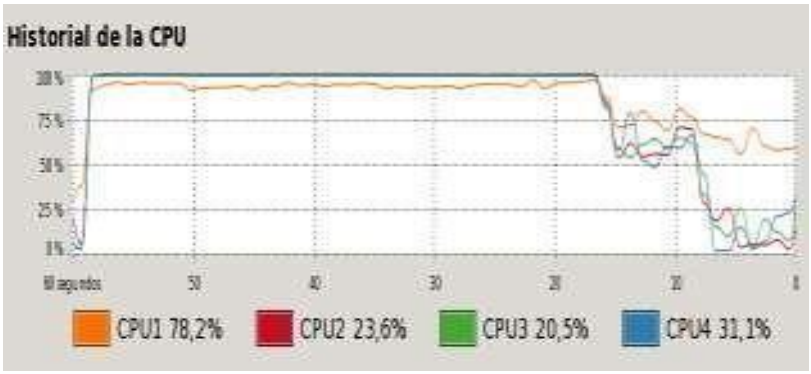


Figura 19: Consumo 4 CPU's

Evaluación del Consumo de Memoria

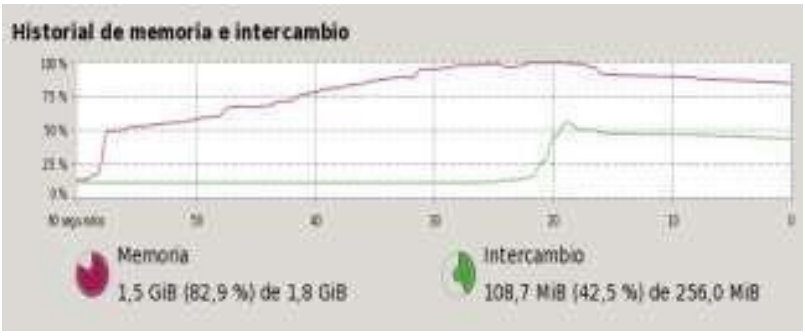


Figura 20: Consumo de memoria

La información más relevante del experimento está en la saturación de los 4 CPU's y de la memoria, donde se puede apreciar el alto consumo de recursos debido al ataque que afectó al cien por ciento de los servicios del servidor, de igual forma se puede apreciar en la gráfica que luego de ejecutar el script de mitigación los niveles de consumo de los 4 CPU's y de la memoria comienza a reducirse y finalmente se estabilizan en los parámetros normales.

El tiempo promedio en que la solución mitigó el ataque fue de 10 segundos, debido a que por motivos prácticos y de demostración se esperó que los 4 CPU's y la memoria estén al límite para ejecutar el Script de mitigación, luego de ello se

puede apreciar en la gráfica que tomó un promedio de 5 minutos estabilizar el consumo de todos los recursos y volver el equipo y los servicios a la normalidad.

Para poder analizar los resultados fue necesario documentar todo el proceso tanto en la identificación del problema, ejecución del ataque como en la implementación de la solución, de igual forma es necesario tener una bitácora de los ataques recibidos con las acciones tomadas para mitigarlo, de esta manera se reduce el RTO garantizando los niveles exigidos de disponibilidad en los servicios.

4. Conclusiones

La implementación de un Firewall mediante IpTables puede ser una herramienta poderosa para filtrado, denegación o aceptación de paquetes siempre y cuando se conozca la estructura, los protocolos de trasmisión y los parámetros de configuración para el filtrado de paquetes a nivel de kernel.

Para contrarrestar los ataques, en el presente proyecto, se desarrolló un demonio en Shell script que detectó, controló y mitigó el ataque mencionado de manera automática y constante, mismo que neutralizó la amenaza del ataque dentro del ambiente de producción.

A pesar de que el proceso de mitigación fue exitoso, el tiempo que necesitó para estabilizar el consumo de recursos del servidor y volver el nivel de prestación de servicios a la normalidad fue de alrededor de 5 minutos, tiempo que consideramos como elevado tomando en cuenta el objeto del negocio de la empresa donde se llevó a cabo este ejercicio, por lo cual el siguiente paso es la búsqueda de la reducción del RTO y la implementación de acciones secundarias que permitan volver más eficiente este procedimiento.

Una investigación a futuro sobre este tema es conocer cómo evitar suceda denegación de servicios a los servidores con tecnología e infraestructura Cloud. El presente trabajo se puede tomar como referencia particularmente para fines académicos.

5. Referencias

1. Alcántara, D., & Tadeo, J. (2011). *Identificación de ataques de DDoS en redes de datos a través de un modelo basado en una red bayesiana (Doctoral dissertation).*
2. Caire, R., (2012). *DDoS Coordinated Attackes Analysus.* http://www.nertfilter.org. Consultado: 25, marzo, 2014.
3. Lemon, J. (2002). *Resisting SYN Flood DoS Attacks with a SYN Cache. In BSDCon.* Vol. 2002, pp. 89-97.
4. Mirkovic, J., & Reiher, P. (2004). *A taxonomy of DDoS attack and DDoS defense mechanisms. ACM SIGCOMM Computer Communication Review, 34*(2),

39-53.
5. Parks, R., & Duggan, D. (2011). Principles of Cyberwarfare. *Security & Privacy, IEEE*, *9*(5), 30-35.
6. Purdy, G. (2009). *Linux iptables pocket reference*. O'Reilly.

V. Biografía

Héctor G. Ávalos, se graduó de Ingeniero Informático en la Universidad Central del Ecuador, trabaja en la Universidad Tecnológica Equinoccial, es docente y desempeña el cargo de Coordinador Académico del Instituto de Informática y Computación.

Estevan Gómez, se graduó de Ingeniero en Sistemas en la Escuela Politécnica del Ejército trabaja como Docente a nivel de Pre-grado y Post- grado, actualmente es estudiante del Doctorado en Ciencias Informáticas en la Universidad Nacional de la Plata en Argentina.