ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

"DISEÑO E IMPLEMENTACIÓN DE UNA SOLUCIÓN DE TELEFONÍA IP APLICANDO HARDENING SOBRE UNA PLATAFORMA BASADA EN SOFTWARE LIBRE EN UNA PYME"

INFORME DE PROYECTO INTEGRADOR

Previa a la obtención del Título de:

LICENCIADO EN REDES Y SISTEMAS OPERATIVOS

ERNESTO ENRIQUE VÁSQUEZ RUBIRA

EVEN ANDRÉS SUÉSCUM TREJOS

GUAYAQUIL – ECUADOR

AÑO: 2015

AGRADECIMIENTO

A Dios por permitirme vivir cada día y guiarme para alcanzar mi meta.

A mis padres Ernesto Vásquez Merizalde y Josefina Rubira Ladines, por ser un apoyo y comprensión porque gracias a sus esfuerzos he podido concluir con mis estudios.

A mi hermana María Isabel Vásquez, por su comprensión y su apoyo en los momentos difíciles.

Al personal docente, por su constante dedicación en todos estos años de estudios en la universidad.

Ernesto Enrique Vásquez Rubira

Gracias a Dios y a la Virgen Santísima, por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente y por haber puesto en mi camino a aquellas personas que han sido mi soporte y compañía durante todo el período de estudio.

A mis padres Rocío Trejos de Suéscum y Juan Alberto Suéscum González, por el apoyo incondicional en mis estudios y por la mejor herencia que me hayan podido dar “la educación”, por hacer de mí una mejor persona a través de su amor, consejo, apoyo y confianza para cumplir con mis metas en la vida.

A mis hermanos Juan Alberto Jr. y David Eduardo Suéscum Trejos, por sus valiosos ejemplos, apoyo, comprensión y alegría que son parte de mi fortaleza para seguir adelante.

Agradezco a los directivos de la Escuela Superior Politécnica del Litoral, ESPOL; por permitirme realizar el trabajo de titulación en sus instalaciones.

De manera particular agradezco, al Ing. José Roberto Patiño tutor de la presente tesis, por su valioso aporte, sugerencia y generosa guía en el desarrollo de esta tesis y la culminación de la misma. Agradezco, a todas aquellas personas que de una u otra manera contribuyeron con la realización de la presente tesis.

Even Andrés Suéscum Trejos

DEDICATORIA

Quiero dedicarles este trabajo a mis Padres que han sido siempre el mejor ejemplo a seguir y que siempre han estado a mi lado apoyándome; en especial a mi madre por impulsarme y aconsejarme en todo momento.

Ernesto Enrique Vásquez Rubira

A Dios y a la Virgen Santísima, por ser mis más valiosas guías y soporte espiritual.

A mis padres Rocío Trejos de Suéscum y Juan Alberto Suéscum González, quienes sembraron en mí el ímpetu, la perseverancia para la consecución de las metas.

A mis hermanos Juan Alberto Jr. y David Eduardo Suéscum Trejos, por ser mi ejemplo a seguir, por estar dispuestos a ayudarme y escucharme en cualquier momento.

A mis sobrinos Juan Andrés, Xavier Francisco y Fabiana, con el ánimo de contribuir con un pequeño aporte en su formación educacional.

Even Andrés Suéscum Trejos

TRIBUNAL DE EVALUACIÓN

Ing. José Patiño, Msc

PROFESOR EVALUADOR

Ing. Rayner Durango, Msig

PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Ernesto Vásquez R

Ernesto Enrique Vásquez Rubira

Even Andrés Suéscum Trejos

RESUMEN

El presente proyecto consiste en el diseño e implementación de una solución de telefonía IP basada en Asterisk sobre un servidor interno para proveer de servicios de telefonía a los usuarios de una red pequeña y verificar que tan factible será la comunicación en la misma, adicional a esto se establecerá configuraciones de seguridad al servidor así como también configuraciones en aplicaciones de código abierto para tratar de mitigar ataques al servidor. Esta solución podrá ser implementada en pequeñas empresas de bajos ingresos para la comunicación por medio de IP.

En cada capítulo se describe en que consiste el proyecto planteado y las pruebas realizadas para verificar la calidad del servicio así como también pruebas para ver qué tan seguro es el mismo.

Para las pruebas de desempeño del servicio se lo realizará en una red de datos pequeña por medio de aplicaciones que nos permitan medir el tráfico que se genera cuando se realizan llamadas entre las extensiones y al mismo tiempo se esté usando otros servicios de red. Para la parte de seguridad se usará aplicaciones de código abierto que permitan probar que tan seguro es el servidor por medio de diferentes ataques.

ÍNDICE GENERAL

AGRADECIMIENTO ii
DEDICATORIA iv
TRIBUNAL DE EVALUACIÓN v
DECLARACIÓN EXPRESA vi
RESUMEN vii
ÍNDICE GENERAL viii
CAPÍTULO 1 1
1. PLANTEAMIENTO 1
1.1. Justificación 1
1.2. Objetivos 1
1.2.1. Objetivos Generales 1
1.2.2. Objetivos Específicos 1
1.3. Descripción del Proyecto 2
CAPÍTULO 2 3
2. METODOLOGÍA E IMPLEMENTACIÓN 3
2.1. Metodología 3
2.2. Implementación 4
2.2.1. Diseño 4
2.2.2. Configuración del Servidor Asterisk 9
2.2.3. Configuración de seguridad para el Servidor Asterisk 14
CAPÍTULO 3 23
3. PRUEBAS Y RESULTADOS. 23
3.1. Pruebas 23

3.2. Resultados 49

CONCLUSIONES Y RECOMENDACIONES 52

BIBLIOGRAFÍA..... 54

ANEXOS 55

CAPÍTULO 1

1. PLANTEAMIENTO

1.1. Justificación

El crecimiento de las TIC's avanza a caminos agigantados, poco a poco la telefonía tradicional está siendo reemplazada por las nuevas tecnología basadas en VOIP. Asterisk es sin duda una de las soluciones más ampliamente usadas a nivel mundial por su bajo costo de implementación, su compatibilidad con otras tecnologías, así como también su rápida configuración con opciones de instalación que incluso hacen que un servidor basado en Asterisk este configurado en poco tiempo dependiendo de los requerimiento del cliente obviamente, por lo cual vamos a realizar un análisis del impacto de este servicio en una red existente para ver si es fiable integrarlo a la misma, así como también decidimos proponer configuraciones mínimas de seguridad y dar a conocimiento de herramientas de código libre que nos permitan asegurar el servidor contra posibles ataques debido a que han crecido las técnicas de ataques a estos servicios y la tecnología actual hace que cualquier persona sin mucho conocimiento pueda causar un gran daño estos servicios. Esto obviamente no es una guía definitiva debido a que constantemente aparecen nuevas amenazas.

1.2. Objetivos

1.2.1. Objetivos Generales

Diseñar e implementar un Servidor de Telefonía IP en una PYME y poder tener mejor un servicio aplicando los métodos de seguridad y calidad de servicio a nivel de Software, en el servidor.

1.2.2. Objetivos Específicos

- ✓ Implementar servicios de VOIP usando software libre a una red existente.

- ✓ Analizar el impacto que va tener en la red de datos y proponer soluciones para la integración de este servicio en una red existente.
- ✓ Aplicar las mejores prácticas de configuración de seguridad en el sistema ante intentos de accesos no autorizados para el servidor VOIP.
- ✓ Definir procedimientos a seguir para brindar mayor seguridad al servidor IP cual sea la plataforma a usar.

1.3. Descripción del Proyecto

Para nuestro proyecto se plantea la instalación y configuración de un servidor de telefonía IP usando software de código abierto “Asterisk” con la cual usaremos una interfaz gráfica llamada “FreePbx” para la configuración de extensiones sobre un sistema operativo en Linux basado en “CentOs” para una red que consta de 20 equipos.

El protocolo de señalización escogido será el protocolo SIP. Los clientes del servidor usaran un programa gratuito llamado softphone que funcionan tanto para computadoras como para celulares inteligentes de diferentes sistemas operativos (Android, Iphone). Con el fin de reducir costos se plante la idea de usar la red de cableado existente.

Se realizará configuraciones basadas en mejores prácticas para asegurar el servidor interno y el servicio**.**

Seguridad en el Servidor

- ✓ Control de acceso al servidor.
- ✓ Control de Servicios.
- ✓ Asegurando SSH.
- ✓ Asegurando el protocolo SIP.

Herramientas de Seguridad

- ✓ Firewall (Iptables).
- ✓ Aplicación de Monitoreo (Fail2ban).

CAPÍTULO 2

2. METODOLOGÍA E IMPLEMENTACIÓN.

2.1. Metodología

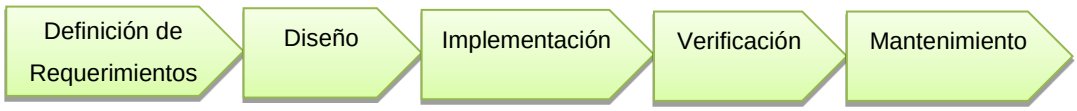


Figura 2.1: Metodología del Proyecto.

La metodología que se usó en este proyecto es “lineal secuencial” que se divide en las varias etapas, como muestra en la Figura 2.1.

Etapa 1: Definición de Requerimientos: en esta etapa se definirá los requisitos que se necesitan para llevar a cabo la solución propuesta como la situación actual de una red de datos existente.

Etapa 2: Diseño: Se definirá el diseño de nuestra solución, es decir qué tipo de dispositivos son necesarios, que programas se van a utilizar, así como también cuáles serán las medidas de seguridad que se le aplicará al servidor.

Etapa 3: Implementación: se realizará la configuración del servidor de telefonía IP creación de las extensiones de los usuarios de la red así como también la configuración en los equipos de los usuario. Por otro lado se realizará las configuraciones mínimas de seguridad en el servidor y la debida configuración de aplicaciones de código abierto como reglas de iptables y fail2ban.

Etapa 4: Verificación: En esta etapa realizaremos las pruebas de desempeño del servicio en una red de datos propuesta y las respectivas pruebas de seguridad basados en diferentes ataques éticos por medio de

aplicaciones de código abierto en dos escenarios uno con seguridad y otro sin seguridad.

Etapa 5: Mantenimiento: En la última etapa del método se realizaría de ser necesario alguna modificación que solicite el cliente es decir, algo que no esté contemplado en el proyecto.

2.2. Implementación

2.2.1. Diseño

Estudio Existente

En estas pruebas se han considerado un escenario de un PYME de 10 equipos, donde se encuentran el uso de Workstations y Smartphones, para los clientes y el servidor con el sistema de telefonía IP ya instalado en una red de datos cableada o inalámbrica.

Como ya se tiene el escenario listo, se pudo probar que todo el flujo de la red era normal y se podía simular el mismo ambiente de una pequeña empresa, donde los usuarios están utilizando la red de forma normal, ya sea consultas internamente, dentro de la red, o de forma externa utilizando la red WAN, tomando en cuenta que es el caso de las empresas, no se están sobrecargando la red, para tener un ejemplo más apegado realidad del uso de las empresas.

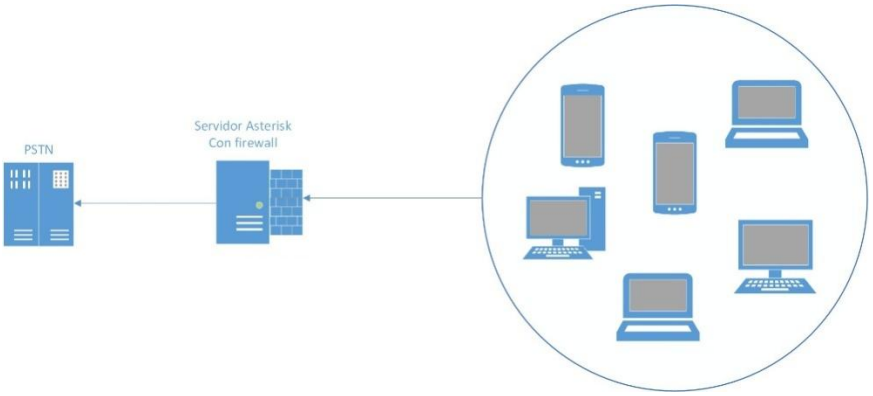


Figura 2.2: Diseño de la Solución.

Una vez ya la red establecida, se pudo implementar el servidor de telefonía IP, como en la Figura 2.2, para así comenzar las pruebas de calidad en el ambiente controlado, además tomando en cuenta el impacto que tendrá este tipo de servicio en la red, así nos podremos dar cuenta de que tan factible es utilizar este tipo de soluciones en las empresas y como superaría a la telefonía convencional.

Seguridad de acceso al servidor.

Para que solo los usuarios asignados puedan acceder a las configuraciones del mismo ya sea un acceso por SSH o por WEB.

Herramienta para monitorear y analizar intentos de accesos fallidos.

Basado en los log del sistemas podremos bloquear cualquier intento de acceso con reiteradas autenticaciones fallidas, con esto evitaremos ataques de acceso, ya sea por diccionario o fuerza bruta.

Aseguramiento de SIP.

Necesarios para no ser víctimas de ataques de fuerza bruta con diccionario a las extensiones configuradas en nuestro servidor.

Diseño e Implementación de Políticas de Firewall.

Para que solo los usuarios de la red interna puedan acceder a los recursos del servidor.

Hardware

Para el servidor IP que contendrá el Sistema Asterisk se tendrá en consideración las siguientes características, como muestra la tabla 1:

Procesador	Intel Core i3
Memoria RAM	4GB
Disco Duro	250GB
1 Tarjetas de Red	1Gbps

Tabla 1: Requisitos del Servidor.

Software

En la central telefónica se ejecutará bajo AsteriskNow 6.12 de 32 bits, dentro de este paquete usaremos la versión de asterisk11 de código abierto.

FreePBX versión 12 para la configuración por medio de la interfaz gráfica.

Los equipos de los clientes contarán con programas que simulan un teléfono IP llamado Softphone estos pueden ser 3CX o X-LITE.

Para la seguridad: Iptables y Fail2ban. Como podemos ver, en la Tabla 2 podemos ver las extensiones y la Tabla 3 el direccionamiento.

Extensiones

Usuario	Extensión
Gerente General	402
Gerente Financiero	403
Coordinador	301
Recepción	101
Ventas1	201
Ventas2	202
Asistente Gerencial	401
Operador 1	302
Operador 2	303
Operador 3	304
Operador 4	305
RRHH	306

Tabla 2: Extensiones.

Direccionamiento IP

Equipo	IP
Gerente General	192.168.0.10
Gerente Financiero	192.168.0.11
Coordinador	192.168.0.12
Recepción	192.168.0.13
Ventas1	192.168.0.14
Ventas2	192.168.0.15
Asistente Gerencial	192.168.0.16
Operador 1	192.168.0.17
Operador 2	192.168.0.18
Operador 3	192.168.0.19
Operador 4	192.168.0.20
RRHH	192.168.0.21
Servidor Asterisk	192.168.0.235
Red Inalámbrica	192.168.0.100-110
Soporte Externo	192.168.0.5 192.168.0.6

Tabla 3: Direccionamiento IP.

2.2.2. Configuración del Servidor Asterisk

Asterisk

Para modificar el mensaje de bienvenida editaremos el archivo de configuración siguiente MOTD.py que está en:

/usr/local/sbin/MOTD.py

Ahora lo modificaremos de modo que quede como rn la Figura 2.3:

Figura 2.3: Mensaje de Bienvenida de Acceso.

FreePBX

Para configurar las extensiones de los respectivos usuarios o clientes del servidor, ingresamos a la interfaz gráfica “FreePBX” [2] usando un cualquier navegador web con la siguiente dirección:

https://192.168.0.235

La primera ventana que nos mostrará será para establecer el usuario y la contraseña de inicio de sesión de la interfaz web, como la Figura 2.4.

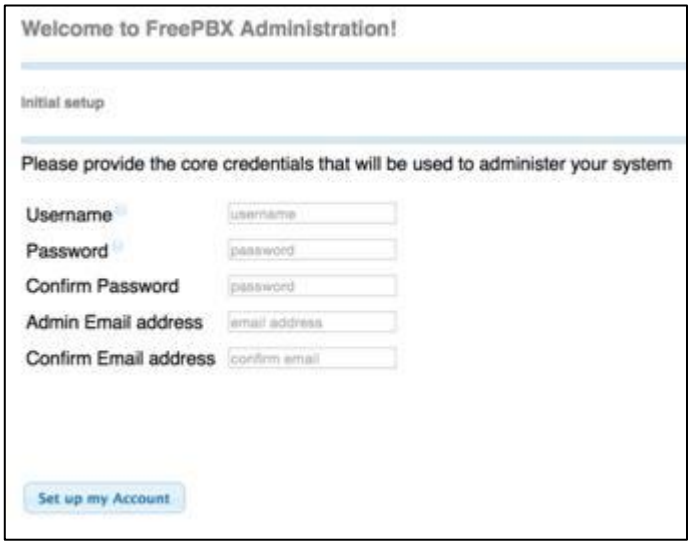


Figura 2.4: Configuración FreePBX parte 1.

Creamos un usuario con el que vamos a conectarnos cada vez que necesitemos configurar alguna extensión o realizar alguna tarea específica en el servidor de telefónica. Por motivos de seguridad lo recomendable es crear un usuario que no sea común y una definir una clave que por lo menos 10 caracteres combinando letras mayúsculas, minúsculas, símbolos y números como la Figura 2.5.

Figura 2.5: Configuración FreePBX parte 2.

Al establecer las credenciales se reinicia y nos pedirá la contraseña, en cual la ingresaremos para ingresar al servidor, como se muestra en la Figura 2.6.

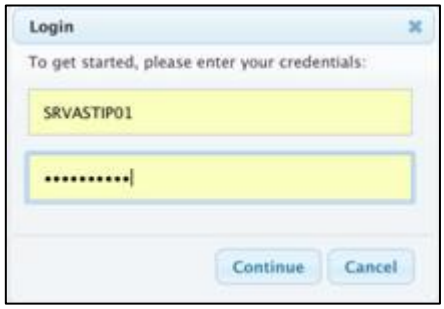


Figura 2.6: Configuración FreePBX parte 3.

Extensiones

Añadimos las extensiones dentro del menú aplicaciones, como se muestra en la Figura 2.7,

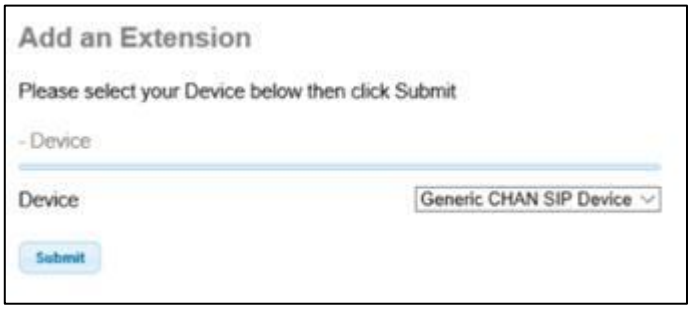


Figura 2.7: Configuración Extensiones parte 1.

Damos clic en **Submit** para agregar una nueva extensión:

Figura 2.8: Configuración Extensiones parte 2.

En la Figura 2.8, nos muestra que estos son los tres parámetros principales para agregar una nueva extensión, donde:

UserExtensión: es el número de la extensión nueva.

DisplayName: es el nombre del usuario que tendrá esa extensión.

Secret: es la contraseña de la extensión que como ya se ha comentado anteriormente debería ser de por lo menos 10 caracteres mínimos combinando letras mayúsculas, minúsculas, símbolos y números una manera de hacer que un atacante le sea difícil averiguar una clave de una extensión. Por defecto no aparece una contraseña establecida de manera opcional al crear una nueva extensión, ilustrado en la Figura 2.9.

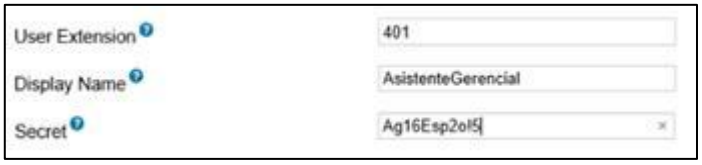


Figura 2.9: Configuración Extensiones parte 3.

Realizamos el mismo procedimiento para cada una de las extensiones como lo muestra la Figura 2.10:

Figura 2.10: Configuración Extensiones parte 4

Softphone

Iniciamos el softphone “X-lite”. Damos clic en el menú Softphone como se muestra en la Figura 2.11:

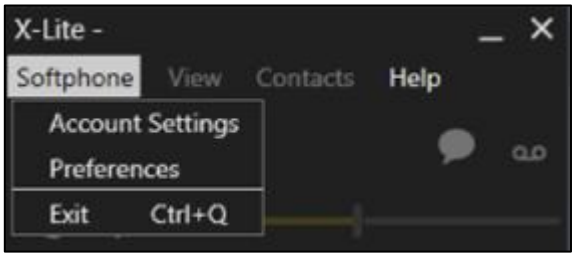


Figura 2.11: Softphone-Xlite parte 1.

Configuramos la cuenta en “AccountSettings”, mostrado en la Figura 2.12

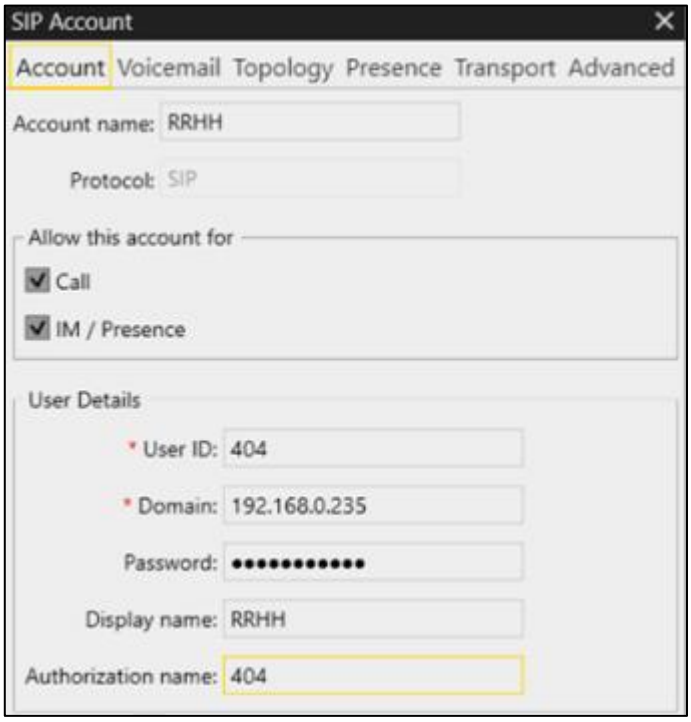


Figura 2.12: Softphone-Xlite parte 2.

UserID: es el número de la extensión.

Domain: es la dirección IP del Servidor de Telefonía IP.

Password: es la contraseña establecida para la extensión.

Display Name: es el nombre de la extensión.

Authorization name: es el número de la extensión autorizada.

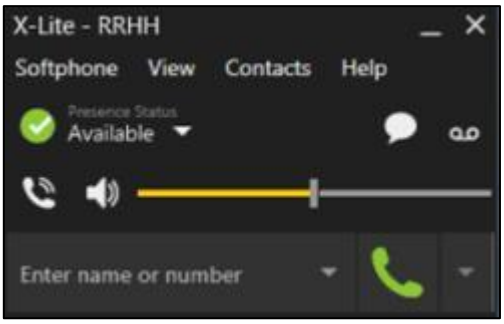


Figura 2.13: Softphone-Xlite parte 3.

La Figura 2.13 nos muestra que se ha establecido correctamente la autenticación del softphone con el servidor VOIP.

2.2.3. Configuración de seguridad para el Servidor Asterisk

Configuración de Acceso

Seguridad en Grub

Para proteger de que un usuario no autorizado modifique la clave del usuario “root”.

Primero creamos una clave cifrada con md5 mostrado en la Figura 2.14:

```
[root@localhost ~]# grub-md5-crypt
Password:
Retype password:
$1$QlerQ$IpKrPySvOTQms1aSltAgK1
[root@localhost ~]# _
```

Figura 2.14: Seguridad en Grub parte 1.

Modificamos el archivo “grub.conf” como se ve en la Figura 2.15

```
[root@localhost grub]# pwd
/boot/grub
[root@localhost grub]# nano grub.conf_
```

Figura 2.15: Seguridad en Grub parte 2.

Agregamos la siguiente línea antes del primer sistema de arranque:

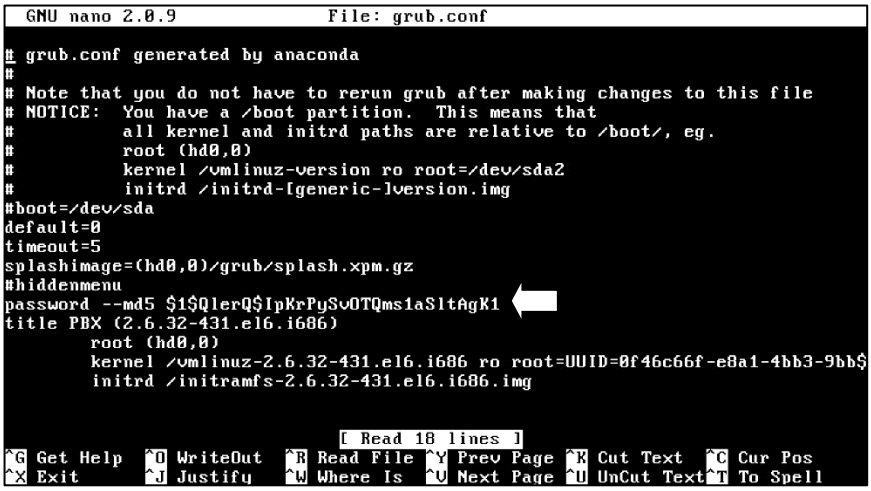


Figura 2.16: Seguridad en Grub parte 3.

Guardamos y cerramos el archivo como se ve en la figura 2.16.

Grupo wheel

El en siguiente fichero vamos a verificar que exista el grupo “wheel” [1], para así crear y agregar un nuevo usuario para el acceso al servidor con privilegios de administrador por medio del comando “su”.

/etc/group

Dentro de este fichero debemos encontrar una línea en que indique el grupo “wheel” y los usuarios que están dentro de dicho grupo, de no encontrarla la agregamos al fichero creando así el grupo.

wheel:*:10:root

Para crear un nuevo usuario y agregarlos dentro de este grupo ingresamos el siguiente comando en el terminal:

useradd –G wheel –m –s /bin/bash –d /home/soportessh01

soportessh01

Luego con el comando “passwd” establecemos una clave para el usuario creado.

passwd soportessh01

Privilegios del comando “su”

En el siguiente fichero se configurará para que solo los usuarios que estén en el grupo “wheel” puedan utilizar el comando “su” y poder tener privilegios de “root”. Para esto vamos a usar una suite de librerías que permiten al administrador del sistema escoger como autenticar a los usuarios y las aplicaciones.

/etc/pam.d/su

En este fichero daremos acceso al comando “su” (“root”) a usuarios sin privilegios que este dentro del grupo “wheel” para esto des comentamos la siguiente línea del fichero:

auth required pam_wheel.so use_uid

Privilegios de Sudo

En el caso de necesitar ejecutar aplicaciones cono “root” lo haremos por medio del comando “sudo”, el cual le daremos al usuario creado permisos para que pueda ejecutar una aplicación determinada.

/etc/sudoers

Editamos este fichero, y al final del mismo digitamos la siguiente línea:

user ALL = NOPASSWD: /usr/sbin/tcpdump

Configuración de SSH

Para configurar los parámetros de seguridad a este protocolo debemos editar el siguiente fichero [1]:

/etc/ssh/sshd_config

Configuramos los siguientes parámetros en la Figura 2.17:

Para que los ajustes sean efectivos debemos reiniciar el servicio para esto ejecutamos el siguiente comando:

/etc/init.d/sshdrestart

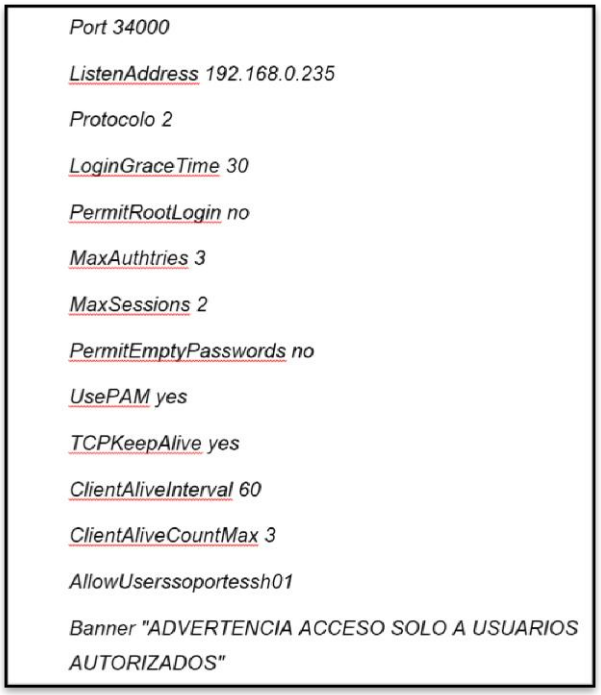


Figura 2.17: Parámetros SSH.

Configuración de Servicios

Una buena práctica en todo servidor es la de desactivar servicios que no son necesarios para el sistema [3], como por ejemplo: ip6tables, KUDZU, ISDN, NETFS, NFSLOCK, PORTMAP, PCGSSD, WANROUTER, VSFTPD, VXINETD, etc.

Para desactivarlos en el arranque usamos el siguiente comando:

chkconfig –-level 345 ip6tables off

Para desactivar los que están en ejecución:

service ip6tables stop

Herramientas de Seguridad

IPTables

Se definieron las siguientes reglas en el servidor [4]:

Borrando Reglas Anteriores

#iptables –F

#iptables -F -t nat

#iptables -X

Política por defecto

#iptables -P INPUT DROP

#iptables -P FORWARD DROP

#iptables -P OUTPUT ACCEPT

Solo tráfico loopback

#iptables -A INPUT -i lo -j ACCEPT

Aceptar todo tráfico de entrada asociado al establecimiento o relacionado a conexión

#iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

Tráfico SSH

#iptables -A INPUT -s 192.168.0.5 -p tcp -i eth0 --dport 34000 -j ACCEPT

#iptables -A INPUT -s 192.168.0.6 -p tcp -i eth0 --dport 34000 -j ACCEPT

Tráfico WEB

#iptables -A INPUT -s 192.168.0.6 -p tcp -i eth0 --dport443 -j ACCEPT

#iptables -A INPUT -s 192.168.0.5 -p tcp -i eth0 –dport443 -j ACCEPT

Aceptando el tráfico SIP

#iptables –A INPUT –s 192.168.0.0/24 –p udp –m udp –i eth0 --dport 5060 –j ACCEPT

#iptables –A INPUT –s 192.168.0.0/24 –p tcp –m tcp –i eth0 --dport 5060 –j ACCEPT

Aceptando el tráfico RTP

#iptables –A INPUT –s 192.168.0.0/24 –p udp –m udp –i eth0 --dport 10000:20000 –j ACCEPT

Aceptando el tráfico ICMP

#iptables –A INPUT –s 192.168.0.0/24 –p icmp –m icmp –-icmp-type echo-request –j ACCEPT

Para ver los logs

#iptables –A INPUT –j LOG

Para Verificar las Reglas

#iptables –L –n –v

Para guardar las reglas

#service iptables save

Fail2ban

El fichero de configuración principal se encuentra en [5]:

/etc/fail2ban/jail.conf

Por lo que haremos una copia del mismo a otro fichero en el cual editaremos con las reglas necesarias:

#cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

Dentro del fichero jail.local vamos a agregar las siguientes líneas que nos permitirán monitorear los logs de cada servicio, ilustrados en la Figura 2.18 y la Figura 2.19:

```
[ssh-iptables]
enable = true
filter = sshd
action = iptables-multiport[name=SSH, port=34000,
protocol=tcp]
logpath = /var/log/secure
maxretry = 3
bantime = 259200

[asterisk-iptables]
enabled = true
filter = asterisk
action = iptables-allports[name=SIP, protocol=all]
logpath = /var/log/asterisk/fail2ban
maxretry = 3
bantime = 259200
```

Figura 2.18: Configuración Fail2ban parte 1.

```
[pbx-gui]
enabled = true
filter = freepbx
action = iptables-allports[name=FreePBX, protocol=all]
logpath = /var/log/asterisk/freepbx_security.log
maxretry = 3
bantime = 259200

[recidive]
enabled = false
filter = recidive
action = iptables-allports[name=recidive, protocol=all]
logpath = /var/log/fail2ban.log*
maxretry = 10
bantime = 345600 ; 4 day
findtime = 86400  ; 1 day
```

Figura 2.19: Configuración Fail2ban parte 2.

El último nos permite bloquear una IP, cuando esta ha tenido reincidencias.

Significado de los parámetros:

- ✓ **[nombre-servicio]:** nos permite establecer el nombre del servicio a monitorear.
- ✓ **enabled**: establece que estará habilitado.
- ✓ **port**: es el puerto que será monitoreado.
- ✓ **filter**: buscara dentro de la carpeta filter.d el script con el cual buscará en los log.
- ✓ **maxretry**: es el número máximo de intentos de accesos fallidos.
- ✓ **bantime**: es el tiempo en segundos que estará bloqueada una IP. En este caso será por 3 días.
- ✓ **logpath**: es la ruta en donde se encuentra el log que analizará fail2ban.
- ✓ **Findtime**: es el tiempo en segundos que se especifica para el número de intentos fallidos.

Reiniciamos el servicio con el siguiente comando:

#service fail2ban restart

Seguridad en SIP

[6] Para asegurar este protocolo debemos seguir las siguientes recomendaciones, como se ve en la Figura 2.20:

- ✓ Asegurar las extensiones estableciendo claves seguras de al menos 10 caracteres usando una mezcla de símbolos, números, mayúsculas y minúsculas.
- ✓ Fijar solo pedidos de autenticación SIP desde la red LAN o redes conocidas.
- ✓ Asegurarse de que el valor “alwaysauthreject = yes”, de esta forma rechazarán pedidos de autenticación fallidos utilizando extensiones válidas, una manera de mitigar ataques de fuerza bruta.

- ✓ El nombre del usuario SIP debe ser diferente a la extensión.
- ✓ Asegurarse de que el valor “allowguest = no”, de esta forma rechazaran llamadas no autenticadas.
- ✓ Se puede fijar en las configuraciones para que una extensión pueda ser usada solo con una determinada dirección IP o denegarla.

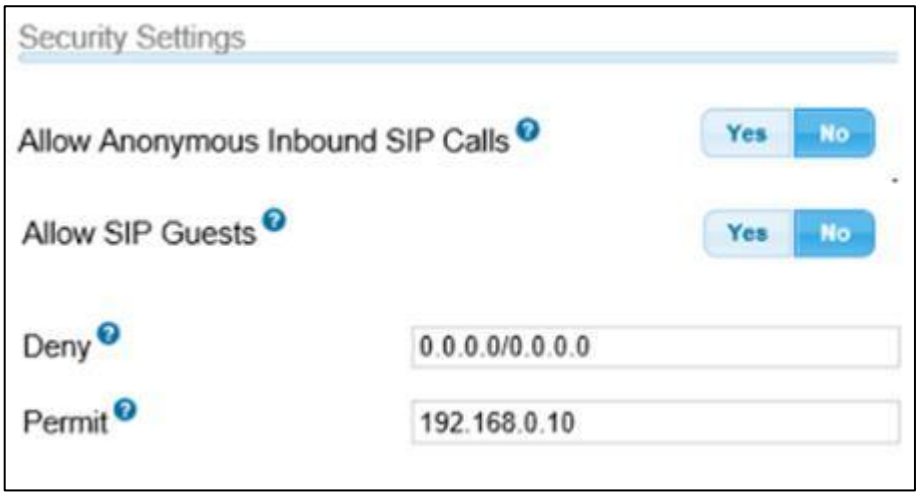


Figura 2.20: Asegurando SIP.

CAPÍTULO 3

3. PRUEBAS Y RESULTADOS.

3.1. Pruebas

Pruebas de llamada

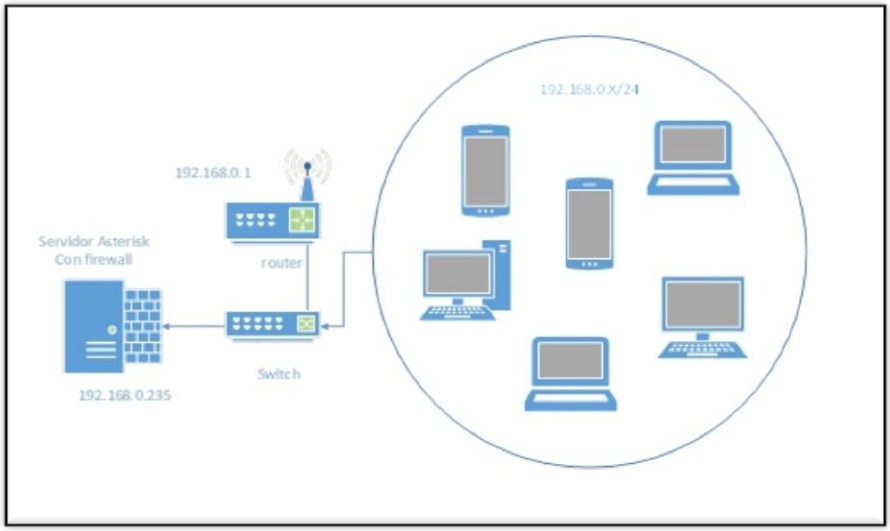


Figura 3.1: Diseño de Red para pruebas de llamada.

Cuando se comenzaron estas pruebas, se tomó en cuenta que los equipos estaban conectados de manera inalámbrica y que las llamadas fueron realizadas por 3 laptops distintas y 3 teléfonos inteligentes algo que hoy en día es muy común de que las empresas den equipos móviles y que el acceso a los recursos sea, de tal manera se podrá tener un escenario actual de la mayoría de empresas, como se puede ver en la Figura 3.1.

En este caso se ha realizado llamadas, se ha medido la capacidad de la red en el servidor, así se puede apreciar que la cantidad de ancho de banda que se va en la llamada, la cual consume una un promedio de 0.2 Mb en la red, tomando en cuenta que la mayoría tiene al uso bajo del servicio, con una buena calidad de comunicación, donde los resultados de la Tabla 4 se ven en la Figura 3.2.

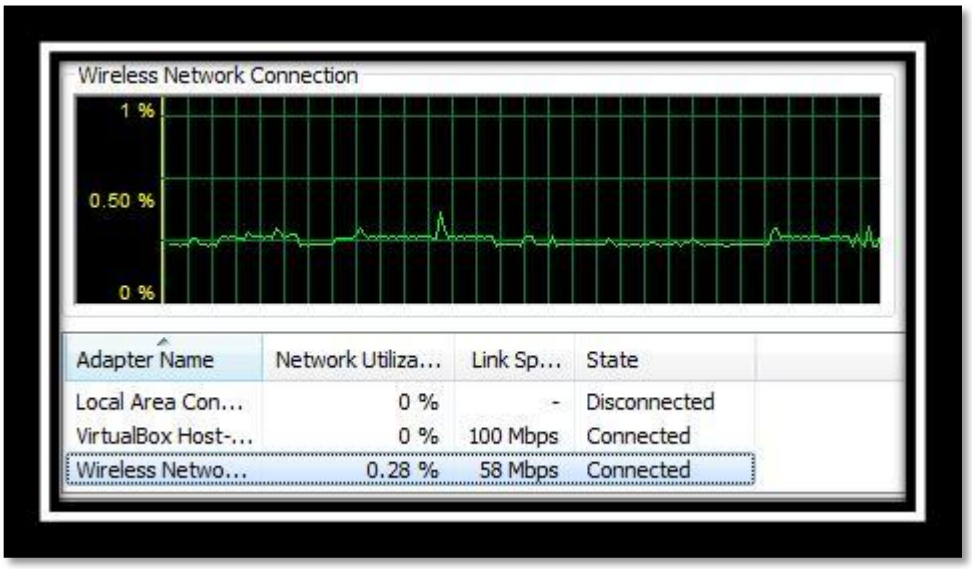


Figura 3.2: Consumo de Llamada

	%	Mb	Kb
Consumo mínimo por llamada	0.28%	0.1484	151.96
Consumo máximo por llamada	1.08%	0.5724	586.14

Tabla 4: Consumo máximo por llamada.

En estos casos también se midió la confiabilidad haciendo un ping constante desde otro equipo en la red, el ping era una señal constante, y tenían un tamaño mucho mayor a lo estándar, con esta señal se realizó igual llamadas para poder apreciar si se sentía alguna baja de la calidad de la voz, la ventaja que nos da la red segura es que igual se le da prioridad a la voz y los procesos como ping o señales que pueden ser consideradas interferencias, como se muestra la Figura 3.3.

Figura 3.3: Confiabilidad de las llamadas.

Ping 192.168.0.235 –t –l 60000

La nueva medición de la red en este caso se pudo hacer con 2 llamadas constantes y con esta señal que se hace desde el otro equipo, muestra que aumento un poco el consumo de la actividad de la red en el servidor, pero igual se puede mantener una buena comunicación, como ya antes mencionado, y se puede trabajar ya que el consumo de los recursos son mínimos, y la de la red también, dándonos a entender que es una solución factible para seguir con un buen desempeño del equipo, en la Figura 3.4.

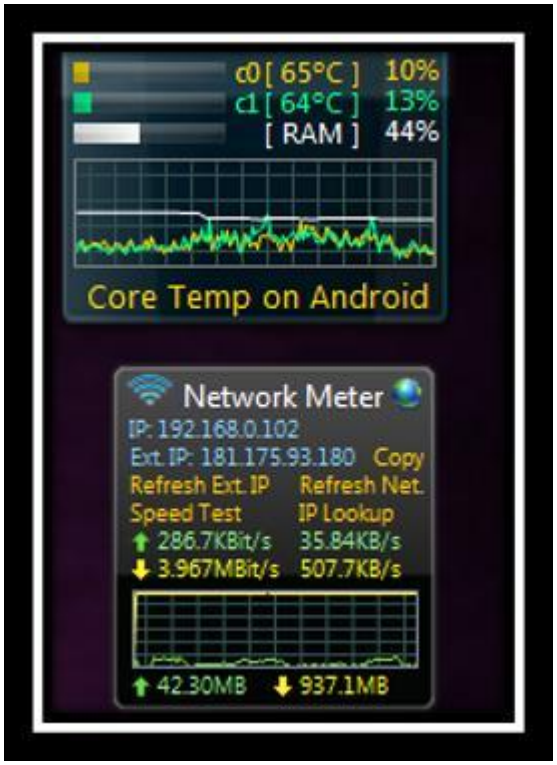


Figura 3.4: Consumo de dos llamadas

Prueba de Seguridad

Descripción

Para las pruebas de seguridad vamos a proponer dos escenario uno con un servidor inseguro y otro con seguridad y fases de los ataques éticos en un ambiente virtual con VMware fusión 7.

Se realizará las pruebas usando el Sistema operativo Kali Linux [7] que cuenta con herramientas que nos permitirán verificar que tan seguro es nuestro servidor ante ataques tales como escaneo de puertos, ataques de fuerza bruta usando diccionarios, así como también ataques a extensiones SIP.

Las herramientas a usar serán las siguientes:

Nmap: permitirá escanear los puertos de un servidor y obtener información necesaria para establecer un tipo de ataque [6].

Hydra: esta herramienta nos permite realizar ataques de fuerza bruta con diccionarios hacia un servicio típicamente al protocolo SSH de acceso remoto [7].

SipVicius: esta herramienta nos permitirá escanear a un servidor en busca de información como las extensiones y las contraseñas del mismo [8].

Putty: una herramienta para la conexión remota a un equipo.

Equipos necesarios:

- ✓ Servidor inseguro con Asterisk: dirección IP 192.168.0.250
- ✓ Servidor Seguro con Asterisk: dirección IP 192.168.0.235
- ✓ Equipo ejecutando Windows.
- ✓ Equipo ejecutando Kali-Linux.

Escenario 1: Servidor sin seguridad

Para los siguientes ataques se ha considerado en un ambiente de pruebas que los mismos se han realizados desde la red interna por equipos que pertenecen a la misma ya sea porque un atacante ha conseguido algún tipo de acceso a unos de estos equipos o por un usuario de la red que quiera irrumpir en la seguridad del mismo, como ilustra la Figura 3.5.

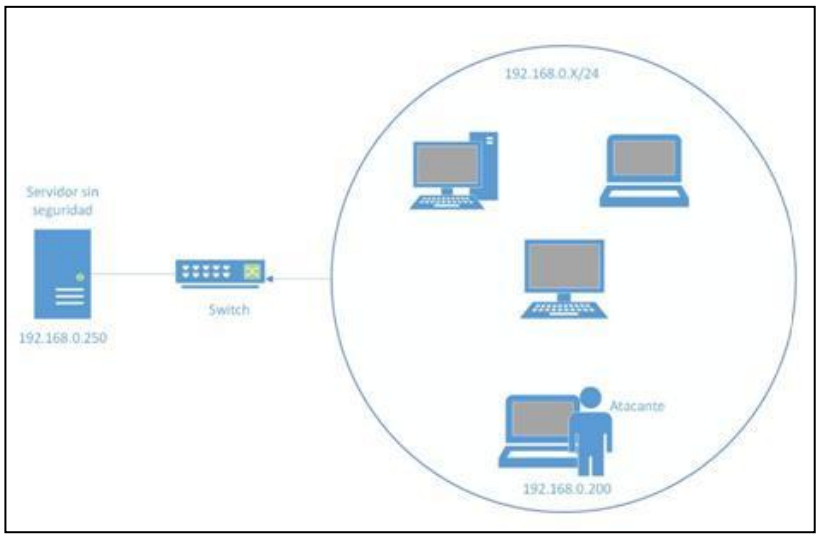


Figura 3.5: Servidor Inseguro Diseño Red de Pruebas.

Fase 1: Escaneo de puertos al servidor

Usando el comando “nmap” en el terminal podemos escanear los puertos de un servidor en este caso usaremos lo siguiente:

nmap –sS -O 192.168.0.250

Figura 3.6: Servidor Inseguro Escaneo de Puertos parte 1.

El resultado de este comando, en la Figura 3.6, nos muestra los puertos su estado y el servicio por lo que podemos ver que por el puerto “22” está ejecutándose el servicio “ssh” y que el estado del mismo es abierto.

Fase 2: Ataque de Fuerza Bruta al protocolo “ssh” del servidor.

Conociendo el puerto podemos realizar un ataque de fuerza bruta por diccionario para averiguar el usuario y la clave de este servicio, para esto usaremos el programa “hydra”.

Archivo que contiene los usuarios, Figura 3.7:

Figura 3.7: Servidor Inseguro Escaneo de Puertos parte 3.

Archivo que contiene las claves, Figura 3.8:

Figura 3.8: Servidor Inseguro Escaneo de Puertos parte 4.

hydra -L users.txt –P lista.txt ssh://192.168.0.250

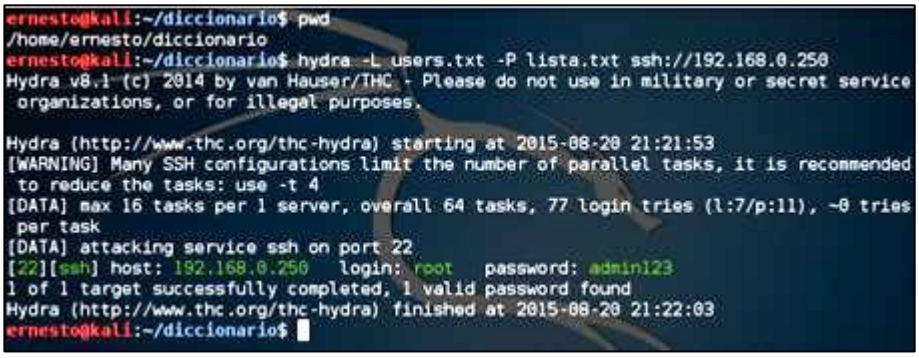


Figura 3.9: Servidor Inseguro Escaneo de Puertos parte 5.

El resultado del comando arrojo, Figura 3.9, el usuario y la contraseña del mismo que es:

Login: root

Password: admin123

Fase 3: Ataque de Fuerza Bruta al protocolo http del servidor.

Con el siguiente comando se intentará obtener una posible clave de acceso hacia la interfaz gráfica de Freepbx:

hydra –l admin –P claves.txt –t 20 –f –vV 192.168.0.250 http-get

El resumen del resultado es el siguiente en la Figura 3.10:

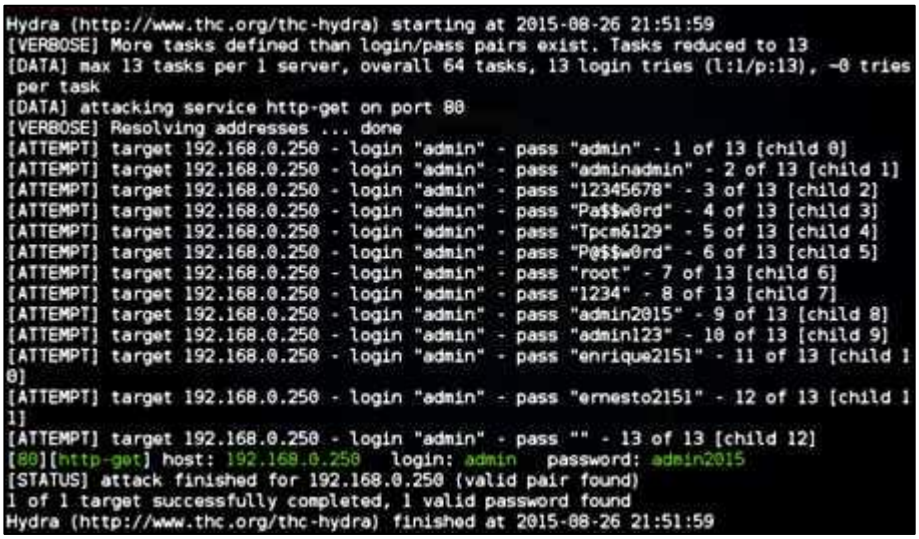


Figura 3.10: Servidor Inseguro Ataque a FreePBX parte 1.

El cual nos muestra la posible clave con la cual podemos intentar acceder por este servicio a las configuraciones del servidor, no es 100% segura pero podemos darnos una idea de cómo podría ser las credenciales. Por otro lado se podría intentar manualmente con contraseñas típicas hasta dar con las credenciales.

Otra manera de hacerlo es por medio de un “sniffer” como Wireshark esta herramienta permite capturar paquetes para un análisis posterior. En este caso nos muestra las credenciales ya que las mismas esta en texto plano sin cifrado por http, como se ve en Figura 3.11.

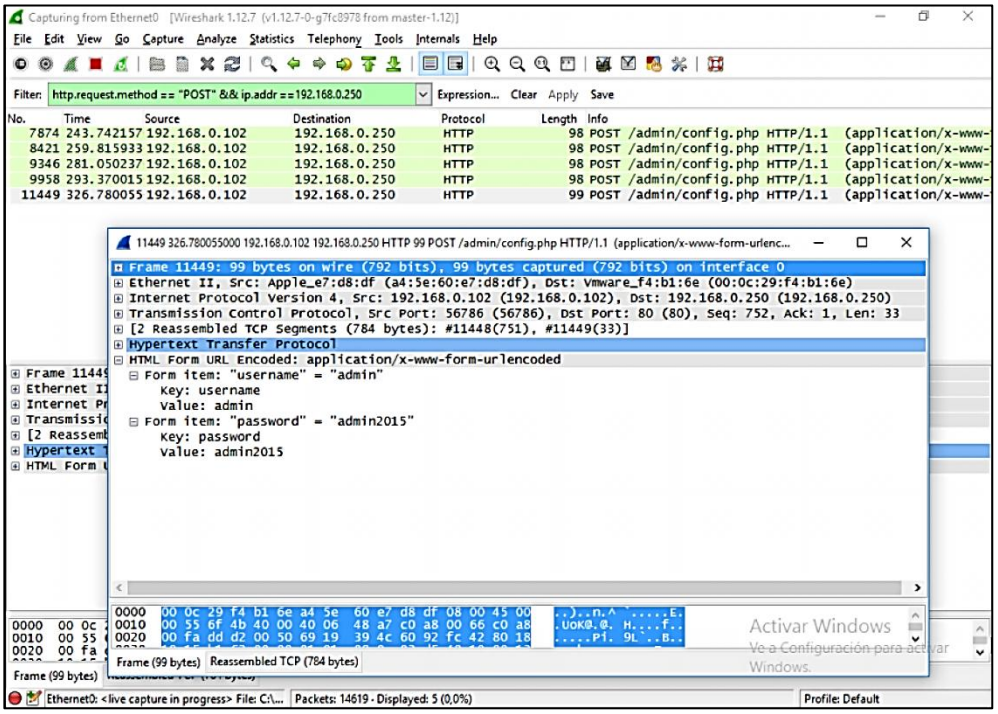


Figura 3.11: Servidor Inseguro Ataque a FreePBX parte 2.

En esta aplicación nos muestra el usuario y contraseña para ingresar a la configuración de servidor por medio de Freepbx.

Fase 4: Escaneo y Ataque a Asterisk

Con el siguiente comando podemos escanear la red ya sea con la IP del servidor, un rango o todo un segmento en busca de información como se aplica en la Figura 3.12:

./svmap.py 192.168.0.0-192.168.0.250

./svmap.py 192.168.0.0/24

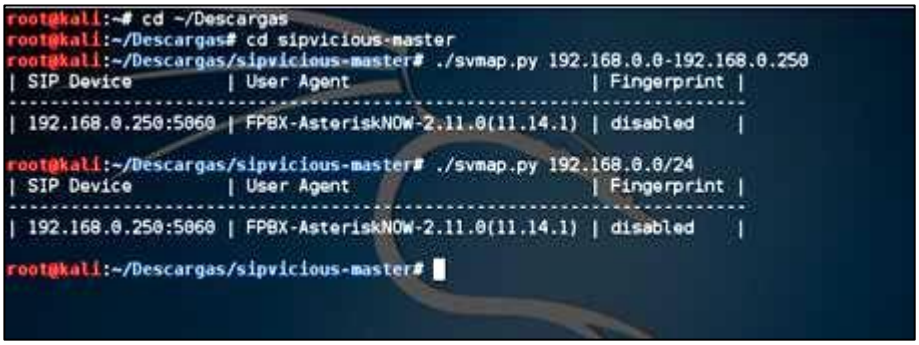


Figura 3.12: Servidor Inseguro Ataque reconocimiento Asterisk.

Una vez identificado la dirección IP del servidor de telefonía, en la Figura 3.13 procedemos a escanearlo en busca de las extensiones que tenga configurado [6]:

./svwar.py 192.168.0.250

Figura 3.13: Servidor Inseguro Ataque enumeración Asterisk.

El resultado del comando anterior nos arrojó que el servidor cuenta con 3 extensiones configuradas por lo cual ahora vamos a hallar la clave de cada extensión. Para esto creamos un archivo de texto llamado “pass.txt”, como se ve en Figura 3.14.

Figura 3.14: Servidor Inseguro Ataque Fuerza Bruta Asterisk parte 1.

Para obtener las claves de cada extensión digitamos el siguiente comando para cada extensión, usando el archivo de texto que contiene la lista de claves posibles [6], como se ve en Figura 3.15.

Figura 3.15: Servidor Inseguro Ataque Fuerza Bruta Asterisk parte 2.

El resultado de cada comando ejecutado nos arroja la contraseña de cada extensión.

Escenario 2: Servidor con Seguridad

Para las pruebas de los siguientes ataques se escogió el mismo escenario anterior aunque la diferencia será que se tendrá las respectivas medidas de seguridad configuradas que se explicaran más adelante.

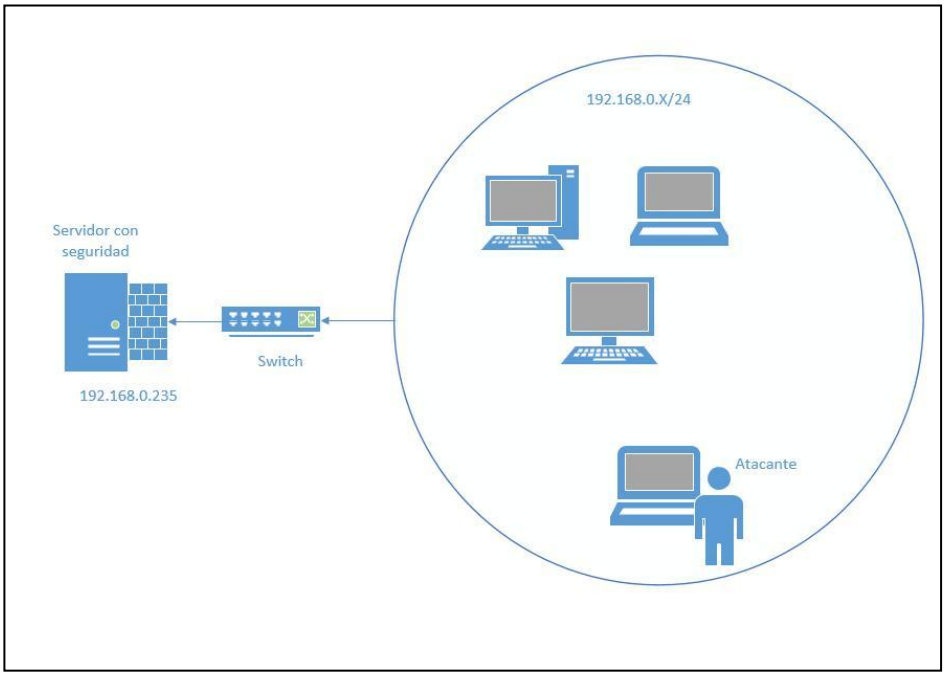


Figura 3.16: Servidor Seguro Diseño Red de pruebas.

En este servidor está configurado y habilitado lo siguiente, tal como en la Figura 3.16:

- ✓ Fail2ban en ssh, asterisk, freepbx bloqueando cada 3 autentificaciones fallidas por un período de 3 minutos en ambiente de pruebas.
- ✓ Reglas de Firewall en el servidor.
- ✓ Bloqueo del usuario “root” para acceso por SSH.

Para verificar que se están aplicando las reglas de seguridad usaremos comandos o estado de los siguientes archivos:

- ✓ #fail2ban-client status
- ✓ #iptables –L

Archivos de eventos (logs)

- ✓ Para SSH: /var/log/secure
- ✓ Para Asterisk: /var/log/asterisk/fail2ban
- ✓ Para FreePBX: /var/log/asterisk/freepbx_security.log

Fase1: Reglas de Firewall de acceso

En este caso se va a probar que las reglas definidas en iptables se estén aplicando correctamente y que se esté filtrando de acuerdo a lo establecido en la implementación.

El acceso por SSH, https al servidor 192.168.0.235 solo estará permitido a las direcciones IP siguientes de la red interna: 192.168.0.5 y 192.168.0.6.

Para las pruebas con ataques de fuerza bruta se lo harán con las direcciones 192.168.0.13 y 192.168.0.21.

Se usará el programa Putty para el acceso remoto y el puerto definido será 34000.

Prueba de seguridad al firewall usando un equipo bloqueado:

IP 192.168.0.254

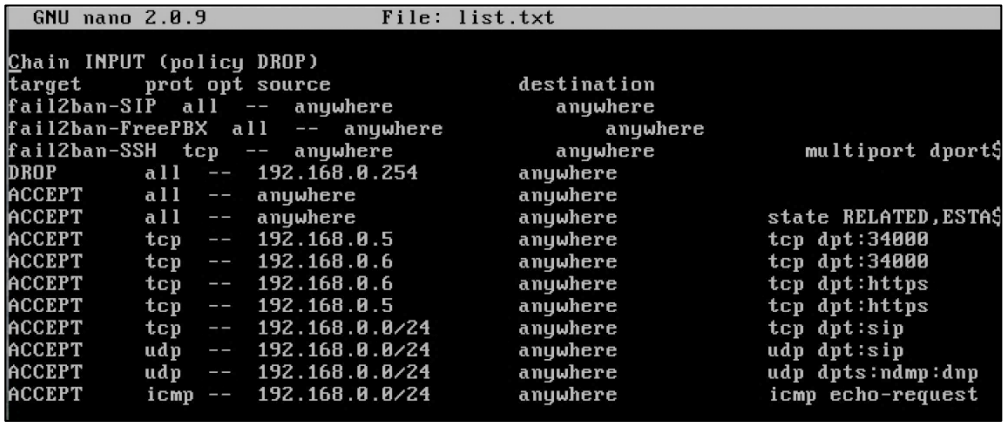


Figura 3.17: Reglas Firewall.

Para esta prueba en la Figura 3.17, se estable en las políticas que la dirección IP 192.168.0.254 no tendrá ningún tipo de acceso con respecto al servidor Voip como una manera de probar como si fuera un atacante externo.

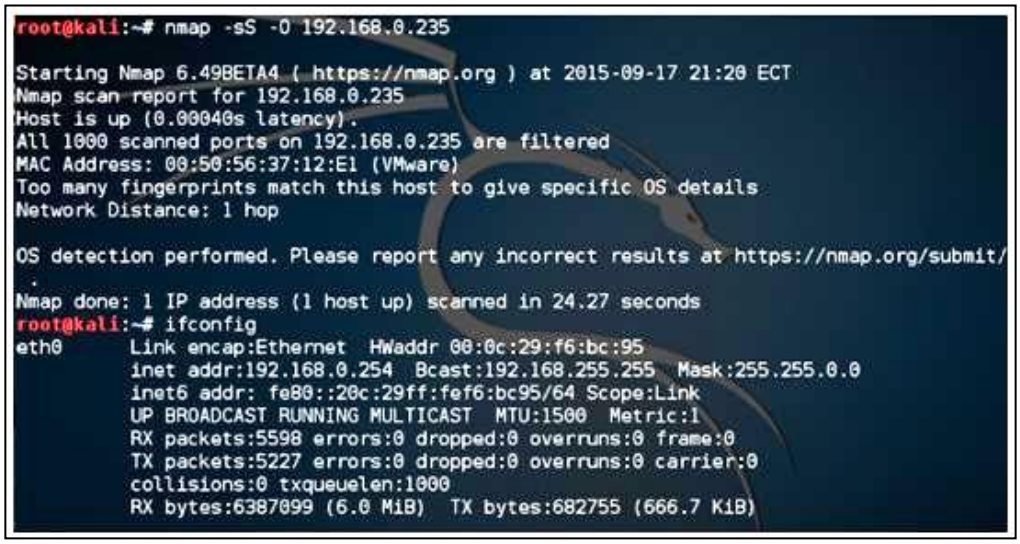


Figura 3.18: Servidor Seguro Ataque Reconocimiento nmap.

La figura 3.18 nos muestra que al realizar un barrido de puertos contra el servidor no encontró ningún puerto debido al bloqueo del firewall, aunque si nos pudo mostrar que el servidor está activo.

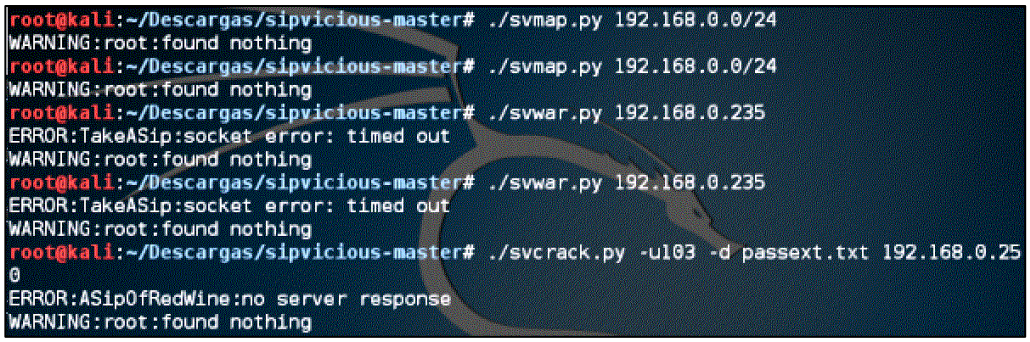


Figura 3.19: Servidor Seguro Ataque a Asterisk.

La figura 3.19 nos muestra que aunque sepa cuál es la dirección IP del servidor no puede encontrar información del mismo por ninguno de los tres ataques demostrando la eficiencia del firewall.

Prueba de acceso por SSH:

Primero probaremos en la Figura 3.20, el acceso con una dirección IP que no tenga permisos.

192.168.0.25

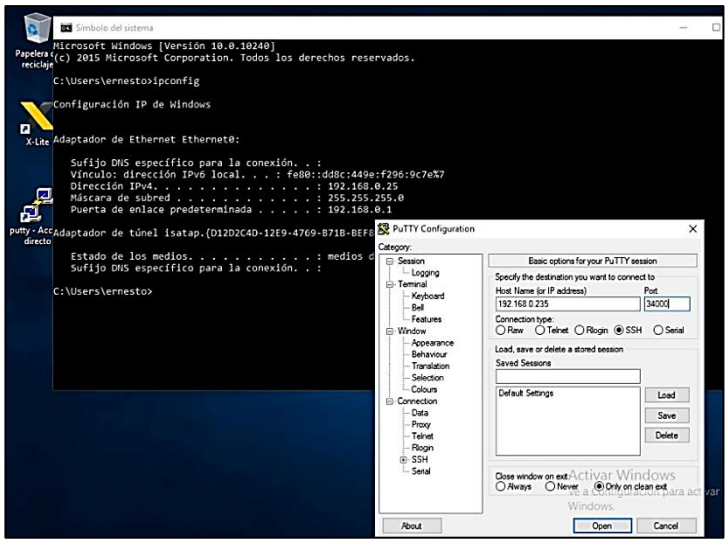


Figura 3.20: Servidor Seguro Firewall Acceso SSH parte 1.

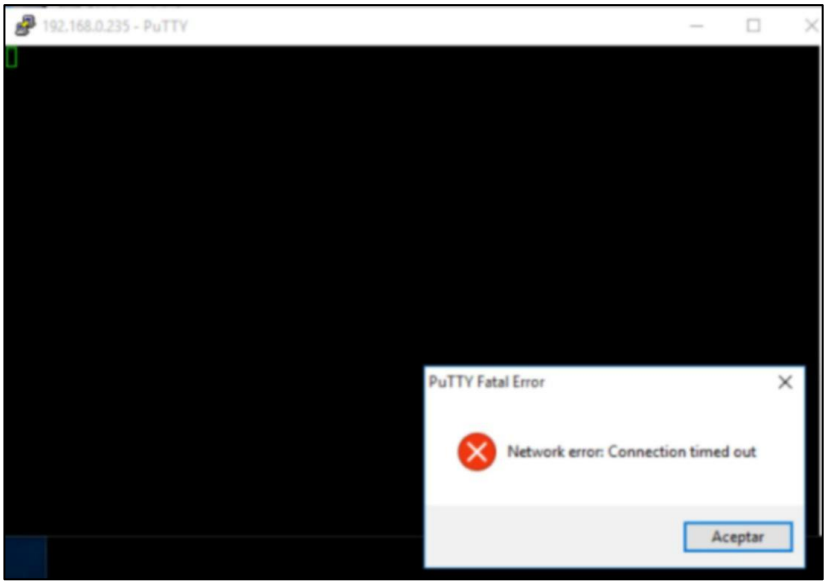


Figura 3.21: Servidor Seguro Firewall Acceso SSH parte 2.

El resultado que se observa en la figura 3.21 nos demuestra que esa dirección IP no tiene permisos para acceder remotamente al servidor.

Ahora probamos con cualquiera de las dos IP que tienen el privilegio, Figura 3.22:

192.168.0.6

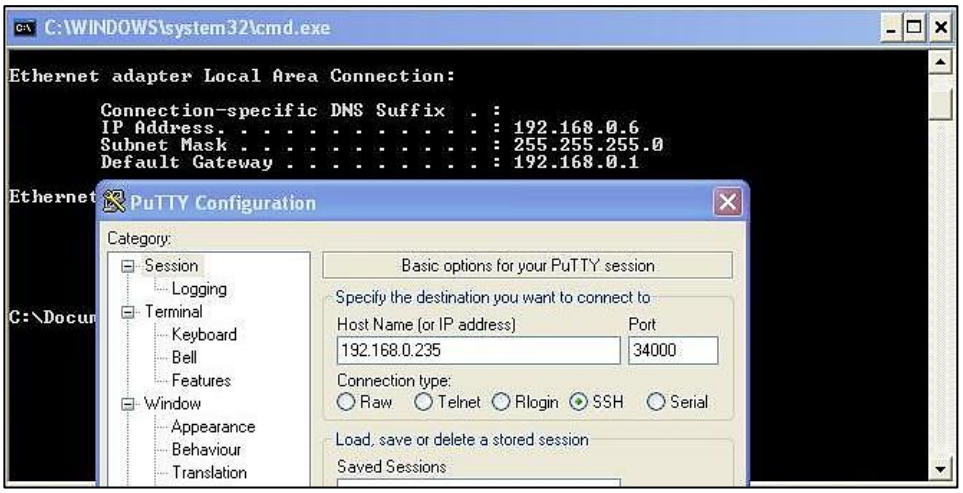


Figura 3.22: Servidor Seguro Firewall Acceso SSH parte 3.

Figura 3.23: Servidor Seguro Firewall Acceso SSH parte 4.

En la Figura 3.23,se ve que a pesar que tenemos acceso por SSH al servidor no podemos ingresar debido a que tenemos desactivado el inicio por SSH al usuario “root” como medida de seguridad para esto vamos a ingresa con el usuario creado para esta función: “soportessh01”

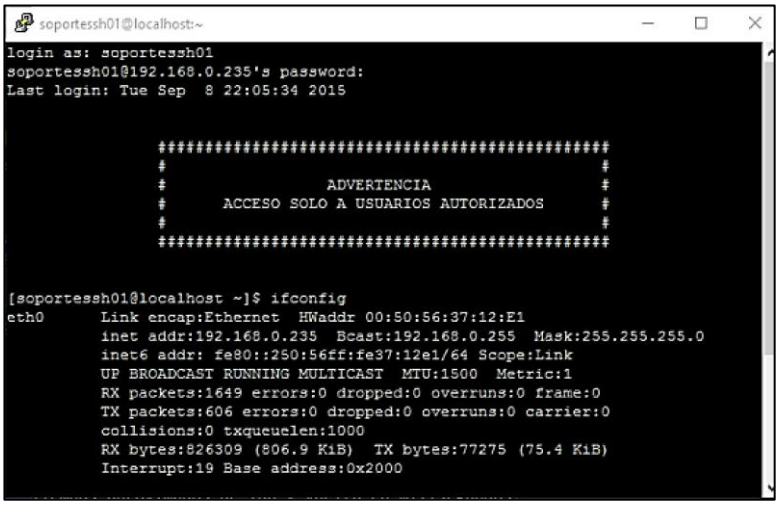


Figura 3.24: Servidor Seguro Firewall Acceso SSH parte 5.

Ahora que hemos accedido al servidor en la Figura 3.24 para poder realizar tareas administrativas deberemos cambiarnos a “root” para eso usamos el comando “su” con la respectiva clave del “root”, como se ve en la Figura 3.25.

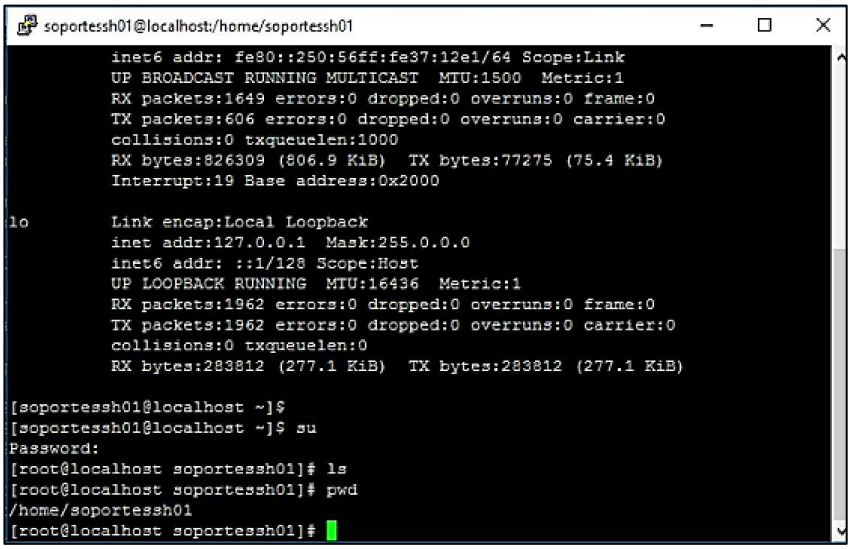


Figura 3.25: Servidor Seguro Firewall Acceso SSH parte 6.

Prueba de acceso por WEB-Freepbx

Primero probaremos el acceso con una dirección IP que tenga permisos, en este caso usaremos la siguiente, Figura 3.26:

192.168.0.6

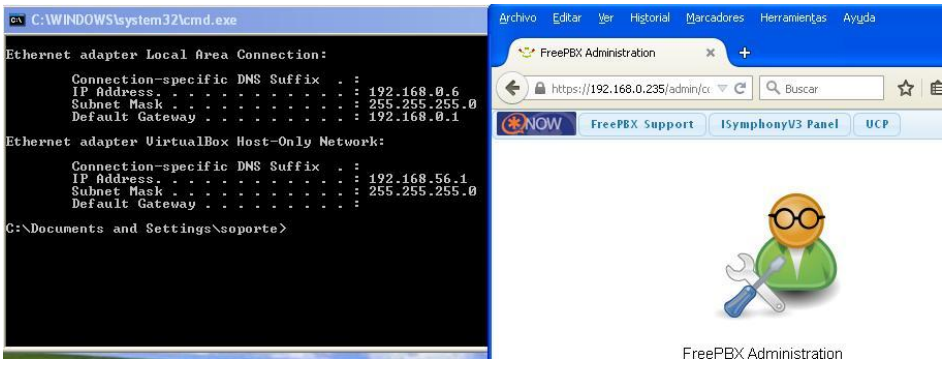


Figura 3.26: Servidor Seguro Acceso Web-FreePBX.

El resultado fue lo esperado, el usuario puede acceder a las configuraciones del servidor por WEB pero solo podrá hacerlo a través de “https” ya que en las reglas del firewall está establecido así.

Fase 3: Ataque por contraseñas

Por medio del comando de la Figura 3.27 y Figura 3.28, podemos ver el estado de Fail2ban:

iptables -L

```
Chain fail2ban-FreePBX (1 references)
target     prot opt source               destination
RETURN     all  --  anywhere             anywhere

Chain fail2ban-SIP (1 references)
target     prot opt source               destination
RETURN     all  --  anywhere             anywhere

Chain fail2ban-SSH (1 references)
target     prot opt source               destination
RETURN     all  --  anywhere             anywhere
```

Figura 3.27: Servidor Seguro Fail2ban Estado 1.

fail2ban-client status

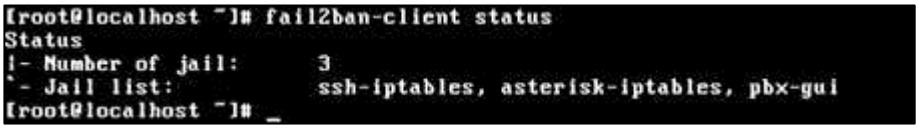


Figura 3.28: Servidor Seguro Fail2ban Estado 2.

Ataque a SSH

Barrido de puertos con el siguiente comando [6] como se ve en la Figura 3.29:

nmap –sS –O 192.168.0.0/24

Figura 3.29: Servidor Seguro Ataque a SSH parte 1.

El resultado del comando anterior nos muestran todos los equipos que están activos en ese momento y su respectiva información.

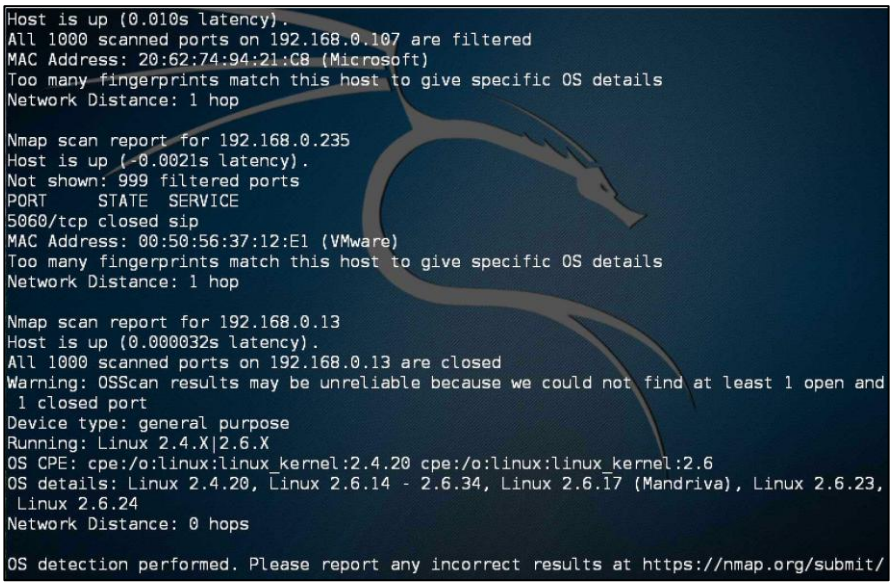


Figura 3.30: Servidor Seguro Ataque a SSH parte 2.

El comando de la Figura 3.30, nos mostrará un resumen de todos los equipos que estén activos en la red que pertenezcan al segmento especificado, cada uno con su respectiva información, en este caso el

servidor es 192.168.0.235 pero no nos muestra el puerto de acceso remoto y esto es debido a que se cambió el puerto por defecto por otro.

Asumiendo que conocemos el puerto podemos realizar un ataque de fuerza bruta para averiguar el usuario y la clave de este servicio, lanzamos el ataque por medio del siguiente comando:

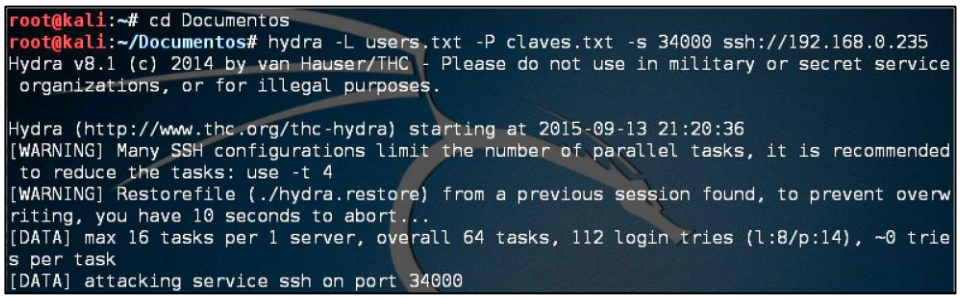


Figura 3.31: Servidor Seguro Ataque a SSH parte 3.

El resultado al realizar el ataque, como se ve en la Figura 3.31, hace que el proceso se paraliza debido a que en el archivo de configuración del servicio SSH “/etc/ssh/sshd_config” se estableció que bloqueará la conexión con un host pasado los 4 intentos fallidos y aunque se lo vuelva a ejecutar ya no podrá atacar porque ha sido bloqueado el host y deberá esperar un cierto tiempo en este caso 3 minutos para volverlo a intentar.

Para mostrar que la IP ha sido bloqueada ingresamos el siguiente comando en la Figura 3.32:

iptables -L

Figura 3.32: Servidor Seguro Ataque a SSH IP bloqueadas.

Ataque a FreePBX

Para este tipo de ataque realizaremos una serie de intentos de accesos a la interfaz gráfica WEB de la página de Freepbx con la finalidad de ver en qué momento seremos bloqueados por el servicio Fail2ban. Usaremos el usuario “srvastpbx01” desde la IP 192.168.0.21, mostrado en la Figura 3.33.

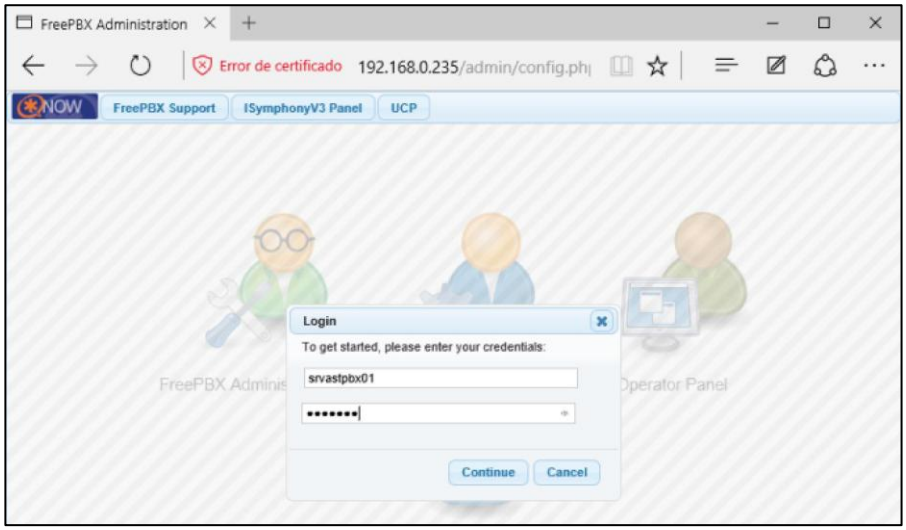


Figura 3.33: Servidor Seguro Ataque a FreePBX parte 1.

Luego de 3 intentos fallidos si intentamos ingresar una vez más nos aparecerá el siguiente mensaje en la Figura 3.34:

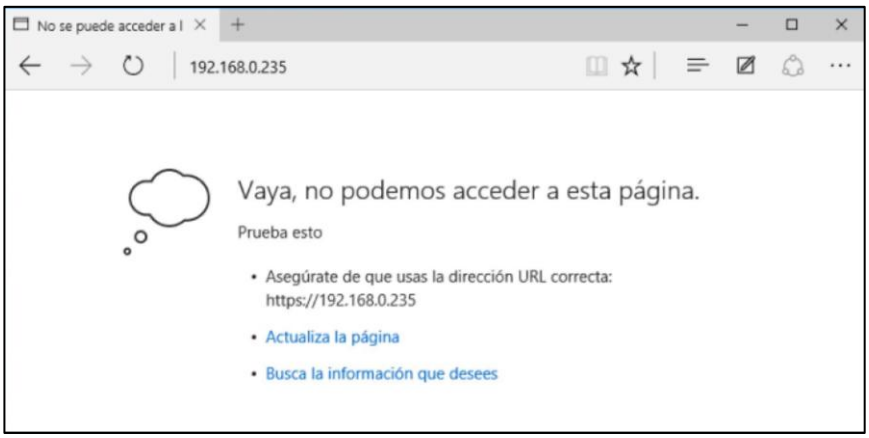


Figura 3.34: Servidor Seguro Ataque a FreePBX parte 2.

Esto es debido a que la dirección IP ha sido bloqueada. Podemos ver la dirección bloqueada por medio del siguiente comando, de la Figura 3.35:

iptables -L

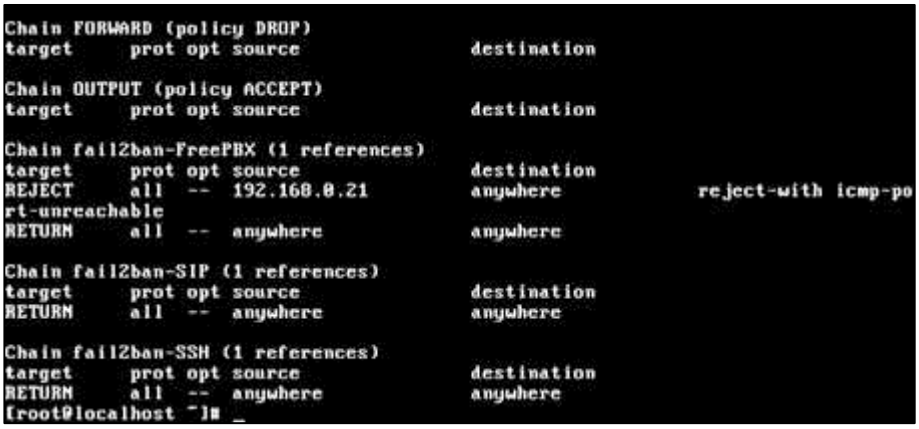


Figura 3.35: Servidor Seguro Ataque a FreePBX parte 3.

Si queremos ver los intentos fallidos, ingresamos al siguiente archivo de la Figura 3.36:

/var/log/asterisk/freepbx_security.log

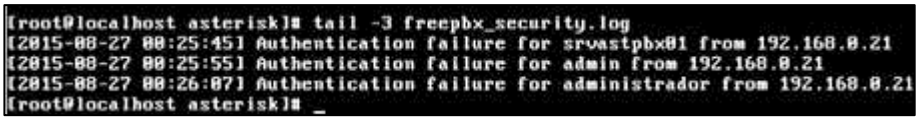


Figura 3.36: Servidor Seguro Ataque a FreePBX parte 4.

Ataque a Asterisk

Buscamos un servidor de telefonía IP en la red por medio del siguiente comando [6]:

./svmap.py 192.168.0.0/24

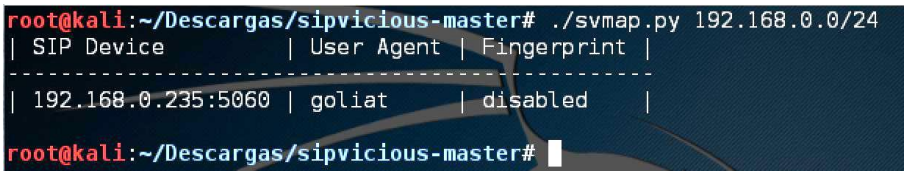


Figura 3.37: Servidor Seguro Ataque reconocimiento Asterisk.

El resultado del comando de la Figura 3.37 nos muestra que en el segmento 192.168.0.0/24 existe un posible servidor de telefonía en donde se está ejecutando una versión de Asterisk.

Una vez identificado la dirección IP del servidor de telefonía procedemos a escanearlo en busca de las extensiones que tenga configurado [6] como en la Figura 3.38 y Figura 3.39:

./svwar.py 192.168.0.235

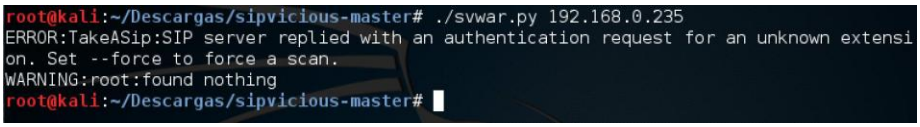


Figura 3.38: Servidor Seguro Ataque enumeración Asterisk parte 1.

./svwar.py –e 100-9999 192.168.0.235

Figura 3.39: Servidor Seguro Ataque enumeración Asterisk parte 2.

Utilizando cualquiera de los dos comandos nos muestra un error en la salida esto es debido a que en el servidor fue configurado el parámetro “alwaysauthreject=yes” el cual previene que el servidor responda al SIP scanner cuales son los números de las extensiones válidas.

Ataque mediante Wireshark

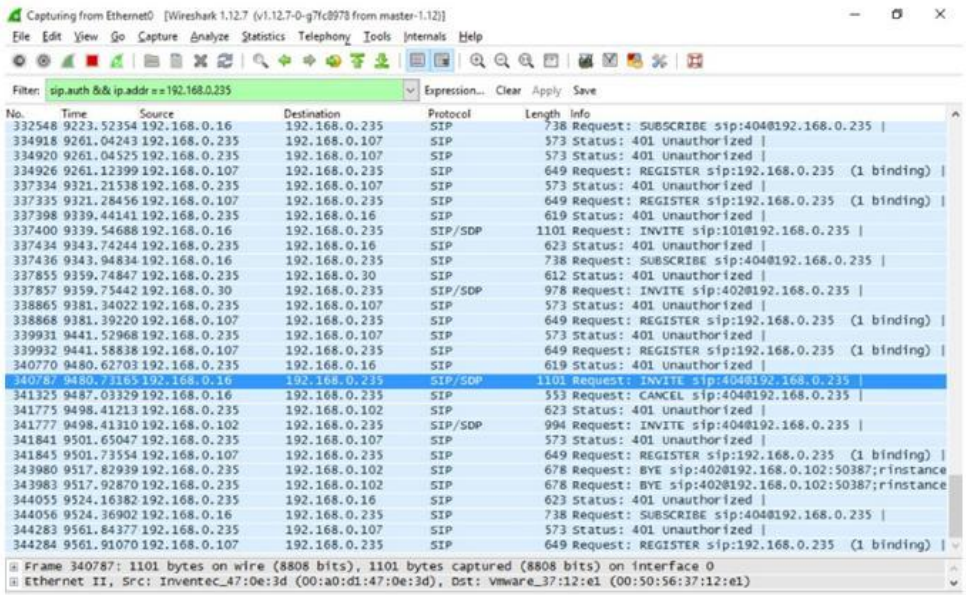


Figura 3.40: Servidor Seguro Ataque enumeración Asterisk parte 3.

En la Figura 3.40 nos muestra equipos de la red que están tratando o estableciendo comunicación con un posible servidor VOIP por medio de mensajes diferentes como REGISTER, INVITE, CANCEL, BYE por medio del protocolo SIP.

Una manera de prevenir seria cifrando la señalización por medio de TLS.

Por este método se podría realizar un ataque de fuerza bruta ya sea por diccionario o por un rango de extensiones con los siguientes comandos o usando un archivo de texto que contiene la lista de claves posibles, como visto en Figura 3.41.

Figura 3.41: Servidor Seguro Ataque fuerza bruta Asterisk parte 1.

Para ver que ha bloqueado la IP ingresamos el siguiente comando mostrado en Figura 3.42:

iptables -L

Figura 3.42: Servidor Seguro Ataque Fuerza Bruta Asterisk parte 2.

Podemos ver los registros en el siguiente fichero en la Figura 3.43:

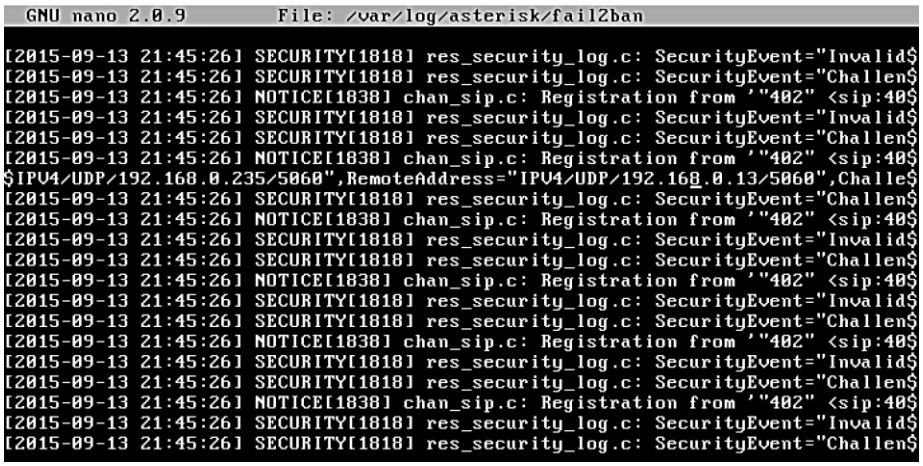


Figura 3.43: Servidor Seguro Ataque Fuerza Bruta Asterisk 3.

Nos muestra los intentos fallidos de autenticación por parte de la dirección 192.168.0.13 a la extensión 402.

3.2. Resultados

En el escenario 1 pudimos constatar los problemas que se pueden llegar a tener por el desconocimiento de mediadas mínimas de seguridad en el servidor VOIP y lo fácil que podría ser para un atacante de hacerse de las credenciales de un determinado servicio así como también de la clave del "root" del servidor que no solo podría crear nuevas extensiones para establecer llamadas gratuitas sino dejar inútil el servidor borrando archivos necesarios para su funcionamiento, mostrados en la Tabla 5 y Tabla 6.

	Reconocimiento	ssh	HTTP(freepbx)
Servidor Sin Seguridad 192.168.0.250	puerto 22 abierto	usuario:root clave:admin123	usuario:admin clave:admin2015

Tabla 5: Servidor sin Seguridad Resultados Ataques Servicios.

	Reconocimiento	Enumeración Extensiones	Fuerza bruta a extensiones
Servidor Sin Seguridad 192.168.0.250	192.168.0.250:5060	1000-1001-1002 las 3 requieren autenticación	recepción1000, ventas1001, coordinador1002

Tabla 6: Servidor sin Seguridad Resultados Ataques Asterisk.

Para el escenario 2, probamos el servidor en varias fases y pudimos notar que aunque el servidor es interno y no es público no está de más establecer algunas medidas mínimas de seguridad al servidor.

Se verificó que la configuración realizada para asegurar uno de los puertos más escaneado, respondió como se esperaba como por ejemplo denegando el acceso al usuario “root” y solo permitiendo su acceso a usuarios autorizados en la Tabla 7.

	Pruebas de Acceso SSH	
IP	Root	soportessh01
192.168.0.6	No accede	Accede

Tabla 7: Servidor con Seguridad Resultado Prueba a SSH

Se demostró que el firewall configurado en el servidor bloqueo a los equipos que no tenían el permiso para acceder a los determinados requerimientos como por ejemplo a servicios de acceso remoto por SSH, acceso a la interfaz gráfica web solo por HTTPS que permite crear las extensiones y configuraciones como se ve en la Tabla 8.

IP	Por SSH	Por HTTPS	Acceso a la red interna
192.168.0.25	No accede	-	-
192.168.0.6	Accede	Accede	-
192.168.0.254	No accede	No accede	No accede

Tabla 8: Servidor con Seguridad Resultado Prueba del Firewall

Las pruebas realizadas con el programa Fail2ban respondieron correctamente para lo que fue configurado bloqueando la conexión a una determinada IP que trataba de hacerse de una credencial mediante el uso de herramientas o de forma manual de uno de los servicios del servidor SSH, HTTPS y extensiones SIP. Estas IP´s fueron bloqueadas en un determinado tiempo de 3 minutos que fue fijado para las de pruebas.

	Reconocimiento	Fuerza bruta a ssh	HTTP(freepbx)
Servidor con Seguridad 192.168.0.235	no muestra los puertos	Se bloqueó la ip: 192.168.0.13	Se bloqueó la ip: 192.168.0.21

Tabla 9: Servidor con Seguridad Resultados Fail2ban Ataques Servicios.

	Reconocimiento	Enumeración Extensiones	Fuerza bruta a extensiones
Servidor con Seguridad 192.168.0.235	192.168.0.235:5060	No se pudo visualizar.	Se bloqueó IP: 192.168.0.13

Tabla 10: Servidor con Seguridad Resultados Ataques a Asterisk.

En las tablas 9 y Tabla 10 podemos visualizar el resultado del ataque usando sipvicious, el cual bloqueo 4 direcciones IP que intentaron autenticarse con credenciales erróneas.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. El impacto que tiene el servicio en la red es manejable, la cual puede ser un proyecto factible para utilizarlo en grandes empresas, donde la cantidad de usuarios crezca mucho más, tomando en cuenta la debida accesibilidad del usuario.

2. Se demostró la importancia de establecer algún tipo de seguridad al implementar un servidor de este tipo basándonos en los resultados obtenidos en un escenario con un servidor configurado por defecto (Escenario 1), el mismo que pudo ser vulnerado mediante ataques de penetración a los diferentes servicios.

3. Basado en los resultados obtenidos en las pruebas realizadas al servidor (Escenario 2), podemos decir que tenemos un sistema capaz de enfrentar diferentes tipos de ataques de accesos y que el mismo será capaz de responder limitando o bloqueando a usuarios no autorizados, pero hay que tener en cuenta de que cada cierto tiempo salen nuevas vulnerabilidades que afectan a los sistemas por lo que se debe tener presente actualizar el mismo con los últimos parches de seguridad.

4. Este proyecto es de bajo costo, debido a que en la solución planteada se utilizó programas de código abierto y que la única inversión para una Pymes será el hardware y la mano de obra.

RECOMENDACIONES

1. Para mitigar ataques físicos al servidor, el mismo debería estar situado en el cuarto de rack con el resto de equipos de redes y protegido por algún medio de seguridad física (cámaras de video, cerrado bajo llave, establecimiento de políticas de acceso).

2. Para mitigar ataques en los cuales se vea involucrado algún equipo de la red interna se recomienda que los mismos tengan instalado un sistema operativo licenciado y actualizado, así mismo un antivirus licenciado y actualizado que cuente con un sistema de prevención de intrusos (IPS) con lo cual se evitará que atacantes externos accedan a la red interna por medio de un equipo afectado.

3. Para cada servicio se debe establecer contraseñas de mínimo 10 caracteres que contengan entre letras mayúsculas, minúsculas, símbolos especiales y números con esto se mitigaría ataques de contraseñas a los servicios del servidor.

4. Para evitar desastres sean estos daños físicos en el equipo o a nivel de software, se recomienda realizar copias de seguridad al servidor estas pueden ser en discos externos como una imagen de todo el disco que contiene la central, así como también el uso de UPS para asegurar el servidor contra problemas eléctricos por cortes de energía.

BIBLIOGRAFÍA

[1] T. B. Christopher Negus, CentOS Bible. USA: John wiley & sons inc, 2009.

[2] N. Andrew. (2015, Mayo 27). FreePBX Distro First Steps After Installation [Online]. Disponible en: http://wiki.freepbx.org/display/PPS/FreePBX+Distro+First+Steps+After+Installation

[3] PaloSanto Solutions. (2006). Seguridad en Implementaciones de Voz Sobre IP [Online]. Disponible en: http://blogs.elastix.org/doc-seguridad/

[4] Creative Commons. (2014, Septiembre 11). IPTables [Online]. Disponible en: https://wiki.centos.org/HowTos/Network/IPTables

[5] DHI Group, Inc. (2013, Mayo 25). MANUAL 0 8 [Online]. Disponible en: http://www.fail2ban.org/wiki/index.php/MANUAL_0_8

[6] N. V. Jose Luis, Hacking y Seguridad VOIP. España: Informatica 64 S.L., 2013.

[7] H. Justin, Kali Linux Network Scanning Cookbook. UK: Packt Publishing, 2014.

[8] Offensive Security. (2015). SIPVicious [Online]. Disponible en: http://tools.kali.org/sniffingspoofing/sipvicious

[9] CNT. (2014). Otros Planes [Online]. Disponible en: https://www.cnt.gob.ec/telefonia/categoria-plan/otros_planes/

[10] Claro. (2013). Planes Comerciales [Online]. Disponible en: http://www.claro.com.ec/portal/ec/sc/corporaciones/telefonia-fija/planes-comerciales/#info_03

ANEXOS

ANEXO A: ÍNDICE DE FIGURAS

Figura 2.1: Metodología del Proyecto. 3
Figura 2.2: Diseño de la Solución. 5
Figura 2.3: Mensaje de Bienvenida de Acceso. 9
Figura 2.4: Configuración FreePBX parte 1. 10
Figura 2.5: Configuración FreePBX parte 2. 10
Figura 2.6: Configuración FreePBX parte 3. 11
Figura 2.7: Configuración Extensiones parte 1. 11
Figura 2.8: Configuración Extensiones parte 2. 11
Figura 2.9: Configuración Extensiones parte 3. 12
Figura 2.10: Configuración Extensiones parte 4 12
Figura 2.11: Softphone-Xlite parte 1. 13
Figura 2.12: Softphone-Xlite parte 2. 13
Figura 2.13: Softphone-Xlite parte 3. 14
Figura 2.14: Seguridad en Grub parte 1. 14
Figura 2.15: Seguridad en Grub parte 2. 14
Figura 2.16: Seguridad en Grub parte 3. 15
Figura 2.17: Parámetros SSH. 17
Figura 2.18: Configuración Fail2ban parte 1. 20
Figura 2.19: Configuración Fail2ban parte 2. 20
Figura 2.20: Asegurando SIP. 22
Figura 3.1: Diseño de Red para pruebas de llamada. 23
Figura 3.2: Consumo de Llamada 24
Figura 3.3: Confiabilidad de las llamadas 25
Figura 3.4: Consumo de dos llamadas 26
Figura 3.5: Servidor Inseguro Diseño Red de Pruebas. 28
Figura 3.6: Servidor Inseguro Escaneo de Puertos parte 1. 29
Figura 3.7: Servidor Inseguro Escaneo de Puertos parte 3. 29
Figura 3.8: Servidor Inseguro Escaneo de Puertos parte 4. 30
Figura 3.9: Servidor Inseguro Escaneo de Puertos parte 5. 30

Figura 3.10: Servidor Inseguro Ataque a FreePBX parte 1. 31
Figura 3.11: Servidor Inseguro Ataque a FreePBX parte 2. 32
Figura 3.12: Servidor Inseguro Ataque reconocimiento Asterisk. 33
Figura 3.13: Servidor Inseguro Ataque enumeración Asterisk..... 33
Figura 3.14: Servidor Inseguro Ataque Fuerza Bruta Asterisk parte 1..... 33
Figura 3.15: Servidor Inseguro Ataque Fuerza Bruta Asterisk parte 2..... 34
Figura 3.16: Servidor Seguro Diseño Red de pruebas. 35
Figura 3.17: Reglas Firewall. 36
Figura 3.18: Servidor Seguro Ataque Reconocimiento nmap. 37
Figura 3.19: Servidor Seguro Ataque a Asterisk. 37
Figura 3.20: Servidor Seguro Firewall Acceso SSH parte 1. 38
Figura 3.21: Servidor Seguro Firewall Acceso SSH parte 2. 38
Figura 3.22: Servidor Seguro Firewall Acceso SSH parte 3. 39
Figura 3.23: Servidor Seguro Firewall Acceso SSH parte 4. 39
Figura 3.24: Servidor Seguro Firewall Acceso SSH parte 5. 40
Figura 3.25: Servidor Seguro Firewall Acceso SSH parte 6. 40
Figura 3.26: Servidor Seguro Acceso Web-FreePBX..... 41
Figura 3.27: Servidor Seguro Fail2ban Estado 1..... 41
Figura 3.28: Servidor Seguro Fail2ban Estado 2..... 42
Figura 3.29: Servidor Seguro Ataque a SSH parte 1..... 42
Figura 3.30: Servidor Seguro Ataque a SSH parte 2..... 42
Figura 3.31: Servidor Seguro Ataque a SSH parte 3..... 43
Figura 3.32: Servidor Seguro Ataque a SSH IP bloqueadas. 43
Figura 3.33: Servidor Seguro Ataque a FreePBX parte 1..... 44
Figura 3.34: Servidor Seguro Ataque a FreePBX parte 2..... 44
Figura 3.35: Servidor Seguro Ataque a FreePBX parte 3..... 45
Figura 3.36: Servidor Seguro Ataque a FreePBX parte 4..... 45
Figura 3.37: Servidor Seguro Ataque reconocimiento Asterisk. 46
Figura 3.38: Servidor Seguro Ataque enumeración Asterisk parte 1. 46
Figura 3.39: Servidor Seguro Ataque enumeración Asterisk parte 2. 46
Figura 3.40: Servidor Seguro Ataque enumeración Asterisk parte 3. 47
Figura 3.41: Servidor Seguro Ataque fuerza bruta Asterisk parte 1. 48

Figura 3.42: Servidor Seguro Ataque Fuerza Bruta Asterisk parte 2. 48
Figura 3.43: Servidor Seguro Ataque Fuerza Bruta Asterisk 3. 48

ANEXO B: ÍNDICE DE TABLAS

Tabla 1: Requisitos del Servidor……..……...6
Tabla 2: Extensiones……..…….....7
Tabla 3: Direccionamiento IP……..8
Tabla 4: Consumo máximo por llamada…….....24
Tabla 5: Servidor sin Seguridad Resultados Ataques Servicios…….….49
Tabla 6: Servidor sin Seguridad Resultados Ataques Asterisk……....49
Tabla 7: Servidor con Seguridad Resultado Prueba a SSH……..……..…50
Tabla 8: Servidor con Seguridad Resultado Prueba del Firewall……..……..…50
Tabla 9: Servidor con Seguridad Resultados Fail2ban Ataques Servicios…….…....51
Tabla 10: Servidor con Seguridad Resultados Ataques a Asterisk…….....51

ANEXO C: Costos del Proyecto

Costos Equipos

Cantidad	Equipos	Precio	Total
1	Servidor Asterisk	$ 400	$ 400
15	Audífono + Micrófono	$ 17	$ 255
1	PatchCord	$ 5	$ 5
			$660

Costos Operación

Mano de Obra	Costo
Configuración del Servidor IP Asterisk y en equipos de usuario final.	$ 900
Configuración de Seguridad al Servidor IP	$ 1,200
Total	$ 2,100

Costos Totales

Costo total de Operación	$ 2,760

Los costos totales del proyecto aproximadamente serían de $ 2,760 dólares sin contar los teléfonos IP que de ser requeridos por parte del cliente se los añadiría al valor final del proyecto cada uno de estos estaría costando aproximadamente $ 70 como precio base.

ANEXO D: Análisis de Costos de PBX-IP conectada a troncales SIP con respecto a troncales E1(valores aproximados).

Costos Troncales Hardware.

	Troncales SIP	Troncales E1
Tarjeta de red Gbps	$ 30	-
Tarjeta E1 PCIe	-	$ 900

La tabla nos muestra la reducción de costos de Hardware adicional que se necesita para poder conectar el servidor de telefonía IP a la red de telefonía pública. En este caso para conectarse a la PSTN con una troncal SIP solo es necesario una tarjeta de red, la cual tiene un costo relativamente bajo en comparación de una tarjeta digital E1.

Costos Troncales Planes

	Inscripción		Tarifa Mensual	
	Troncales SIP	Troncales E1	Troncales SIP	Troncales E1
5 Canales	$150	$1,800	$60	$300
10 Canales	$150	$1,800	$120	$300

En esta tabla se muestra la diferencia de costos entre E1 y SIP por canales para la comunicación con la PSTN dichas tarifas están sin IVA, una vez más los costos de troncales SIP son menores que los de troncales E1. Para una pymes con pocos usuarios se podría tener unos 10 canales aproximadamente, siendo la troncal SIP una solución más recomendada para una empresa pequeña.

Nota: Todos los datos de los costos de troncales E1 y SIP, fueron obtenidos de la siguiente página web, como una referencia de los costos de este tipo de tecnología.[9]

ANEXO E: Costos Troncales Por Proveedor SIP

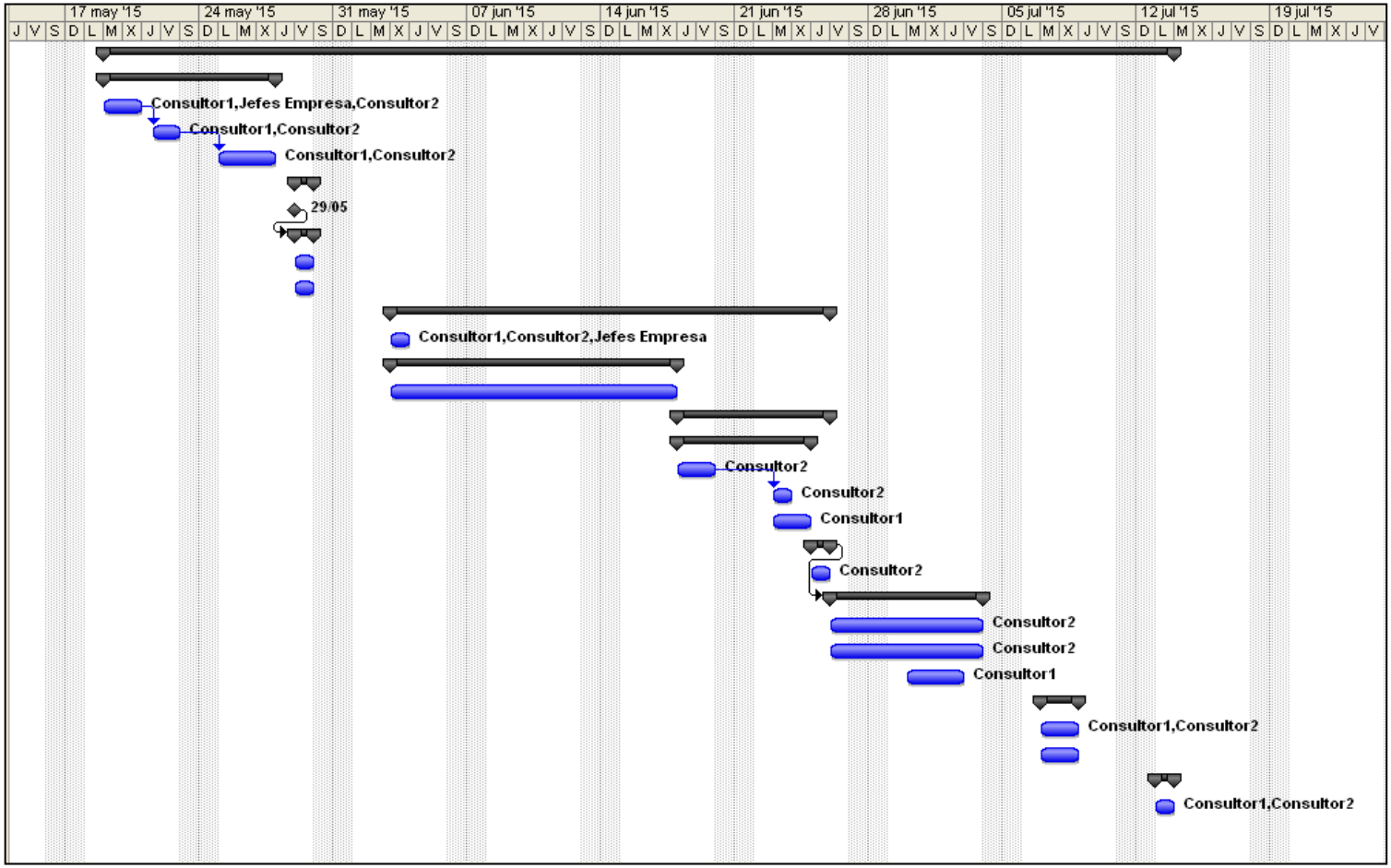
[10]

	Troncales SIP			
	CNT	Claro	CNT	Claro
Canales	5	5	10	10
Costo Mensual	$ 60	$ 35	$ 120	$ 70
Instalación	$ 150	$ 300	$ 150	$ 600
Servicio Nacional, Celular	Si	Si	Si	Si

ANEXO F: Plan de Trabajo

Diagrama de Gantt

	Nombre de tarea	Duración	Comienzo	Fin	Predec	Nombres de los recursos
1	**Solución de telefonía IP de bajo costo aplicando Hardening**	**40 días?**	**mar 19/05/15**	**lun 13/07/15**		
2	**Datos Iniciales**	**7 días?**	**mar 19/05/15**	**mié 27/05/15**		
3	Reunión con Directivos de la Empresa	2 días?	mar 19/05/15	mié 20/05/15		Consultor1,Jefes Empresa,Consultor2
4	Requerimientos	1,5 días?	jue 21/05/15	vie 22/05/15	3	Consultor1,Consultor2
5	Realización de la Propuesta de Diseño y Solución	3 días?	lun 25/05/15	mié 27/05/15	4	Consultor1,Consultor2
6	**Presentación de Solución con Directivos de la Empresa**	**1 día?**	**vie 29/05/15**	**vie 29/05/15**		**Consultor1,Jefes Empresa,Consultor2**
7	Presentación del Diseño	0 días?	vie 29/05/15	vie 29/05/15		
8	**Presentacón del Plan de Costos totales**	**1 día?**	**vie 29/05/15**	**vie 29/05/15**	**7**	
9	Costo Total del Proyecto	1 día?	vie 29/05/15	vie 29/05/15		
10	Presentación del Plan de trabajo	1 día?	vie 29/05/15	vie 29/05/15		
11	**Aprobación del Proyecto**	**17 días?**	**mié 03/06/15**	**jue 25/06/15**		
12	Firma del Contrato	1 día?	mié 03/06/15	mié 03/06/15		Consultor1,Consultor2,Jefes Empresa
13	**Recaudación del 70% del costo total del Proyecto**	**11 días?**	**mié 03/06/15**	**mié 17/06/15**		
14	Compra de Equipos	11 días?	mié 03/06/15	mié 17/06/15		
15	**Ejecución del Proyecto**	**6 días?**	**jue 18/06/15**	**jue 25/06/15**		
16	**Configuración de Equipos**	**5 días?**	**jue 18/06/15**	**mié 24/06/15**		
17	Configuración Servidor Asterisk	2 días?	jue 18/06/15	vie 19/06/15		Consultor2
18	Configuración Equipos de usuarios	1 día?	mar 23/06/15	mar 23/06/15	17	Consultor2
19	Configuración de Seguridad para el Servidor	2 días?	mar 23/06/15	mié 24/06/15		Consultor1
20	**Instalación de Equipos en la red**	**1 día?**	**jue 25/06/15**	**jue 25/06/15**		
21	Servidor Asterisk	1 día?	jue 25/06/15	jue 25/06/15		Consultor2
22	**Pruebas**	**6 días?**	**vie 26/06/15**	**vie 03/07/15**	**20**	**Consultor1,Consultor2**
23	Servidor Asterisk	6 días?	vie 26/06/15	vie 03/07/15		Consultor2
24	Equipos clientes del Servidor Asterisk	6 días?	vie 26/06/15	vie 03/07/15		Consultor2
25	Pruebas de Seguridad	3 días?	mar 30/06/15	jue 02/07/15		Consultor1
26	**Mantenimiento**	**2 días?**	**mar 07/07/15**	**mié 08/07/15**		
27	Otros Requerimientos	2 días?	mar 07/07/15	mié 08/07/15		Consultor1,Consultor2
28	Pago del 30 % del costo total del Proyecto	2 días?	mar 07/07/15	mié 08/07/15		
29	**Entrega de Documento**	**1 día?**	**lun 13/07/15**	**lun 13/07/15**		
30	Informes del Proyecto	1 día?	lun 13/07/15	lun 13/07/15		Consultor1,Consultor2



En este Diagrama de Gantt nos muestra el plan de trabajo del proyecto que tiene una duración a aproximada de 40 días y define el proceso de cada etapa en el desarrollo del mismo.