

Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Laboratorio de Comunicación y Redes



**Conjunto de Lineamientos
para la Implantación de una
Solución VoIP Segura**

Trabajo Especial de Grado
presentado ante la Ilustre
Universidad Central de Venezuela
por el Bachiller:

Amelia M. Araneo M.
V-19.874.251
aaraneo7@gmail.com

para optar al título de Licenciado en Computación

Tutores: Prof. Dedaniel Urribarri y Prof. Eric Gamess

Caracas, Octubre 2015



Universidad Central de Venezuela
Facultad de Ciencias
Escuela de Computación
Laboratorio de Comunicación y Redes



ACTA DE VEREDICTO

Quienes suscriben, miembros del jurado designado por el Consejo de Escuela de Computación, para examinar el Trabajo Especial de Grado, presentado por la Bachiller Amelia Araneo, identificada con la cédula de identidad V-19.874.251, con el título **“Conjunto de Lineamientos para la Implantación de una Solución VoIP Segura”**, a los fines de cumplir con el requisito legal para optar al título de Licenciado en Computación, dejan constancia de lo siguiente:

Leído el nombrado trabajo por cada uno de los miembros del jurado, éste fijó el día 15 de octubre de 2015, para que su autora lo defendiera en forma pública, en el laboratorio Internet II de la Escuela de Computación, mediante una exposición oral de su contenido, luego de la cual respondió satisfactoriamente a las preguntas que le fueron formuladas por el jurado, todo ello conforme a lo dispuesto en la Ley de Universidades y demás normativas vigentes de la Universidad Central de Venezuela.

Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobarlo.

En fe de lo cual se levanta la presente acta, en Caracas a los quince días del mes de Octubre del año dos mil quince, dejándose constancia que actuaron el Profesor Eric Gamess actuó como coordinador de jurado.

Prof. Eric Gamess
Tutor

Prof. Dedaniel Urribarri
Tutor

Prof. Fernando Crema
Jurado Principal

Prof. Francisco Sans
Jurado Principal

Dedicatoria

Este Trabajo Especial de Grado lo dedico con amor y cariño a mis padres, a mi hermano y a mi nonna. Su amor y fortaleza me hicieron siempre querer ser una mejor persona.

Agradecimientos

Quiero agradecer especialmente ese Trabajo Especial de Grado a mis padres, mi hermano y a todas aquellas personas que me ayudaron en su realización.

Agradezco a mi madre María por siempre ayudarme, estar conmigo en todo momento y por siempre querer lo mejor para mí.

A mi padre Roberto por nunca dejarme sola, por siempre creer en mí y por sus buenos consejos.

A mi hermano Roberto que siempre me ayudó en lo que pudo.

A mi tutor Eric Gamess por ayudarme mucho, apoyarme, darme buenos consejos y por ser un excelente profesor.

A mi tutor Dedaniel Urribarri por brindarme la oportunidad de realizar el Trabajo Especial de Grado en su empresa y por ayudarme.

A mi querida Universidad Central de Venezuela por permitirme estudiar allí y formarme como profesional.

Por último quiero agradecer a todas aquellas personas que a lo largo de mi carrera compartieron especiales momentos conmigo, me hicieron reír, me ayudaron y me hicieron crecer como persona.

Amelia Araneo.

Resumen

TÍTULO

Propuesta de un Conjunto de Lineamientos para la Implantación de una Solución VoIP Segura.

AUTOR

Amelia M. Araneo M.

TUTORES

Prof. Dedaniel Urribarri y Prof. Eric Gamess.

Resumen

En este Trabajo Especial de Grado, se propone un conjunto de lineamientos para implantar de forma más segura soluciones VoIP en escenarios controlados.

VoIP es una de las tecnologías que está actualmente disponible en el mundo para la transmisión del tráfico de voz y video sobre una red IP. Esta permite que las llamadas telefónicas sean soportadas sobre las redes de datos ya existentes en lugar de utilizar una red paralela o líneas telefónicas tradicionales, lo cual ofrece muchos beneficios a los proveedores de servicios así como a los usuarios finales.

La seguridad de las comunicaciones y la telefonía IP no es una tarea fácil. Al apoyarse en otras capas y protocolos existentes de las redes de datos, VoIP hereda ciertos problemas de seguridad de las capas y protocolos ya existentes. Muchos de los ataques hacia las redes VoIP tienen como objetivo principal el hardware y software de los dispositivos VoIP debido a que son tan vulnerables como lo es el sistema operativo o el firmware que ejecutan. Diferentes protocolos para señalización son utilizados por VoIP, siendo SIP uno de los más usados actualmente por sus ventajas en cuanto a servicios ofrecidos y por ser un protocolo abierto. El problema de su utilización está en que a pesar de las ventajas que este protocolo ofrece, hay que estar consciente de los riesgos que conlleva su uso, dado que posee grandes vulnerabilidades en cuanto a su seguridad y por tanto está propenso a diferentes ataques. Si bien este problema en muchos casos no parece preocupar demasiado al usuario, garantizar la seguridad en el entorno VoIP es fundamental.

Elastix es un software de código abierto para el establecimiento de comunicaciones unificadas (servidor de fax, mensajería instantánea, servidor de correos, video conferencia y voz sobre IP). Kali Linux es una distribución de Linux basada en Debian GNU/Linux, adaptada para ser usada en evaluaciones de seguridad y realizar ataques.

En este trabajo se implantaron 3 escenarios de pruebas, que representan arquitecturas comunes de redes con soporte para VoIP. Se utilizó Elastix 2.4.0 como el servidor de comunicaciones unificadas y la herramienta Kali Linux 1.1.0 para llevar a cabo los ataques que generaron problemas significativos de seguridad en cada una de las arquitecturas propuestas. Finalmente se establece un conjunto de lineamientos para combatir los riesgos de seguridad que se presentaron.

Palabras Claves: Lineamientos, VoIP, Implantación, Seguridad, SIP, Elastix, Kali Linux.

Tabla de Contenido

Índice de Figuras	15
Índice de Tablas.....	19
1. Introducción.....	21
2. El Problema.....	23
2.1 Planteamiento del Problema.....	23
2.2 Justificación del Problema	23
2.3 Objetivos	23
2.3.1 Objetivo General.....	23
2.3.2 Objetivos Específicos	24
2.3.3 Alcances.....	24
3. VoIP	25
3.1 Introducción.....	25
3.2 Evolución.....	25
3.3 Arquitectura	26
3.3.1 Teléfonos IP	27
3.3.2 Gateways y Adaptadores Analógicos	28
3.3.3 Dispositivos GSM/UMTS/HSDPA/HSUPA/LTE.....	30
3.3.4 Softphones	30
3.3.5 Proxys y Enrutadores	31
3.4 Internet Protocol	32
3.5 Protocolos de Transporte.....	32
3.5.1 Transmission Control Protocol.....	32
3.5.2 User Datagram Protocol	33
3.5.3 Stream Control Transmission Protocol.....	34
3.6 Protocolos de Señalización.....	34
3.6.1 Session Initiation Protocol.....	35
3.6.2 H.323.....	35
3.6.3 Inter-Asterisk eXchange	36
3.7 Session Description Protocol	36
3.8 Protocolos de Transporte de Audio y Video en Tiempo Real	36
3.8.1 Real Time Protocol	37
3.8.2 Real Time Control Protocol.....	37
3.8.3 Real Time Streaming Protocol	38
3.9 Resource Reservation Protocol	38
3.10 Codificación de la Voz	38
4. Session Initiation Protocol.....	41
4.1 Introducción.....	41
4.2 Direccionamiento.....	41
4.3 Elementos	42
4.3.1 User Agent	42
4.3.2 Intermediary Entity.....	42
4.3.3 Redirect Server.....	42
4.3.4 Location Server	43
4.3.5 Back-to-Back User Agent.....	43
4.4 Peticiones.....	43
4.5 Respuestas.....	44
4.6 Campos de los Mensajes SIP	46

5. Problemas de Seguridad de SIP y Técnicas de Mitigación.....	49
5.1 Introducción.....	49
5.2 Problemas de Seguridad de SIP.....	49
5.2.1 Accesos Desautorizados y Fraudes.....	50
5.2.2 Ataques de Denegación de Servicio.....	50
5.2.3 Ataques a los Dispositivos.....	52
5.2.4 Ataques y Vulnerabilidades a Nivel de Red IP.....	52
5.2.5 Enumeración y Descubrimiento.....	53
5.2.6 Ataques a Nivel de Aplicación.....	53
5.2.7 Ingeniería Social.....	53
5.2.8 Fuerza Bruta.....	53
5.3 Técnicas de Mitigación.....	54
5.3.1 BackTrack Linux.....	54
5.3.2 BackBox.....	54
5.3.3 Snort.....	55
5.3.4 Kali Linux.....	55
6. Elastix.....	57
6.1 Introducción.....	57
6.2 Historia.....	57
6.3 Arquitectura.....	58
6.4 Características.....	58
6.4.1 General.....	59
6.4.2 PBX.....	59
6.4.3 Fax.....	60
6.4.4 Colaboración.....	60
6.4.5 Mensajería Instantánea.....	60
6.4.6 Email.....	60
7. Arquitectura.....	63
7.1 Introducción.....	63
7.1.1 Conexión a un PSTN.....	63
7.1.2 Conexión a un ITSP.....	63
7.1.3 Conexión Remota a una Red LAN.....	64
7.2 Servidor VoIP.....	64
7.3 Ataques y Vulnerabilidades.....	65
7.4 Especificaciones Técnicas.....	65
8. Trabajos Relacionados.....	67
8.1 Seguridad en Implementaciones VoIP.....	67
8.2 Seguridad en Servidores Elastix.....	68
8.3 Seguridad en el Protocolo SIP.....	68
9. Marco Metodológico.....	69
9.1 Adaptación de la Metodología de Desarrollo.....	69
9.1.1 Análisis de Vulnerabilidades.....	69
9.1.2 Pruebas de Penetración Internas.....	69
9.1.3 Pruebas de Penetración Externas.....	70
9.1.4 Mitigación.....	70
9.1.5 Verificación y Análisis de los Resultados.....	70
10. Ataques y Mitigación.....	71
10.1 Introducción.....	71
10.2 Análisis de Vulnerabilidades.....	71

10.3	Pruebas de Penetración Internas.....	72
10.3.1	Ingeniería Social.....	73
10.3.2	Port Scanning.....	73
10.3.3	Man-in-the-Middle.....	75
10.3.4	Eavesdropping.....	77
10.3.5	Ataque de DoS.....	79
10.3.6	Ataque de Fuerza Bruta.....	81
10.4	Pruebas de Penetración Externas a Escenario 1.....	83
10.4.1	Ingeniería Social.....	84
10.4.2	Ataque de DoS.....	84
10.4.3	Port Scanning.....	84
10.5	Pruebas de Penetración Externas a Escenario 2.....	85
10.5.1	Ingeniería Social.....	86
10.5.2	Ataque de DoS.....	87
10.6	Pruebas de Penetración Externas a Escenario 3.....	87
10.6.1	Ingeniería Social.....	87
10.6.2	Ataque de DoS.....	88
10.6.3	Ataque de Fuerza Bruta.....	88
10.7	Mitigación.....	89
10.7.1	Control de Servicios.....	89
10.7.2	Configuración de SSH.....	90
10.7.3	Fail2ban.....	91
10.7.4	Firewall.....	101
10.7.5	PortSentry.....	103
10.7.6	chkrootkit.....	114
10.7.7	TCPWrapper.....	116
10.7.8	Shellshock.....	117
10.7.9	SELinux.....	118
10.7.10	Configuraciones en Elastix.....	119
10.7.11	Soluciones para Ataques de DoS y DDoS.....	122
10.7.12	Soluciones para ARP Spoofing y Ataques Relacionados.....	123
10.7.13	OpenVPN.....	126
10.7.14	IDS/IPS.....	126
10.7.15	Buenas Prácticas y Recomendaciones.....	126
11.	Conclusiones y Trabajos Futuros.....	129
	Referencias Bibliográficas.....	131

Índice de Figuras

Figura 3.1: Arquitectura VoIP	27
Figura 3.2: Teléfono de Gama Baja Marca Pheenet	28
Figura 3.3: Teléfono IP de Gama Media Marca Linksys	28
Figura 3.4: Teléfono IP Nortel IP Phone 2007.....	28
Figura 3.5: Ejemplo Básico de Conexión entre un Teléfono y un Puerto FXS	29
Figura 3.6: Adaptador Telefónico para Analógico y SIP (Linksys PAP2)	29
Figura 3.7: Adaptador Analógico para Interconexión con la RTC, Marca Grandstream....	29
Figura 3.8: Gateway GSM-IP (Fabricante 2N, Modelo VoiceBlue Enterprise)	30
Figura 3.9: Softphone ZoIPer.....	31
Figura 3.10: Softphone X-lite de CounterPath.....	31
Figura 3.11: Esquema de Enrutado en un Entorno SIP.....	32
Figura 3.12: Cabecera del Paquete IP	32
Figura 3.13: Formato de un Datagrama UDP.....	33
Figura 4.1: Establecimiento de la Comunicación entre el usuario A y el usuario B.....	46
Figura 6.1: Esquema General de los Componentes de Elastix	58
Figura 7.1: Conexión a un PSTN	63
Figura 7.2: Conexión a un ITSP	64
Figura 7.3: Conexión Remota a una Red LAN	64
Figura 10.1: Análisis de Vulnerabilidades de OpenVAS	72
Figura 10.2: Análisis de Vulnerabilidades de Nessus.....	72
Figura 10.3: Ataque a una LAN con Soporte VoIP	73
Figura 10.4: Descubrimiento de los Hosts de una Red con Nmap	74
Figura 10.5: Descubrimiento de los Puertos Abiertos de un Host con Nmap	75
Figura 10.6: Descubrimiento de Información Adicional sobre un Puerto Particular con Nmap.....	75
Figura 10.7: Tabla ARP del Servidor Elastix antes del Ataque.....	76
Figura 10.8: Ataque ARP Spoofing con Ettercap	76
Figura 10.9: Tabla ARP del Servidor Elastix Antes y Después del Ataque.....	77
Figura 10.10: Activación de IP Forwarding en Kali Linux	77
Figura 10.11: Captura de Tráfico VoIP en Wireshark.....	78
Figura 10.12: Análisis del Tráfico SIP de la Llamada Interceptada con Wireshark	78
Figura 10.13: Escucha de la Llamada Interceptada con el Reproductor RTP de Wireshark	79
Figura 10.14: Ejecución de la Herramienta inviteflood	80
Figura 10.15: Ejecución de la Herramienta rtpflood	81
Figura 10.16: Captura de Tráfico SIP en Wireshark.....	81
Figura 10.17: Ejecución de la Herramienta SIPDump	82
Figura 10.18: Archivo passhash.txt Generado por SIPDump	82
Figura 10.19: Visualización de Algunas Palabras del Diccionario Utilizado en SIPcrack..	82
Figura 10.20: Ejecución de la Herramienta SIPcrack	83
Figura 10.21: Ataque a una Red LAN con VoIP y una Conexión a un PSTN	84
Figura 10.22: Descubrimiento de los Puertos Abiertos de un Host y su Dirección IP Pública con Nmap.....	85
Figura 10.23: Ataque a una Red LAN con VoIP y una Conexión a un ITSP	86
Figura 10.24: Formulario para la Creación de Troncal SIP con el ITSP	86
Figura 10.25: Ataque a una Red LAN con VoIP y una Conexión Remota	87
Figura 10.26: Diccionario Utilizado en Medusa	88
Figura 10.27: Ejecución de la Herramienta Medusa.....	89

Figura 10.28: Lista de Servicios en Elastix.....	89
Figura 10.29: Desactivación de Servicios en Elastix	90
Figura 10.30: Detención de Servicios en Elastix	90
Figura 10.31: Acceso Rechazado por SSH en el puerto 23022 TCP al Usuario root.....	91
Figura 10.32: Directorio Inicial Fail2ban	92
Figura 10.33: Configuración Inicial de iptables.....	92
Figura 10.34: Configuración del Tipo de Bloqueo	93
Figura 10.35: Filtro para Asterisk 1.8	94
Figura 10.36: Configuración del jail de Asterisk	94
Figura 10.37: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Autenticación Fallida a una Extensión por el Puerto 5060 UDP	95
Figura 10.38: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Autenticación Fallida a una Extensión por el Puerto 5060 UDP	95
Figura 10.39: Configuración del jail de SSH.....	96
Figura 10.40: Intento Fallidos de Autenticación al Puerto 23022 TCP del Servidor VoIP utilizando PuTTY	96
Figura 10.41: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 23022 TCP	97
Figura 10.42: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 23022 TCP	97
Figura 10.43: Filtro para Apache.....	98
Figura 10.44: Configuración del jail de Apache	98
Figura 10.45: Configuración del jail de apache-noscript.....	99
Figura 10.46: Configuración del jail de apache-overflows	99
Figura 10.47: Configuración del jail de apache-nohome	99
Figura 10.48: Interfaz Gráfica de Usuario de Elastix 2.4.0	100
Figura 10.49: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 443 TCP	100
Figura 10.50: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 443 TCP	101
Figura 10.51: Intento de Visualizar la Interfaz Gráfica de Usuario de Elastix 2.4.0.....	101
Figura 10.52: Instalación de PortSentry	103
Figura 10.53: Configuración de Puertos en Modo "advanced"	104
Figura 10.54: Configuración de Acción luego de Detectar Escaneo de Puertos.....	105
Figura 10.55: Configuración de Regla para iptables	105
Figura 10.56: Configuración del jail de PortSentry	105
Figura 10.57: Filtro para PortSentry	106
Figura 10.58: Template para Interpretación de Fecha de PortSentry	106
Figura 10.59: Intento de Conexión al Puerto 22 TCP del Servidor VoIP utilizando PuTTY	107
Figura 10.60: Restricción del Tráfico Entrante por PortSentry a una Dirección IP que Intentó Conexión al Puerto 22 TCP.....	107
Figura 10.61: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Conexión al Puerto 22 TCP	107
Figura 10.62: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que intentó Conexión al Puerto 22 TCP	108
Figura 10.63: Descubrimiento Fallido de los Puertos Abiertos de un Host con Nmap	108
Figura 10.64: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó un Port Scanning	109
Figura 10.65: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó un Port Scanning.....	109

Figura 10.66: Captura de Paquetes en Wireshark Antes del Bloqueo de Dirección IP Atacante	110
Figura 10.67: Restricción del Tráfico Entrante por PortSentry a una Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades.....	110
Figura 10.68: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades	111
Figura 10.69: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades.....	111
Figura 10.70: Resumen del Análisis de Vulnerabilidades Realizado por OpenVAS	112
Figura 10.71: Resumen del Análisis de Vulnerabilidades Realizado por Nessus	112
Figura 10.72: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Enviar Script	112
Figura 10.73: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Enviar Script.....	113
Figura 10.74: Ataque de Fuerza Bruta al Puerto 22 TCP No Satisfactoria	113
Figura 10.75: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Ataque de Fuerza Bruta al Puerto 22 TCP.....	114
Figura 10.76: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que intentó Ataque de Fuerza Bruta al Puerto 22 TCP	114
Figura 10.77: Instalación de chkrootkit.....	115
Figura 10.78: Programación de chkrootkit para Ejecutar Todos los Días a las 3 a.m.	115
Figura 10.79: Ejecución de chkrootkit	116
Figura 10.80: Configuración del Archivo "hosts.allow" para SSH	117
Figura 10.81: Configuración del Archivo "hosts.deny" para SSH.....	117
Figura 10.82: Conexión al Puerto SSH del Servidor VoIP utilizando el Bash de Kali Linux	117
Figura 10.83: Prueba de Vulnerabilidad de Shellshock	118
Figura 10.84: Prueba de la No Vulnerabilidad de Shellshock.....	118
Figura 10.85: Ruta Saliente "Route 9_outside" con Opción a Eliminación.....	119
Figura 10.86: ACL en Elastix para el Registro de Extensión con Dirección IP Específica	119
Figura 10.87: ACL en Elastix para el Registro de Extensión perteneciente a un Segmento de Red.....	119
Figura 10.88: Configuraciones en "/etc/asterisk/manager.conf"	120
Figura 10.89: Opciones Avanzadas del Módulo de Seguridad del GUI de Elastix.....	120
Figura 10.90: Configuraciones en "/etc/amportal.conf".....	121
Figura 10.91: Configuraciones en "/etc/asterisk/sip_general_custom.conf"	121
Figura 10.92: Contextos Personalizados en FreePBX	122
Figura 10.93: Configuración del Contexto Personalizado "PSTN-Locales".....	122
Figura 10.94: Archivo de Configuración de ARPWatch	123
Figura 10.95: Iniciación del Servicio de ARPWatch y Configuración de Arranque.....	124
Figura 10.96: Registros de Nuevas Estaciones y Cambios de Direcciones MAC en las Estaciones	124
Figura 10.97: Correo Electrónico Enviado por ARPWatch como Resultado del Descubrimiento de una Nueva Estación	125
Figura 10.98: Correo Electrónico Enviado por ARPWatch como Resultado del Cambio de Dirección MAC en una Estación	125

Índice de Tablas

Tabla 3.1: Protocolos VoIP	35
Tabla 4.1: Ejemplos de Direcciones SIP	42
Tabla 4.2: Peticiones SIP.....	43
Tabla 4.3: Posibles Respuestas del Protocolo SIP	45
Tabla 4.4: Campos más Frecuentes de los Mensajes SIP	47
Tabla 7.1: Comparación de Distribuciones para Pruebas de Penetración.....	65
Tabla 7.2: Detalle de las Versiones de los Paquetes de Elastix 2.4.0	66
Tabla 10.1: Funcionamiento de Elastix ante Número de Peticiones INVITE	80
Tabla 10.2: Calidad de la Llamada ante Número de Paquetes RTP	80
Tabla 10.3: Términos de Fail2ban.....	91

1. Introducción

Desde hace algunos años, gracias a la evolución tecnológica, se han desarrollado nuevas tecnologías y nuevos dispositivos que han revolucionado totalmente el campo de las telecomunicaciones. Sin duda alguna, Internet ha sido una de las mayores revoluciones. Poco a poco Internet se fue convirtiendo en un gigante en donde todos comparten información y se permite algo hasta entonces imposible, la comunicación a través del PC.

Pronto apareció la necesidad humana de comunicarse con la voz, que unido a gran cantidad de avances tecnológicos (procesado digital de señales, desarrollo de mecanismos de control, priorización y calidad de servicio en redes de datos) hicieron centrar las miradas en el desarrollo de la telefonía sobre la red de datos utilizando esa gran red de comunicaciones que es Internet.

La tecnología VoIP trata de transportar la voz, previamente procesada, encapsulándola en paquetes para poder ser enviada sobre redes de datos sin necesidad de disponer de una infraestructura telefónica convencional. A medida que crece su popularidad, aumentan las preocupaciones por la seguridad de las comunicaciones y la telefonía IP. VoIP es una tecnología que ha de apoyarse necesariamente en muchas otras capas y protocolos ya existentes de las redes de datos. Por eso en cierto modo la telefonía IP va a heredar ciertos problemas de las capas y protocolos ya existentes, siendo algunas de las amenazas más importantes de VoIP, los problemas clásicos de seguridad que afectan al mundo de las redes de datos.

Este trabajo tiene la siguiente estructura:

Capítulo 2: Presenta la descripción del problema.

Capítulo 3: Contiene una breve introducción a VoIP, su evolución, arquitectura y los protocolos que intervienen.

Capítulo 4: Describe el funcionamiento del protocolo SIP.

Capítulo 5: Presenta el análisis de los problemas de seguridad de SIP y sus técnicas de mitigación.

Capítulo 6: Este capítulo se enfoca en el servidor de comunicaciones unificadas, su historia y características.

Capítulo 7: Da una descripción de los escenarios propuestos de redes con soporte para VoIP y muestra las especificaciones técnicas de los escenarios.

Capítulo 8: Presenta los trabajos relacionados recopilados durante la investigación.

Capítulo 9: Describe el marco metodológico.

Capítulo 10: Se aplica el Marco Metodológico a los Ataques Internos y a los Ataques Externos del Escenario 1, 2 y 3.

Capítulo 11: Contiene las conclusiones y los trabajos futuros.

2. El Problema

En este capítulo se exponen los argumentos que justifican el planteamiento de un conjunto de lineamientos para tratar de disminuir los riesgos de seguridad, así como el planteamiento de nuevos objetivos y alcances para obtener una solución eficaz.

2.1 Planteamiento del Problema

Mucha gente no asegura correctamente sus servidores Asterisk o Elastix y existen ciertos inconvenientes para el desarrollo de la telefonía sobre IP como lo es la seguridad. Desde el punto de vista de la seguridad, las llamadas en VoIP se transmiten por Internet o por redes potencialmente inseguras. Lo cual plantea riesgos de privacidad y seguridad que no surgen con un servicio telefónico tradicional. VoIP es vulnerable además en muchos otros puntos, ya sea en los protocolos utilizados, en los dispositivos que intervienen, o debilidades en la red por la que se transmite.

No hay un sistema lo suficientemente seguro y además Asterisk o Elastix no disponen de todos los medios necesarios para hacer las comunicaciones suficientemente seguras, por lo que se opta por hacer más difíciles los ataques.

Para defender una red que implanta VoIP de ataques, es importante adoptar medidas o lineamientos que permitan un esquema seguridad robusto.

2.2 Justificación del Problema

La seguridad es un aspecto que en una arquitectura VoIP se suele dejar para el final y que puede llegar a ser un factor crítico en el correcto funcionamiento de implementaciones VoIP.

Se pueden realizar multitud de ataques contra una red de VoIP e información sensible e importante podría quedar expuesta. También es posible utilizar ataques de flooding con el objetivo de saturar la red y ocasionar la interrupción del servicio. Como consecuencia, el contenido de las comunicaciones VoIP es vulnerable a los ataques, el hackeo, las modificaciones y las interceptaciones.

Con la exposición a serios riesgos de seguridad se puede disminuir considerablemente los ahorros ganados al reducir los cargos de las llamadas. Es incalculable el costo que representa para su empresa el robo de información clave o la pérdida de tiempo productivo de sus empleados.

2.3 Objetivos

En esta sección se definen los objetivos que se quieren conseguir con la realización del trabajo propuesto.

2.3.1 Objetivo General

Para las 3 arquitecturas de red que implementan VoIP, que serán descritas más adelante, proponer un conjunto de lineamientos que permitan mitigar o prevenir ataques en las arquitecturas propuestas.

2.3.2 Objetivos Específicos

- Estudiar el área de seguridad en redes VoIP.
- Aprender a utilizar y configurar el servidor de comunicaciones unificadas (Elastix 2.4.0).
- Realizar ataques en escenarios controlados y proponer soluciones para su mitigación.
- Conocer herramientas de seguridad en IP y VoIP.

2.3.3 Alcances

Los ataques propuestos deberán generar problemas de seguridad significativos en cada una de las arquitecturas de red sugeridas (descritas más adelante). Adicionalmente, se hará una lista de lineamientos que ayudarán a combatir estos problemas.

3. VoIP

3.1 Introducción

En la década de los 90, un grupo de personas pertenecientes al entorno de la investigación, tanto de instituciones educativas como empresariales, comenzaron a mostrar un cierto interés por transportar voz y video sobre redes IP, especialmente a través de intranets corporativas e Internet [1]. Esta tecnología es conocida hoy día como VoIP (Voice Over Internet Protocol) y es el proceso de dividir el audio y el vídeo en paquetes de datos, transmitir dichos paquetes a través de una red IP y reensamblar esos paquetes en el destino final permitiendo de esta manera que la gente pueda comunicarse. Esto significa que se envía la señal de voz en forma digital, en lugar de enviarla en forma analógica a través de circuitos utilizables sólo por telefonía convencional como las redes PSTN (Public Switched Telephone Network).

VoIP consigue desarrollar una única red homogénea en la que se envía todo tipo de información ya sea voz, video o datos. Es evidente que la utilización de una única red para la transmisión de voz y datos presenta gran cantidad de ventajas. Para un proveedor de servicio de telefonía y datos, por un lado, obtiene mayores beneficios ya que con una sola línea puede ofrecer más servicios. Y por otro lado le supone un ahorro de gastos tanto de infraestructura como de mantenimiento. Una llamada telefónica requiere una gran red de centrales conectadas entre sí con cableado, fibra óptica, satélites de telecomunicación o cualquier otro medio, que equivale a una enorme inversión para crear y mantener estas infraestructuras. En cambio, una llamada telefónica sobre IP supone comprimir la voz y enviarla en paquetes de datos por una línea en la que pueden viajar diferentes llamadas e incluso diferentes datos, sin necesidad de líneas dedicadas ni desaprovechamiento del ancho de banda [2].

Existen también ciertos inconvenientes para el desarrollo de la telefonía sobre IP que se podrían resumir en los siguientes tres conceptos: (1) seguridad, (2) fiabilidad y (3) calidad de servicio (QoS). VoIP al basarse sobre el protocolo IP (y en muchos casos usando UDP en la capa de transporte) asume la posibilidad de que los paquetes puedan perderse, otro problema es que no hay una garantía absoluta en el tiempo que tardan en llegar los paquetes al otro extremo de la comunicación aunque se utilicen técnicas de priorización. Estos problemas de calidad de servicio telefónico y dependencia de la red de datos suponen uno de los principales problemas para la difusión total de la telefonía por IP. Pero es cierto que, poco a poco dichos problemas se van solucionando con la evolución de las tecnologías involucradas [2].

3.2 Evolución

Hay patentes y publicaciones de investigaciones de décadas atrás que nos muestran que la idea de la VoIP no es nueva. En 1995, la compañía israelí Vacoltec anunció el lanzamiento del primer teléfono software para Internet. Este software era únicamente útil para entablar una comunicación de PC a PC y para ello necesitaba hacer uso de diversos requisitos de hardware tales como micrófono, altavoces, tarjeta de sonido y módem. Básicamente el funcionamiento de este software es igual al de hoy día, transformar la señal de voz en paquetes IP una vez comprimida. Sin embargo, esta alternativa a la comunicación telefónica tradicional fue comercialmente un fracaso ya que las conexiones a Internet que se disponían ofrecían un ancho de banda muy escaso [1].

Durante los años siguientes, la tecnología asociada a las redes de datos y las comunicaciones continuó mejorando, y en 1998 se dieron los primeros pasos desde un punto de vista comercial. En este año diversas compañías lanzaron al mercado adaptadores que permitían hacer uso de los teléfonos tradicionales en un entorno VoIP. Ello facilitó el acercamiento a los clientes a la hora de poder hacer uso de la tecnología VoIP, por lo que algunas empresas importantes se lanzaron al mercado ofreciendo productos y servicios relacionados con esta tecnología [1].

En 1999, compañías dedicadas a las redes de datos tales como Cisco crearon las primeras plataformas corporativas para VoIP, utilizando el protocolo de señalización H.323. Estas plataformas eran destinadas a empresas capaces de tratar con tráfico VoIP. Esto supuso un nuevo impulso a la VoIP ya que comenzó a implantarse en muchas empresas.

Las redes de datos siguieron mejorando y evolucionando en los años posteriores. Alrededor del año 2005, ya era fácil para cualquier persona de países desarrollados adquirir una conexión a Internet que reduzca al mínimo las posibles interrupciones que se pudieran producir durante la conversación. Esta conexión cumplía con los requisitos mínimos para ofrecer una buena calidad de voz y una comunicación fiable a través de VoIP.

Esto supuso otro gran impulso a la VoIP y provocó que al día de hoy existan muchas soluciones que hacen uso de esta tecnología. Un ejemplo claro es Asterisk, una central telefónica de software libre que se distribuye bajo licencia GPL. Este producto, soportado comercialmente por Digium, se ha convertido en pocos años en una de las soluciones IP más extendidas en diversos ámbitos, como el empresarial o el educativo. Otro ejemplo destacable de producto VoIP es Skype, que fue creado por dos jóvenes universitarios en el año 2003. A diferencia de Asterisk, Skype hace uso de un protocolo privado que no está basado en un estándar, lo que a largo plazo se piensa que limitará a sus usuarios. Al día de hoy, Skype se puede emplear en multitud de plataformas y su uso se encuentra también ampliamente extendido [1].

3.3 Arquitectura

Uno de los beneficios que aporta la VoIP es que la arquitectura, desde el punto de vista de su distribución, puede ser centralizada o distribuida. El enfoque centralizado es criticado porque al estar todo localizado en un mismo punto las futuras innovaciones tecnológicas se verán entorpecidas. Por otro lado, la arquitectura distribuida es más compleja que la arquitectura centralizada. Independientemente de un enfoque u otro, lo que VoIP permite es una gran flexibilidad [1].

En la Figura 3.1 se muestra una arquitectura de VoIP muy general, donde se pueden observar los distintos dispositivos que la compone:

- Teléfono IP: es un teléfono similar a un teléfono tradicional con la diferencia que está adaptado para ser utilizado en entornos IP.
- Softphone: es un teléfono similar al del punto anterior con la peculiaridad de que este es software.
- Adaptador ATA: es un adaptador que permite conectar un teléfono convencional a una red IP.

- SIP: es un protocolo de señalización encargado de, entre otras funciones, iniciar y finalizar las llamadas VoIP.
- B2BUA: es una entidad intermediaria encargada de procesar las comunicaciones VoIP y retransmitirlas a su destino.

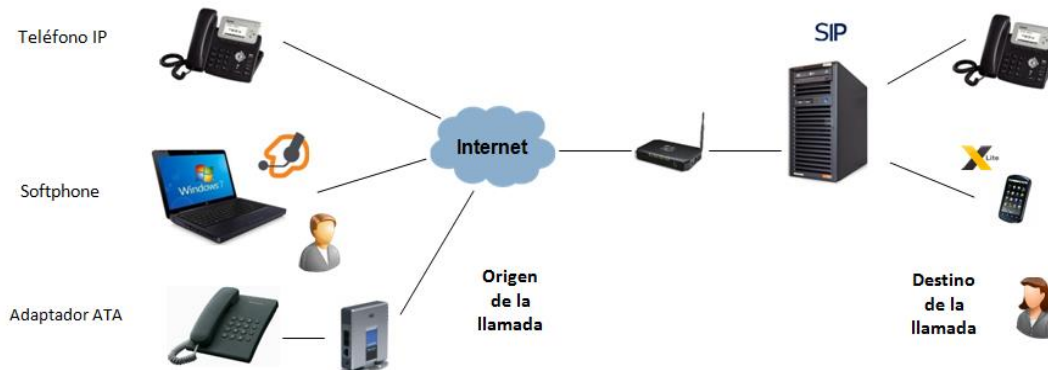


Figura 3.1: Arquitectura VoIP

A continuación, se presentan de manera más detallada los elementos más significativos de un entorno de VoIP.

3.3.1 Teléfonos IP

Una parte importante en la arquitectura de VoIP son los teléfonos IP. En la actualidad, los teléfonos IP son, en su gran mayoría, muy similares al resto de teléfonos tradicionales. Sin embargo, en su aspecto exterior se puede apreciar que existen ciertas diferencias:

- Disponen de al menos un puerto de conexión RJ-45 en lugar del tradicional RJ-11. El puerto RJ-45 es un puerto Ethernet que sirve para conectar el teléfono IP a la red.
- Suelen disponer de pantalla para mostrar información relevante.
- Incorporan varios botones programables que pueden usarse para diferentes funcionalidades.
- Conector de auriculares.

Dependiendo de sus características y posibilidades los teléfonos IP, se pueden clasificar en tres categorías:

- Gama baja: constituyen la mayoría y son aquellos que se parecen más a los teléfonos tradicionales (ver Figura 3.2). Proporcionan un buen servicio para realizar llamadas por VoIP a otros terminales de la red o a través de proxys, aunque disponen de pocas funcionalidades extra. Entre las mismas cabe destacar el soporte para varios idiomas o la personalización de tonos de llamada y melodías.



Figura 3.2: Teléfono de Gama Baja Marca Pheenet

- Gama media: son muy parecidos a los teléfonos IP gama baja, pero incorporan nuevas funcionalidades que los anteriores no poseen. Además suelen tener una pantalla más avanzada y grande (ver Figura 3.3), así como más conexiones hardware de las que tienen los básicos.



Figura 3.3: Teléfono IP de Gama Media Marca Linksys

- Gama avanzada: estos teléfonos suelen incluir pantallas a color (ver Figura 3.4) y muchas otras funciones extras como la posibilidad de configurar el acceso a un servidor LDAP (Lightweight Directory Access Protocol) de una organización o acceso web a través de la pantalla del teléfono.



Figura 3.4: Teléfono IP Nortel IP Phone 2007

3.3.2 Gateways y Adaptadores Analógicos

Un ATA (Analog Telephone Adaptor) se puede describir brevemente como un dispositivo que permite transformar la señal de telefonía clásica a VoIP. En resumen, estos dispositivos se emplean para convertir una señal digital (ya sea IP o propietaria) a una señal analógica (o viceversa) que pueda ser conectada a teléfonos o faxes tradicionales [1].

Existen diferentes versiones de ATA en función de que desee conectar un puerto FXO o un puerto FXS. La interfaz FXS (Foreign eXchange Subscriber) es el puerto por el cual el abonado accede a la línea telefónica mientras que la interfaz FXO (Foreign eXchange Office) es el puerto por el cual se recibe a la línea telefónica (ver Figura 3.5). En otras palabras, la interfaz FXS es el puerto que efectivamente envía la línea analógica al abonado y la interfaz FXO es el puerto que recibe la línea analógica. Un FXS siempre va conectado a un FXO y viceversa.

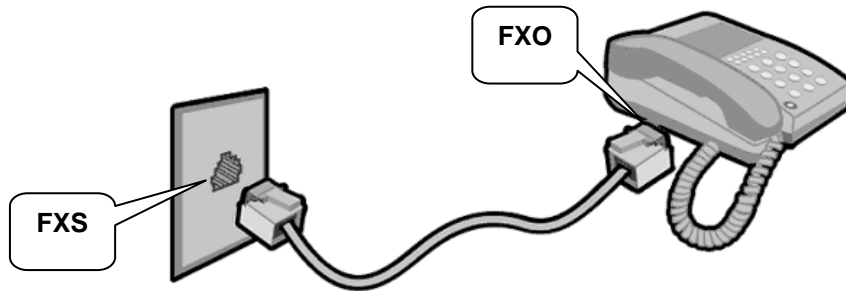


Figura 3.5: Ejemplo Básico de Conexión entre un Teléfono y un Puerto FXS

En la Figura 3.6 se muestra un adaptador telefónico para analógico y SIP (Linksys PAP2) que dispone de 2 puertos analógicos (puertos FXS) para conectar dos teléfonos y un puerto RJ45 para conectividad Ethernet.



Figura 3.6: Adaptador Telefónico para Analógico y SIP (Linksys PAP2)

Estos adaptadores podrían ser descritos como gateways, ya que su función es justamente la de pasarela entre el mundo analógico y el IP. Sin embargo, el uso popular del término gateway de telefonía describiría mejor un adaptador telefónico multipuerto, generalmente con funciones de enrutamiento más complejas [1] (ver Figura 3.7).



Figura 3.7: Adaptador Analógico para Interconexión con la RTC, Marca Grandstream

3.3.3 Dispositivos GSM/UMTS/HSDPA/HSUPA/LTE

Los teléfonos móviles o también llamados teléfonos celulares son dispositivos electrónicos de pequeño tamaño que son empleados para realizar comunicaciones de voz o datos a través de una conexión a una estación base que pertenece a una determinada red de telefonía móvil. Estos pequeños dispositivos son una auténtica revolución en la forma en que nos comunicarnos.

Existe una gran variedad de teléfonos móviles, desde los más básicos hasta los teléfonos que ofrecen mayores funcionalidades, como los smartphones (teléfonos inteligentes), musicphones (teléfonos con posibilidad de reproducir música) o cameraphones (teléfonos con cámara integrada).

Desde el punto de vista de VoIP se pueden encontrar dispositivos que integran ambas tecnologías, por ejemplo, SIP y GSM. Estos dispositivos permiten una mayor integración que la tecnología analógica tradicional ya que no dependen de conversiones intermedias a analógico y, por tanto, permiten enviar toda la señalización existente entre ambas redes, de forma transparente y fiable [1].

En la Figura 3.8 se observa un Gateway GSM-IP para interconexión directa de redes GSM e IP del fabricante 2N, modelo VoiceBlue Enterprise con 4 líneas GSM, Proxy SIP y H.323.



Figura 3.8: Gateway GSM-IP (Fabricante 2N, Modelo VoiceBlue Enterprise)

3.3.4 Softphones

Los softphones son teléfonos implementados por software. Éstos proporcionan a un dispositivo, como un computador, las funcionalidades de un teléfono VoIP. El objetivo principal de un softphone es realizar llamadas telefónicas a otros softphones o a otros teléfonos convencionales. Para que esto sea posible, no es necesario que el dispositivo en cuestión sea muy potente. Simplemente se necesita un equipo de audio adecuado y alguna forma de conectarse a una red TCP/IP.

La principal ventaja de un softphone sobre un teléfono VoIP hardware es el costo. En muchas empresas se debe instalar como mínimo en cada escritorio un computador y un teléfono. Si se pudiese quitar el teléfono de todos esos escritorios se produciría un gran ahorro [1].

Se pueden encontrar softphones que funcionan con diferentes protocolos de señalización, aunque el más usado es SIP. Entre ellos, los más conocidos y usados son X-lite y ZoIPer,

aunque también existen muchos otros que presentan buenas funcionalidades. En la Figura 3.9 se detalla un softphone ZoIPer. Este softphone puede trabajar con el protocolo SIP o IAX, y dispone de una versión gratuita y otra comercial. En la Figura 3.10 se puede apreciar un softphone X-lite de CounterPath, uno de los softphones más empleados hoy día y se encuentra disponible para varios sistemas operativos.



Figura 3.9: Softphone ZoIPer



Figura 3.10: Softphone X-lite de CounterPath

Los teléfonos IP de hardware son una combinación entre un softphone y un hardware dedicado. Aunque la capacidad de computación de un computador exceda de forma considerable a la de un teléfono IP, siempre se debe tomar en cuenta que estos últimos son dispositivos especialmente creados para realizar llamadas y transportar voz [1].

3.3.5 Proxys y Enrutadores

En una arquitectura de VoIP, es necesario incluir el uso de ciertos elementos que permitan ordenar el tráfico telefónico y a la vez conectar a los diferentes usuarios de las redes implicadas. Tal y como trabajan los enrutadores con los datos en general, recibiendo y enviando peticiones desde y hacia otras máquinas, los diferentes protocolos de señalización necesitan igualmente que alguien o algo encamine sus peticiones hacia los usuarios finales, a fin de establecer una conversación. Esta tarea la realizan los proxys

o enrutadores, encargándose de redirigir la señalización hacia los sitios adecuados en función de las indicaciones pertinentes que cada protocolo implementa [1]. En la Figura 3.11, puede verse un esquema de enrutamiento en un entorno SIP.

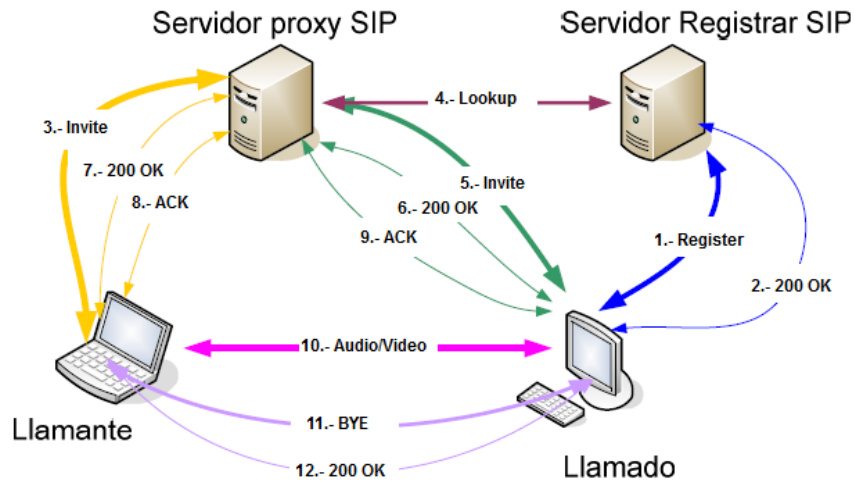


Figura 3.11: Esquema de Enrutado en un Entorno SIP

3.4 Internet Protocol

El protocolo IP (Internet Protocol) es un protocolo que trabaja a nivel de red y el mismo se encuentra definido en el RFC 791 [7] de septiembre de 1981. La información se envía en paquetes llamados paquetes IP. Este protocolo ofrece un servicio “sin garantías” también llamado del “mejor esfuerzo”. Es decir que no garantiza que los paquetes lleguen al destino, sin embargo se hará lo posible por hacerlos llegar. La Figura 3.12 muestra la cabecera del paquete IP.

0	3 4	7 8	15 16 17 18 19	23 24	31
Version		IHL	Type of Service	Total Length	
Identification			0	D F	M F
Time To Live		Protocol	Header Checksum		
Source Address					
Destination Address					

Figura 3.12: Cabecera del Paquete IP

3.5 Protocolos de Transporte

Proporciona una transferencia transparente y fiable de datos entre los puntos finales; además, proporciona procedimientos de recuperación de errores y control de flujo origen-destino [5]. En la arquitectura de protocolos TCP/IP se han especificado tres protocolos para la capa de transporte: TCP, UDP y SCTP.

3.5.1 Transmission Control Protocol

La mayor parte de aplicaciones que se ejecutan usando la arquitectura TCP/IP usan como protocolo de transporte TCP (Transmission Control Protocol). TCP proporciona una

conexión fiable para transferir los datos entre las aplicaciones. Una conexión es simplemente una asociación lógica de carácter temporal entre dos entidades de sistemas distintos. Cada PDU de TCP, denominado segmento TCP, contiene en la cabecera la identificación de los puertos origen y destino, los cuales corresponden con los puntos de acceso al servicio (SAP) de la arquitectura OSI. Los valores de los puertos identifican a los respectivos usuarios (aplicaciones) de las dos entidades TCP. Una conexión lógica alude a un par de puertos. Durante la conexión, cada entidad seguirá la pista de los segmentos TCP que vengán y vayan hacia la otra entidad, para así regular el flujo de segmentos y recuperar aquellos que se pierdan o dañen [5].

3.5.2 User Datagram Protocol

UDP (User Datagram Protocol) es otro protocolo de transporte de la arquitectura TCP/IP que no garantiza la entrega, la conservación del orden secuencial, ni la protección frente a duplicados y que hace posible el envío de mensajes entre aplicaciones con la complejidad mínima. Debido a su carácter no orientado a conexión, UDP en realidad tiene poca tarea que hacer. Básicamente, su cometido es añadir a IP la capacidad de identificar los puertos [5].

UDP es el más simple de los protocolos de transporte de TCP/IP. Es un protocolo de nivel de transporte extremo a extremo que añade sólo direcciones de puertos, control de errores mediante sumas de comprobación y la información de longitud de los datos del nivel superior. El paquete creado por el protocolo UDP se denomina datagrama de usuario y se puede apreciar en la Figura 3.13.

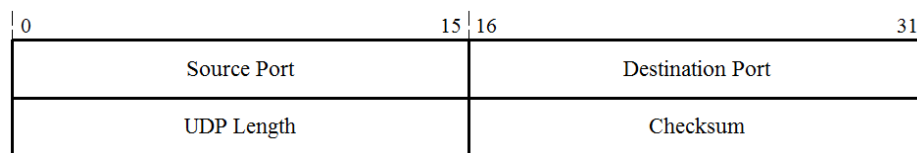


Figura 3.13: Formato de un Datagrama UDP

A continuación se realiza una breve descripción de los campos de un datagrama UDP:

- Source Port: es la dirección del programa de aplicación que ha creado el mensaje.
- Destination Port: es la dirección del programa de aplicación que recibirá el mensaje.
- UDP Length: este campo define la longitud total del datagrama de usuario en bytes.
- Checksum: esta suma de comprobación es un campo de 16 bits utilizado para la detección de errores.

UDP proporciona sólo las funciones básicas necesarias para la entrega extremo a extremo de una transmisión. No ofrece funciones de secuenciamiento ni de reordenación y no puede especificar el paquete dañado cuando se informa de un error (por lo que debe usarse con ICMP). UDP puede descubrir que ha ocurrido un error; ICMP puede, a continuación, informar al emisor de que un datagrama de usuario se ha dañado o se ha descartado. Tampoco tiene la capacidad para especificar qué paquete se ha perdido. UDP contiene solo una suma de comprobación; no contiene un identificador o número de secuencia para un segmento de datos concreto.

3.5.3 Stream Control Transmission Protocol

El protocolo SCTP (Stream Control Transmission Protocol) es un protocolo de transporte confiable y orientado a mensaje (similar al envío de datagramas UDP). El protocolo está especificado en el RFC 4960 [7] y el RFC 3286 [8] en donde se brinda una introducción al mismo. SCTP es una alternativa a los protocolos de transporte TCP y UDP pues provee confiabilidad, control de flujo y secuenciación como TCP. Sin embargo, SCTP opcionalmente permite el envío de mensajes fuera de orden.

SCTP fue diseñado inicialmente por el grupo SIGTRAN (SIGnaling TRANsport) para transportar señalización de control de la telefonía pública SS7 (Signalling System No. 7) por redes IP. La intención fue la de proveer en IP de algunas de las características de confiabilidad de SS7. Por su versatilidad luego se ha propuesto utilizarlo en otras áreas, como por ejemplo para transportar mensajes de los protocolos DIAMETER o SIP.

3.6 Protocolos de Señalización

La realización de una llamada entre dos teléfonos cualesquiera implica la utilización de diversos equipos electrónicos, los cuales deben comunicarse entre sí. Las reglas y/o normas que garantizan que la comunicación entre los equipos se realiza adecuadamente es lo que se conoce como protocolo de señalización.

En las redes analógicas o redes de conmutación de circuitos antes de que ambos extremos puedan comunicarse, se produce la reserva de recursos necesarios para que la comunicación tenga éxito. Si por cualquier circunstancia no puede llevarse a cabo esta reserva de camino entre ambos extremos se informa al emisor de este hecho. A la acción de “reservar un camino de recursos entre ambos extremos” es lo que se llama señalización [1].

En conmutación de paquetes los protocolos de señalización realizan acciones muy similares a los protocolos de señalización en conmutación de circuitos, es decir, tareas de establecimiento de sesión, control del progreso de la llamada, entre otras. Adicionalmente, estos protocolos cuidan de que se cumplan ciertas garantías de calidad y los mismos se encuentran en la capa de sesión del modelo OSI [1].

La Tabla 3.1 muestra los protocolos VoIP organizados por orden de antigüedad.

Protocolo VoIP	Descripción
H.323	Protocolo definido por la ITU-T.
SIP	Protocolo definido por la IETF (RFC 3261).
Megaco (H.248)	Protocolo de control definido por la IETF (RFC 3015).
MGCP	Protocolo de control definido por la IETF (RFC 3435).
UNISTim	Protocolo propiedad de Nortel (Avaya).
SCCP	Protocolo propiedad de Cisco.
MiNet	Protocolo propiedad de Mitel.
CorNet-IP	Protocolo propiedad de Siemens.
IAX	Protocolo original para la comunicación entre PBXs Asterisk.
Skype	Protocolo propietario peer-to-peer utilizado en la aplicación Skype.
IAX2	Protocolo para la comunicación entre PBXs Asterisk en reemplazo de IAX.
Jingle	Protocolo abierto utilizado en tecnología XMPP.

weSIP	Protocolo licencia gratuita de VozTelecom.
-------	--

Tabla 3.1: Protocolos VoIP

A continuación se presentan los protocolos de señalización más utilizados en conmutación de paquetes (SIP y H.323) y uno de los protocolos desarrollado para Asterisk (IAX/IAX2).

3.6.1 Session Initiation Protocol

Numerosos protocolos han sido escritos para transportar de diferentes formas los datos de sesión multimedia en tiempo real, tales como voz, vídeo o mensajes de texto. El protocolo SIP (Session Initiation Protocol) trabaja en conjunto con estos protocolos al permitir que los puntos finales de Internet (llamados agentes de usuario) se descubran el uno al otro y se pongan de acuerdo en una sesión a la que le gustaría compartir. SIP es una herramienta de uso general ágil para crear, modificar y finalizar sesiones que funciona de forma independiente de los protocolos de transporte y sin depender del tipo de sesión que se está estableciendo [3]. Este protocolo será descrito de forma más detallada en el Capítulo 4.

3.6.2 H.323

El protocolo H.323 fue diseñado por ITU-T (International Telecommunication Union) en el año 1996. Fue diseñado para ser un estándar en la transmisión de datos, audio y video a través de las redes IP en las cuales no existe garantía en la calidad del servicio. Ha evolucionado rápidamente para dirigir las crecientes necesidades de las redes de VoIP. Fue el primer estándar de VoIP en adoptar el estándar de IETF de RTP (Real-time Transport Protocol) para transportar audio y vídeo sobre redes IP. Es más complejo que SIP, y por tanto no es tan fácilmente expansible como éste, por lo cual actualmente predominan las aplicaciones basadas en SIP.

El estándar H.323 ofrece control y señalización de la llamada, control y transporte multimedia, control del ancho de banda punto-a-punto y conferencias. La señalización de H.323 es muy rápida, sobre todo comparada a la de SIP, la cual utiliza paquetes de gran tamaño. Esto es debido a que el formato de los mensajes en H.323 es binario, mientras que en los mensajes SIP el formato es texto plano [1]. Tiene referencias hacia algunos otros protocolos de ITU-T como:

- H.225 para describir la señalización de llamada, el medio (audio y video), el empaquetamiento de las tramas, la sincronización de tramas de medio y los formatos de los mensajes de control.
- H.245 para describir los mensajes y procedimientos utilizados para abrir y cerrar canales lógicos para audio, video y datos, capacidad de intercambio, control e indicaciones.
- H.450 es usado para describir los servicios suplementarios.
- H.235 para describir la seguridad de H.323.
- H.239 para describir el uso de la doble trama en videoconferencia, normalmente uno para video en tiempo real y la otra para presentación.
- H.281 para describir el control de cámara lejana para movimientos PTZ (Pan-Tilt-Zoom).

A pesar de estar muy extendido, actualmente H.323 está siendo sustituido por SIP, el cual es modular y por tanto mucho más flexible.

3.6.3 Inter-Asterisk eXchange

A pesar de que Asterisk reconoce y soporta muchos protocolos, introduce un protocolo propio para la señalización de VoIP en Asterisk. Este protocolo se denominó IAX (Inter-Asterisk eXchange) y es utilizado por Asterisk para comunicaciones de VoIP entre servidores Asterisk y entre servidores y clientes que también utilizan protocolo IAX. La principal diferencia entre IAX y SIP o H.323 es que IAX no utiliza RTP, sino que en su lugar implementa su propio mecanismo de transmisión de voz. Actualmente se utiliza la versión 2 de este protocolo (IAX2) ya que la primera versión se considera obsoleta. IAX2 es robusto, lleno de novedades y relativamente simple en comparación con otros protocolos. Permite manejar una gran cantidad de códecs y un gran número de flujos (*streams*), lo que significa que puede ser utilizado para transportar virtualmente cualquier tipo de datos. Esta capacidad lo hace muy útil para realizar videoconferencias o realizar presentaciones remotas. Puede empaquetar múltiples sesiones dentro de un flujo de datos, así requiere de menos ancho de banda y permite mayor número de canales entre terminales.

Un inconveniente de IAX/IAX2 es que no es un estándar, sino un protocolo independiente creado por Mark Spencer, creador de Digium. A pesar de ser un protocolo propietario, es abierto y ha sido aceptado por la comunidad de VoIP [1].

IAX/IAX2 trabaja junto a UDP con una característica muy especial: todas las comunicaciones (registro, señalización de llamada, transmisión de voz) hacen uso de un único puerto UDP (generalmente 4569). Por lo tanto, NAT (Network Address Translation) no es un problema en IAX a diferencia de SIP, ya que tanto los datos de señalización como el audio viajan por puertos distintos.

Los mensajes en SIP son en formato de texto mientras que en IAX son en formato binario, esto hace que IAX consuma menos ancho de banda.

3.7 Session Description Protocol

SDP (Session Description Protocol) se encuentra definido en el RFC 4566 [13], se utiliza durante la negociación que lleva a cabo SIP entre los dos agentes. Su función es la de detallar cómo se va a realizar el intercambio de comunicación posterior mediante protocolos como, por ejemplo, RTP. Para ello indica toda la información relacionada con el tráfico multimedia tal como la dirección IP y el puerto donde espera cada agente recibir el audio, el códec a utilizar entre ambos, etc.

3.8 Protocolos de Transporte de Audio y Video en Tiempo Real

El transporte de voz y datos por una red IP no sólo significa mover por la misma secuencias de bits empaquetados, sino que también se requiere adicionarle a éstos cierta información de control, por ejemplo: tipo de información transportada, número de secuencia, marcas de tiempo, etc., de manera que esto supone una forma de “formatear” los paquetes IP y hacerlos aptos para el transporte de información con requerimientos de tiempo real [4].

Para establecer un flujo de comunicación media es necesario un protocolo que intercambie la información entre los extremos de dicha comunicación, es decir, que transporte la información entre un origen y su destino, además de proveer de las técnicas necesarias para enviar los problemas que se pueden presentar durante el intercambio, tales como: jitter, retardo, etc [1].

Los protocolos más utilizados para esta finalidad de transporte de audio y video en tiempo real son RTP y RTCP. Para la transmisión de datos en streaming se utiliza en conjunto los protocolos RTP/RTCP y RTSP debido a que el protocolo SIP se integra perfectamente con muchos protocolos.

Los protocolos RTP y RTCP son protocolos a nivel de aplicación que no ejercen ningún tipo de influencia en las condiciones de la red IP, no controlan la calidad de servicio, sólo posibilitan que los receptores puedan manejar apropiadamente las perturbaciones (jitter, pérdida de secuencia) a que son sometidos los paquetes IP con contenidos de tiempo real al atravesar la red.

3.8.1 Real Time Protocol

El protocolo RTP (Real Time Protocol) está definido en el RFC 3550 [10] y es el encargado de transportar tanto audio como vídeo en tiempo real. Fue concebido para manejar el jitter y la pérdida de secuencia de los paquetes que introducen las redes IP, pudiendo ser empleado para el transporte de flujos (streams) sensibles al tiempo, por ejemplo voz y vídeo. Utiliza UDP como protocolo de transporte, ya que el uso de TCP y su control de flujo y congestión darían lugar a un retardo elevado durante la comunicación a causa de las retransmisiones.

Para llevar a cabo su función, el protocolo RTP hace uso de un número de secuencia, marcas de tiempo, envío de paquetes sin retransmisión, identificación del origen, identificación del contenido, sincronización, etc., lo que le permite en presencia de pérdidas, jitter o retardo poder continuar con la reproducción del flujo de paquetes. Por lo tanto no puede garantizar que la entrega de tráfico se haga en tiempo real, aunque sí garantiza al menos que lo hará de forma sincronizada [1].

RTP no garantiza reserva de recursos ni da garantía de calidad de servicio, tampoco provee ningún mecanismo para garantizar la entrega a tiempo, y asume que la red es segura y entrega los paquetes en secuencia [4].

Para comunicaciones seguras se dispone de SRTP (Secure Real Time Transport Protocol), una extensión de RTP que puede usarse opcionalmente para proporcionar confidencialidad, autenticación de mensajes y protección de reenvío para flujos de audio y video. SRTP fue publicado por primera vez por el IETF en marzo de 2004 bajo el RFC 3711 [11].

3.8.2 Real Time Control Protocol

El protocolo RTCP (Real Time Control Protocol) es el protocolo compañero de RTP. RTCP es el encargado de monitorizar el flujo de los paquetes RTP. Obtiene estadísticas sobre el jitter, RTT, latencia, pérdida de paquetes, etc. Fundamentalmente está relacionado con la calidad de servicio. Aunque realice una monitorización de la calidad de servicio de RTP, no se poseen mecanismos como reservar ancho de banda o control de la congestión para intentar apaciguar una situación en la que la calidad de la transmisión

no es suficiente. Es por ello por lo que la utilización de RTCP es opcional, aunque también recomendable [1].

3.8.3 Real Time Streaming Protocol

El protocolo RTSP (Real Time Streaming Protocol) fue desarrollado por RealNetworks, Netscape Communications y Columbia University y está publicado en el RFC 2326 [12], es un protocolo a nivel de aplicación para el envío de datos con propiedades de tiempo real que puede trabajar junto a otros protocolos como RTP y RSVP. Proporciona un entorno para el envío de datos de tiempo real bajo demanda, como lo son el audio y el vídeo. Las fuentes de datos pueden incluir tanto datos en directo, como almacenados. Este protocolo puede funcionar sobre UDP, UDP multicast y TCP [6].

En sintaxis y funcionamiento, es similar al protocolo HTTP, donde tanto el cliente y el servidor pueden hacer peticiones. No obstante, a diferencia de HTTP, el protocolo RTSP necesita mantener información de estado. Entre sus principales ventajas, se puede destacar que debido a sus similitudes con HTTP, hace que sea adaptable a proxys y firewalls, y es compatible con el modo de difusión multicast, siendo capaz de enviar la información a un grupo de clientes en un solo paso. Además, es independiente de la capa de transporte usada: puede utilizar tanto TCP como UDP.

3.9 Resource Reservation Protocol

El protocolo RSVP (Resource Reservation Protocol) se encuentra descrito en el RFC 2205 [14], es un protocolo de la capa de transporte diseñado para reservar recursos de una red bajo la arquitectura de servicios integrados (IntServ).

Es utilizado en VoIP para la gestión de QoS (Quality of Service), ya que hay que tener en cuenta que los paquetes IP son de longitud variable y el tráfico de datos suele ser a ráfagas. El propósito de RSVP es eliminar aquellas situaciones en las que la voz se pierde porque se tiene una ráfaga de datos en la red. Para ello, solicita ancho de banda, divide los paquetes de datos grandes y da prioridad a los paquetes de voz cuando hay una congestión en un router. Si bien este protocolo ayudará considerablemente al tráfico multimedia por la red, hay que tener en cuenta que RSVP no garantiza una calidad de servicio como sucede en redes avanzadas como ATM, que proporcionan servicios de QoS de forma estándar.

3.10 Codificación de la Voz

Para transportar la voz se utilizan algunos protocolos como SIP, IAX y otros como RTP o RTCP. La voz es una onda analógica que necesita transformarse a digital en algún formato antes de ser transmitida por la red IP. Para ello se hace uso de códecs que garanticen la codificación y compresión del audio o del video para su posterior decodificación y descompresión antes de poder generar un sonido o imagen utilizable.

La palabra códec proviene de abreviar las palabras CODificación y DECODificación. Según el códec utilizado en la transmisión, se utilizará más o menos ancho de banda. La cantidad de ancho de banda utilizada suele ser directamente proporcional a la calidad de los datos transmitidos. Los códecs realmente no son exclusivos de la VoIP, pues también se usan en otros tipos de comunicaciones digitales.

Entre los códecs utilizados en VoIP se encuentran: G.711, G.723.1 y G.729 (especificados por la ITU-T). Estos códecs tienen la siguiente tasa de bits en su señalización:

- G.711: bit-rate de 56/64 kbps.
- G.722: bit-rate de 48/56/64 kbps.
- G.723: bit-rate de 24/40kbps.
- G.726: bit-rate de 16/24/32/40 kbps.
- G.728: bit-rate de 16 kbps.
- G.729: bit-rate de 8 kbps.
- GSM: bit-rate de 13 kbps.
- iLBC: bit-rate de 15 kbps.

4. Session Initiation Protocol

4.1 Introducción

El protocolo SIP (Session Initiation Protocol) es un protocolo de señalización a nivel de aplicación encargado de la iniciación, modificación y terminación de sesiones multimedia, las cuales se llevan a cabo de manera interactiva. Por sesiones multimedia se refiere a aplicaciones de mensajería instantánea, aplicaciones de video, de audio, conferencias y aplicaciones similares.

SIP se definió en el RFC 2543 [15] en marzo de 1999 por el grupo de trabajo MMSC perteneciente a IETF. En junio de 2002, el IETF publicó una nueva revisión de SIP con el RFC 3261 [3].

El protocolo SIP es una parte de una arquitectura multimedia, ya que la única finalidad es la de gestionar las sesiones multimedia: iniciarlas, modificarlas, finalizarlas, etc. Sin embargo, se integra perfectamente con otros protocolos como RVSP, RTP o RTSP gracias al protocolo SDP que puede formar una completa arquitectura multimedia.

El protocolo SIP posee cuatro características que lo hacen muy recomendable para cumplir esta función:

- Localización del usuario: SIP posee la capacidad de poder conocer en todo momento la localización de los usuarios. De esta manera no importa en qué lugar se encuentre un determinado usuario. En definitiva la movilidad de los usuarios no se ve limitada.
- Negociación de los parámetros: Posibilidad de negociar los parámetros necesarios para la comunicación como lo son los puertos para el tráfico SIP así como el tráfico multimedia, direcciones IP para el tráfico multimedia, códec, etc.
- Disponibilidad del usuario: SIP permite determinar si un determinado usuario está disponible o no para establecer una comunicación.
- Gestión de la comunicación: Permite la modificación, transferencia, finalización de la sesión activa. Además informa del estado de la comunicación que se encuentra en progreso.

4.2 Direccionamiento

El protocolo es similar a HTTP por la forma en que funciona (protocolo basado en texto) y es similar a SMTP en la forma en la que se especifican las direcciones SIP. Las direcciones SIP identifican a un usuario de un determinado dominio. A estas direcciones SIP habitualmente se les llama URI (Uniform Resource Identifier). Una URI se puede especificar de las siguientes maneras:

SIP:usuario@dominio[:port]
SIP:usuario@direcciónIP[:port]

El dominio representa el nombre del proxy SIP que conoce la dirección IP del terminal identificado por el usuario de dicho dominio. El puerto por defecto para SIP es 5060,

aunque es posible especificar otros adicionales si es necesario. En la Tabla 4.1 se pueden ver algunos ejemplos de direcciones SIP.

Dirección SIP	Descripción
200@ciens.ucv.ve	El usuario "200" pertenece al dominio "ciens.ucv.ve".
200@192.168.1.2	El usuario "200" pertenece al dominio con dirección IP 192.168.1.2. La dirección IP puede referirse a la IP del usuario, en un momento determinado, o a su dominio. Por lo tanto es posible hacer uso de una dirección IP si no se dispone de un dominio registrado para este propósito.

Tabla 4.1: Ejemplos de Direcciones SIP

4.3 Elementos

Para una comunicación con el protocolo SIP es necesaria la intervención de varios elementos, donde cada uno desempeña su función. Los elementos de la comunicación definidos en el RFC 3261 [3] son:

4.3.1 User Agent

Los UAs (User Agent) manejan la señalización de SIP y estos se pueden dividir en dos categorías:

- User Agent Client (UAC): es un elemento que realiza peticiones SIP y acepta respuestas SIP provenientes de UAS. Un ejemplo de UAC es un teléfono VoIP ya que realiza peticiones SIP [1].
- User Agent Server (UAS): es el elemento encargado de aceptar las peticiones SIP realizadas por el UAC y enviar a este la respuesta conveniente. Un teléfono VoIP también es un ejemplo de UAS, ya que acepta las peticiones de inicio de comunicación enviadas por otro teléfono (UAC). Un servidor SIP o proxy también es un UAS [1].

4.3.2 Intermediary Entity

Los intermediarios son necesarios para que la comunicación entre dos UAs sea posible. Entre ellos se tiene: los proxys o proxy servers. Un proxy es un intermediario que actúa como un servidor y un cliente con el propósito de realizar peticiones en nombre de otros clientes [3], en otras palabras se encarga de reenviar las peticiones SIP provenientes de un UAC al UAS destino que corresponda, así como de encaminar las respuestas del UAS destino al UAC origen. Se puede hacer una similitud con el encaminamiento que realizan los router con los paquetes a nivel IP, es decir, sería como el encargado de enrutar los paquetes SIP [1].

4.3.3 Redirect Server

Es un UAS que genera respuestas 3xx a las solicitudes que recibe, en otras palabras envía al cliente un conjunto alternativo de URIs [3] que pueden indicar la dirección del destino o de otro servidor que lo acerque al destino. Estos servidores ayudan a reducir la carga de procesamiento en los Proxy Servers que son responsables de peticiones de enrutamiento.

4.3.4 Location Server

Es utilizado por un SIP redirect o por un proxy server para obtener información sobre la posible ubicación del destinatario de la llamada [3]. Es decir, acepta las peticiones de registro de los UACs, guardando toda la información referente a la localización física del UAC, para que así posteriormente llegue una petición con destino al UAC y sea posible localizarlo (traduce su dirección a la forma usuario@direcciónIP donde dirección se refiere a la IP del usuario) [1].

4.3.5 Back-to-Back User Agent

Un B2BUA (Back-to-Back User Agent) es una entidad lógica que procesa peticiones SIP entrantes comportándose como un UAS, y responde a estas actuando como un UAC regenerando por completo la petición SIP entrante en una nueva petición SIP que va a ser enviada. A diferencia de un proxy server, se mantiene el estado de diálogo y debe participar en todas las solicitudes enviadas en los cuadros de diálogo que tiene establecidos. Dado que es una concatenación de un UAC y UAS, no se necesitan definiciones explícitas para su comportamiento [3].

4.4 Peticiones

Son mensajes SIP enviados desde un cliente a un servidor, con el propósito de invocar una operación en particular. En la Tabla 4.2 se muestran las distintas peticiones SIP definidas en el RFC 3261 [3]:

Petición SIP	Descripción
INVITE	Permite invitar un usuario o servicio para participar en una sesión o para modificar parámetros en una sesión ya existente.
ACK	Confirma el establecimiento de una sesión. Esta petición es enviada por el usuario origen que envió la petición INVITE para hacer saber al usuario destino que su respuesta OK ha sido recibida. Es el momento en que ambos pueden empezar a enviar tráfico multimedia.
OPTION	Solicita información sobre las capacidades de un servidor. Un UA puede enviar peticiones OPTIONS a un UAS para solicitar cierta información sobre este.
BYE	Indica la terminación de la sesión establecida anteriormente con INVITE.
CANCEL	Cancela una petición previamente enviada por un cliente. Específicamente, se pide a los UAS cesar el procesamiento de la petición y generar una respuesta de error a esa petición. CANCEL no tiene efecto sobre peticiones en las que el UAS ya ha dado una respuesta definitiva, es por ello que es útil para cancelar peticiones en las que el servidor tarda mucho tiempo en responder. INVITE es un buen ejemplo de este tipo de peticiones.
REGISTER	Registra al User Agent. Un UAC envía peticiones REGISTER a un servidor de registro localización para informar de la posición actual en la que se encuentra en un momento determinado. Esto hace posible que el UAC pueda ser localizado haciendo uso de su misma dirección user@dominio sin importar donde el UAC se encuentre físicamente.

Tabla 4.2: Peticiones SIP

4.5 Respuestas

Son mensajes SIP enviados desde un servidor a un cliente, para indicar el estado de una petición enviada desde el cliente al servidor [3]. Cada petición SIP lleva asociada una respuesta enumerada con un código que la identifica. Estos códigos van desde el identificador 100 hasta el identificador 699 [3], siendo además agrupadas en grupos de respuestas tales como: 1xx, 2xx, 3xx, 4xx, 5xx y 6xx [1]. En la Tabla 4.3, se muestran las distintas respuestas SIP definidas en el RFC 3261 [3]:

Tipo de Respuesta	Identificador	Significado
Informan del estado provisional de la comunicación	100	Trying – Intentando
	180	Ringing – Sonando
	181	Call Is Being Forwarded – Llamada Está Siendo Transferida
	182	Queued – Encolada
	183	Session Progress – Llamada en Progreso
Informan del éxito de la comunicación	200	OK – OK
	202	Accepted – Aceptada
Informan del reenvío necesario de la petición SIP	300	Multiple Choices – Múltiples Opciones
	301	Moved Permanently – Movido Permanentemente
	302	Moved Temporarily – Movido temporalmente
	305	Use Proxy – Usar Proxy
	380	Alternative Service – Servicio Alternativo
Informan de errores del cliente	400	Bad Request – Mala Petición
	401	Unauthorized – No Autorizado
	402	Payment Required – Se Requiere Pago
	403	Forbidden – Prohibido
	404	Not Found – No Encontrado
	405	Method Not Allowed – Método no Permitido
	406	Not Acceptable – No es Aceptable
	407	Proxy Authentication Required – Se Requiere Autenticación
	408	Request Timeout – Tiempo Agotado para la Petición
	410	Gone – Ya no Existe
	413	Request Entity Too Large – Petición Demasiado Grande
	414	Request URI Too Long – URI Demasiado Largo
	415	Unsupported Media Type – Tipo de Medio no Soportado
	416	Unsupported URI Scheme – Esquema URI no Soportado
	420	Bad Extension – Extensión Incorrecta
421	Extension Required – Se Requiere Extensión	

	423	Interval Too Brief – Intervalo Demasiado Corto
	480	Temporarily Unavailable – No Disponible Temporalmente
	481	Call/Transaction Does Not Exist – No Existe la Llamada/Transacción
	482	Loop Detected – Bucle Detectado
	483	Too Many Hops – Demasiados Saltos
	484	Address Incomplete – Dirección Incompleta
	485	Ambiguous – Ambiguo
	486	Busy Here – Ocupado
	487	Request Terminated – Petición Terminada
	488	Not Acceptable Here – No Aceptable en este Momento o Aquí
	491	Request Pending – Petición Pendiente
	493	Undecipherable – Indescifrable
Informan de errores del servidor	500	Server Internal Error – Error Interno del Servidor
	501	Not Implemented – No Implementado
	502	Bad Gateway – Pasarela (Gateway) Equivocada
	503	Service Unavailable – Servicio no Disponible
	504	Server Time-out – Tiempo Agotado en el Servidor
	505	Version Not Supported – Versión no Soportada
	513	Message Too Large – Mensaje Demasiado Largo
Informan de errores generales	600	Busy Everywhere – Ocupado en todos Sitios
	603	Decline – Rechazado
	604	Does Not Exist Anywhere – No Existe en ningún Sitio
	606	Not Acceptable – No Aceptable

Tabla 4.3: Posibles Respuestas del Protocolo SIP

En la Figura 4.1 se muestra como se establece la comunicación entre el usuario A y el usuario B. Se asume que ambos usuarios se encuentran registrados en el proxy server.

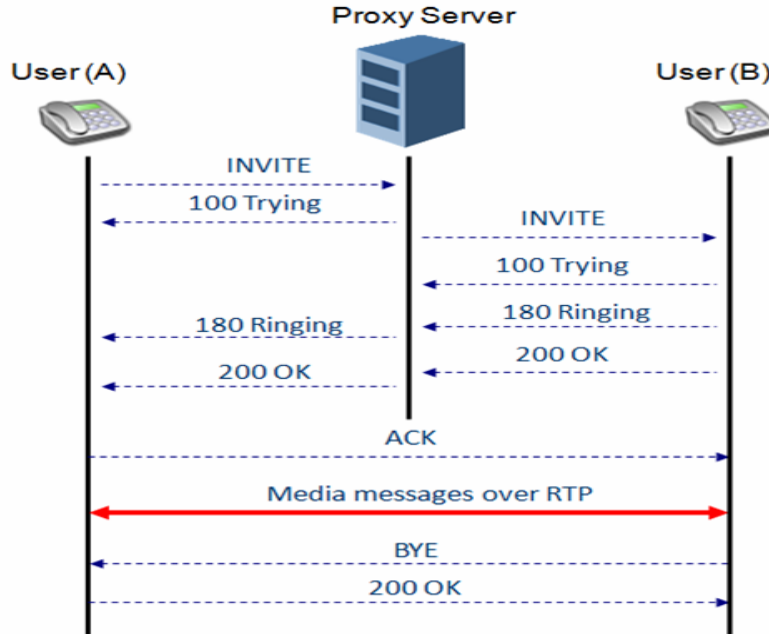


Figura 4.1: Establecimiento de la Comunicación entre el usuario A y el usuario B

4.6 Campos de los Mensajes SIP

Los campos que aparecen en los mensajes SIP dependen del tipo de mensaje y de los parámetros necesarios en cada uno de ellos. Siguen el mismo formato de la cabecera genérica de la Sección 2.2 del RFC 2822 [16] (Internet Message Format). Cada campo consiste en un nombre de campo seguido de dos puntos (":") y el valor del campo. Estos campos son similares a los campos de cabecera HTTP tanto en sintaxis como en semántica [3].

En la Tabla 4.4 se muestran los campos más frecuentes de los mensajes SIP tomados del RFC 3261 [3].

Campo	Descripción
Subject	Proporciona un resumen o indica la naturaleza de la llamada, lo que permite el filtrado de llamadas sin tener que analizar la descripción de la sesión.
Via	En este campo se almacena cada uno de los elementos por los que va pasando la petición. Almacenar el camino que sigue la petición desde su origen al destino es muy útil para las respuestas, ya que estas simplemente tienen que seguir el camino inverso.
Max-Forwards	Número máximo de saltos permitidos a la petición para llegar a su destino. Ese valor será decrementado en cada uno de los saltos por los que va pasando.
From	En ella se indica la entidad origen que envió la petición SIP. Se especifica mediante la URI o mejor dicho mediante el AOR (Address of Record) que es una URI global y pública que puede ser enrutada desde cualquier punto.
To	Hace referencia a la URI de destino o AOR de destino de la petición. No se utiliza para enrutar el paquete hacia el próximo

	salto, sino que siempre mantiene el destino de la petición inicial.
Call-ID	Es un identificador único y global que se forma mediante combinación de una cadena aleatoria, el nombre de la máquina o la dirección IP del teléfono. La combinación de los tags que se indican en TO, FROM junto con el Call-ID definen e identifican de manera unívoca un diálogo SIP entre dos extremos.
CSeq	Es un número de secuencia, donde cada nueva petición que se envía en un mismo diálogo incrementa en una unidad su valor. Es un contador de peticiones pertenecientes a un mismo diálogo.
Contact	En él se indica la SIP URI de la forma usuario@direcciónIP[:puerto] o usuario@dominio[:puerto] que representa la dirección de contacto directo con el emisor de la petición. La finalidad de esta dirección de contacto directo es la de que las futuras peticiones se puedan enviar directamente al emisor, evitando a la petición seguir el mismo camino, como sucede con las respuestas, que siguen el camino inverso dictaminado por las cabeceras VIA.
Content-type	Tipo del cuerpo del mensaje. No siempre tiene cuerpo el mensaje, pero si lo tuviera por lo general sería "application/SDP", esto es, el protocolo multimedia SDP.
Content-length	Tamaño del cuerpo del mensaje.

Tabla 4.4: Campos más Frecuentes de los Mensajes SIP

5. Problemas de Seguridad de SIP y Técnicas de Mitigación

5.1 Introducción

Recientemente han aparecido una serie de nuevas herramientas que hacen posible a cualquier persona atacar y cometer fraudes en equipos SIP, incluyendo en los sistemas basados en Asterisk. Estas herramientas hacen un barrido de redes en busca de hosts que ofrezcan servicios SIP. Una vez encontrados, las herramientas realizan un barrido en busca de extensiones y contraseñas. Existen ciertas reglas y técnicas fáciles de implementar que eliminan muchos de los problemas de seguridad, protegiendo de barridos masivos y de ataques posteriores.

Cabe destacar que la creciente utilización de redes inalámbricas supone en muchos casos una vía más a explotar por parte de los intrusos. Redes WiFi mal configuradas junto con una infraestructura de red insegura pueden facilitar el trabajo de los intrusos a la hora de acceder a la red VoIP para lanzar sus ataques.

5.2 Problemas de Seguridad de SIP

Paradójicamente una de las principales debilidades de la tecnología VoIP es apoyarse sobre una red potencialmente insegura como son las redes IP. VoIP hereda problemas de las capas y protocolos ya existentes.

Gran cantidad de ataques hacia las infraestructuras IP van a afectar irremediablemente a la telefonía: (1) ataques de denegación de servicio, (2) inundación de paquetes y (3) cualquier otro tipo de ataque que intente limitar la disponibilidad de la red suponen un gran problema para la telefonía IP. Además VoIP será vulnerable a ataques a bajo nivel: (1) secuestro de sesiones (hijacking), (2) interceptación, (3) fragmentación IP, (4) paquetes IP malformados y (5) spoofing.

VoIP no solo es susceptible a problemas heredados de las redes IP, también posee una gran cantidad de ataques específicos: (1) SPIT (Spam over Internet Telephony), (2) vishing (voice phising), (3) fuzzing, (4) secuestro de sesiones, (5) entre otros.

A continuación se detallan los ataques y vulnerabilidades de VoIP y de las infraestructuras IP:

- Contraseñas Débiles.
- Mala Política de Privilegios.
- Fuerza Bruta.
- Accesos Permisivos a Datos Comprometidos.
- Acceso Físico a Dispositivos Sensibles.
- Reinicio de Máquinas.
- DoS (Denial of Service).
- DDoS (Distributed Denial of Service).
- ICMP Unreachable.
- SYN Floods.

- Gran Variedad de Floods.
- SQL Injections.
- Denegación en DHCP.
- Buffer Overflows.
- Gusanos y Virus.
- Malas Configuraciones.
- Fraudes.
- SPIT (Spam over Internet Telephony).
- Vishing (voice phishing).
- Fuzzing.
- SIP Floods (INVITE, REGISTER, entre otros).
- Secuestro de Sesiones (hijacking).
- Interceptación (eavesdropping).
- Redirección de Llamadas (CALL redirection).
- Reproducción de Llamadas (CALL replay).
- Port Scanning.
- Footprinting.
- Sniffing.
- Ingeniería Social.
- Man-in-the-Middle.

En las secciones siguientes, se clasifican y se describen los ataques, amenazas y vulnerabilidades más significativas que afectan a la telefonía sobre redes IP:

5.2.1 Accesos Desautorizados y Fraudes

Los sistemas VoIP incluyen múltiples sistemas para el control de la llamada, administración, facturación y otras funciones telefónicas. Cada uno de estos sistemas debe contener datos (de facturación, registros, datos de cuentas, etc.) que si son comprometidos pueden ser utilizados para realizar fraudes. El costo de usar fraudulentamente esos datos VoIP a nivel empresarial puede ser devastador.

Una de las más importantes amenazas de las redes VoIP son los fraudes consecuencia de un acceso desautorizado a una red legal VoIP (por ejemplo obtener datos de cuentas). Una vez obtenido el acceso, el usuario desautorizado realiza llamadas (nacionales o internacionales) que generan grandes gastos a la empresa. Un control y registro estricto de las llamadas puede disminuir el problema [2].

Cabe destacar que las técnicas utilizadas por estos individuos son descendientes de las que utilizaban los famosos “phreakers” en las antiguas líneas telefónicas.

5.2.2 Ataques de Denegación de Servicio

Los ataques de denegación de servicio o DoS son intentos malintencionados de degradar seriamente el rendimiento de la red o un sistema incluso llegando al punto de impedir la utilización del mismo por parte de usuarios legítimos. Algunas técnicas se basan en el

envío de paquetes especialmente contruidos para explotar alguna vulnerabilidad en el software o en el hardware del sistema, saturación de los flujos de datos y de la red o sobrecarga de procesos en los dispositivos [2].

Los ataques distribuidos de denegación de servicio o DDoS son ataques DoS simples pero realizados desde múltiples computadores de forma coordinada. Las redes y sistemas VoIP son especialmente vulnerables a los ataques DDoS por diversas razones: (1) por la dependencia y la necesidad de garantías en la calidad de servicio, que hacen que las redes IP donde se mantienen llamadas telefónicas tengan una tolerancia mucho menor a problemas de rendimiento, y (2) porque en una red VoIP existen multitud de dispositivos con funciones muy específicas por lo que ataques contra casi cualquier dispositivo de la red pueden afectar seriamente los servicios de telefonía IP. Muchos de estos dispositivos son muy susceptibles de no manejar, priorizar o enrutar el tráfico de forma fiable si presentan un consumo de CPU alto. Por lo que muchos de los ataques de DoS se centran en atacar los dispositivos de red y/o inundar la red de tráfico inútil para degradar su funcionamiento y que los paquetes pertenecientes a comunicaciones telefónicas se pierdan o retrasen [2].

Las aplicaciones y los dispositivos de telefonía IP suelen trabajar sobre ciertos puertos específicos, bombardear dichos puertos con tráfico innecesario pero aparentemente real puede causar una denegación de servicio y que usuarios legítimos no puedan hacer uso del sistema. Modificaciones y ataques al servidor DNS pueden afectar de manera directa al servicio de voz. El robo o suplantación de identidad (del destinatario de la llamada o de algún otro dispositivo VoIP) generalmente deriva en una denegación de servicio. El acceso SNMP a los dispositivos, además de ofrecer una gran cantidad de información, permite potencialmente al atacante afectar al servicio de Voz sobre IP. En redes VoIP basadas en el protocolo SIP es posible enviar mensajes CANCEL, BYE o ICMP Port Unreacheable, con el objetivo de desconectar ciertos usuarios de sus respectivas llamadas o evitar que se produzcan por no permitir la correcta configuración inicial de la llamada (señalización) [2].

Las redes VoIP siguen siendo vulnerables a los tradicionales ataques de DoS como pueden ser los SYN flood, UDP flood, etc. Existen también ataques específicos de protocolos como SIP. Inundación de SIP o SIP Flooding es un tipo de ataque de denegación de servicio que consiste en atacar las capacidades de un servidor SIP como Asterisk o Elastix con peticiones SIP hasta conseguir que dejen de prestar el servicio. Las inundaciones con peticiones INVITE a toda la red VoIP puede llegar a colapsar las líneas y los servidores. Otros ataques de inundación similares se pueden producir también con las peticiones REGISTER y OPTIONS. Las aplicaciones VoIP escuchan en ciertos puertos determinados lo cual hace posible atacar esos servicios para causar un ataque DoS.

Existen una gran cantidad de programas disponibles en Internet para degradar el servicio de VoIP. Entre ellos se encuentran:

- Gran cantidad de flooders como por ejemplo: UDP flooder.
- Generadores de paquetes convencionales como por ejemplo: Scapy.
- Generadores de peticiones SIP como por ejemplo: SIP Tester.

En general existe una gran cantidad de métodos para sobrecargar la red y los servidores con el fin de conseguir una denegación de servicio. El problema se agrava con el hecho

que en una infraestructura IP pueden coexistir una gran cantidad de protocolos y de dispositivos que son vulnerables de una forma diferente cada uno.

5.2.3 Ataques a los Dispositivos

Muchos de los ataques realizados hoy en día por hackers y crackers hacia las redes de datos tienen como objetivo principal el hardware y el software de los dispositivos. Por lo tanto, en redes VoIP, los gateways, call managers, proxy servers y los teléfonos IP serán potencialmente objetivos a explotar por parte de un intruso [2].

El Fuzzing consiste en el envío masivo de paquetes malformados a un dispositivo. Muchas veces la implementación SIP de los dispositivos posee fallas, y es imposible utilizar un dispositivo mandando paquetes erróneos a propósito, ya que podrían darse overflows y se sobrescribirían posiciones de memoria que harían que el dispositivo dejase de responder. Si un dispositivo ha quedado inutilizable por este tipo de ataque, basta con reiniciarlo para que recupere su estado habitual [1].

Otra forma de atacar un dispositivo consiste en aprovechar errores en la configuración de los mismos. No es tan difícil encontrarse con las contraseñas por defecto, o la administración por telnet habilitada, lo que puede facilitar la labor de hackers que desean interrumpir el servicio de ese dispositivo, o conseguir credenciales del usuario [1].

5.2.4 Ataques y Vulnerabilidades a Nivel de Red IP

Se sabe que la tecnología VoIP se apoya sobre redes IP y que muchos ataques hacia IP la afectan de forma significativa. Uno de los mayores problemas es quizás la interceptación o eavesdropping. Es el término con el que se conoce a la captura de información (cifrada o no) por parte de un intruso al que no iba dirigida dicha información. En términos de telefonía IP, se habla de la interceptación de las conversaciones VoIP por parte de individuos que no participan en la conversación. El impacto de esta técnica es obtener toda clase de información sensible y altamente confidencial. Y aunque en principio se trata de un técnica puramente pasiva, razón por la cual hace difícil su detección, es posible intervenir también de forma activa en la comunicación insertando nuevos datos (que en el caso de VoIP se trataría de audio), redireccionar o impedir que los datos lleguen a su destino [2].

Sniffing es una práctica que consiste en escuchar los paquetes que circulan por la red. En algunos casos un atacante puede incluso escuchar el tráfico entre otros equipos ajenos al equipo donde se tiene acceso [18]. Para poder capturar el tráfico se utiliza una aplicación llamada Sniffer y existen algunas alternativas disponibles en Internet por lo que esta práctica se ha convertido en una tarea no tan difícil de llevar a cabo por hackers. Existen excelentes sniffers como Wireshark que permiten capturar todo el tráfico de un segmento de la red [2]. Si por el contrario se está dentro de redes conmutadas es necesario emplear otras técnicas para interceptar el tráfico de red como "Man-in-the-Middle" o MitM utilizando envenenamiento ARP (ARP Spoofing o ARP Poisoning). La técnica MitM es un ataque en el que el enemigo adquiere la capacidad de leer, insertar y modificar a voluntad los mensajes entre dos partes sin que ninguna de ellas conozca que el enlace entre ellos ha sido comprometido.

5.2.5 Enumeración y Descubrimiento

Una vez que el hacker ha seleccionado una red como su próximo objetivo, sus primeros pasos consistirán en obtener la mayor información posible de su víctima. Cuando el intruso tenga información suficiente evaluará sus siguientes pasos eligiendo el método de ataque más adecuado para alcanzar su objetivo. Normalmente el método de obtención de información se realiza con técnicas de menos a más nivel de intrusión. De este modo en las primeras etapas el atacante realizará un footprinting u obtención de toda la información pública posible del objetivo. Más adelante una de las acciones más comunes consiste en obtener la mayor información posible de las máquinas y servicios conectados en la red atacada (Port Scanning). Después de tener un listado de servicios y direcciones IP consistente, tratará de buscar agujeros de seguridad, vulnerabilidades y obtener la mayor información sensible de esos servicios (Enumeración) para poder explotarlos y conseguir una vía de entrada [2].

5.2.6 Ataques a Nivel de Aplicación

El nivel de aplicación de la red IP es quizás uno de los más vulnerables, debido en parte a que VoIP engloba gran cantidad de protocolos y estándares añadiendo cada uno de ellos su propio riesgo de seguridad. Un ejemplo claro de ellos es el protocolo SIP, muy discutido desde el punto de vista de la seguridad. Entre los ataques específicos contra el nivel de aplicación de VoIP encontramos ataques de secuestro de sesión, desconexiones ilegales, inundación de peticiones, generación de paquetes malformados, falsificación de llamadas, entre otros [2].

5.2.7 Ingeniería Social

El término “ingeniería social” hace referencia al arte de manipular personas para eludir los sistemas de seguridad. Esta técnica consiste en obtener información de los usuarios por teléfono, correo electrónico, correo tradicional o contacto directo.

El principio que sustenta la ingeniería social es el que en cualquier sistema “los usuarios son el eslabón débil”. Los atacantes de la ingeniería social usan la fuerza persuasiva y se aprovechan de la inocencia del usuario haciéndose pasar por un compañero de trabajo, un técnico o un administrador para poder obtener lo que se desea.

Phishing o suplantación de identidad es un tipo de ingeniería social caracterizado por intentar adquirir información confidencial de un usuario de forma fraudulenta.

5.2.8 Fuerza Bruta

La fuerza bruta es un ataque que busca de forma exhaustiva contraseñas, números, nombres de usuarios y cualquier otro tipo de información útil. El atacante prueba todas las combinaciones posibles hasta dar con la correcta.

Un ejemplo de fuerza bruta contra servidores VoIP es obtener una lista de extensiones telefónicas válidas. Información que sería extremadamente útil para ejecutar otros ataques como inundaciones INVITE o secuestro de registro.

5.3 Técnicas de Mitigación

Para proteger una arquitectura de red que implementa VoIP de algunos de los ataques antes mencionados se recomienda adoptar las siguientes medidas:

- Utilizar VLANs separadas para voz y datos, de manera que resulte más complicado (aunque no imposible) el acceso a la red de VoIP.
- No utilizar un gran número de softphones. Dado que los softphones son programas de escritorio, posibilitar su uso implica permitir el acceso de la red de datos a la red de VoIP, perdiendo así las ventajas logradas con la separación de redes.
- Mantener el firmware de los terminales actualizado. Es importante estar al día con los boletines de seguridad de los fabricantes, y en caso de anunciarse una vulnerabilidad realizar la actualización pertinente para evitar posibles ataques.
- Encriptar el audio de las llamadas mediante SRTP o ZRTP. Esto todavía no es posible con Asterisk, pero debería estar disponible en breve.
- Utilizar VPNs cifradas al realizar enlaces a través de Internet.
- Limitar el acceso al servidor VoIP.

Además, es posible utilizar herramientas de seguridad para tratar de mitigar estos ataques. En las siguientes secciones se describen algunas de estas herramientas:

5.3.1 BackTrack Linux

BackTrack Linux es una distribución GNU/Linux en formato LiveCD pensada y diseñada para la auditoría de seguridad y relacionada con la seguridad informática en general¹. Actualmente tiene una gran popularidad y aceptación en la comunidad que se mueve en torno a la seguridad informática.

Se deriva de la unión de dos grandes distribuciones orientadas a la seguridad, el Auditor + WHAX. Incluye una larga lista de herramientas de seguridad aptas para el uso, entre las que destacan numerosos escáneres de puertos y vulnerabilidades, archivos de exploits, sniffers, herramientas de análisis forense y herramientas para la auditoría Wireless.

5.3.2 BackBox

BackBox es una distribución Linux basada en Ubuntu, adaptada para ser usada en pruebas de seguridad². Se ha diseñado y desarrollado con el objetivo de ser rápida y sencilla de utilizar. Mantiene su propio repositorio que pretende estar siempre actualizado con la última versión estable de las herramientas más conocidas y utilizadas en las pruebas de hacking.

Entre las herramientas que incluye se pueden destacar: Zenmap, hping3, Xprobe, Wireshark, Ettercap-gtk, dsniff, Scapy, Yersinia, OpenVAS, WhatWeb, Plecost, SQLmap, Paros, Nikto, w3af, Aircrack-ng y Kismet.

¹ <http://www.backtrack-linux.org>

² <http://www.backbox.org>

5.3.3 Snort

Snort es un sniffer de paquetes y un detector de intrusos basado en red (se monitoriza todo un dominio de colisión) [19]. Es un software muy flexible que ofrece capacidades de almacenamiento de sus bitácoras tanto en archivos de texto como en bases de datos abiertas como lo es MySQL. Implementa un motor de detección de ataques y barrido de puertos que permite registrar, alertar y responder ante cualquier anomalía previamente definida³. Así mismo existen herramientas de terceros para mostrar informes en tiempo real (ACID) o para convertirlo en un Sistema Detector de Intrusos (IDS).

5.3.4 Kali Linux

Kali Linux es una distribución de Linux de software libre avanzada que se adhiere completamente a los estándares de desarrollo de Debian GNU/Linux y que sirve para pruebas de penetración y auditorías de seguridad. Kali Linux se distribuye en imágenes ISO compiladas para diferentes arquitecturas (32/64 bits y ARM) y las mismas pueden ser descargadas desde su página web oficial⁴.

Posee más de 600 herramientas de pruebas de penetración y agrega miles de paquetes de software libre en su sección principal. Como derivado de Debian GNU/Linux todo el software cumple con las guías de software libre de Debian GNU/Linux.

En términos de seguridad informática, BackTrack Linux⁵ ha sido una de las distribuciones de Linux que más se ha utilizado para realizar auditorías de seguridad. Offensive Security, quién también desarrolló BackTrack Linux, lanzó una nueva distribución llamada “Kali Linux” que promete ser la más “avanzada, robusta y estable” distribución de seguridad basada en Debian GNU/Linux.

Durante los últimos años BackTrack Linux se ha ganado el lugar como una de las mejores distribuciones para profesionales de la seguridad informática, pero con cada nueva versión se volvía más lento, pesado e incluía cosas que realmente muy pocas personas usaban, esto permitió que otras distribuciones crecieran en popularidad y tomaran fuerza. Offensive Security estaba consciente de esta realidad y anunció hace tiempo que la versión 5 R3 sería la última versión de BackTrack y que la versión 6 de BackTrack pasaría a llamarse Kali Linux. La versión 1.0.0 de Kali Linux fue lanzada el 13 de marzo de 2013.

Kali Linux posee muchas herramientas de seguridad instaladas y muchas otras se pueden descargar de los repositorios de Kali Linux. A continuación se listan algunas de las herramientas que Kali Linux ofrece:

- Análisis DNS: dnsdict6, dnsenum, dnsmap, dnsrecon, dnsrevenue6, dnstracer, dnswalk, Fierce, Maltego, Nmap, URLLCrazy.
- Análisis de Tráfico: cdpsnarf, intrace, irpas-ass, irpass-cdp, p0f, tcpflow, Wireshark, Ettercap.
- Análisis de VoIP: ACE, enumIAX.
- Herramientas para VoIP: iaxflood, inviteflood, ohrwurm, protos-sip, rtpbreak, rtpflood, rtpinsertsound, rtpmixsound, sctpscan, SIPArmyKnife, SIPp, SIPsak, svcrack, svcraash, svmap, svreport, svwar, VoIPHopper, SIPDump, SIPcrack.

³ <https://www.snort.org>

⁴ <https://www.kali.org>

⁵ <http://www.backtrack-linux.org>

- Herramientas de Tunneling: cryptcat, dbd, dns2tcp, dns2tcpd, iodine, miredo, ncat, proxychains, proxytunnel, ptunnel, pwnat, sbd socat, sslh, stunnel4.

6. Elastix

6.1 Introducción

Elastix es un software de código abierto para el establecimiento de comunicaciones unificadas (servidor de fax, mensajería instantánea, servidor de correos, video conferencia y Voz sobre IP). Su objetivo principal es incorporar en una única solución todos los medios y alternativas de comunicación existentes en el ámbito empresarial.

Implementa gran parte de su funcionalidad sobre cuatro programas de software muy importantes como son: (1) Asterisk, (2) HylaFAX, (3) openfire y (4) Postfix. Estos brindan las funciones de PBX, fax, mensajería instantánea y correo electrónico, respectivamente. Elastix corre sobre CentOS como sistema operativo y actualmente su versión más estable es Elastix 2.5.0.

6.2 Historia

Asterisk es un programa de software libre (bajo licencia GPL) que proporciona funcionalidades de una central telefónica (PBX). Como cualquier PBX, se puede conectar un número determinado de teléfonos para hacer llamadas entre sí e incluso conectar a un proveedor de VoIP. Fue creado por Mark Spencer, de Digium y actualmente es su principal desarrollador junto con otros programadores que han contribuido a corregir errores y añadir novedades y funcionalidades. Reconoce muchos protocolos de VoIP como: (1) H.323, (2) SIP, (3) IAX/IAX2 y (4) MGCP.

Elastix fue creado y es permanentemente desarrollado por la empresa Ecuatoriana PaloSanto Solutions. El proyecto Elastix se inició como una interfaz para mostrar el registro de llamadas en Asterisk y fue liberado en marzo del 2006. Posteriormente el proyecto evolucionó hasta convertirse en una distribución basada en Asterisk que permite el establecimiento de comunicaciones unificadas.

A medida que la demanda de clientes de Asterisk incrementaba, PaloSanto Solutions necesitaba estandarizar su instalación para la implementación de telefonía IP. Con este objetivo decide elaborar una solución que sea capaz de instalar Asterisk partiendo de la instalación del sistema operativo sobre un servidor y agregando otras funcionalidades que incluían una interfaz gráfica para administración y configuraciones básicas. Inicialmente fue implementada en varios clientes con mucho éxito, por lo cual deciden lanzar la solución bajo la licencia GPLv2, siguiendo los principios de la compañía que apuntaban al open source como estructura principal de negocios.

Al principio el número de descargas no fue tan alto, pero con más experiencia PaloSanto Solutions lanza una nueva versión incluyendo algunas mejoras y poco a poco varias personas empiezan a colaborar en el proyecto. En el año 2007, la comunidad de Elastix crece, incrementando el número de descargas y usuarios alrededor del mundo. Actualmente Elastix ha superado el millón de descargas y ha incorporado varios socios tecnológicos que colaboran en la compatibilidad del software con hardware de telefonía.

En el año 2010, PaloSanto Solutions lanza Elastix 2.0 incorporando Asterisk 1.6, un módulo de Addons, su propio panel de operador (Elastix Operator Panel), Faxing basado en web, entre otras mejoras y desarrollos. Ese año lanza también ElastixWorld, un evento

destinado a reunir a la comunidad de Elastix a nivel mundial y cuya primera edición se realizó en Quito, Ecuador.

Elastix se distribuye como imagen ISO y puede ser descargada desde www.elastix.org o desde la página del proyecto en SourceForge.

6.3 Arquitectura

Debido a que la telefonía es el medio tradicional que ha liderado las comunicaciones durante el siglo pasado, muchas empresas y usuarios centralizan sus requerimientos únicamente en sus necesidades de establecer telefonía en su organización confundiendo distribuciones de comunicaciones unificadas con equipos destinados a ser centrales telefónicas. Sin embargo Elastix no solamente provee telefonía, integra otros medios de comunicación para hacer más eficiente y productivo su entorno de trabajo. Elastix incluye en su solución los siguientes medios de comunicación: (1) telefonía IP, (2) servidor de correo, (3) servidor de fax, (4) conferencias, (5) servidor de mensajería instantánea. Al tener una integración de diferentes sistemas de comunicaciones se logra una mejor productividad (ahorro de tiempo y papel, y mejor acceso a la información compartida) en cualquier ambiente de trabajo. En la Figura 6.1 (tomado de [17]) se muestra la capa de comunicaciones dentro de un esquema general de Elastix.

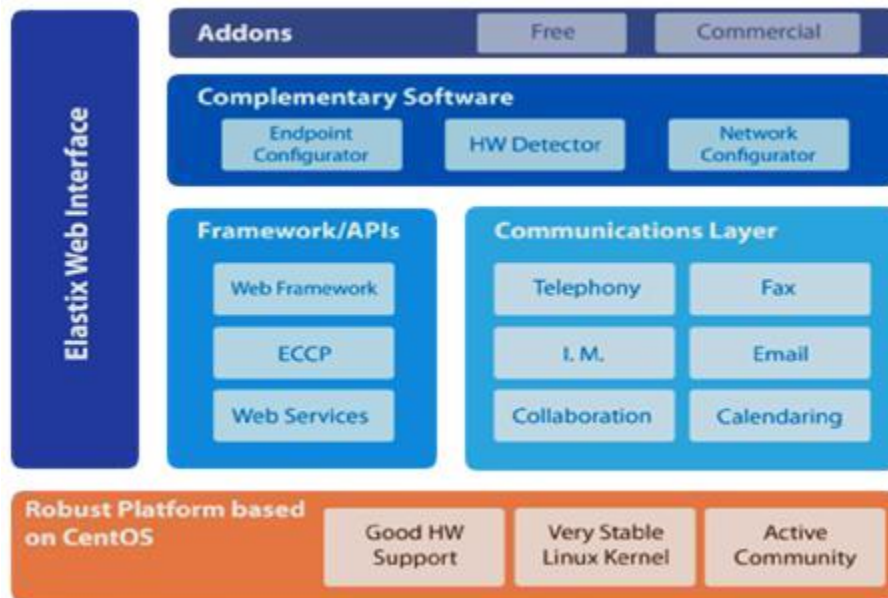


Figura 6.1: Esquema General de los Componentes de Elastix

6.4 Características

Elastix tiene múltiples características y funcionalidades relacionadas con los servicios que presta. Nuevas características, funcionalidades y servicios son añadidos en el desarrollo de nuevas versiones. A continuación se detallan las características y funcionalidades de Elastix⁶:

⁶ <http://www.elastix.org>

6.4.1 General

- Ayuda en línea embebida.
- Administración centralizada de actualizaciones.
- Monitor de recursos del sistema.
- Soporte para backup/restore a través de web.
- Configurador de parámetros de red.
- Soporte para temas o skins.
- Control de apagado/re-encendido de la central vía web.
- Soporte para configuración de fechas en el servidor, horas y zonas horarias.
- Control de acceso a la interfaz, basado en ACLs.

6.4.2 PBX

- Grabación de llamadas.
- Centro de conferencias con salas virtuales.
- Correo de voz.
- Soporte para protocolos SIP e IAX, entre otros.
- Códecs soportados: ADPCM, G.711 (A-Law & μ -Law), G.722, G.723.1 (pass through), G.726, G.728, G.729, GSM, iLBC (opcional) entre otros.
- IVR configurable y flexible.
- Soporte para interfaces análogas como FXS/FXO (PSTN/POTS).
- Soporte para sintetización de voz.
- Soporte para interfaces digitales E1/T1/J1 a través de los protocolos PRI/BRI/R2.
- Herramienta para la creación de extensiones por lote.
- Identificación de llamadas.
- Cancelador de eco integrado.
- Troncalización.
- Provee teléfonos vía web.
- Rutas entrantes y salientes con configuración por coincidencia de patrones de marcado.
- Soporte para videófonos.
- Soporte para follow-me.
- Interfaz de detección de hardware.
- Soporte para grupos de timbrado.
- Servidor DHCP para asignación dinámica de IP.
- Soporte para paging e intercom.
- Panel de operador basado en web.
- Soporte para condiciones de tiempo.
- Reporte de detalle de llamadas (CDR).
- Soporte para DISA (Direct Inward System Access).
- Tarifación con reporte de consumo por destino.

- Soporte para callback.
- Reportes de uso de canales.
- Soporte para interfaces tipo bluetooth a través de teléfonos celulares (chan_mobile).
- Soporte para colas de llamadas.

6.4.3 Fax

- Servidor fax basado en HylaFAX.
- Personalización de faxes-a-email.
- Visor de faxes integrado con PDFs descargables.
- Control de acceso para clientes de fax.
- Aplicación fax-a-email.
- Puede ser integrada con Winprint HylaFAX.

6.4.4 Colaboración

- Calendario integrado con PBX con soporte para recordatorios de voz.
- Libreta telefónica (Phonebook) con capacidad click to call.
- Dos productos CRM integrados a la interfaz (vtigerCR and SugarCRM).
- Web Conference.
- Nuevas características en el Módulo Calendario.

6.4.5 Mensajería Instantánea

- Servidor de mensajería instantánea basado en openfire.
- Reporte de sesiones de usuarios.
- Inicio de llamadas desde cliente de mensajería.
- Soporte Jabber.
- Servidor de mensajería es configurable desde web.
- Soporte de plugins.
- Soporta grupos de usuarios.
- Soporte LDAP.
- Soporta conexión a otras redes de mensajería como MSN, Yahoo Messenger, GTalk, ICQ.
- Soporta conexiones server-to-server para compartir usuarios.

6.4.6 Email

- Servidor de Email con soporte multidominio.
- Soporte para cuotas.
- Administración centralizada vía web.
- Soporte antispam.
- Interfaz de configuración de relay.
- Basado en Postfix para un alto volumen de correos.

- Cliente de email basado en web.
- Módulo de SMTP remoto.
- Administración de listas de emails.

7. Arquitectura

7.1 Introducción

En este capítulo se propone describir las arquitecturas que representan los escenarios de pruebas para el estudio de seguridad en implementaciones VoIP. Se presentan 3 escenarios de pruebas con diferentes características, que representan arquitecturas de redes con soporte para VoIP, ampliamente usadas:

- Escenario 1: una red LAN con soporte VoIP y conexiones a la PSTN (Public Switched Telephone Network) a través de tarjetas analógicas/digitales y al Internet.
- Escenario 2: una red LAN con soporte VoIP y una conexión a un ITSP (Internet Telephone Service Provider) a través de una de las redes WAN.
- Escenario 3: una red LAN con soporte VoIP. Adicionalmente, se pueden conectar remotamente estaciones de trabajo y teléfonos IP.

A continuación se describen con más detalles cada uno de estos casos.

7.1.1 Conexión a un PSTN

La estructura corresponde a una red LAN donde se tienen computadores, teléfonos IP y computadores conectados en puente a través de un teléfono IP. Se estima un número máximo de 100 teléfonos para la red. Adicionalmente, esta red LAN posee una conexión a la PSTN a través de una tarjeta analógica OpenVox modelo A400E y al Internet (ver Figura 7.1).

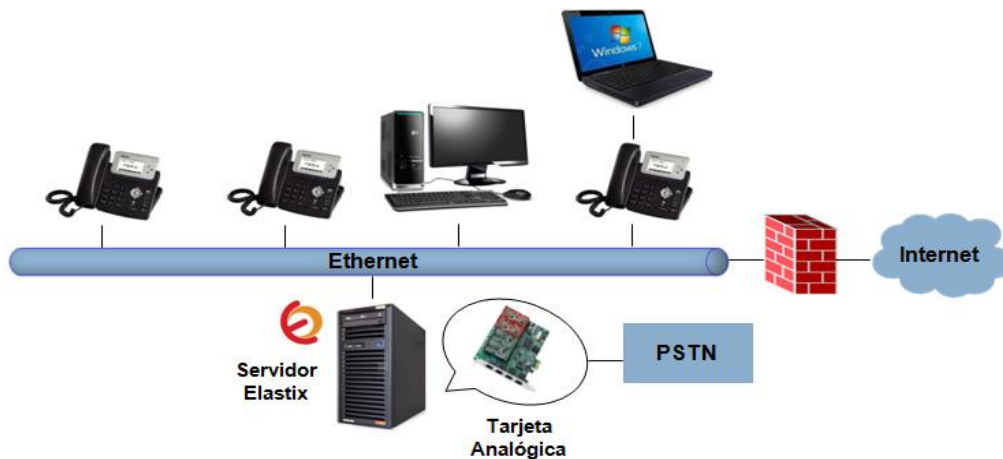


Figura 7.1: Conexión a un PSTN

7.1.2 Conexión a un ITSP

La estructura incluye a una red LAN donde se tienen computadores, teléfonos IP y computadores conectados en puente a través de un teléfono IP. Se estima igualmente un número máximo de 100 teléfonos para la red. Esta red LAN establece una conexión a un ITSP para las llamadas telefónicas exteriores en lugar de ser directamente conectada a la PSTN como se puede ver en la Figura 7.2.

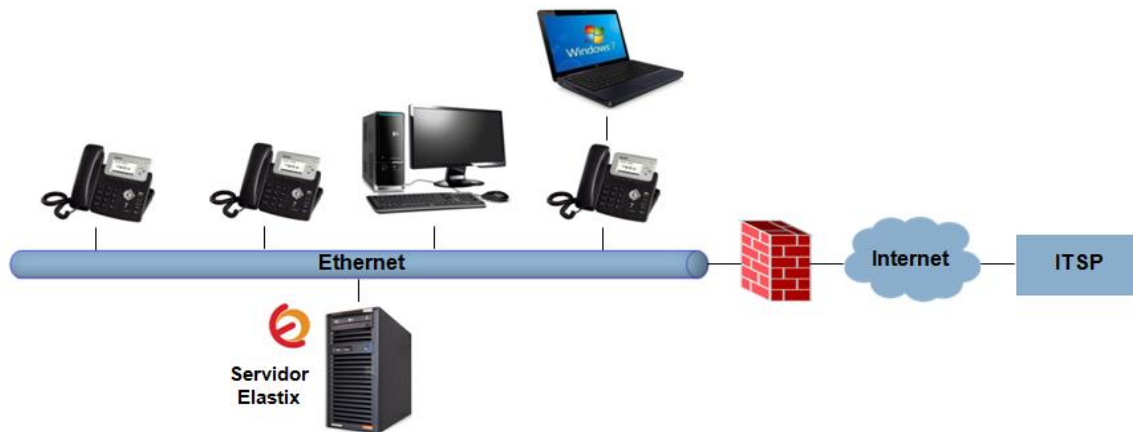


Figura 7.2: Conexión a un ITSP

7.1.3 Conexión Remota a una Red LAN

La estructura de la red es una LAN donde se tienen computadores, teléfonos IP y computadores conectados en puente a través de un teléfono IP. Se estima un máximo de 100 teléfonos. Adicionalmente a esta red pueden conectarse estaciones de trabajo remotas a través de Internet. En cada estación remota puede utilizarse un softphone o un teléfono IP como se muestra en la Figura 7.3.

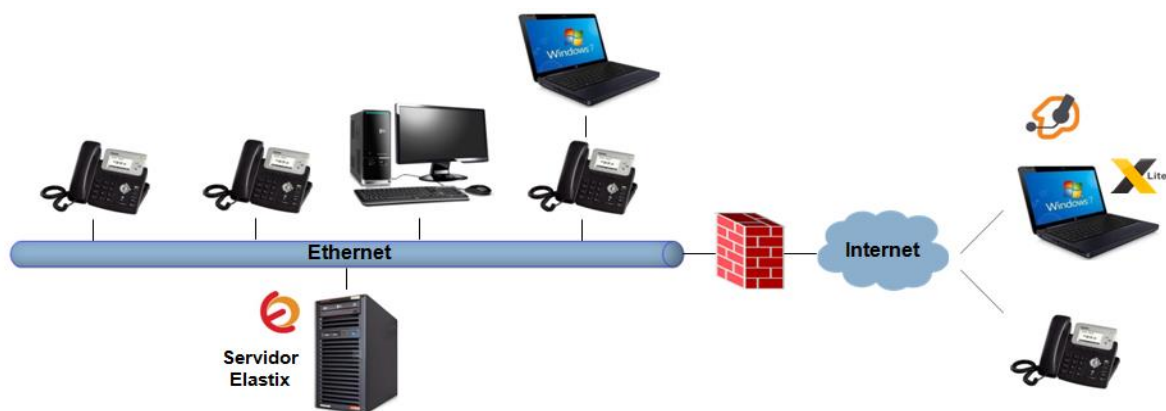


Figura 7.3: Conexión Remota a una Red LAN

7.2 Servidor VoIP

Para las arquitecturas descritas se propone el uso de un servidor de comunicaciones unificadas (Elastix 2.4.0) para la red LAN. De acuerdo a la red, se implementará una configuración básica de Elastix que involucra la configuración de extensiones y teléfonos IP. Se elige Elastix porque es una solución de telefonía VoIP y comunicaciones unificadas de código abierto que integra las mejores herramientas disponibles y funcionalidades en una sencilla interfaz de fácil administración⁷.

⁷ <http://www.elastix.org>

7.3 Ataques y Vulnerabilidades

Los ataques a las arquitecturas expuestas se pueden generar con una distribución Linux que permita hacer pruebas de penetración para VoIP. En la Tabla 7.1 se muestra una comparación de algunas distribuciones Linux conocidas que son orientadas a pruebas de penetración y auditorías de seguridad.

Distribución	BackTrack Linux	BackBox	Kali Linux
Licencia	Software Libre	Software Libre	Software Libre
Nro. De Herramientas	Más de 700 herramientas de pruebas de penetración	Más de 100 herramientas incluidas	Más de 600 herramientas de pruebas de penetración
Soporte VoIP	Gran variedad de herramientas	Pocas herramientas	Gran variedad de herramientas
Documentación	Libros, trabajos, tutoriales, manuales, entrenamientos.	Libros, tutoriales, guías.	Libros, trabajos, tutoriales, manuales, entrenamientos.
Estado Actual	Reemplazado por Kali Linux	Activo	Activo

Tabla 7.1: Comparación de Distribuciones para Pruebas de Penetración

Kali Linux⁸ posee una gran variedad de herramientas para las pruebas de penetración y un número significativo de herramientas para el uso en redes VoIP. El libro “Mastering Kali Linux for Advanced Penetration Testing” [30], dedicado a la utilización de Kali Linux en la realización de pruebas de penetración en redes se menciona a Kali Linux como la distribución con las herramientas más importantes para la seguridad de las redes y datos. Debido a que las pruebas de penetración deben hacerse en arquitecturas con soporte VoIP se propone a Kali Linux como la distribución responsable de los ataques por ser un proyecto activo que reemplaza a Backtrack⁹ y por poseer una mayor variedad de herramientas¹⁰ para VoIP que BackBox¹¹. Adicionalmente, Kali Linux es un proyecto con bastante documentación en la red y posee sitios web oficiales para servir a los usuarios.

7.4 Especificaciones Técnicas

El servidor de comunicaciones unificadas seleccionado es Elastix 2.4.0 basado en la distribución Linux CentOS 5.9 para la arquitectura Intel x86 en un equipo con procesador Intel(R) Core(TM) i3-3250 CPU de 3.50GHz, 2 GB de RAM y una tarjeta analógica OpenVox A400E con 2 puertos FXS y 2 puertos FXO sobre un slot PCI Express X1. El Elastix 2.4.0 integra varios paquetes de software y en la Tabla 7.2 se listan cuáles son estos paquetes y sus versiones:

⁸ <https://www.kali.org>

⁹ <http://www.backtrack-linux.org>

¹⁰ <http://tools.kali.org/>

¹¹ <http://www.backbox.org>

Nombre	Paquete	Versión
Elastix	elastix	2.4.0-1
	elastix-my_extension	2.4.0-1
	elastix-system	2.4.0-1
	elastix-im	2.4.0-1
	elastix-a2billing	1.9.4-5
	elastix-firstboot	2.4.0-1
	elastix-addons	2.4.0-1
	elastix-extras	2.4.0-1
	elastix-vtigercrm	5.2.1-7
	elastix-security	2.4.0-1
	elastix-framework	2.4.0-1
	elastix-fax	2.4.0-1
	elastix-reports	2.4.0-1
	elastix-email_admin	2.4.0-1
	elastix-portknock	0.0.1-0
	elastix-agenda	2.4.0-1
elastix-asterisk-sounds	1.2.3-1	
elastix-pbx	2.4.0-1	
RounCubeMail	RoundCubeMail	0.3.1-12
Mail	postfix	2.3.3-6.el5
	cyrus-imapd	2.3.7-12.el5_7.2
IM	openfire	3.7.1-1
FreePBX	freePBX	2.8.1-16
Asterisk	asterisk	1.8.20.0-0
	asterisk-perl	0.10-2
	asterisk-addons	1.8.20.0-0
FAX	hylafax	4.3.10-2rhel5
	iaxmodem	1.2.0-2
DRIVERS	dahdi	2.10.0.1-3
	rhino	0.99.6-0.b2
	wanpipe-util	3.5.28-0

Tabla 7.2: Detalle de las Versiones de los Paquetes de Elastix 2.4.0

Los ataques en las diferentes arquitecturas de red con soporte VoIP se llevan a cabo con la distribución Kali Linux 1.1.0 basada en Debian GNU/Linux para arquitectura Intel x86. Se utiliza una máquina virtual en VirtualBox de Oracle para virtualizar Kali Linux con 60 GB de disco duro, 2048 MB de memoria RAM y tarjeta de red en modo bridge.

8. Trabajos Relacionados

En este capítulo se abordará una serie de trabajos que se relacionan con la seguridad en ambientes VoIP. Estos trabajos abordan la seguridad en implementaciones VoIP, en servidores Elastix y en el protocolo SIP.

8.1 Seguridad en Implementaciones VoIP

En el trabajo “Security Considerations for Voice Over IP Systems” [20], se revisan las tecnologías y soluciones de VoIP. Con la introducción de VoIP, la necesidad de seguridad se agrava porque se tiene que proteger dos recursos muy valiosos: (1) los datos y (2) la voz. Agencias del Gobierno Federal por ley están obligadas a proteger a una gran cantidad de información, incluso si no es clasificada. La arquitectura actual de Internet no proporciona la misma seguridad del cable físico como las líneas telefónicas. La clave para asegurar VoIP es utilizar los mecanismos de seguridad de las redes de datos (firewalls, encriptación, etc.) para emular el nivel de seguridad que actualmente gozan los usuarios de la red PSTN. Este trabajo investiga los ataques y defensas pertinentes para VoIP y explora maneras de proporcionar niveles adecuados de seguridad para las redes de VoIP a un costo razonable [20].

La calidad de servicio es fundamental para el funcionamiento de una red VoIP. A pesar del gran ahorro que VoIP proporciona a sus usuarios, si no se puede entregar la misma calidad en el establecimiento de la llamada y la transmisión de voz que una red telefónica tradicional entonces se proporcionará poco valor añadido. La aplicación de diversas medidas de seguridad puede degradar la calidad de servicio. Estas medidas van desde el retardo o bloqueos de llamadas por firewalls hasta la latencia o retardo variable producido por la encriptación. Debido a la naturaleza crítica de VoIP y su poca tolerancia a la pérdida y alteración de paquetes, muchas de las medidas de seguridad implementadas en las redes de datos tradicionales simplemente no son aplicables a VoIP en su forma actual. Los principales problemas de calidad de servicio asociados con VoIP que afectan la seguridad: (1) latencia, (2) retardo variable, (4) ancho de banda y (3) paquetes perdidos [20].

Otro trabajo orientado a la seguridad de sistemas VoIP es “Seguridad en Implementaciones de Voz Sobre IP” [21]. Dicho trabajo se centra en mostrar un panorama general de los lineamientos y consideraciones a tomar en cuenta para brindar seguridad a plataformas de Voz sobre IP [21]. Para ello se describen algunas amenazas potenciales del protocolo de señalización SIP y se especifican soluciones de seguridad en plataformas VoIP.

Algunas amenazas no son muy diferentes de las que existen actualmente en las redes de datos. Las soluciones propietarias tienen una amplia gama de productos para cada necesidad que a menudo representan tendencias en la tecnología y servicios en el mercado que luego se convertirán en las necesidades del cliente. Soluciones de código abierto no están libres de los informes de seguridad. La ventaja es que por un lado hay empresas que están detrás del desarrollo y por otro lado se encuentran apoyadas por comunidades de desarrolladores [21].

8.2 Seguridad en Servidores Elastix

El aumento en las instalaciones de servidores de comunicaciones se debe en gran parte al avance continuo de la VoIP y la telefonía. De una misma manera los ataques y vulnerabilidades son mayores cada día. Muchos artículos son elaborados para tratar de mitigar los problemas de seguridad que se pueden presentar en los servidores Elastix. En el artículo “Seguridad en Servidores CentOS con Elastix” [22], se presentan ciertas recomendaciones y configuraciones básicas del sistema operativo CentOS, extraídos de libros, manuales y experiencias personales del autor que ayudan a mantener seguro un servidor Elastix.

Muchos trabajos y artículos se encargan de detallar especificaciones de seguridad al momento que se implanta VoIP dado que constantemente aparecen nuevas vulnerabilidades y ataques. Por ejemplo, el artículo “Asegurando Elastix” [23], se enfoca en proporcionar estas especificaciones de seguridad.

Existen ciertas reglas, de aplicación inmediata, que eliminan muchos de los problemas de seguridad, protegiendo al servidor Asterisk y/o Elastix de los barridos masivos y los ataques posteriores. Estos métodos y herramientas de protección ya existen, simplemente hay que aplicarlos [23].

8.3 Seguridad en el Protocolo SIP

La seguridad del protocolo SIP es un aspecto fundamental a la hora de establecer y finalizar una sesión. El trabajo “Seguridad del Protocolo SIP en la VoIP” [24], tiene como objetivo mejorar la seguridad del protocolo SIP en VoIP. Para ello se presentan dos propuestas que tienen como línea fundamental el cifrado de este protocolo. En el trabajo se abordan las características fundamentales del protocolo de señalización SIP y los mecanismos de establecimiento y liberación de una sesión en este. Se exponen además las principales características, propósitos, ventajas y desventajas que se obtienen con la implementación de las dos propuestas citadas anteriormente [24].

AES [26] es el algoritmo criptográfico que se propone para garantizar la seguridad del protocolo SIP en el trabajo “Seguridad del Protocolo SIP en la VoIP” [24]. El mismo fue escogido principalmente por ser un algoritmo simétrico siendo el proceso de encriptar y desencriptar mucho más rápido que en los asimétricos. El detalle anterior se ha tenido en cuenta ya que la VoIP requiere una alta inmediatez. En comparación con su predecesor DES [27], AES es rápido tanto en software como en hardware y requiere poca memoria para efectuar el proceso de encriptar y desencriptar. También proporciona mayor rapidez y menor costo computacional que 3DES [28], pudiendo ser implementado en equipos con bajo requerimiento de memoria, por ejemplo, un teléfono VoIP [24].

El protocolo SIP tiene un número de mecanismos de seguridad y algunos de ellos se han integrado en el propio protocolo, por ejemplo, la autenticación HTTP. Estos mecanismos tienen algoritmos y parámetros alternativos. Tres campos en la cabecera de SIP se utilizan para la negociación de los mecanismos de seguridad entre un UAC y un UAS [20]. Es un estándar propuesto en el RFC 3329 [25] y actualmente se encuentran soportados los mecanismos: (1) TLS, (2) HTTP digest authentication, (3) IPsec with IKE, (4) manually keyed IPsec without IKE y (5) S/MIME.

9. Marco Metodológico

Para lograr cumplir con los objetivos planteados en el Capítulo 2, es necesario definir un esquema o metodología de trabajo que permita la elaboración de un conjunto de lineamientos de manera estructurada y organizada. A continuación se presenta la especificación de la metodología utilizada.

9.1 Adaptación de la Metodología de Desarrollo

Una prueba de penetración consiste en el análisis sistemático de todas las medidas de seguridad establecidas con el propósito de preservar la confidencialidad, integridad y disponibilidad de la información. Señala vulnerabilidades y documenta cómo esas debilidades pueden ser tratadas. Existen muchas metodologías de pruebas de penetración y entre ellas se destaca la metodología “Penetration Testing: Procedures and Methodologies” definida en el libro de EC-Council [29].

Se definen tres tipos de pruebas de penetración:

- Pruebas Caja-Negra (pruebas de conocimiento cero): con el fin de simular ataques del mundo real y minimizar los falsos positivos se opta por realizar las pruebas caja negra. En este tipo de pruebas no se posee conocimiento o información del sistema a perpetrar.
- Pruebas Caja-Blanca (pruebas de conocimiento completo): si se necesita evaluar la seguridad frente a un tipo específico de ataque o un objetivo determinado se puede proporcionar la información completa sobre la red para las pruebas de penetración. La información proporcionada puede incluir documentos de la topología de red, inventario de activos e información de valor. Típicamente se opta por esto cuando se quiere una auditoría completa de seguridad.
- Pruebas Caja-Gris: es el método más común para probar las vulnerabilidades que un atacante puede encontrar y explotar. El funcionamiento del proceso de pruebas es similar a las pruebas caja negra. Tanto el equipo de ataque como los usuarios normales se proporcionan con los mismos privilegios. El objetivo de estas pruebas es simular un ataque de un malicioso empleado.

9.1.1 Análisis de Vulnerabilidades

El proceso de evaluación de vulnerabilidades implica reconocer, medir y priorizar las vulnerabilidades de un sistema. Ayuda a conocer las amenazas y vulnerabilidades de la infraestructura del sistema. Antes de iniciar una prueba de penetración es esencial para identificar las vulnerabilidades usar un escáner de vulnerabilidades. Realizar un análisis de vulnerabilidades ayuda a un equipo de pruebas de penetración evaluar si se lleva a cabo o no la prueba de penetración e identificar las áreas que pueden ser el blanco en una prueba de penetración.

9.1.2 Pruebas de Penetración Internas

Este tipo de pruebas se llevan a cabo dentro de una red. Se asume un escenario de pruebas Caja-Gris. Involucra pruebas de las debilidades de seguridad y fortalezas de los equipos y dispositivos dentro de la red interna. Se incluye la conexión a la red interna y se realiza principalmente para comprobar las vulnerabilidades conocidas que podrían ser explotadas.

En las pruebas de penetración interna se destacan los siguientes elementos de una red:

- Vulnerabilidades de la infraestructura del protocolo y de la red.
- Sistema operativo del servidor y vulnerabilidades de las aplicaciones.
- Controles y procedimientos internos.
- Privilegios de usuario inadecuados.

9.1.3 Pruebas de Penetración Externas

Este tipo de pruebas se llevan a cabo fuera de la red. Se asume un ambiente de pruebas Caja-Negra. El principal objetivo de las pruebas de penetración externa es identificar las vulnerabilidades existentes que pueden ser explotadas desde fuera de la red. Hacer esto ayuda al administrador del sistema a identificar vulnerabilidades y evitar que sean explotadas. Utiliza casi las mismas herramientas y métodos de las pruebas de penetración interna para probar la red externa. La diferencia es que las pruebas de penetración externa se llevan a cabo desde fuera de la red interna, por lo que generalmente se centra en vulnerabilidades desconocidas.

Las pruebas de penetración externa se llevan a cabo durante un período de evaluación predeterminado y debe incluir las siguientes tareas:

- Recompilar información disponible sobre las configuraciones de acceso externo.
- Analizar e identificar los servicios y topologías de redes de clientes externos.
- Comprobar los puertos y los servicios de cliente de servicios de Internet que son propensos a ser atacados.

El principal objetivo de las pruebas de penetración externa es proporcionar la capacidad de prever los ataques externos. También se hace énfasis en los medios prácticos para reducir al mínimo los riesgos relacionados con los sistemas y redes de información. La implementación de este enfoque mejora la seguridad de los recursos y ayuda a mitigar los costos e incertidumbres que surgen de posibles ataques externos.

9.1.4 Mitigación

La mitigación es el conjunto de lineamientos o medidas que se pueden tomar para contrarrestar o minimizar los riesgos o vulnerabilidades en las implementaciones VoIP. El objetivo es añadir capas de seguridad que le hagan el trabajo más difícil a un atacante. Es importante que el conjunto de lineamientos propuestos sea tanto para un ataque que surge dentro de la red interna como fuera.

9.1.5 Verificación y Análisis de los Resultados

La verificación brinda la oportunidad de saber si el conjunto de lineamientos propuesto que representa la solución a los problemas encontrados en arquitecturas VoIP reducen los riesgos o vulnerabilidades en las implementaciones VoIP. El análisis de resultados contiene la información encontrada durante el proceso de verificación. Por medio del análisis de los resultados se permite corroborar o comprobar que la solución proporcionada permite mitigar o prevenir ataques en las arquitecturas propuestas.

10. Ataques y Mitigación

10.1 Introducción

Un intruso puede llevar a cabo ataques a una red con el fin de tomar el control, desestabilizar o dañar un sistema. Estos ataques pueden clasificarse por el lugar de su realización, es decir, pueden ser internos o externos y de forma activa o pasiva. Los ataques pasivos no provocan una modificación, alteración o daño físico a los bienes, sino que únicamente se dedican a observar, escuchar o monitorear los lugares donde se encuentran los bienes, ocasionando que no sean detectados de manera rápida. Por otra parte, los ataques activos provocan una alteración o daño físico a los dispositivos, equipos, información o lugares en donde se encuentran los bienes y por lo tanto son más fácilmente detectables.

Para poder elaborar un conjunto de lineamientos se sigue la metodología presentada en el Capítulo 9. Se plantea realizar un análisis de vulnerabilidades para determinar los puntos débiles del sistema y dependiendo del lugar donde se encuentre el atacante se realizarán pruebas de penetración internas o externas. Posteriormente, se elabora un conjunto de lineamientos generales debido a que los Escenarios 1, 2 y 3 heredan por igual los problemas de seguridad de la red interna y se especifican las soluciones adicionales. Finalmente, se verifican y analizan los resultados obtenidos.

10.2 Análisis de Vulnerabilidades

Se hizo uso de OpenVAS una herramienta de Kali Linux y se incorporó la herramienta Nessus para realizar los análisis de vulnerabilidades.

OpenVAS muestra las vulnerabilidades encontradas y provee la posibilidad de exportar los resultados a un documento detallado. En dicho documento se detalla que el puerto de SIP (5060) se encuentra abierto, se listan las opciones soportadas por el protocolo y se muestra información relevante. La Figura 10.1 muestra el documento generado por la herramienta con el detalle del puerto UDP 5060.

La herramienta Nessus informó de varios servicios de VoIP que se ejecutan en el servidor Elastix. De esta forma el atacante se entera que el host atacado provee telefonía VoIP. En la Figura 10.2 se muestran los servicios VoIP encontrados por Nessus.

Ambas herramientas sirvieron para revelar al atacante que el host atacado es un servidor que provee diferentes servicios VoIP y para evidenciar las vulnerabilidades que posee. Adicionalmente, se identificó el Elastix utilizado por el servidor y la versión en la que se encuentra. Una vez realizado el análisis de vulnerabilidades, el atacante procedió a elaborar una lista de ataques al servidor de manera que pudiese cumplir sus objetivos e intereses.

```

Log (CVSS: 0.0)
NVT: Detect SIP Compatible Hosts

Summary
A Voice Over IP service is listening on the remote port.
Description :
The remote host is running SIP (Session Initiation Protocol), a protocol
used for Internet conferencing and telephony.
Make sure the use of this program is done in accordance with your corporate
security policy.

OID of test routine: 1.3.6.1.4.1.25623.1.0.11963

Vulnerability Detection Result
Summary:
A Voice Over IP service is listening on the remote port.
Description :
The remote host is running SIP (Session Initiation Protocol), a protocol
used for Internet conferencing and telephony.
Make sure the use of this program is done in accordance with your corporate
security policy.
Solution:
If this service is not needed, disable it or filter incoming traffic
to this port.
Plugin output :
FPBX-2.8.1(1.8.20.0)
Supported Options:
INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH

Solution
If this service is not needed, disable it or filter incoming traffic
to this port.

Log Method
Details:Detect SIP Compatible Hosts
OID:1.3.6.1.4.1.25623.1.0.11963
Version used: $Revision: 920 $

```

Figura 10.1: Análisis de Vulnerabilidades de OpenVAS

Details		
Severity	Plugin Id	Name
Info	20834	Inter-Asterisk eXchange Protocol Detection
Info	21642	Session Initiation Protocol Detection
Info	63202	Asterisk Detection
Info	76347	HylaFAX Installed

Figura 10.2: Análisis de Vulnerabilidades de Nessus

10.3 Pruebas de Penetración Internas

Los ataques internos son lo más comunes y peligrosos. Estos son iniciados por alguien con acceso autorizado a una red, es decir, son aquellos que se originan dentro de la propia organización. La mayoría de los incidentes reportados provienen de acceso interno o del mal uso de una cuenta. En muchas ocasiones estos ataques provienen de empleados y socios descontentos.

Dar más importancia a las amenazas externas que a los peligros internos es un error, y puede acrecentar de sobremanera la vulnerabilidad de las defensas de una organización.

Se hace uso de las pruebas de penetración internas debido a que estas serán ejecutadas por alguien que posee acceso autorizado dentro de la red LAN con soporte VoIP. Los 3 escenarios de prueba propuestos son susceptibles a ataques internos y vulnerabilidades. En la Figura 10.3 se muestra la red LAN con soporte VoIP presente en los 3 escenarios y la posible ubicación del atacante dentro de la red LAN.

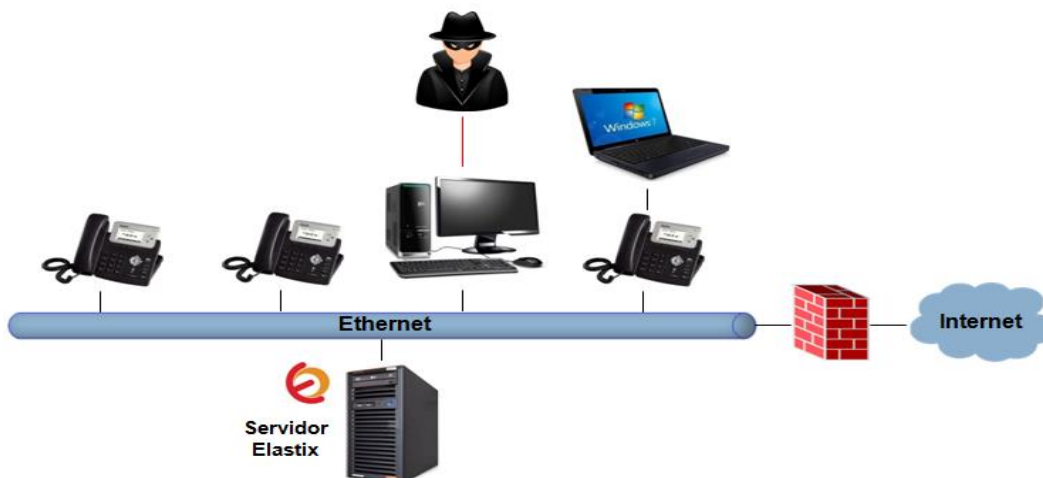


Figura 10.3: Ataque a una LAN con Soporte VoIP

Para este tipo de pruebas se utilizaron diferentes herramientas de Kali Linux. A continuación se detallan los ataques realizados y se describen las vulnerabilidades en la red LAN.

10.3.1 Ingeniería Social

Al encontrarse el atacante dentro de la propia red interna es posible que conozca a los otros usuarios e intente obtener información vital para completar su ataque. En este caso el atacante podría conseguir las direcciones IP de otras extensiones pertenecientes a usuarios y del servidor Elastix con el fin de hacer más fácil su ataque dentro de la red.

10.3.2 Port Scanning

La herramienta Nmap (Network Mapper) es un escáner de seguridad que sirve para escanear hosts, puertos y servicios de una red de computadores. Al ejecutar esta herramienta con la dirección y máscara de red fue posible obtener una lista con las direcciones IP con la dirección MAC asociada. En la Figura 10.4 se visualiza la ejecución de la herramienta con la dirección y máscara de red de la red interna donde se encuentra el servidor Elastix y sus respectivas extensiones.

```
root@kaliLinux: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~# nmap -v -sn 192.168.1.0/24

Starting Nmap 6.47 ( http://nmap.org ) at 2015-07-04 20:28 VET
Initiating ARP Ping Scan at 20:28
Scanning 255 hosts [1 port/host]
Completed ARP Ping Scan at 20:29, 3.76s elapsed (255 total hosts)
Initiating Parallel DNS resolution of 255 hosts. at 20:29
Completed Parallel DNS resolution of 255 hosts. at 20:29, 0.10s elapsed
Nmap scan report for 192.168.1.0 [host down]
Nmap scan report for 192.168.1.1 [host down]
Nmap scan report for 192.168.1.2
Host is up (0.0011s latency).
MAC Address: 00:1A:92:1E:8A:AF (Asustek Computer)
Nmap scan report for 192.168.1.3
Host is up (0.0019s latency).
MAC Address: 08:00:27:DE:DF:4F (Cadmus Computer Systems)
Nmap scan report for 192.168.1.4
Host is up (0.034s latency).
MAC Address: 98:0C:82:D3:56:04 (Samsung Electro Mechanics)
Nmap scan report for 192.168.1.5
Host is up (0.031s latency).
MAC Address: 84:38:38:EE:6F:DE (Samsung Electro Mechanics co.)
Nmap scan report for 192.168.1.6
Host is up (0.032s latency).
MAC Address: 50:F5:20:92:CF:64 (Samsung Electronics Co.)
Nmap scan report for 192.168.1.7
Host is up (0.00033s latency).
MAC Address: 98:4B:E1:BC:91:72 (Hewlett-Packard Company)
Nmap scan report for 192.168.1.9 [host down]
Nmap scan report for 192.168.1.10 [host down]
Nmap scan report for 192.168.1.11
Host is up (0.025s latency).
MAC Address: 94:EB:CD:ED:4E:38 (Research In Motion Limited)
```

Figura 10.4: Descubrimiento de los Hosts de una Red con Nmap

Cuando se ejecuta la herramienta Nmap con la dirección IP del servidor de Elastix se evidenció que es posible obtener información del sistema operativo, servicios, puertos TCP/UDP y su estado. En la Figura 10.5 se muestra la salida de la herramienta Nmap donde se evidencia la información de todos los puertos TCP/UDP abiertos en el servidor Elastix y en la Figura 10.6 se observa la información del puerto UDP de SIP (5060).

```

root@kaliLinux: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~# nmap -sS -P0 -sV -O 192.168.1.3

Starting Nmap 6.47 ( http://nmap.org ) at 2015-07-04 20:41 VET
Nmap scan report for 192.168.1.3
Host is up (0.0012s latency).
Not shown: 987 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 4.3 (protocol 2.0)
25/tcp    open  smtp         Postfix smtpd
80/tcp    open  http         Apache httpd 2.2.3 ((CentOS))
110/tcp   open  pop3         Cyrus pop3d 2.3.7-Invoca-RPM-2.3.7-12.el5_7.2
111/tcp   open  rpcbind      2 (RPC #100000)
143/tcp   open  imap         Cyrus imapd 2.3.7-Invoca-RPM-2.3.7-12.el5_7.2
443/tcp   open  ssl/http     Apache httpd 2.2.3 ((CentOS))
801/tcp   open  status       1 (RPC #100024)
993/tcp   open  ssl/imap     Cyrus imapd
995/tcp   open  pop3         Cyrus pop3d
3306/tcp  open  mysql        MySQL 5.0.95
4445/tcp  open  upnotifyp?
5060/tcp  open  sip          Zoiper VoIP software rev.14736
MAC Address: 08:00:27:DE:DF:4F (Cadmus Computer Systems)
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6
OS details: Linux 2.6.18 - 2.6.32
Network Distance: 1 hop
Service Info: Hosts: elastix.org, example.com

OS and Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 164.20 seconds

```

Figura 10.5: Descubrimiento de los Puertos Abiertos de un Host con Nmap

```

root@kaliLinux: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~# nmap -sT 192.168.1.3 -p 5060

Starting Nmap 6.47 ( http://nmap.org ) at 2015-07-04 20:40 VET
Nmap scan report for 192.168.1.3
Host is up (0.00094s latency).
PORT      STATE SERVICE
5060/tcp  open  sip
MAC Address: 08:00:27:DE:DF:4F (Cadmus Computer Systems)

Nmap done: 1 IP address (1 host up) scanned in 0.23 seconds

```

Figura 10.6: Descubrimiento de Información Adicional sobre un Puerto Particular con Nmap

10.3.3 Man-in-the-Middle

Para elaborar este tipo de ataque en la red interna se debe interceptar y reenviar el tráfico de red. Para lograr que el atacante se encuentre en medio de la comunicación se intercepta el tráfico VoIP entre el servidor Elastix y una extensión dentro de la red interna con el ataque “ARP Spoofing” y el mecanismo “IP Forwarding” para reenviar los paquetes recibidos por una interfaz física. El “IP Forwarding” se utiliza para mantener comunicación entre el servidor Elastix y la extensión previamente atacados con “ARP Spoofing”, de esta forma se les hace creer que se están comunicando directamente. En la Figura 10.7 se muestra la tabla ARP del servidor Elastix antes del ataque “ARP Spoofing”.

```
root@elastix240:~# arp -a
? (192.168.1.6) at 50:F5:20:92:CF:64 [ether] on eth0
? (192.168.1.4) at 98:0C:82:D3:56:04 [ether] on eth0
? (192.168.1.5) at 84:38:38:EE:6F:DE [ether] on eth0
? (192.168.1.252) at 58:6D:8F:7E:F0:72 [ether] on eth0
```

Figura 10.7: Tabla ARP del Servidor Elastix antes del Ataque

Ettercap es una herramienta que permite realizar ataques “Man-in-the-Middle” en redes LAN. Inicialmente se usó Ettercap para escanear y listar los hosts de una interfaz de red. Una vez obtenida la lista de hosts se seleccionaron como objetivos la dirección IP del servidor Elastix y una extensión registrada al servidor. Ambas direcciones se les realizó un ataque “ARP Spoofing”. La Figura 10.8 detalla la interfaz de Ettercap y la selección de las direcciones IP.

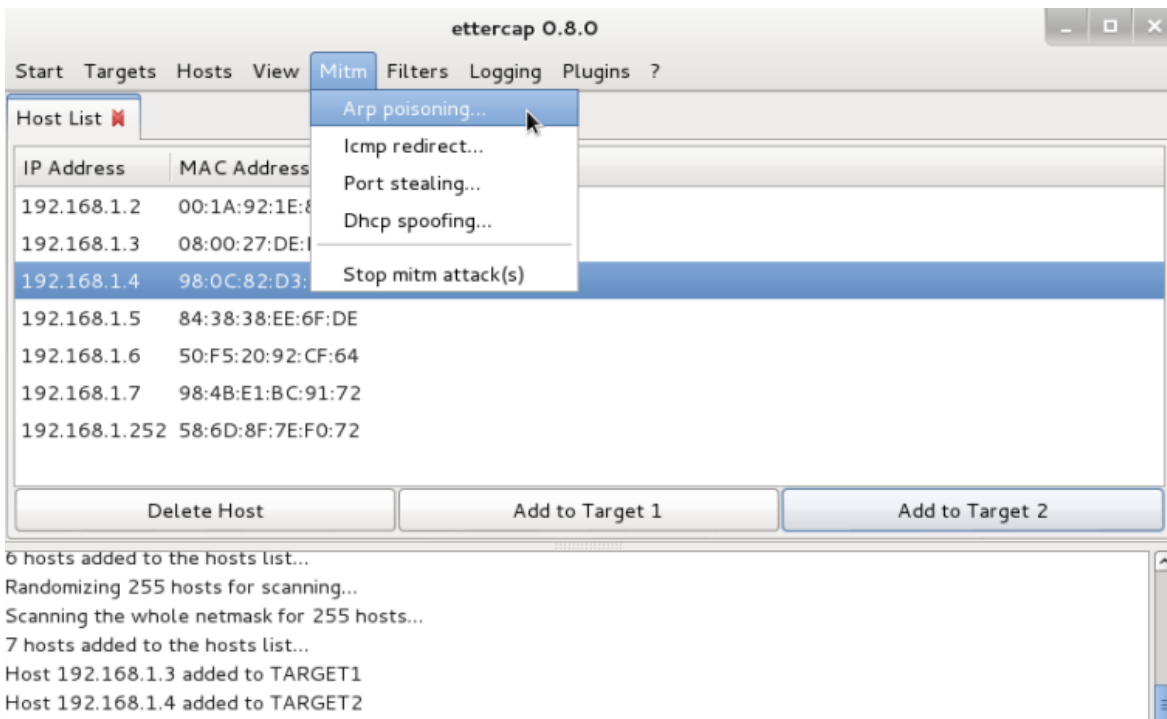


Figura 10.8: Ataque ARP Spoofing con Ettercap

Realizado el ataque “ARP Spoofing” y manteniéndolo en ejecución se evidencia que la tabla ARP del servidor Elastix cambia la dirección MAC de la extensión que se seleccionó previamente para el ataque. En la Figura 10.9 se visualiza la tabla ARP antes y después del ataque.



```
root@elastix240:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@elastix240 ~]# arp -a  
? (192.168.1.6) at 50:F5:20:92:CF:64 [ether] on eth0  
? (192.168.1.4) at 98:0C:82:D3:56:04 [ether] on eth0  
? (192.168.1.5) at 84:38:38:EE:6F:DE [ether] on eth0  
? (192.168.1.252) at 58:6D:8F:7E:F0:72 [ether] on eth0  
[root@elastix240 ~]# arp -a  
? (192.168.1.6) at 50:F5:20:92:CF:64 [ether] on eth0  
? (192.168.1.4) at 08:00:27:77:CB:F1 [ether] on eth0  
? (192.168.1.5) at 84:38:38:EE:6F:DE [ether] on eth0  
? (192.168.1.252) at 58:6D:8F:7E:F0:72 [ether] on eth0
```

Figura 10.9: Tabla ARP del Servidor Elastix Antes y Después del Ataque

La extensión atacada con “ARP Spoofing” deja de establecer comunicación con el genuino servidor Elastix. Su tabla ARP fue envenenada y su comunicación es ahora con el atacante. Para mantener a la extensión comunicada con el servidor se hizo uso de “IP Forwarding” para reenviar los paquetes que recibe el atacante al servidor y viceversa. En la Figura 10.10 se presenta el comando ejecutado en Kali Linux para activar el “IP Forwarding”.



```
root@kaliLinux: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
root@kaliLinux:~# echo 1 > /proc/sys/net/ipv4/ip_forward  
root@kaliLinux:~# cat /proc/sys/net/ipv4/ip_forward  
1
```

Figura 10.10: Activación de IP Forwarding en Kali Linux

10.3.4 Eavesdropping

Para llevar a cabo este tipo de ataque se utilizó la herramienta Wireshark para capturar todo el tráfico VoIP (peticiones, respuestas, etc) entre el servidor Elastix y una extensión e incluso escuchar conversaciones activas. Antes de realizar el eavesdropping con Wireshark es necesario realizar un ataque “Man-in-the-Middle” que permita la interceptación y reenvío del tráfico VoIP. Una vez realizado el ataque “Man-in-the-Middle” se podrá capturar todo el tráfico de red entre el servidor Elastix y una de sus extensiones. En la Figura 10.11 se muestra el tráfico VoIP capturado con la herramienta, mientras que en la Figura 10.12 se puede observar la captura de una llamada realizada en VoIP entre dos extensiones. Por último, en la Figura 10.13 se evidencia cómo es escuchada la llamada por el atacante.

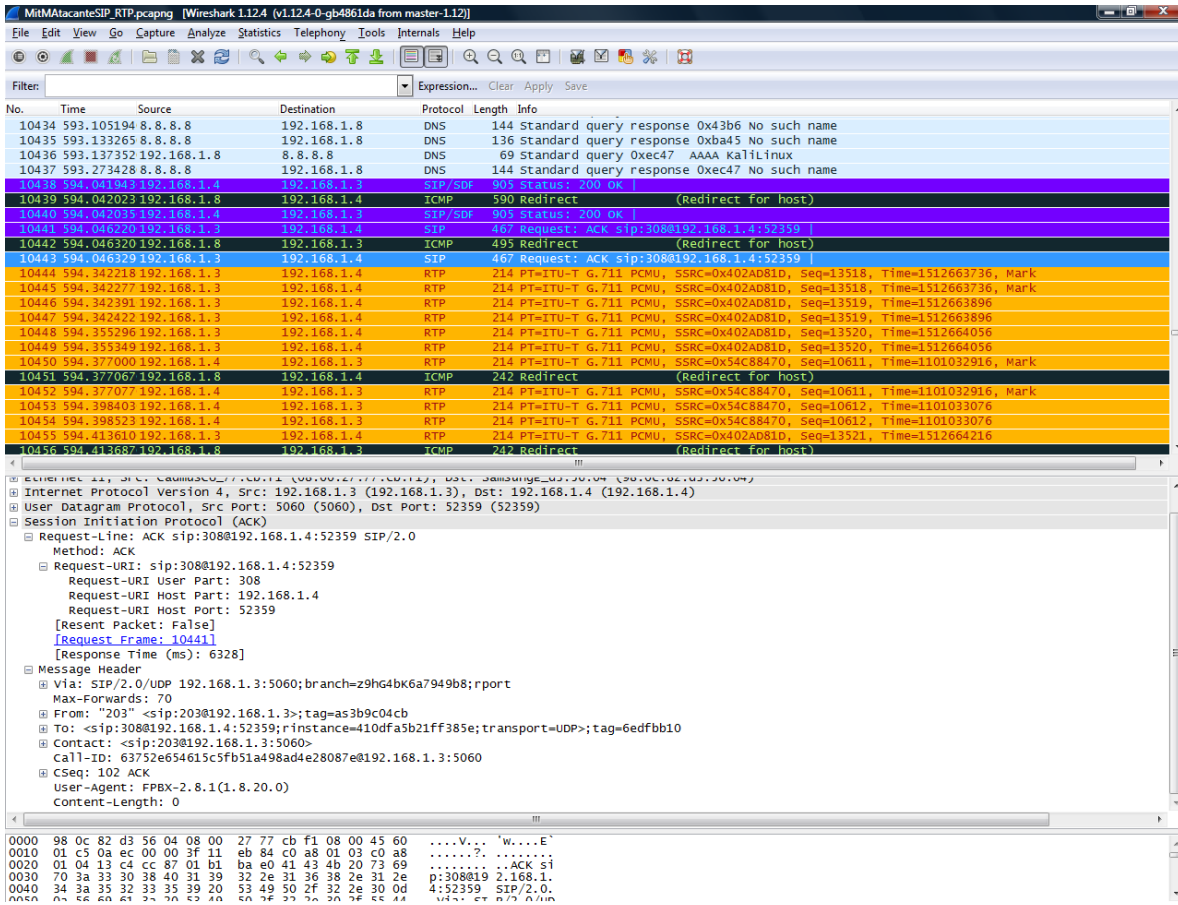


Figura 10.11: Captura de Tráfico VoIP en Wireshark

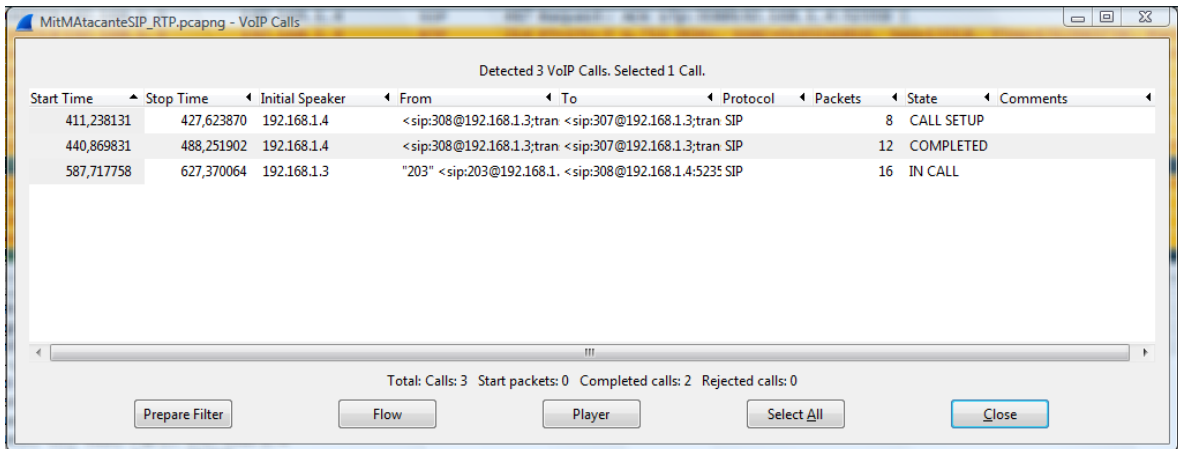


Figura 10.12: Análisis del Tráfico SIP de la Llamada Interceptada con Wireshark



Figura 10.13: Escucha de la Llamada Interceptada con el Reproductor RTP de Wireshark

10.3.5 Ataque de DoS

Para el ataque de DoS se hizo uso de la herramienta inviteflood (ver Figura 10.14) presente en Kali Linux. Para esta prueba se enviaron diferentes números de peticiones INVITE al servidor Elastix. En la Tabla 10.1 se muestra la cantidad de peticiones enviadas y el estado del funcionamiento del Elastix. Se evidencia que luego de enviar 10000 peticiones el servidor Elastix comienza a comportarse de forma anormal.

Nro. de Peticiones	Funcionamiento del Elastix
100	Funcionamiento normal del Elastix. Las llamadas se realizan de forma correcta. Si se establece una llamada, esta sigue establecida y con buen audio durante el ataque.
1000	Funcionamiento normal del Elastix. Las llamadas se realizan de forma correcta pero tardan algunos segundos adicionales en establecerse. Si se establece una llamada, esta sigue establecida y con buen audio durante el ataque.
10000	Funcionamiento anormal del Elastix. Durante el envío de peticiones INVITE generadas por inviteflood, no se logra establecer la llamada sino hasta que el envío de peticiones por inviteflood finalice. Si una llamada se encuentra establecida, esta sigue establecida y con buen audio durante el ataque.
100000	Funcionamiento anormal del Elastix. Debido a la gran cantidad de peticiones INVITE generadas por inviteflood, ninguna extensión logra establecer la llamada mientras se haga el envío de peticiones. Si una llamada se encuentra establecida,

	esta sigue establecida pero presenta en ocasiones eco y retraso.
1000000	Funcionamiento anormal del Elastix. Ninguna extensión logra establecer una llamada incluso después de finalizado el envío de peticiones INVITE por inviteflood. Por algunos momentos las extensiones dejan de estar registradas en el servidor Elastix. Durante el envío de peticiones, las extensiones pasaron por los estados "OK", "LAGGED", "UNREACHABLE" y "UNKNOWN". Si una llamada se encuentra establecida, esta sigue establecida pero se obtuvo retraso y ruido.

Tabla 10.1: Funcionamiento de Elastix ante Diferentes Números de Peticiones INVITE

```

root@kaliLinux: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@kaliLinux:~# inviteflood eth0 308 192.168.1.3 192.168.1.3 100

inviteflood - Version 2.0
              June 09, 2006

source IPv4 addr:port = 192.168.1.8:9
dest   IPv4 addr:port = 192.168.1.3:5060
targeted UA          = 308@192.168.1.3

Flooding destination with 100 packets
sent: 100

```

Figura 10.14: Ejecución de la Herramienta inviteflood

También se realizó un ataque con la herramienta rtpflood. Esta herramienta permite el envío de paquetes RTP a una dirección IP con el puerto UDP abierto durante una llamada VoIP. Para realizar el ataque es necesario capturar el tráfico VoIP entre una extensión y el servidor Elastix. Los ataques "Man-in-the-Middle" y eavesdropping se usaron para capturar el tráfico VoIP y para conocer los puertos UDP abiertos cuando se estableció una llamada entre dos extensiones. Los paquetes RTP generados por rtpflood se enviaron a la extensión atacada y al servidor Elastix para que el mismo reenviara los paquetes a la otra extensión (ver Figura 10.15). De esta forma ambas extensiones presenciaron anomalías en la llamada establecida. En la Tabla 10.2 se describe la calidad de la llamada de acuerdo al número total de paquetes RTP enviados a la extensión y al servidor Elastix en un mismo tiempo.

Nro. de Paquetes	Calidad de la Llamada
100	En la llamada establecida se escucha ruido durante el corto tiempo que tarda el envío de paquetes RTP.
1000	En la llamada establecida se escucha ruido en un período mayor de tiempo que tarda el envío de paquetes RTP.
10000	En la llamada establecida se escucha mucho más ruido pudiéndose entender pocas palabras durante el envío de paquetes RTP.
100000	En la llamada establecida se escucha mucho más ruido, pocas palabras entendibles y en ocasiones ausencia total de la voz humana.
1000000	En la llamada establecida se escucha mucho más ruido y se experimenta ausencia total de la voz humana.

Tabla 10.2: Calidad de la Llamada ante Número de Paquetes RTP

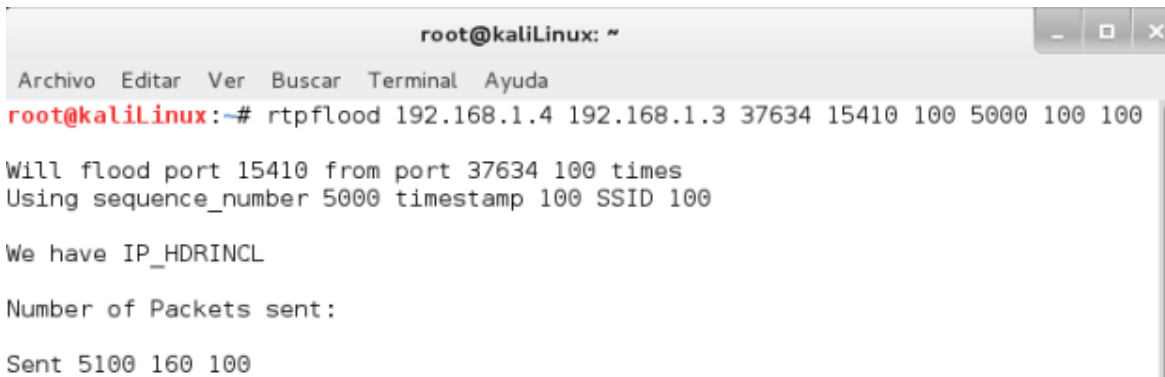


Figura 10.15: Ejecución de la Herramienta rtpflood

10.3.6 Ataque de Fuerza Bruta

Las herramientas SIPDump y SIPcrack permitieron ejecutar un ataque de fuerza bruta para encontrar la contraseña de una extensión registrada en el servidor Elastix a partir de una captura del tráfico VoIP y un diccionario.

Antes de realizar el ataque de fuerza bruta es necesario realizar un ataque “Man-in-the-Middle” que permita la captura del tráfico VoIP. En la Figura 10.16 se presenta la captura de peticiones y respuestas SIP.

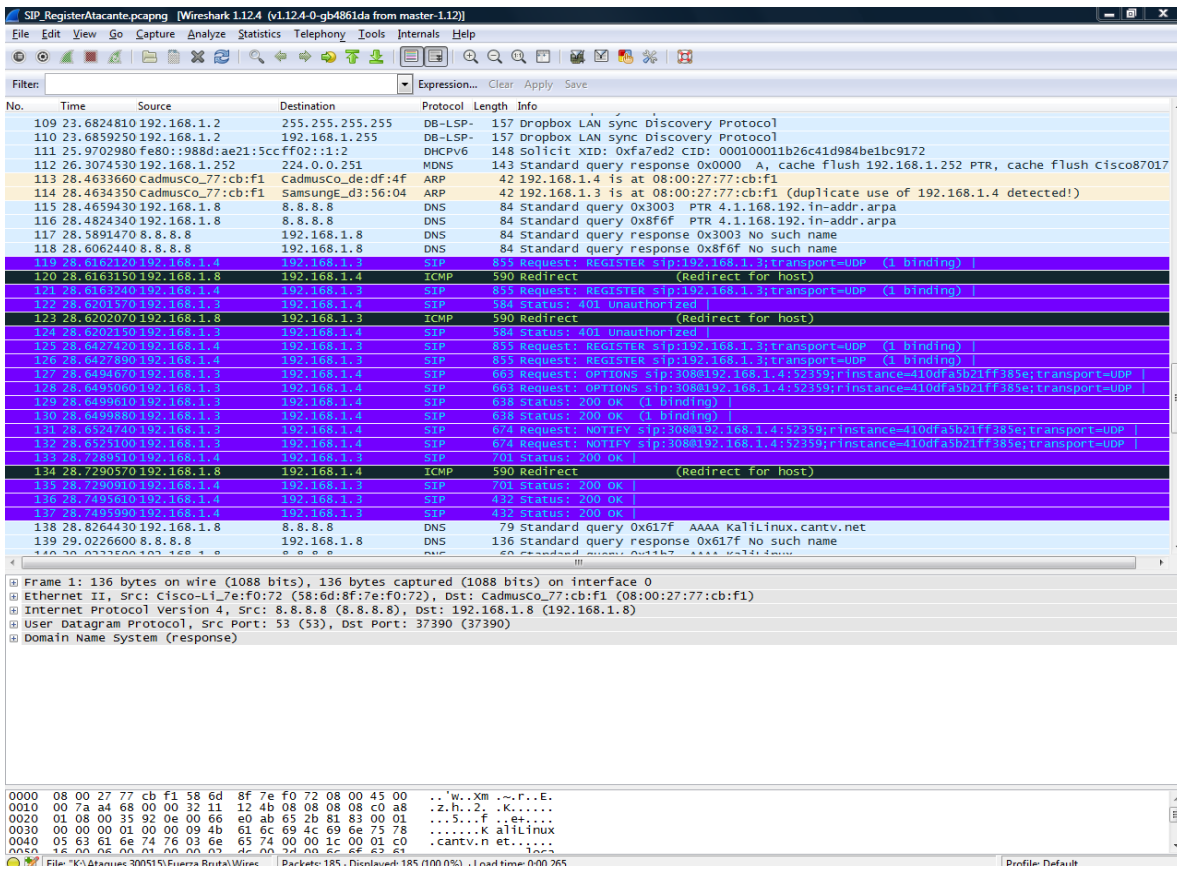


Figura 10.16: Captura de Tráfico SIP en Wireshark

La captura del tráfico VoIP (peticiones y respuestas SIP) ayuda al atacante a crear un archivo de Wireshark que es utilizado por la herramienta SIPDump. SIPDump captura la autenticación HTTP digest en caso de existir un intercambio de contraseña entre el servidor Elastix y la extensión para crear un archivo con formato dump. En la Figura 10.17, se puede apreciar la ejecución de la herramienta SIPDump y en la Figura 10.18 se visualiza el archivo generado luego de la ejecución de la herramienta.

```

root@kaliLinux: ~/Desktop
Archivo Editar Ver Buscar Terminal Ayuda
root@kaliLinux:~/Desktop# sipdump passhash.txt -p SIP_RegisterAtacante.pcapng

SIPdump 0.2 ( MaJoMu | www.codito.de )
-----

* Using pcap file 'SIP_RegisterAtacante.pcapng' for sniffing
* Starting to sniff with packet filter 'tcp or udp'

* Dumped login from 192.168.1.3 -> 192.168.1.4 (User: '308')
* Dumped login from 192.168.1.3 -> 192.168.1.4 (User: '308')
* Dumped login from 192.168.1.3 -> 192.168.1.4 (User: '308')
* Dumped login from 192.168.1.3 -> 192.168.1.4 (User: '308')

* Exiting, sniffed 4 logins

```

Figura 10.17: Ejecución de la Herramienta SIPDump

```

1 192.168.1.4"192.168.1.3"308"asterisk"REGISTER"sip:192.168.1.3;transport=UDP"21dcbf0b""PLAIN"abc308
2 192.168.1.4"192.168.1.3"308"asterisk"REGISTER"sip:192.168.1.3;transport=UDP"21dcbf0b""PLAIN"abc308
3 192.168.1.4"192.168.1.3"308"asterisk"REGISTER"sip:192.168.1.3;transport=UDP"0382296a""MD5"258f93635bc8a96591f43b249d47c14a
4 192.168.1.4"192.168.1.3"308"asterisk"REGISTER"sip:192.168.1.3;transport=UDP"0382296a""MD5"258f93635bc8a96591f43b249d47c14a
5

```

Figura 10.18: Archivo passhash.txt Generado por SIPDump

La herramienta SIPcrack utiliza el archivo generado por la herramienta SIPDump y un diccionario para intentar encontrar la contraseña de un usuario legítimo. Si una contraseña en texto plano presente en el diccionario coincide con el hash MD5 capturado en la autenticación HTTP digest, la herramienta encontrará la contraseña del usuario legítimo. La Figura 10.19 muestra un fragmento del diccionario utilizado y en la Figura 10.20 se presenta la ejecución de la herramienta SIPcrack.

```

1 abc200
2 acb200
3 bac200
4 bca200
5 cab200
6 cba200
7 abc201
8 acb201
9 bac201
10 bca201
11 cab201
12 cba201
13 abc202
14 acb202
15 bac202
16 bca202
17 cab202
18 cba202
19 abc203
20 acb203
21 bac203
22 bca203
23 cab203
24 cba203
25 abc204

```

Figura 10.19: Visualización de Algunas Palabras del Diccionario Utilizado en SIPcrack

```

root@kaliLinux: ~/Desktop
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~/Desktop# sipcrack passhash.txt -w diccionario.txt

SIPcrack 0.2 ( MaJoMu | www.codito.de )
-----

* Found Accounts:

Num      Server      Client      User      Hash|Password
1        192.168.1.4  192.168.1.3  308      840d8323e47edfa98abdcad0b3c11f18
2        192.168.1.4  192.168.1.3  308      840d8323e47edfa98abdcad0b3c11f18
3        192.168.1.4  192.168.1.3  308      258f93635bc8a96591f43b249d47c14a
4        192.168.1.4  192.168.1.3  308      258f93635bc8a96591f43b249d47c14a

* Select which entry to crack (1 - 4): 1

* Generating static MD5 hash... 2286b754a26fc03178dc238760f9bad6
* Loaded wordlist: 'diccionario.txt'
* Starting bruteforce against user '308' (MD5: '840d8323e47edfa98abdcad0b3c11f18')
* Tried 109 passwords in 0 seconds

* Found password: 'abc308'
* Updating dump file 'passhash.txt'... done

```

Figura 10.20: Ejecución de la Herramienta SIPcrack

10.4 Pruebas de Penetración Externas a Escenario 1

Los ataques externos son iniciados por individuos o grupos desde afuera de la organización. Estos no tienen acceso autorizado al sistema o red de computadores de la organización. Se encargan de reunir información para así poder abrir camino dentro de la red.

Un PSTN permite la transmisión de la voz humana a través de líneas telefónicas. Un servidor Elastix puede conectarse a un PSTN por medio de una tarjeta analógica o digital. Los equipos con interconexión a un PSTN que no se encuentren protegidos representan un grave peligro para una empresa u organización debido a que si aceptan llamadas desde cualquier origen estos podrían enviar tráfico a destinos bastante costosos.

El Escenario 1 es susceptible a múltiples ataques externos. Se utilizan pruebas de penetración externas y se evalúan las vulnerabilidades presentes en la arquitectura propuesta. Un servidor Elastix mal configurado puede generar grandes gastos a una organización, como por ejemplo puede permitir INVITEs externos y ser enrutados a través del PSTN. En la Figura 10.21 se muestra la red LAN con soporte VoIP con una conexión a un PSTN y la ubicación del atacante fuera de la red LAN.

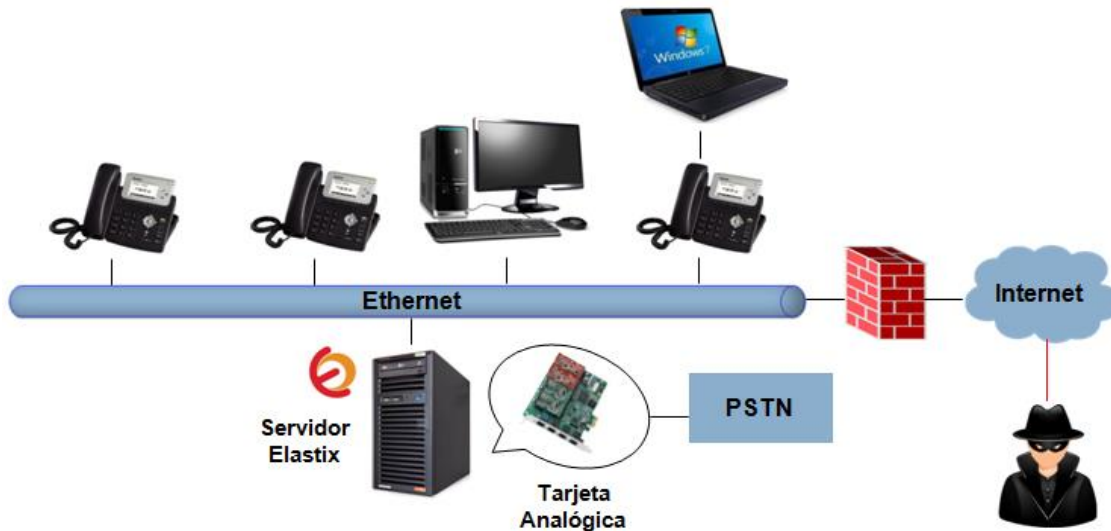


Figura 10.21: Ataque a una Red LAN con VoIP y una Conexión a un PSTN

A continuación se detallan los ataques que se realizaron y se describen las vulnerabilidades en el escenario.

10.4.1 Ingeniería Social

Al encontrarse el atacante fuera de la red LAN es posible que el atacante utilice los diferentes tipos de la ingeniería social, como por ejemplo el phishing para comprometer la seguridad del firewall. En este caso el atacante podría conseguir el usuario y contraseña del firewall para posteriormente obtener acceso al servidor Elastix y realizar llamadas a través del PSTN. Muchos firewalls proporcionan servicios de seguridad, como por ejemplo antivirus, IPS, anti-spam y filtrado de contenido bajo licencias que duran 1, 2 o 3 años. Al vencerse las licencias el servidor Elastix queda menos seguro y haciéndolo susceptible a múltiples ataques.

10.4.2 Ataque de DoS

Si un atacante dispone del usuario y de la contraseña del firewall puede entonces modificar las reglas y políticas de acceso al servidor Elastix para posteriormente realizar un ataque de DoS. Adicionalmente, una mala configuración de las reglas y políticas del firewall podría permitir al atacante acceder al servidor Elastix. Con el acceso al servidor se podrían realizar otros ataques con el objetivo de efectuar llamadas a través del PSTN.

10.4.3 Port Scanning

Se utiliza la herramienta Nmap para escanear hosts, puertos y servicios del firewall debido a que es el dispositivo final que interconecta la red LAN con el Internet y que además se encarga de la seguridad dentro de la propia red. Al ejecutar esta herramienta con su dirección IP pública, se evidenció que es posible obtener información de los servicios, puertos TCP/UDP y su estado. Adicionalmente, el atacante determina que la dirección IP pública pertenece a un firewall de Fortinet (ver Figura 10.22).

```
root@kaliLinux: ~
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~# nmap -sS -P0 -sV -O 190.153.104.146

Starting Nmap 6.47 ( http://nmap.org ) at 2015-08-11 22:14 VET
Nmap scan report for nsl.cgtscorp.com (190.153.104.146)
Host is up (0.12s latency).
Not shown: 991 filtered ports
PORT      STATE SERVICE      VERSION
80/tcp    open  http        Apache httpd
110/tcp   open  pop3        Dovecot pop3d
143/tcp   open  imap        Dovecot imapd
443/tcp   open  ssl/http    Fortinet FortiGate 50B or FortiWifi 80C firewall http co
nfig
587/tcp   open  smtp        Postfix smtpd
993/tcp   open  ssl/imap    Dovecot imapd
995/tcp   open  ssl/pop3    Dovecot pop3d
2022/tcp  open  ssh         OpenSSH 4.3 (protocol 2.0)
9080/tcp  open  http        Apache httpd 2.2.15 ((CentOS))
Warning: OSScan results may be unreliable because we could not find at least 1 o
pen and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Linux 2.6.X|3.X (90%)
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3
Aggressive OS guesses: 2.6.32 (90%), Linux 2.6.39 (90%), Linux 3.4 (90%), Linux
2.6.32 (90%), Linux 3.1 - 3.2 (90%), Linux 2.6.32 - 2.6.39 (87%)
No exact OS matches for host (test conditions non-ideal).
Service Info: Host: mail.cgtscorp.com; Device: firewall; CPE: cpe:/h:fortinet:f
ortiwifi:80c

OS and Service detection performed. Please report any incorrect results at http:
//nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 37.05 seconds
```

Figura 10.22: Descubrimiento de los Puertos Abiertos de un Host y su Dirección IP Pública con Nmap

Los ataques de Port Scanning se encuentran presentes en los 3 escenarios propuestos debido a que poseen el firewall como dispositivo final que interconecta la red LAN con el Internet.

10.5 Pruebas de Penetración Externas a Escenario 2

Los ITSPs ofrecen servicios de telecomunicaciones digitales basados en VoIP que son provistos vía Internet. Proveen los mismos servicios que una operadora de telecomunicaciones tradicional, como por ejemplo, realizar llamadas nacionales e internacionales haciendo uso del Internet. Entre muchas cosas, un ITSP puede proveer servicios a usuarios finales directamente.

El Escenario 2 es susceptible a muchos ataques externos provenientes de Internet. Se utilizan pruebas de penetración externas y se revisan las vulnerabilidades presentes en el escenario. En la Figura 10.23, se puede apreciar la red LAN con soporte VoIP con una conexión a un ITSP y la ubicación del atacante fuera de la red LAN.



Figura 10.23: Ataque a una Red LAN con VoIP y una Conexión a un ITSP

A continuación se describen los ataques que se realizaron y se detallan las vulnerabilidades en el Escenario 2.

10.5.1 Ingeniería Social

El servidor Elastix siempre será susceptible a la ingeniería social para la obtención de información valiosa. En este escenario, el servidor VoIP establece una troncal SIP con el ITSP (ver Figura 10.24). La troncal utiliza la dirección IP del ITSP, usuario y contraseña. Un mal uso de las credenciales o un servidor sin protección podría permitir a un atacante utilizarlos para establecer una troncal con el ITSP para realizar llamadas que generan costos a la empresa u organización.

Trunk Name:

PEER Details:

```
host=***provider ip address***
username=***userid***
secret=***password***
type=peer
```

Incoming Settings

USER Context:

USER Details:

```
secret=***password***
type=user
context=from-trunk
```

Figura 10.24: Formulario para la Creación de Troncal SIP con el ITSP

10.5.2 Ataque de DoS

El servidor Elastix en este escenario posee un firewall que por licencias vencidas o una mala configuración de reglas y políticas podría permitir o redirigir el tráfico entrante de una dirección IP atacante hacia el servidor. Un gran número de paquetes maliciosos ocasionaría un ataque de DoS.

10.6 Pruebas de Penetración Externas a Escenario 3

La conexión remota permite el acceso a la red LAN con soporte VoIP para el establecimiento de llamadas VoIP entre una extensión que se encuentra en la red interna y otra en la red externa. Habilitar el acceso a la red interna puede generar grandes problemas de seguridad y costos para una empresa u organización.

El Escenario 3 es susceptible a varios ataques externos. Se realizan pruebas de penetración externas y se revisan las vulnerabilidades en el escenario propuesto. En la Figura 10.25, se puede observar la red LAN con soporte VoIP con una conexión remota hacia la misma y la ubicación del atacante fuera de la red LAN.



Figura 10.25: Ataque a una Red LAN con VoIP y una Conexión Remota

A continuación se explican los ataques que se realizaron y se detallan las vulnerabilidades en el Escenario 3.

10.6.1 Ingeniería Social

No se necesita ser un experto de la ingeniería social para tratar de obtener datos privados de los usuarios legítimos. El robo de información para fines indebidos es cada vez más común. Conocer al usuario o simplemente enviar correos electrónicos fraudulentos (phishing) para la obtención de información es el inicio de esta práctica. Con la dirección IP pública, usuario y contraseña de una de las extensiones del servidor de VoIP sin importar el método de ingeniería social utilizado, es posible efectuar llamadas VoIP en ese servidor y que con un gran número de llamadas representaría grandes costos a una empresa u organización.

10.6.2 Ataque de DoS

Las extensiones se conectan por medio de una dirección IP pública al servidor Elastix sin utilizar una VPN. El firewall tiene establecidas reglas y políticas de entrada que redirigen el tráfico SIP y RTP al servidor Elastix cuando es utilizada la dirección IP pública en el puerto 5060 UDP. Si el firewall no tiene activado el módulo de IPS, dejará pasar todas las peticiones al servidor Elastix, que en gran número ocasionan un ataque de DoS.

10.6.3 Ataque de Fuerza Bruta

La finalidad de este ataque es ingresar al servidor Elastix, a través de Internet y haciendo uso de una conexión remota (SSH). Para poder llevar a cabo este ataque, se necesita el nombre de un usuario legítimo y su correspondiente contraseña. En los sistemas operativos del tipo Unix, "root" es el nombre convencional de la cuenta de usuario que posee todos los derechos que también es conocida como la cuenta de administrador. Muchos servidores utilizan el puerto 22 TCP para SSH con el usuario "root". En la Figura 10.26 se observa un fragmento del diccionario creado con las posibles combinaciones de contraseñas para este usuario con la herramienta John The Ripper en Kali Linux. Se utilizó la herramienta Medusa debido a que es una herramienta disponible en Kali Linux que permite realizar ataques de fuerza bruta contra un variado conjunto de protocolos. Se realizó un ataque con el nombre de usuario "root", puerto 22 TCP y el diccionario generado. La Figura 10.27 muestra la ejecución de la herramienta Medusa con un resultado satisfactorio, ya que encontró que una de las contraseñas presentes en el diccionario corresponde a la contraseña del usuario "root".

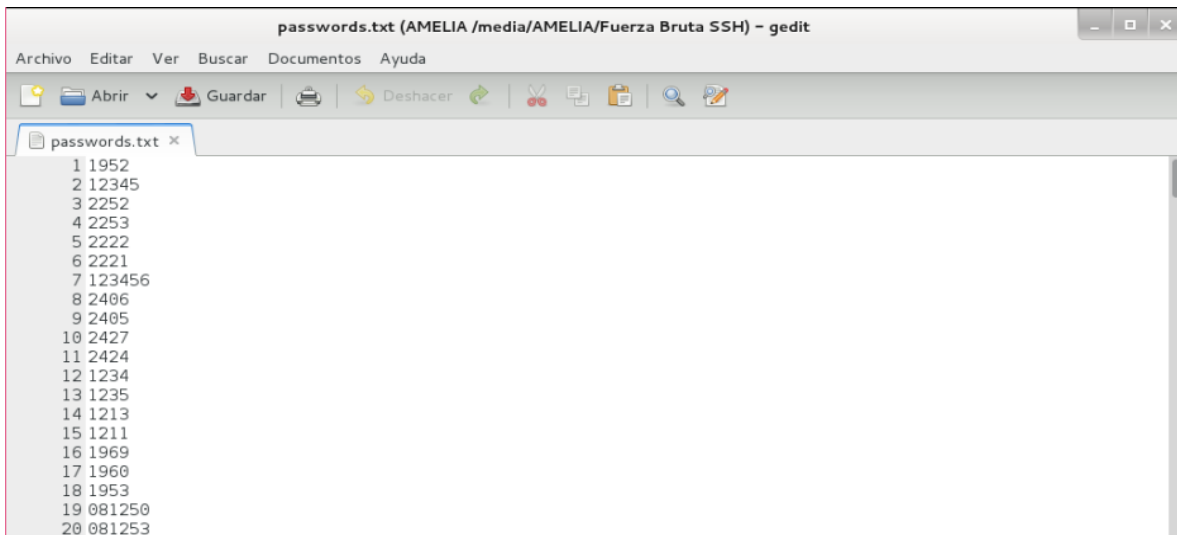


Figura 10.26: Diccionario Utilizado en Medusa


```

root@kaliLinux: ~/Desktop/Ataques
Archivo  Editar  Ver  Buscar  Terminal  Ayuda
root@kaliLinux:~/Desktop/Ataques# medusa -h 192.168.1.3 -u root -P passwords.txt
-M ssh
Medusa v2.0 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.n
et>

ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 1952 (1 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 12345 (2 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 2252 (3 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 2253 (4 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 2222 (5 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 2221 (6 of 1111110 complete)
ACCOUNT CHECK: [ssh] Host: 192.168.1.3 (1 of 1, 0 complete) User: root (1 of 1,
0 complete) Password: 123456 (7 of 1111110 complete)
ACCOUNT FOUND: [ssh] Host: 192.168.1.3 User: root Password: 123456 [SUCCESS]

```

Figura 10.27: Ejecución de la Herramienta Medusa

10.7 Mitigación

10.7.1 Control de Servicios

Al momento de administrar un servidor, se debe tener conocimiento y control de los servicios que se ejecutan. Muchos servicios son instalados por defecto cuando se realiza la instalación del servidor. A continuación se muestra en la Figura 10.28 los servicios que se ejecutan al inicio en un servidor Elastix luego de una instalación:

```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[root@elastix240 ~]# chkconfig --list
LCDd          0:desactivado 1:desactivado 2:desactivado 3:desactivado 4
:desactivado 5:desactivado 6:desactivado
NetworkManager 0:desactivado 1:desactivado 2:desactivado 3:desactivado 4
:desactivado 5:desactivado 6:desactivado
acpid         0:desactivado 1:desactivado 2:activo      3:activo      4
:activo 5:activo 6:desactivado
asterisk      0:desactivado 1:desactivado 2:activo      3:activo      4
:activo 5:activo 6:desactivado
atd           0:desactivado 1:desactivado 2:desactivado 3:activo      4
:activo 5:activo 6:desactivado
auditd       0:desactivado 1:desactivado 2:activo      3:activo      4
:activo 5:activo 6:desactivado
avahi-daemon  0:desactivado 1:desactivado 2:desactivado 3:activo      4
:activo 5:activo 6:desactivado
avahi-dnssconfd 0:desactivado 1:desactivado 2:desactivado 3:desactivado 4
:desactivado 5:desactivado 6:desactivado
crond        0:desactivado 1:desactivado 2:activo      3:activo      4
:activo 5:activo 6:desactivado
cups         0:desactivado 1:desactivado 2:activo      3:activo      4
:activo 5:activo 6:desactivado
cyrus-imapd  0:desactivado 1:desactivado 2:desactivado 3:activo      4
:activo 5:activo 6:desactivado
dahdi        0:desactivado 1:desactivado 2:activo      3:activo      4

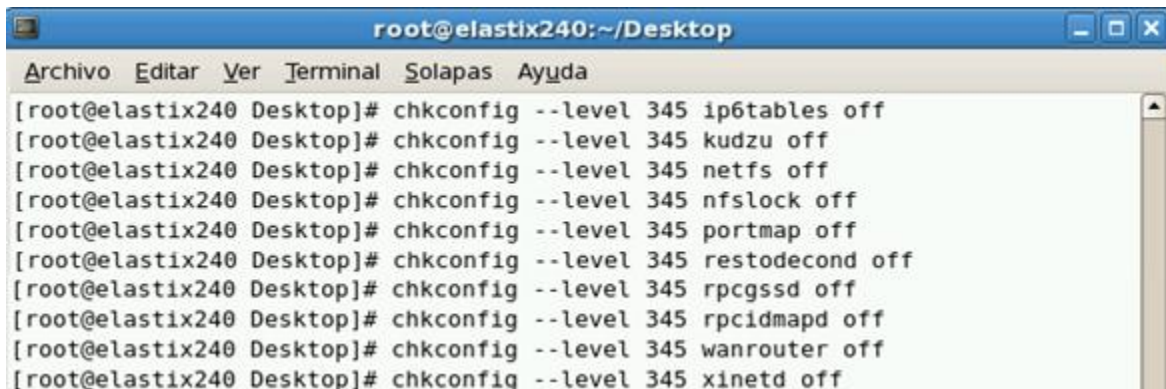
```

Figura 10.28: Lista de Servicios en Elastix

En la lista de servicios se debe evaluar cuáles servicios son necesarios y desactivar aquellos servicios que no se usaran en el servidor de comunicaciones unificadas. Esta acción se realiza con la ejecución del comando:

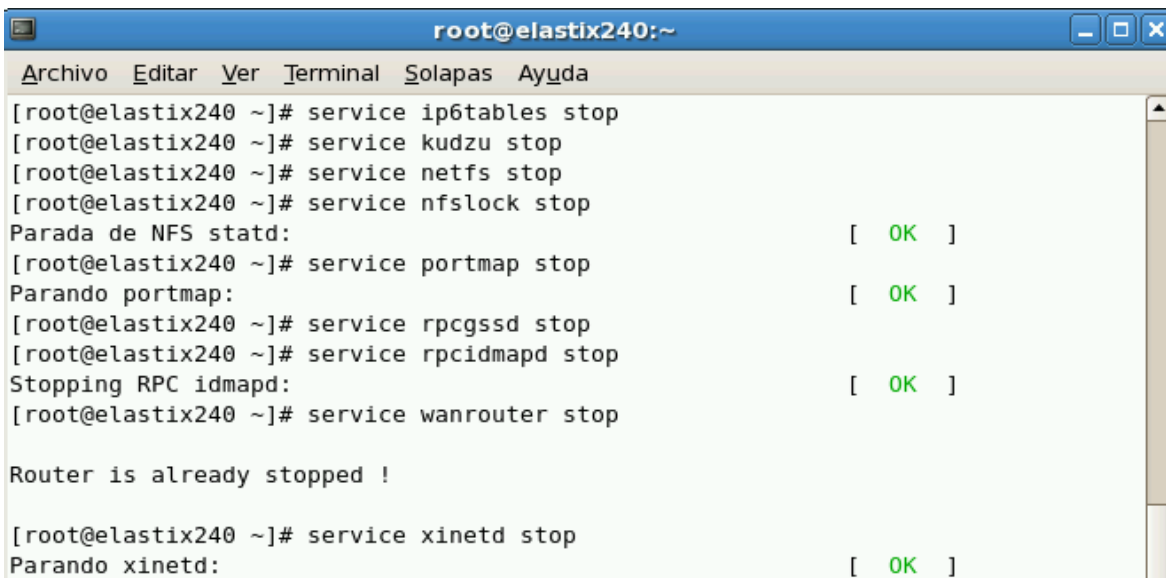
```
chkconfig --level 345 <servicio> off
```

En la Figura 10.29 se muestra la desactivación de algunos servicios de Elastix. Estos servicios quedan deshabilitados al momento del arranque del servidor de comunicaciones unificadas, pero se siguen ejecutando y se deben detener. La Figura 10.30 muestra la detención de los servicios que se desactivaron previamente. De esta forma solo quedan habilitados los servicios necesarios para el correcto funcionamiento del servidor de comunicaciones unificadas.



```
root@elastix240:~/Desktop
Archivo Editar Ver Terminal Solapas Ayuda
[root@elastix240 Desktop]# chkconfig --level 345 ip6tables off
[root@elastix240 Desktop]# chkconfig --level 345 kudzu off
[root@elastix240 Desktop]# chkconfig --level 345 netfs off
[root@elastix240 Desktop]# chkconfig --level 345 nfslock off
[root@elastix240 Desktop]# chkconfig --level 345 portmap off
[root@elastix240 Desktop]# chkconfig --level 345 restodecond off
[root@elastix240 Desktop]# chkconfig --level 345 rpcgssd off
[root@elastix240 Desktop]# chkconfig --level 345 rpcidmapd off
[root@elastix240 Desktop]# chkconfig --level 345 wanrouter off
[root@elastix240 Desktop]# chkconfig --level 345 xinetd off
```

Figura 10.29: Desactivación de Servicios en Elastix



```
root@elastix240:~
Archivo Editar Ver Terminal Solapas Ayuda
[root@elastix240 ~]# service ip6tables stop
[root@elastix240 ~]# service kudzu stop
[root@elastix240 ~]# service netfs stop
[root@elastix240 ~]# service nfslock stop
Parada de NFS statd: [ OK ]
[root@elastix240 ~]# service portmap stop
Parando portmap: [ OK ]
[root@elastix240 ~]# service rpcgssd stop
[root@elastix240 ~]# service rpcidmapd stop
Stopping RPC idmapd: [ OK ]
[root@elastix240 ~]# service wanrouter stop
Router is already stopped !
[root@elastix240 ~]# service xinetd stop
Parando xinetd: [ OK ]
```

Figura 10.30: Detención de Servicios en Elastix

10.7.2 Configuración de SSH

SSH (Secure Shell) es el nombre de un protocolo y del programa que lo implementa, sirve para acceder a máquinas remotas a través de una red. Permite hacer sesiones remotas.

La configuración del servidor SSH se encuentra almacenada en el archivo “/etc/ssh/sshd_config”. En este archivo se recomienda modificar algunas líneas con el fin de mejorar la seguridad en el protocolo SSH, por ejemplo deshabilitar el acceso de root, cambiar el puerto por defecto y la versión del protocolo:

```
PermitRootLogin no
Port 23022
Protocol 2
```

Después de configurar el archivo “/etc/ssh/sshd_config” para que no permita la autenticación con el usuario root, se realiza una conexión por SSH al servidor en el puerto 23022 TCP con la contraseña correcta de ese usuario y en la Figura 10.31 se observa que el acceso es denegado.



Figura 10.31: Acceso Rechazado por SSH en el puerto 23022 TCP al Usuario root

10.7.3 Fail2ban

Fail2ban es un analizador de archivos de registros que bloquea direcciones IP cuando detecta un comportamiento malicioso, por ejemplo, intentos fallidos de contraseñas. Adicionalmente, Fail2ban permite notificar vía email cuando una dirección IP ha sido bloqueada y bloquea una dirección IP por determinados puertos. Fail2ban es de software libre que se distribuye bajo licencia GPL. La Tabla 10.3 muestra algunos términos utilizados por Fail2ban para su configuración y la Figura 10.32 enseña la estructura inicial del directorio de Fail2ban. La instalación de Fail2ban se hace con el comando “yum install fail2ban”.

Término	Significado en Fail2ban
filter	Es un filtro que define una expresión regular que debe coincidir con un patrón correspondiente a un inicio de sesión fallido o cualquier otra expresión. Estos filtros se declaran en archivos separados en el directorio “/etc/fail2ban/filter.d” y se invocan desde el archivo jail.conf.
action	Es una acción que define varios comandos que son ejecutados en diferentes momentos. Estas acciones se declaran en archivos separados en el directorio “/etc/fail2ban/action.d” y se invocan desde el archivo jail.conf.
jail	Es una combinación de un filtro y una o varias acciones. Fail2ban puede manejar varios jails al mismo tiempo en un archivo llamado jail.conf en el directorio “/etc/fail2ban”.

Tabla 10.3: Términos de Fail2ban

```

/etc/fail2ban/
├── action.d
│   ├── dummy.conf
│   ├── hostsdeny.conf
│   ├── iptables.conf
│   ├── mail-whois.conf
│   ├── mail.conf
│   └── shorewall.conf
├── fail2ban.conf
├── fail2ban.local
├── filter.d
│   ├── apache-auth.conf
│   ├── apache-noscript.conf
│   ├── couriersmtp.conf
│   ├── postfix.conf
│   ├── proftpd.conf
│   ├── qmail.conf
│   ├── sasl.conf
│   ├── sshd.conf
│   └── vsftpd.conf
├── jail.conf
└── jail.local

```

Figura 10.32: Directorio Inicial Fail2ban

Antes de configurar Fail2ban para el análisis de registros se debe verificar que iptables está funcionando y que está configurado para arrancar cuando se inicie el servidor. Para ello se listan y se guardan las reglas existentes. Inicialmente no hay reglas y se permite todo el tráfico. Posteriormente se enciende el firewall y se le indica al firewall que siempre cargue estas reglas al momento de iniciar el servidor. La Figura 10.33 detalla el proceso de configuración inicial de iptables.

```

root@elastix240:~
Archivo Editar Ver Terminal Solapas Ayuda
[root@elastix240 ~]# iptables -L
Chain INPUT (policy ACCEPT)
target    prot opt source                destination

Chain FORWARD (policy ACCEPT)
target    prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target    prot opt source                destination
[root@elastix240 ~]# service iptables save
Guardando las reglas del cortafuegos a /etc/sysconfig/iptables: [ OK ]
[root@elastix240 ~]# service iptables start
Expurgar reglas del cortafuegos: [ OK ]
Configuración de cadenas a la política ACCEPT: filter [ OK ]
Descargando módulos iptables: [ OK ]
Aplicando reglas del cortafuegos iptables: [ OK ]
[root@elastix240 ~]# service iptables status
Tabla: filter
Chain INPUT (policy ACCEPT)
num target    prot opt source                destination

Chain FORWARD (policy ACCEPT)
num target    prot opt source                destination

Chain OUTPUT (policy ACCEPT)
num target    prot opt source                destination
[root@elastix240 ~]# chkconfig iptables on

```

Figura 10.33: Configuración Inicial de iptables

Antes de configurar los jails, filtros y acciones se debe definir el tipo de bloqueo que utilizará Fail2ban luego de detectar un comportamiento indebido. En el archivo “/etc/fail2ban/action.d/iptables-blocktype.conf” se configura un bloqueo del tipo “DROP” para que los paquetes del atacante sean desechados sin generar algún paquete de respuesta (ver Figura 10.34).

```

1# Fail2Ban configuration file
2#
3# Author: Daniel Black
4#
5# This is a included configuration file and includes the defination for the blocktype
6# used in all iptables based actions by default.
7#
8# The user can override the default in iptables-blocktype.local
9
10 [INCLUDES]
11
12 after = iptables-blocktype.local
13
14 [Init]
15
16# Option: blocktype
17# Note: This is what the action does with rules. This can be any jump target
18# as per the iptables man page (section 8). Common values are DROP
19# REJECT, REJECT --reject-with icmp-port-unreachable
20# Values: STRING
21
22
23 blocktype = DROP

```

Figura 10.34: Configuración del Tipo de Bloqueo

Para que Fail2ban pueda analizar los registros de Asterisk, se debe configurar el archivo “/etc/asterisk/logger.conf”. Al comienzo se agregan las dos líneas que cambian el formato de la hora reportada:

```
[general]
dateformat=%F %T
```

Al final se incluye la línea que permite que el registro de seguridad sea activado:

```
messages => security, notice,warning,error
```

Una vez editado el archivo se reinicia el módulo de registro de Asterisk:

```
asterisk -rx "logger reload"
```

Se crea un filtro para Asterisk (ver Figura 10.35) proporcionado en la wiki¹² de Fail2ban con el nombre “asterisk.conf”. Este filtro posteriormente será invocado por el jail de Asterisk.

¹² <http://www.fail2ban.org/wiki>

```

1# Fail2Ban configuration file
2# $Revision: 251 $
3
4 [INCLUDES]
5
6# Read common prefixes. If any customizations available -- read them from
7# common.local
8before = common.conf
9
10 [Definition]
11
12#_daemon = asterisk
13
14# Option: failregex
15# Notes.: regex to match the password failures messages in the logfile. The
16#         host must be matched by a group named "host". The tag "<HOST>" can
17#         be used for standard IP/hostname matching and is only an alias for
18#         (?:::f{4,6}:)?(?P<host>\S+)
19# Values: TEXT
20#
21# Asterisk 1.8 uses Host:Port format which is reflected here
22
23 failregex = NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - Wrong password
24             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - No matching peer found
25             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - No matching peer found
26             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - Username/auth name mismatch
27             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - Device does not match ACL
28             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - Peer is not supposed to register
29             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - ACL error (permit/deny)
30             NOTICE.* *: Registration from '.*' failed for '<HOST>:.*' - Device does not match ACL
31             NOTICE.* *: Registration from '\.*\.*' failed for '<HOST>:.*' - No matching peer found
32             NOTICE.* *: Registration from '\.*\.*' failed for '<HOST>:.*' - Wrong password
33             NOTICE.* <HOST> failed to authenticate as '.*'$
34             NOTICE.* *: No registration for peer '.*' \ (from <HOST>\)
35             NOTICE.* *: Host <HOST> failed MD5 authentication for '.*' (.*)
36             NOTICE.* *: Failed to authenticate user '.*@<HOST>.*'
37             NOTICE.* *: <HOST> failed to authenticate as '.*'
38             NOTICE.* *: <HOST> tried to authenticate with nonexistent user '.*'

```

Figura 10.35: Filtro para Asterisk 1.8

A este filtro, se le agrega una expresión regular para que Fail2ban pueda detectar el envío masivo de peticiones sin autenticación, como por ejemplo las peticiones inviteflood:

```
NOTICE.* *: Sending fake auth rejection for (device|user) \<sip:<HOST>:.*\>;tag=.*
```

Se crea un jail para Asterisk que permite configurar a Fail2ban para que bloquee por 24 horas direcciones IP luego de 3 autenticaciones fallidas y notificar por correo electrónico al administrador del Elastix el bloqueo de la dirección IP, como se puede observar en la Figura 10.36.

```

79 [asterisk-iptables]
80
81# detect password authentication failures
82# if more than 3 attempts are made within 6 hours, ban for 24 hours
83 enabled = true
84 filter = asterisk
85 action = iptables-allports[name=ASTERISK, protocol=all]
86         sendmail-whois[name=ASTERISK, dest=admr.elastix240@gmail.com,
87         sender=elastix240@gmail.com]
87 logpath = /var/log/asterisk/messages
88 maxretry = 3
89 findtime = 21600
90 bantime = 86400

```

Figura 10.36: Configuración del jail de Asterisk

Los intentos fallidos a Asterisk quedan escritos en el archivo “/var/log/asterisk/messages”. Al encontrar 3 intentos fallidos Fail2ban bloquea la dirección IP que los realizó. Se hicieron intentos de autenticación con usuarios y contraseñas erróneas. En la Figura 10.37 se evidencia la restricción del tráfico a la dirección IP atacante por parte de Fail2ban con la identificación “fail2ban-ASTERISK” y en la Figura 10.38 se muestra la notificación enviada al correo electrónico del administrador del servidor.

```

root@elastix240:~
Archivo Editar Ver Terminal Solapas Ayuda
7/0x02 state NEW
16 DROP      all  -f  0.0.0.0/0      0.0.0.0/0
17 DROP      tcp  --  0.0.0.0/0      0.0.0.0/0      tcp flags:0x3F
/0x3F
18 DROP      tcp  --  0.0.0.0/0      0.0.0.0/0      tcp flags:0x3F
/0x00
19 DROP      all  --  0.0.0.0/0      0.0.0.0/0

Chain FORWARD (policy ACCEPT)
num target      prot opt source          destination

Chain OUTPUT (policy ACCEPT)
num target      prot opt source          destination

Chain fail2ban-APACHE (1 references)
num target      prot opt source          destination
1 RETURN        all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-ASTERISK (1 references)
num target      prot opt source          destination
1 DROP          all  --  192.168.1.6    0.0.0.0/0
2 RETURN        all  --  0.0.0.0/0      0.0.0.0/0
  
```

Figura 10.37: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Autenticación Fallida a una Extensión por el Puerto 5060 UDP

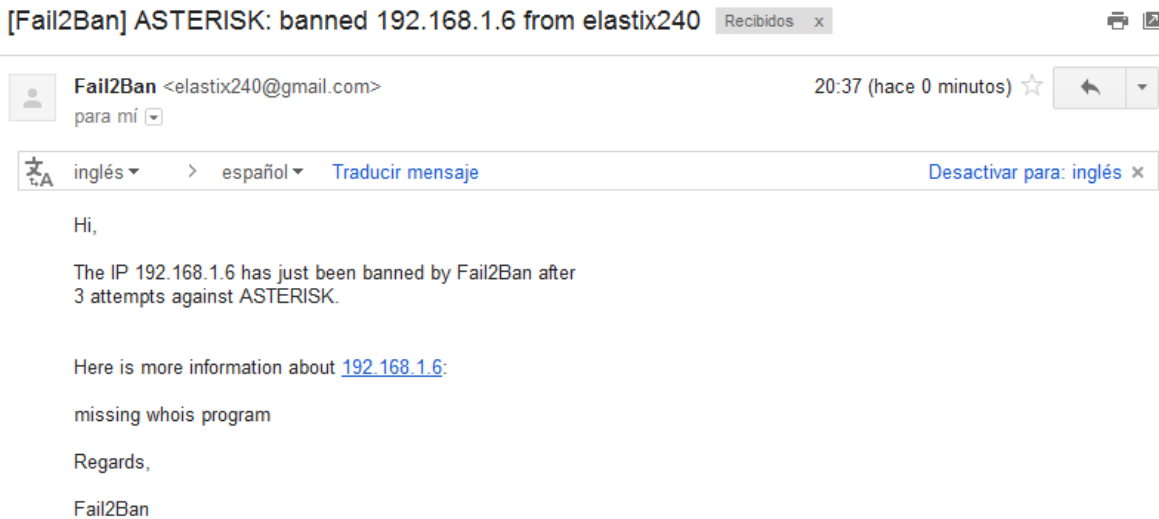


Figura 10.38: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Autenticación Fallida a una Extensión por el Puerto 5060 UDP

Se crea un jail para SSH con una configuración parecida al jail de Asterisk, bloquea por 24 horas direcciones IP luego de 3 autenticaciones fallidas por SSH y notifica por correo electrónico al administrador del Elastix el bloqueo de la dirección IP. Se utiliza el filtro “sshd.conf” que proporciona Fail2ban para SSH. La Figura 10.39 muestra el jail creado para SSH con el puerto 23022 debido a que se cambió el puerto por defecto.



```
165
166 [ssh-iptables]
167
168 # detect password authentication failures
169 # if more than 3 attempts are made within 6 hours, ban for 24 hours
170 enabled = true
171 filter = sshd
172 action = iptables[name=SSH, port=23022, protocol=tcp]
173         sendmail-whois[name=SSH, dest=admr.elastix240@gmail.com,
174         sender=elastix240@gmail.com]
174 logpath = /var/log/secure
175 maxretry = 3
176 bantime = 86400
```

Figura 10.39: Configuración del jail de SSH

Al realizar tres intentos fallidos de autenticación al puerto de SSH, Fail2ban bloquea la dirección IP de donde provienen los ataques. El puerto SSH fue cambiado al 23022 por seguridad. Se intenta autenticar por PuTTY tres veces a un usuario con contraseñas erróneas en el servidor VoIP. En la Figura 10.40 se muestra los intentos fallidos y la respuesta del servidor al usuario atacante. Adicionalmente, se aprecia en la Figura 10.41 el DROP a la dirección IP del atacante que hace Fail2ban con la identificación de “fail2ban-SSH” y en la Figura 10.42 la recepción del correo electrónico enviado por Fail2ban.

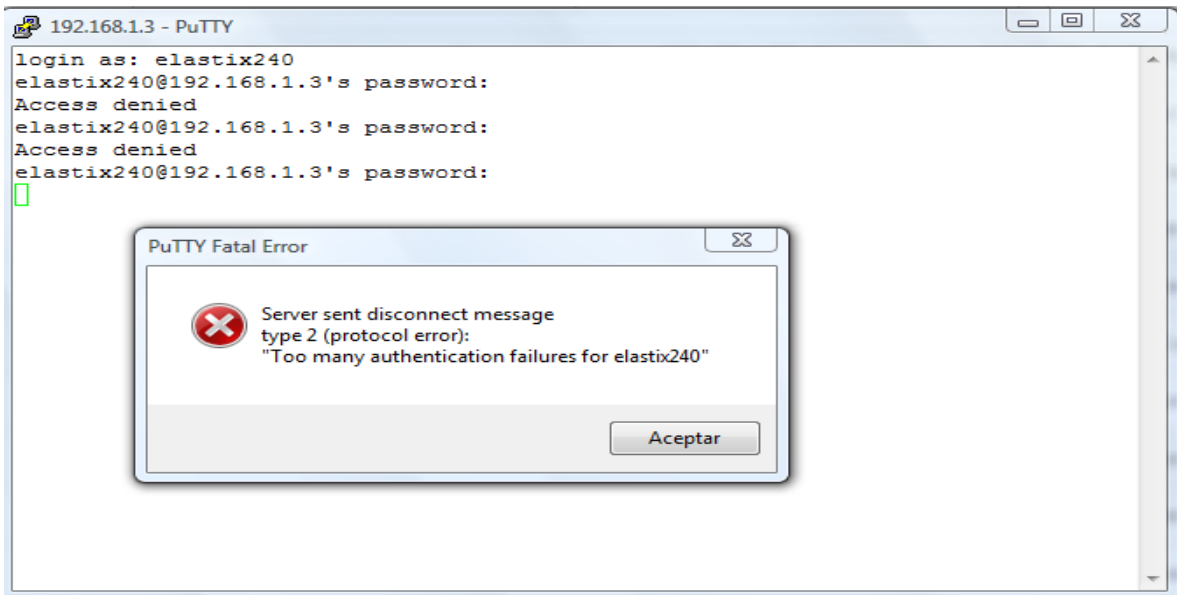


Figura 10.40: Intento Fallidos de Autenticación al Puerto 23022 TCP del Servidor VoIP utilizando PuTTY


```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
12  ACCEPT  icmp  --  0.0.0.0/0  0.0.0.0/0  icmp type 8 state NEW
13  ACCEPT  tcp  --  0.0.0.0/0  0.0.0.0/0  tcp dpt:25
14  DROP    tcp  --  0.0.0.0/0  0.0.0.0/0  tcp flags:!0x17/0x02 state NEW
15  DROP    all  -f  0.0.0.0/0  0.0.0.0/0
16  DROP    tcp  --  0.0.0.0/0  0.0.0.0/0  tcp flags:0x3F/0x3F
17  DROP    tcp  --  0.0.0.0/0  0.0.0.0/0  tcp flags:0x3F/0x00
18  DROP    all  --  0.0.0.0/0  0.0.0.0/0

Chain FORWARD (policy ACCEPT)
num target      prot opt source      destination
Chain OUTPUT (policy ACCEPT)
num target      prot opt source      destination
Chain fail2ban-SSH (1 references)
num target      prot opt source      destination
1  DROP    all  --  192.168.1.2  0.0.0.0/0
2  RETURN  all  --  0.0.0.0/0    0.0.0.0/0

```

Figura 10.41: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 23022 TCP

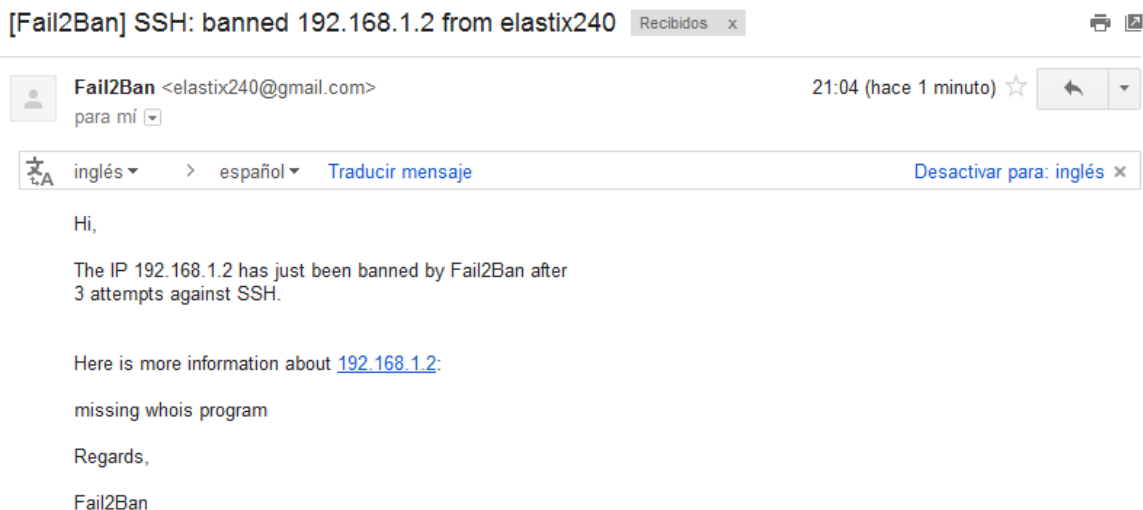
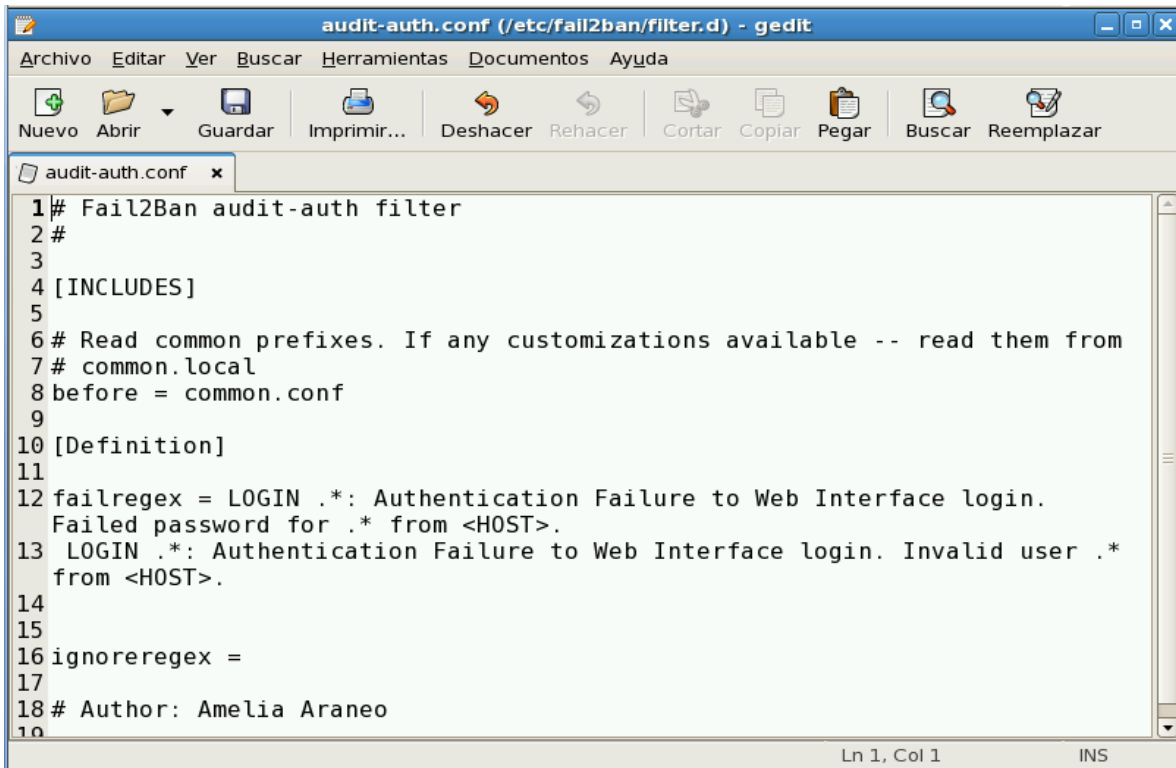


Figura 10.42: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 23022 TCP

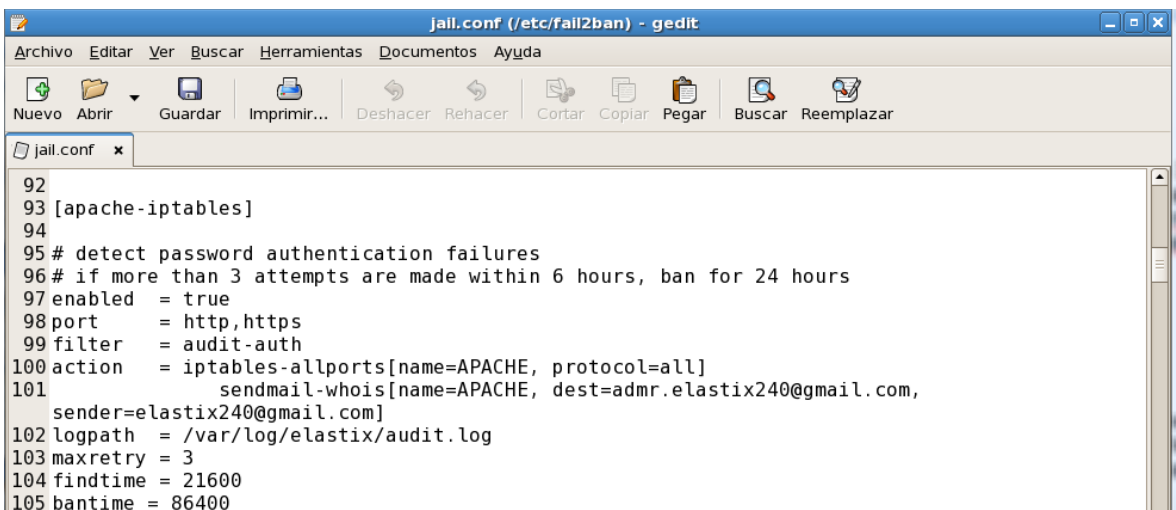
Se crea un filtro que permita reconocer los intentos fallidos de autenticación en la interfaz web del registro ubicado en el archivo “/var/log/elastix/audit.log”. En la Figura 10.43, se muestra el filtro creado para Apache.



```
audit-auth.conf (/etc/fail2ban/filter.d) - gedit
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Nuevo  Abrir  Guardar  Imprimir...  Deshacer  Rehacer  Cortar  Copiar  Pegar  Buscar  Reemplazar
audit-auth.conf x
1 # Fail2Ban audit-auth filter
2 #
3
4 [INCLUDES]
5
6 # Read common prefixes. If any customizations available -- read them from
7 # common.local
8 before = common.conf
9
10 [Definition]
11
12 failregex = LOGIN .*: Authentication Failure to Web Interface login.
13             Failed password for .* from <HOST>.
14 LOGIN .*: Authentication Failure to Web Interface login. Invalid user .*
15             from <HOST>.
16
17 ignoreregex =
18 # Author: Amelia Araneo
19
Ln 1, Col 1  INS
```

Figura 10.43: Filtro para Apache

Se crea un jail para Apache de forma que bloquee por 24 horas direcciones IP luego de 3 autenticaciones fallidas en la interfaz web de Elastix y notifique por correo electrónico al administrador del Elastix el bloqueo de la dirección IP. Se utiliza el filtro ubicado en el archivo “/etc/fail2ban/filter.d/audit-auth.conf” para identificar los intentos fallidos de autenticación. La Figura 10.44 muestra el jail creado para Apache. Adicionalmente, se crea un jail para la búsqueda de exploits y vulnerabilidades de PHP (ver Figura 10.45), un jail para los intentos de desbordamientos de Apache (ver Figura 10.46) y un jail para detectar fallos al encontrar el directorio home del servidor (ver Figura 10.47).



```
jail.conf (/etc/fail2ban) - gedit
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Nuevo  Abrir  Guardar  Imprimir...  Deshacer  Rehacer  Cortar  Copiar  Pegar  Buscar  Reemplazar
jail.conf x
92
93 [apache-iptables]
94
95 # detect password authentication failures
96 # if more than 3 attempts are made within 6 hours, ban for 24 hours
97 enabled = true
98 port = http,https
99 filter = audit-auth
100 action = iptables-allports[name=APACHE, protocol=all]
101          sendmail-whois[name=APACHE, dest=admr.elastix240@gmail.com,
102          sender=elastix240@gmail.com]
103 logpath = /var/log/elastix/audit.log
104 maxretry = 3
105 findtime = 21600
106 bantime = 86400
```

Figura 10.44: Configuración del jail de Apache

```
jail.conf (/etc/fail2ban) - gedit
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Nuevo  Abrir  Guardar  Imprimir...  Deshacer  Rehacer  Cortar  Copiar  Pegar  Buscar  Reemplazar
jail.conf x
108 [apache-noscript-iptables]
109
110 # detect potential search for exploits and php vulnerabilities
111 # if more than 3 attempts are made within 6 hours, ban for 24 hours
112 enabled = true
113 port = http,https
114 filter = apache-noscript
115 action = iptables-allports[name=APACHE-NOSCRIPT, protocol=all]
116         sendmail-whois[name=APACHE-NOSCRIPT, dest=admr.elastix240@gmail.com,
117         sender=elastix240@gmail.com]
117 logpath = /var/log/httpd/ssl_error_log.log
118 maxretry = 3
119 findtime = 21600
120 bantime = 86400
```

Figura 10.45: Configuración del jail de apache-noscript

```
jail.conf (/etc/fail2ban) - gedit
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Nuevo  Abrir  Guardar  Imprimir...  Deshacer  Rehacer  Cortar  Copiar  Pegar  Buscar  Reemplazar
jail.conf x
122
123 [apache-overflows-iptables]
124
125 # detect Apache overflow attempts
126 # if more than 2 attempts are made within 6 hours, ban for 24 hours
127 enabled = true
128 port = http,https
129 filter = apache-overflows
130 action = iptables-allports[name=APACHE-OVERFLOWS, protocol=all]
131         sendmail-whois[name=APACHE-OVERFLOWS, dest=admr.elastix240@gmail.com,
132         sender=elastix240@gmail.com]
132 logpath = /var/log/httpd/ssl_error_log.log
133 maxretry = 2
134 findtime = 21600
135 bantime = 86400
```

Figura 10.46: Configuración del jail de apache-overflows

```
jail.conf (/etc/fail2ban) - gedit
Archivo  Editar  Ver  Buscar  Herramientas  Documentos  Ayuda
Nuevo  Abrir  Guardar  Imprimir...  Deshacer  Rehacer  Cortar  Copiar  Pegar  Buscar  Reemplazar
jail.conf x
137
138 [apache-nohome-iptables]
139
140 # detect failures to find a home directory on a server
141 # if more than 2 attempts are made within 6 hours, ban for 24 hours
142 enabled = true
143 port = http,https
144 filter = apache-nohome
145 action = iptables-allports[name=APACHE-NOHOME, protocol=all]
146         sendmail-whois[name=APACHE-NOHOME, dest=admr.elastix240@gmail.com,
147         sender=elastix240@gmail.com]
147 logpath = /var/log/httpd/ssl_error_log.log
148 maxretry = 2
149 findtime = 21600
150 bantime = 86400
```

Figura 10.47: Configuración del jail de apache-nohome

Las autenticaciones fallidas realizadas en la interfaz web de usuario (ver Figura 10.48) que utiliza el puerto 443 TCP son registradas para que posteriormente Fail2ban tome medidas de seguridad. Si se realizan tres autenticaciones fallidas, a la dirección IP del atacante se le restringirá el acceso a la interfaz web de usuario. Se realizaron tres autenticaciones fallidas y la dirección IP que pertenecen los ataques fue bloqueada por Fail2ban bajo la identificación de “fail2ban-APACHE” (ver Figura 10.49). En la Figura 10.50 se muestra el correo electrónico de notificación que envía Fail2ban al administrador del servidor y seguido de esta acción el atacante deja de visualizar la interfaz gráfica (ver Figura 10.51).



Figura 10.48: Interfaz Gráfica de Usuario de Elastix 2.4.0

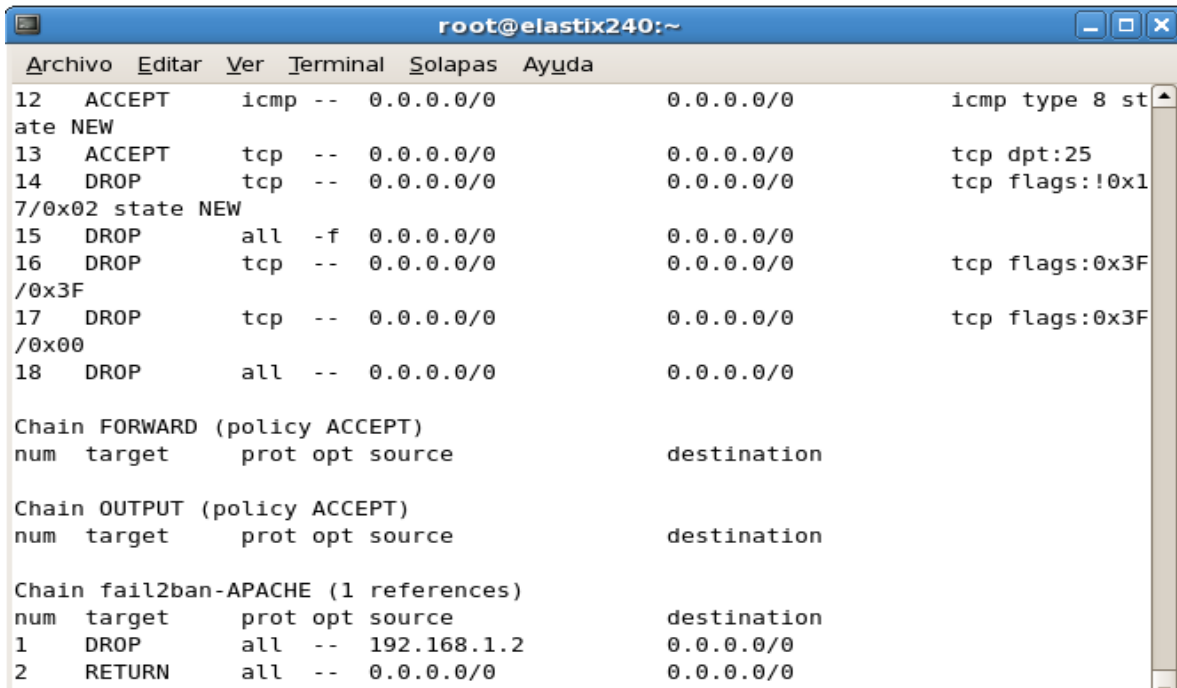


Figura 10.49: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 443 TCP

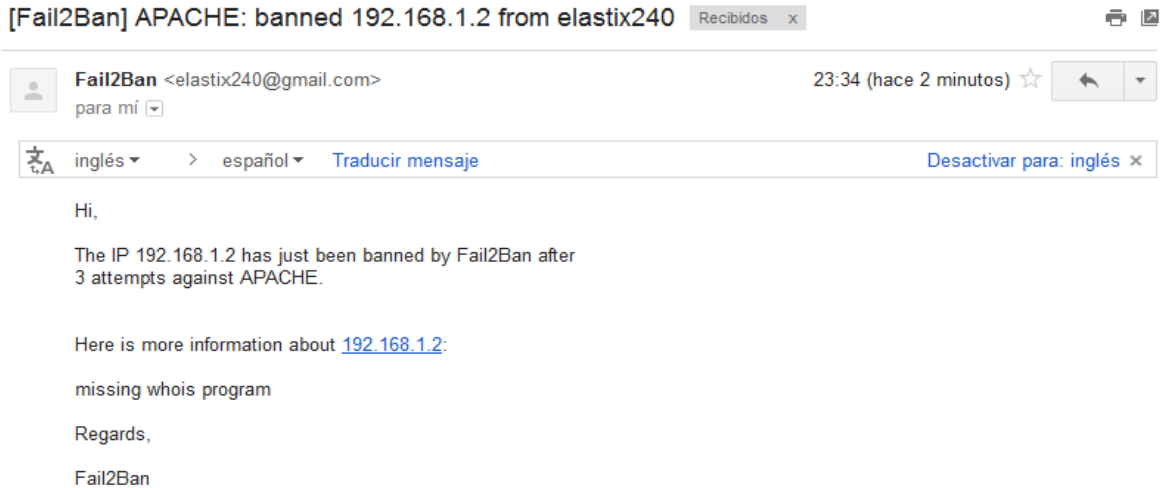


Figura 10.50: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó tres Autenticaciones Fallidas al Puerto 443 TCP



Figura 10.51: Intento de Visualizar la Interfaz Gráfica de Usuario de Elastix 2.4.0

10.7.4 Firewall

Netfilter es un framework disponible en el Kernel de Linux a partir de la versión 2.4, permitiendo interceptar y manipular paquetes de red. Netfilter es también el nombre que recibe el proyecto que se encarga de ofrecer herramientas open source para firewall basados en Linux.

El componente más popular construido sobre Netfilter es iptables, una herramienta de firewall que permite no solamente filtrar paquetes, sino también realizar traducción de direcciones de red (NAT) para IPv4 o mantener registros de log. Su funcionamiento se basa en la definición de reglas con las cuales se especifica qué hacer con el tráfico de red. Cuando se recibe o se envía un paquete, antes que dicho evento sea ejecutado, se consultan las reglas definidas por iptables y se determina si el paquete pasa el filtro definido en las reglas.

Para asegurar el tráfico que procesará el servidor, es necesario establecer reglas en las cuales se especifiquen la aceptación del tráfico VoIP en ambos sentidos (INPUT y OUTPUT) y rechazar el tráfico de entrada (INPUT) que pueda comprometer al servidor (vulnerabilidades en SIP, en mensajes ICMP, en paquetes IP fragmentados, en mensajes mal formados, etc).

Primero se debe restringir el acceso total del tráfico de entrada y permitir la salida de cualquier tráfico en el servidor mediante el uso de políticas. Esto nos permitirá realizar conexiones hacia el exterior con cualquier puerto pero sólo se permitirán conexiones de entrada según los puertos especificados:

```
iptables --policy INPUT DROP
iptables --policy OUTPUT ACCEPT
```

Se acepta el tráfico de entrada de SSH en el puerto 23022 TCP y se permite el tráfico del puerto 443 TCP para poder utilizar la interfaz gráfica de usuario del Elastix desde afuera:

```
iptables -A INPUT -p tcp -m tcp --dport 23022 -j ACCEPT
iptables -A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
```

Se acepta el tráfico de entrada de SIP en el puerto 5060 UDP y se acepta el tráfico RTP en los puertos comprendidos del 10000 al 20000 UDP:

```
iptables -A INPUT -p udp -m udp --dport 5060 -j ACCEPT
iptables -A INPUT -p udp -m udp --dport 10000:20000 -j ACCEPT
```

Las siguientes reglas permiten la entrada de tráfico establecido y relacionado, paquetes ICMP para querying y SMTP:

```
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -p icmp --icmp-type 8 -m state --state NEW -j ACCEPT
iptables -A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
```

Se debe rechazar el tráfico entrante que contribuya a ataques de SYN Flooding, paquetes mal formados y nulos:

```
iptables -A INPUT -p tcp ! --syn -m state --state NEW -j DROP
iptables -A INPUT -f -j DROP
iptables -A INPUT -p tcp --tcp-flags ALL ALL -j DROP
iptables -A INPUT -p tcp --tcp-flags ALL NONE -j DROP
```

Por último se rechaza el resto del tráfico entrante:

```
iptables -A INPUT -p all -j DROP
```

10.7.5 PortSentry

PortSentry es un software libre que se utiliza para monitorear puertos especificados que deben permanecer siempre inactivos. En caso de llegar una conexión a uno de los puertos monitoreados se puede escribir en el registro del sistema, bloquear la dirección IP del intruso o ejecutar un comando externo. En la Figura 10.52 se detalla la instalación de PortSentry en el servidor de comunicaciones unificadas.

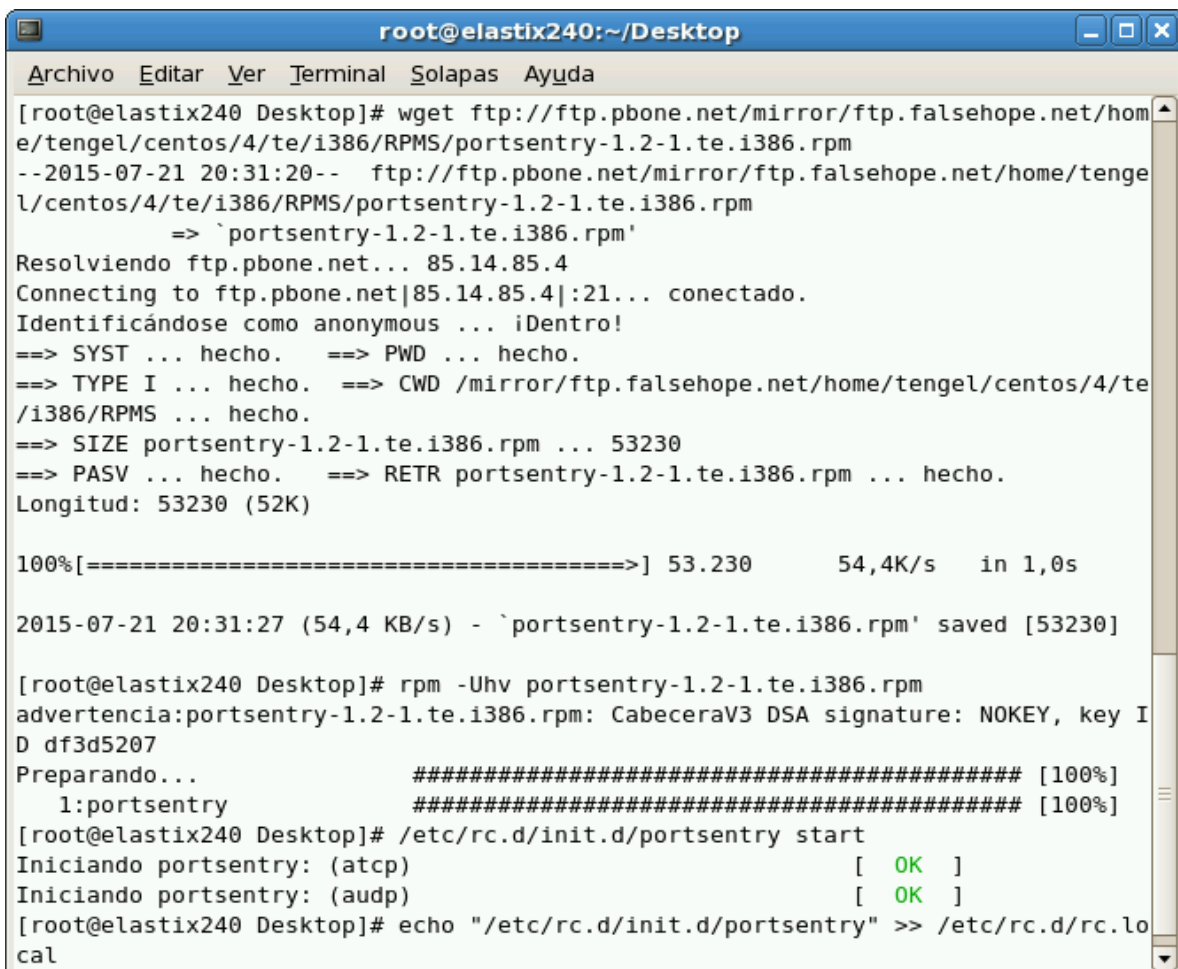
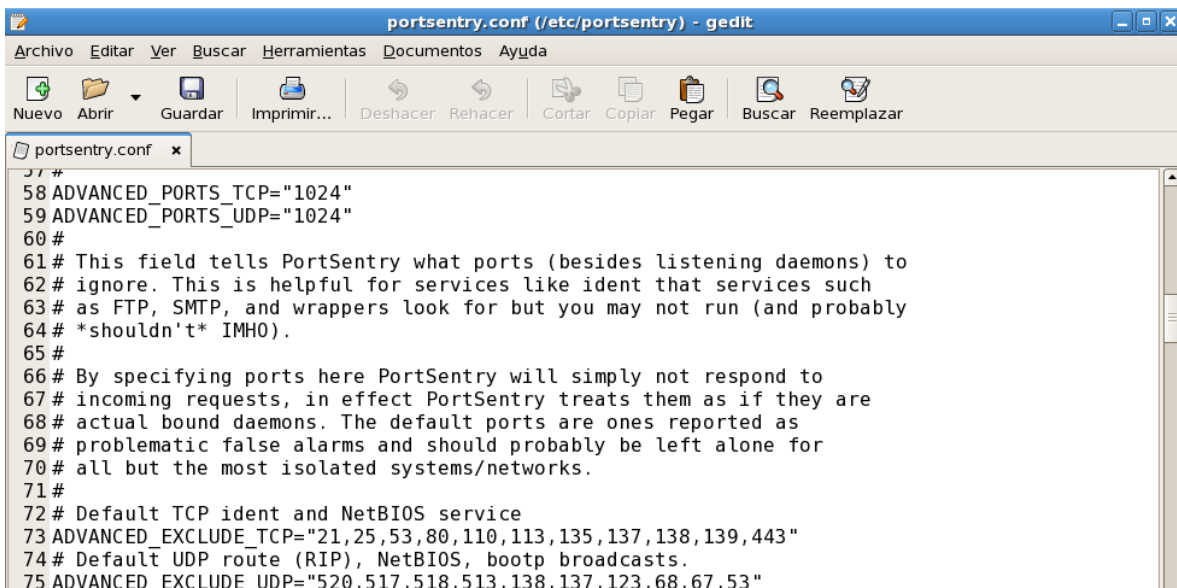


Figura 10.52: Instalación de PortSentry

Este software posee varios modos de operación. El más común es el modo "basic" en donde se especifica a PortSentry que escuche en determinados puertos TCP y UDP que son poco solicitados con las opciones UDP_PORTS y TCP_PORTS dentro del archivo de configuración "portsentry.conf". PortSentry abre el socket de los puertos especificados sin proporcionar servicio alguno, registrando cualquier intento de conexión y tomando la acción definida.

En el modo llamado "stealth", se abren sockets que permiten detectar una mayor cantidad de barridos de tipo SYN, FIN, NULL y XMAS.

Por último, existe el modo "advanced", considerado hasta cierto punto dentro de la categoría "stealth" con la diferencia de que no monitorea un número de puertos seleccionados sino que lo hace con un rango específico. En éste modo, PortSentry no abre ningún puerto, sino que le solicita al kernel que le notifique si llega alguna petición a algún puerto menor al especificado en las opciones "ADVANCED_PORTS_TCP" y "ADVANCED_PORTS_UDP" en el archivo de configuración "portsentry.conf". Las opciones "ADVANCED_EXCLUDE_TCP" y "ADVANCED_EXCLUDE_UDP" permiten excluir algunos puertos, con frecuencia los activos. La Figura 10.53 muestra la configuración de los puertos en modo "advanced". Si un intruso ataca un puerto TCP/UDP menor al especificado en "ADVANCED_PORTS_TCP" o "ADVANCED_PORTS_UDP", se toma la acción de bloquear el escaneo de puertos (ver Figura 10.54) y agregar una regla en iptables que descarte todas las conexiones entrantes de ese host (ver Figura 10.55).



```
58 ADVANCED_PORTS_TCP="1024"
59 ADVANCED_PORTS_UDP="1024"
60 #
61 # This field tells PortSentry what ports (besides listening daemons) to
62 # ignore. This is helpful for services like ident that services such
63 # as FTP, SMTP, and wrappers look for but you may not run (and probably
64 # *shouldn't* IMHO).
65 #
66 # By specifying ports here PortSentry will simply not respond to
67 # incoming requests, in effect PortSentry treats them as if they are
68 # actual bound daemons. The default ports are ones reported as
69 # problematic false alarms and should probably be left alone for
70 # all but the most isolated systems/networks.
71 #
72 # Default TCP ident and NetBIOS service
73 ADVANCED_EXCLUDE_TCP="21,25,53,80,110,113,135,137,138,139,443"
74 # Default UDP route (RIP), NetBIOS, bootp broadcasts.
75 ADVANCED_EXCLUDE_UDP="520,517,518,513,138,137,123,68,67,53"
```

Figura 10.53: Configuración de Puertos en Modo "advanced"


```

128 # 0 = Do not block UDP/TCP scans.
129 # 1 = Block UDP/TCP scans.
130 # 2 = Run external command only (KILL_RUN_CMD)
131
132 BLOCK_UDP="1"
133 BLOCK_TCP="1"

```

Figura 10.54: Configuración de Acción luego de Detectar Escaneo de Puertos

```

204 #
205 # iptables support for Linux
206 KILL_ROUTE="/sbin/iptables -I INPUT -s $TARGET$ -j DROP"

```

Figura 10.55: Configuración de Regla para iptables

Se combina PortSentry con Fail2ban para tener un control del tiempo en el que el host permanecerá bloqueado y para enviar un correo de notificación al administrador. Se crea un jail para PortSentry que bloquea por 24 horas las direcciones IP que se encuentren registradas en el archivo “portsentry.history” y notifica por correo electrónico al administrador del Elastix el bloqueo de la dirección IP. La Figura 10.56 muestra el jail creado para PortSentry.

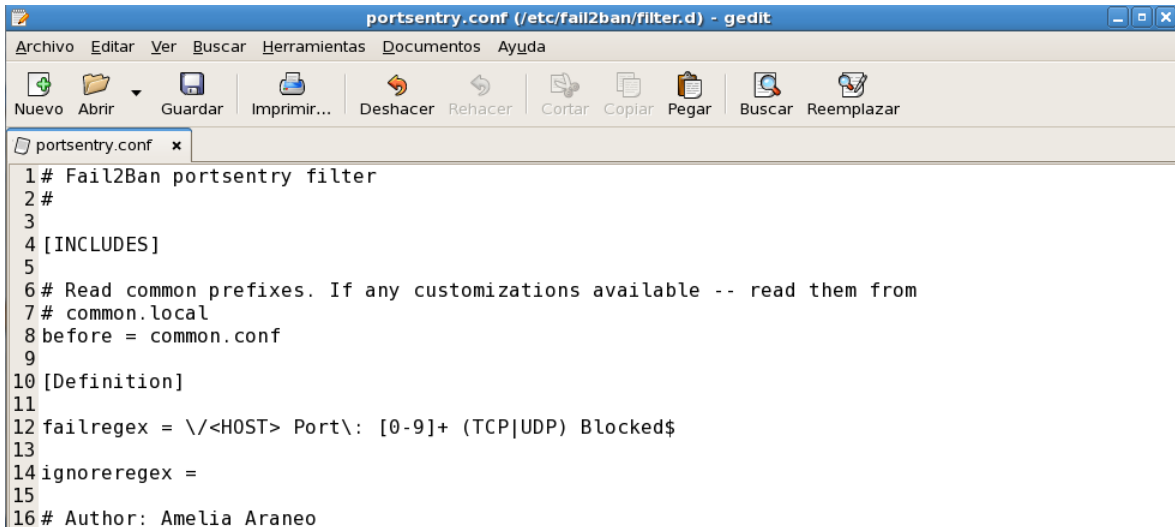
```

152
153 [portsentry-iptables]
154
155 # detect hosts that have been denied
156 # if more than 1 attempt are made within 6 hours, ban for 24 hours
157 enabled = true
158 filter = portsentry
159 action = iptables-allports[name=PORTSENTRY, protocol=all]
160 sender=elastix240@gmail.com]
161 logpath = /etc/portsentry/portsentry.history
162 maxretry = 1
163 findtime = 21600
164 bantime = 86400

```

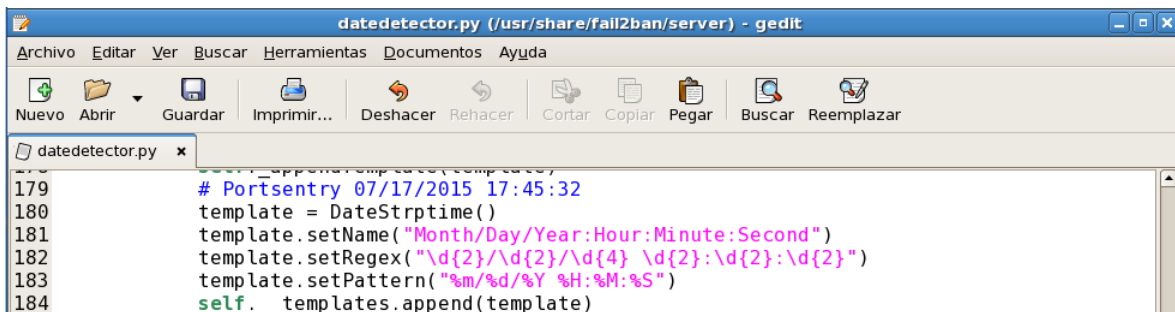
Figura 10.56: Configuración del jail de PortSentry

Se crea un filtro que permita a Fail2ban determinar los hosts que han sido bloqueados por PortSentry en el archivo “portsentry.history” ubicado en el directorio “/etc/portsentry”. La Figura 10.57 muestra el filtro creado. Para que Fail2ban pueda interpretar el formato de fecha del archivo “portsentry.history” se debe agregar un template en el archivo “datedetector.py” de Fail2ban (ver Figura 10.58).



```
1 # Fail2Ban portsentry filter
2 #
3
4 [INCLUDES]
5
6 # Read common prefixes. If any customizations available -- read them from
7 # common.local
8 before = common.conf
9
10 [Definition]
11
12 failregex = \/<HOST> Port\.: [0-9]+ (TCP|UDP) Blocked$
13
14 ignoreregex =
15
16 # Author: Amelia Araneo
```

Figura 10.57: Filtro para PortSentry



```
179 # Portsentry 07/17/2015 17:45:32
180 template = DateTime()
181 template.setName("Month/Day/Year:Hour:Minute:Second")
182 template.setRegex("\d{2}/\d{2}/\d{4} \d{2}:\d{2}:\d{2}")
183 template.setPattern("%m/%d/%Y %H:%M:%S")
184 self.__templates.append(template)
```

Figura 10.58: Template para Interpretación de Fecha de PortSentry

PortSentry se encarga de mantener bajo supervisión a todos los puertos TCP y UDP que se encuentren por debajo a 1024. Fail2ban se utiliza para bloquear por 24 horas la dirección IP del atacante que registró PortSentry y para enviar un correo de notificación al administrador. Al intentar autenticarse por SSH en el puerto 22 TCP se registra el intento fallido debido a que este puerto se encuentra bajo monitoreo y además no proporciona ningún servicio SSH. La Figura 10.59 muestra el mensaje de error de PuTTY cuando se intenta autenticar un usuario por SSH en el puerto 22 y en la Figura 10.60 se detalla en el número 2 de la lista de iptables la acción tomada por PortSentry, en este caso hizo un DROP a la dirección IP del atacante. La Figura 10.61 muestra el DROP a la dirección IP reportada por PortSentry que hace Fail2ban con la identificación de “fail2ban-PORTSENTRY” y en la Figura 10.62 se ve la recepción del correo electrónico enviado por Fail2ban.

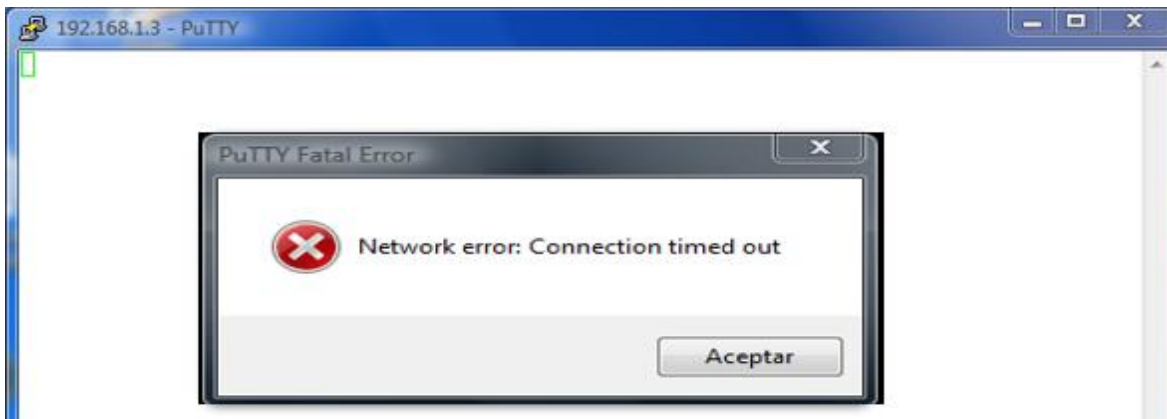


Figura 10.59: Intento de Conexión al Puerto 22 TCP del Servidor VoIP utilizando PuTTY

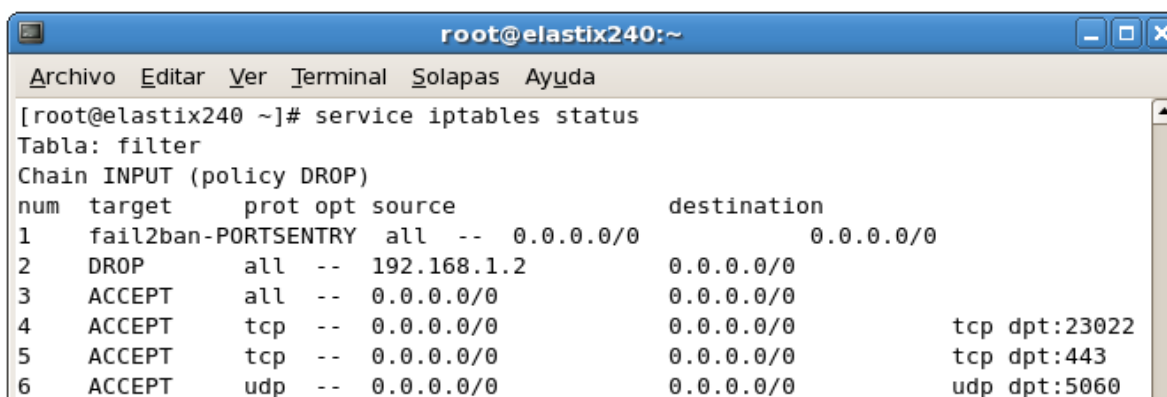


Figura 10.60: Restricción del Tráfico Entrante por PortSentry a una Dirección IP que Intentó Conexión al Puerto 22 TCP

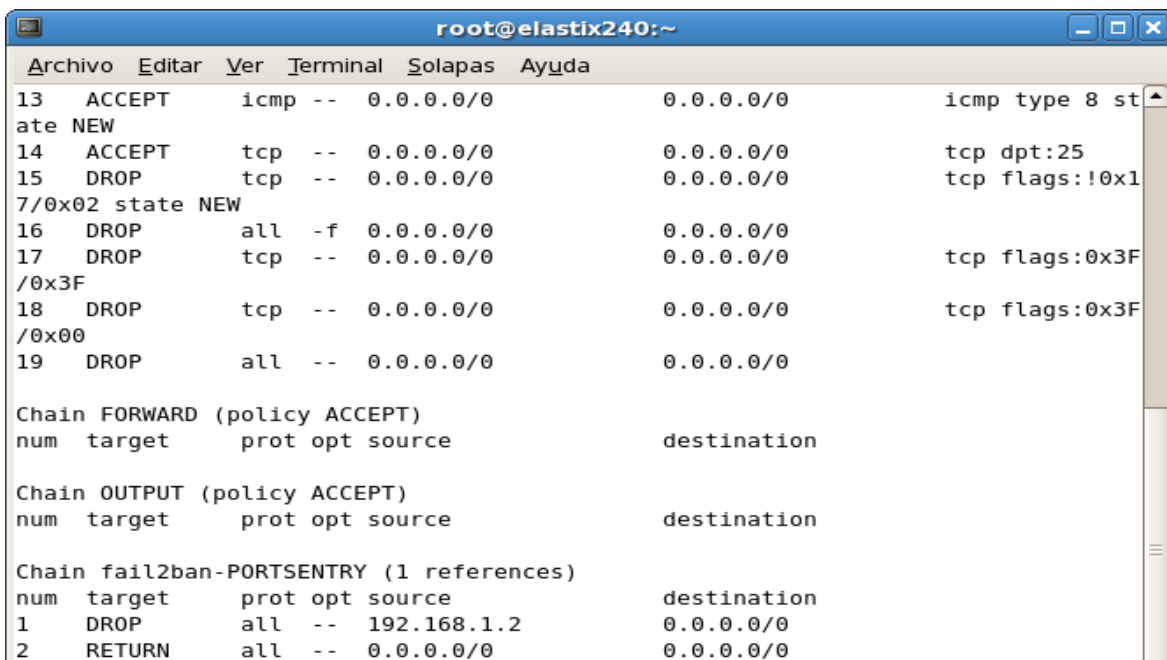


Figura 10.61: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Conexión al Puerto 22 TCP

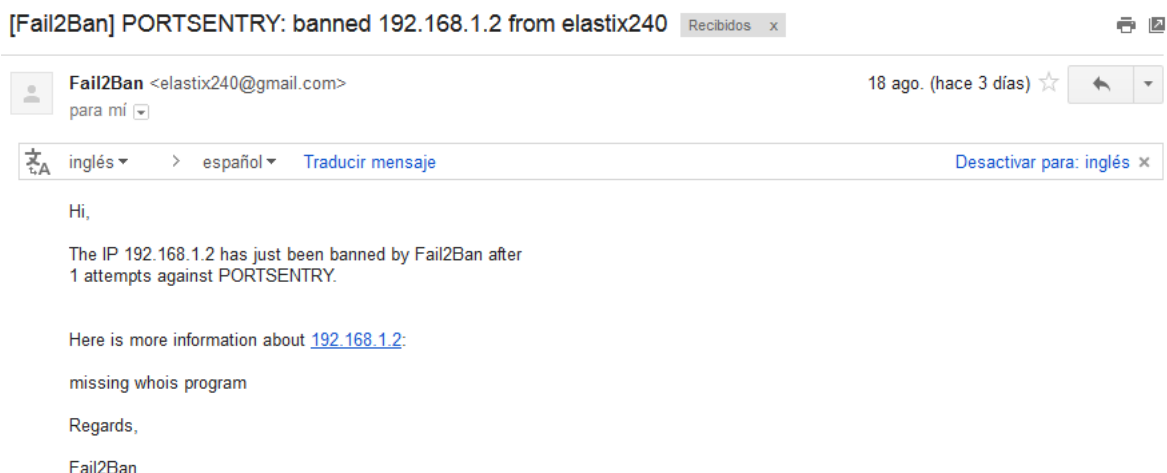


Figura 10.62: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que intentó Conexión al Puerto 22 TCP

Se realiza un Port Scanning con la herramienta Nmap (ver Figura 10.63) para descubrir puertos y servicios del servidor Elastix. La Figura 10.64 se muestra la restricción del tráfico entrante a la dirección IP reportada por PortSentry que hace Fail2ban con la identificación de jail "fail2ban-PORTSENTRY" y en la Figura 10.65 la recepción del correo electrónico enviado por Fail2ban.

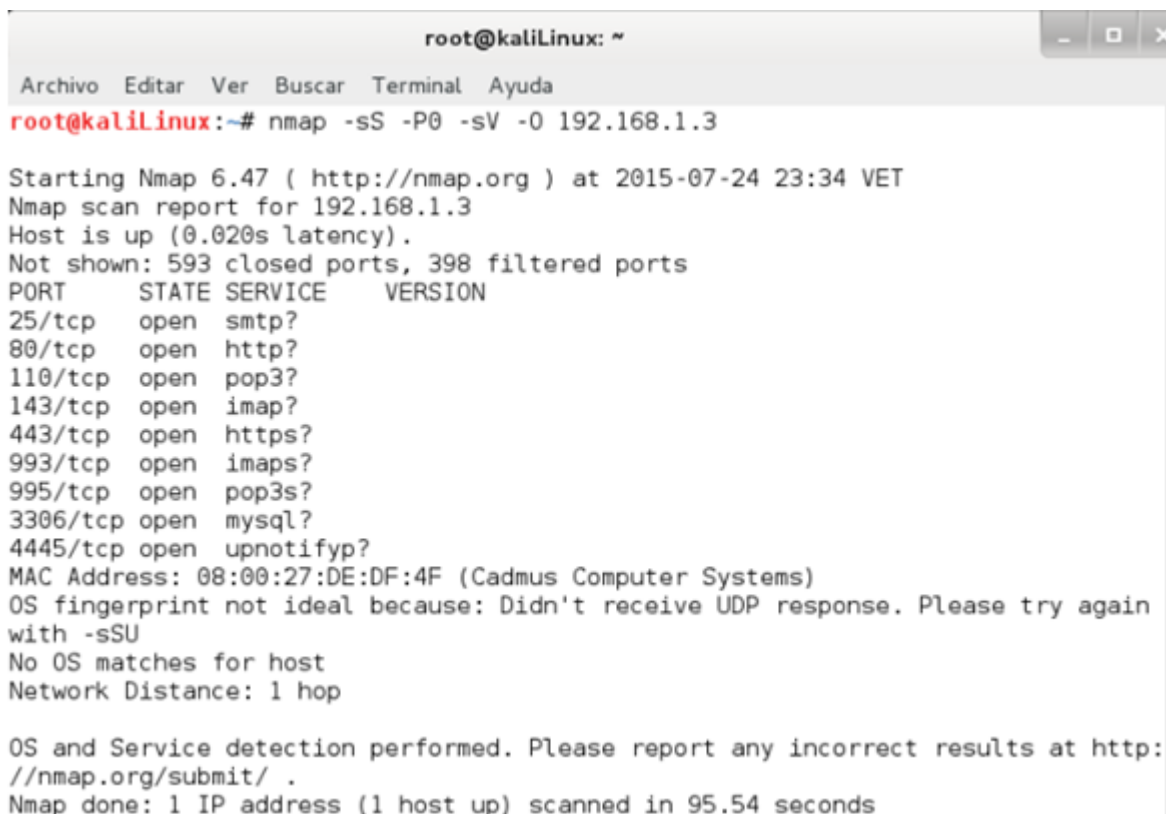


Figura 10.63: Descubrimiento Fallido de los Puertos Abiertos de un Host con Nmap

```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
7/0x02 state NEW
17  DROP      all  -f  0.0.0.0/0      0.0.0.0/0
18  DROP      tcp  --  0.0.0.0/0      0.0.0.0/0      tcp flags:0x3F
/0x3F
19  DROP      tcp  --  0.0.0.0/0      0.0.0.0/0      tcp flags:0x3F
/0x00
20  DROP      all  --  0.0.0.0/0      0.0.0.0/0

Chain FORWARD (policy ACCEPT)
num target      prot opt source          destination
Chain OUTPUT (policy ACCEPT)
num target      prot opt source          destination

Chain fail2ban-PORTSENTRY (1 references)
num target      prot opt source          destination
1  DROP      all  --  192.168.1.23    0.0.0.0/0
2  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-SSH (1 references)
num target      prot opt source          destination
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

```

Figura 10.64: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó un Port Scanning

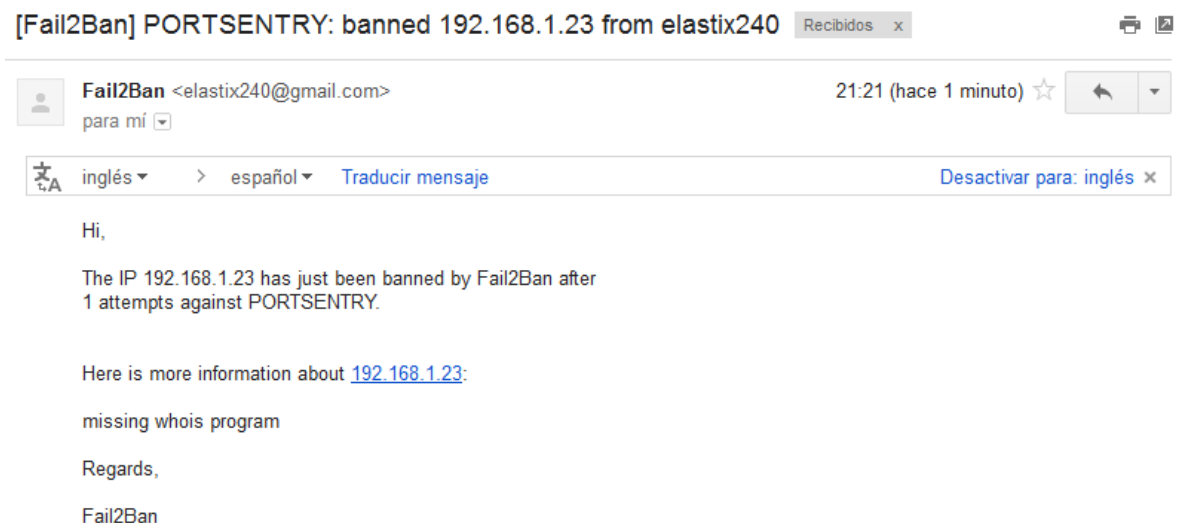


Figura 10.65: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó un Port Scanning

Por lo general antes de comenzar a realizar ataques a un servidor se hace un análisis de vulnerabilidades que permite al atacante conocer las amenazas y vulnerabilidades del sistema. En la Figura 10.66 se muestra tráfico de paquetes capturado con la herramienta Wireshark cuando se está realizando un análisis de vulnerabilidades. Se resalta en azul cuando el servidor VoIP responde a los paquetes enviados por el atacante.

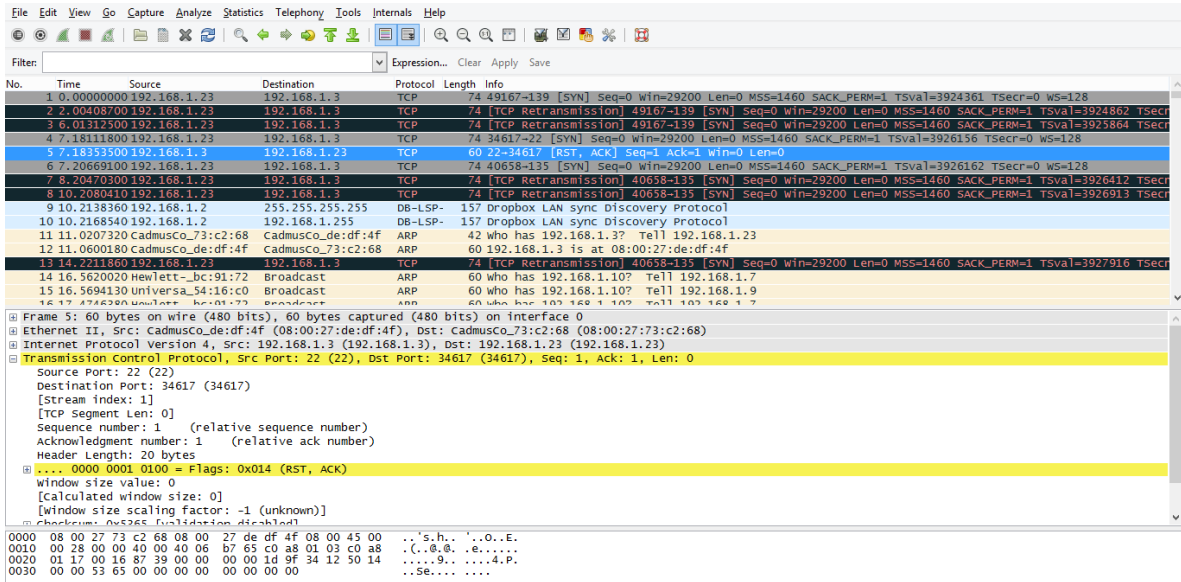


Figura 10.66: Captura de Paquetes en Wireshark Antes del Bloqueo de Dirección IP Atacante

El análisis de vulnerabilidades realizado por OpenVAS y Nessus involucra un Port Scanning que es detectado por las herramientas PortSentry y Fail2ban. Al detectar el ataque ambas herramientas bloquean el tráfico entrante que proviene de la dirección IP del atacante (ver Figura 10.67 y Figura 10.68) y se notifica por correo electrónico al administrador (ver Figura 10.69).

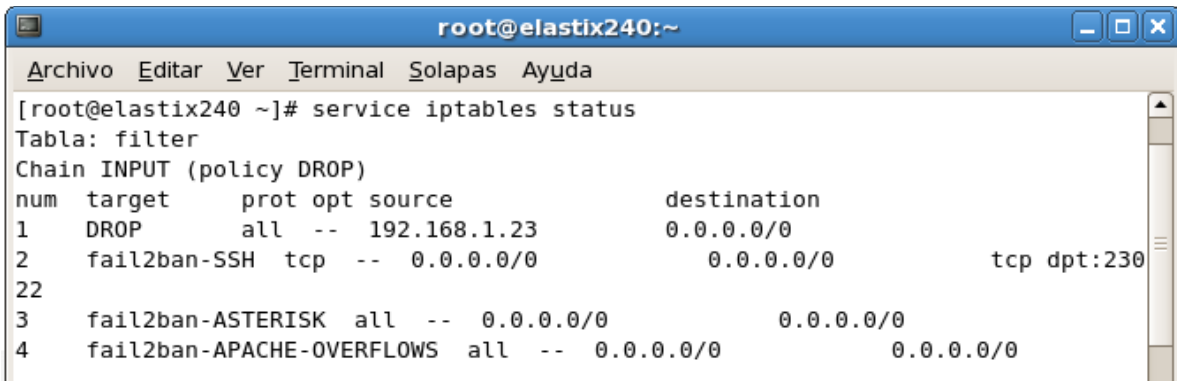


Figura 10.67: Restricción del Tráfico Entrante por PortSentry a una Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades

```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-APACHE-NOScript (1 references)
num target    prot opt source          destination
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-APACHE-OVERFLOWS (1 references)
num target    prot opt source          destination
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-ASTERISK (1 references)
num target    prot opt source          destination
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-PORTSENTRY (1 references)
num target    prot opt source          destination
1  DROP     all  --  192.168.1.23   0.0.0.0/0
2  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

Chain fail2ban-SSH (1 references)
num target    prot opt source          destination
1  RETURN    all  --  0.0.0.0/0      0.0.0.0/0

```

Figura 10.68: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades

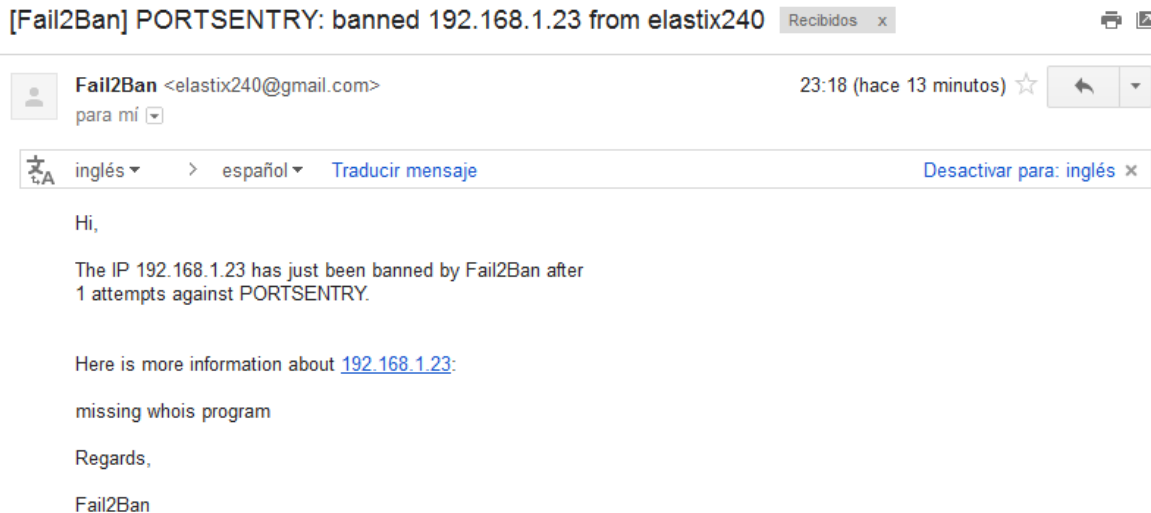


Figura 10.69: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Realizar un Análisis de Vulnerabilidades

Al restringir el tráfico entrante a la dirección IP del atacante, el servidor VoIP deja de contestar los paquetes enviados. La ausencia de respuesta provoca que el análisis de vulnerabilidades realizado por OpenVAS y Nessus no sea preciso. La Figura 10.70 muestra el resumen del análisis de vulnerabilidades realizado por OpenVAS y en la Figura 10.71 el resumen del análisis de Nessus. Ambos análisis aportaron muy poca información de interés del servidor VoIP al atacante.

Host	High	Medium	Low	Log	False Positive
192.168.1.3	0	0	1	8	0
Total: 1	0	0	1	8	0

Figura 10.70: Resumen del Análisis de Vulnerabilidades Realizado por OpenVAS

192.168.1.3					
Summary					
Critical	High	Medium	Low	Info	Total
0	0	0	0	4	4
Details					
Severity	Plugin Id	Name			
Info	10919	Open Port Re-check			
Info	11219	Nessus SYN scanner			
Info	19506	Nessus Scan Information			
Info	35716	Ethernet Card Manufacturer Detection			

Figura 10.71: Resumen del Análisis de Vulnerabilidades Realizado por Nessus

Al momento de realizar el análisis de vulnerabilidades, la herramienta OpenVAS envía un script al servidor para detectar si es vulnerable a ataques XSS (Cross-Site Scripting). Fail2ban detecta el intento, bloquea la dirección IP (ver Figura 10.72) y notifica por correo electrónico al administrador del sistema (ver Figura 10.73).

```

root@elastix240:~
Archivo Editar Ver Terminal Solapas Ayuda
Chain INPUT (policy ACCEPT)
num target      prot opt source                destination
1  fail2ban-APACHE-NOSCRIPT  all  --  0.0.0.0/0              0.0.0.0/0

Chain FORWARD (policy ACCEPT)
num target      prot opt source                destination

Chain OUTPUT (policy ACCEPT)
num target      prot opt source                destination

Chain fail2ban-APACHE-NOSCRIPT (1 references)
num target      prot opt source                destination
1  DROP          all  --  192.168.1.23          0.0.0.0/0
2  RETURN        all  --  0.0.0.0/0             0.0.0.0/0

```

Figura 10.72: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Enviar Script

[Fail2Ban] APACHE-NOSCRIP: banned 192.168.1.23 from elastix240 Recibidos x 🖨️ 🗑️

 **Fail2Ban** <elastix240@gmail.com> 23:02 (hace 1 minuto) ☆ ↶ ⌵
para mí ⌵

 inglés ▾ > español ▾ [Traducir mensaje](#) [Desactivar para: inglés](#) ✕

Hi,

The IP 192.168.1.23 has just been banned by Fail2Ban after 4 attempts against APACHE-NOSCRIP.

Here is more information about [192.168.1.23](#):

missing whois program

Regards,

Fail2Ban

Figura 10.73: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que Intentó Enviar Script

Se realiza un ataque de fuerza bruta con la herramienta Medusa con el nombre de usuario “root”, puerto 22 TCP y un diccionario. En la Figura 10.74 muestra el ataque realizado que no obtiene un resultado satisfactorio debido a que el puerto utilizado es supervisado por PortSentry y la dirección IP del atacante es bloqueada por Fail2ban (ver Figura 10.75) y se notifica por correo electrónico al administrador (ver Figura 10.76).

```
root@kaliLinux: ~/Desktop/Ataques
Archivo Editar Ver Buscar Terminal Ayuda
root@kaliLinux:~/Desktop/Ataques# medusa -h 192.168.1.3 -u root -P passwords.txt
-M ssh
Medusa v2.0 [http://www.foofus.net] (C) JoMo-Kun / Foofus Networks <jmk@foofus.net>

ERROR: Thread B6339B70: Host: 192.168.1.3 Cannot connect [unreachable], retrying
(1 of 3 retries)
ERROR: Thread B6339B70: Host: 192.168.1.3 Cannot connect [unreachable], retrying
(2 of 3 retries)
ERROR: Thread B6339B70: Host: 192.168.1.3 Cannot connect [unreachable], retrying
(3 of 3 retries)
NOTICE: ssh.mod: failed to connect, port 22 was not open on 192.168.1.3
```

Figura 10.74: Ataque de Fuerza Bruta al Puerto 22 TCP No Satisfactoria

```

root@elastix240:~
Archivo Editar Ver Terminal Solapas Ayuda
1 RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain fail2ban-APACHE-NOSCRYPT (1 references)
num target prot opt source destination
1 RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain fail2ban-APACHE-OVERFLOWS (1 references)
num target prot opt source destination
1 RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain fail2ban-ASTERISK (1 references)
num target prot opt source destination
1 RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain fail2ban-PORTSENTRY (1 references)
num target prot opt source destination
1 DROP all -- 192.168.1.8 0.0.0.0/0
2 RETURN all -- 0.0.0.0/0 0.0.0.0/0

Chain fail2ban-SSH (1 references)
num target prot opt source destination
1 RETURN all -- 0.0.0.0/0 0.0.0.0/0

```

Figura 10.75: Restricción del Tráfico Entrante por Fail2ban a una Dirección IP que Intentó Ataque de Fuerza Bruta al Puerto 22 TCP

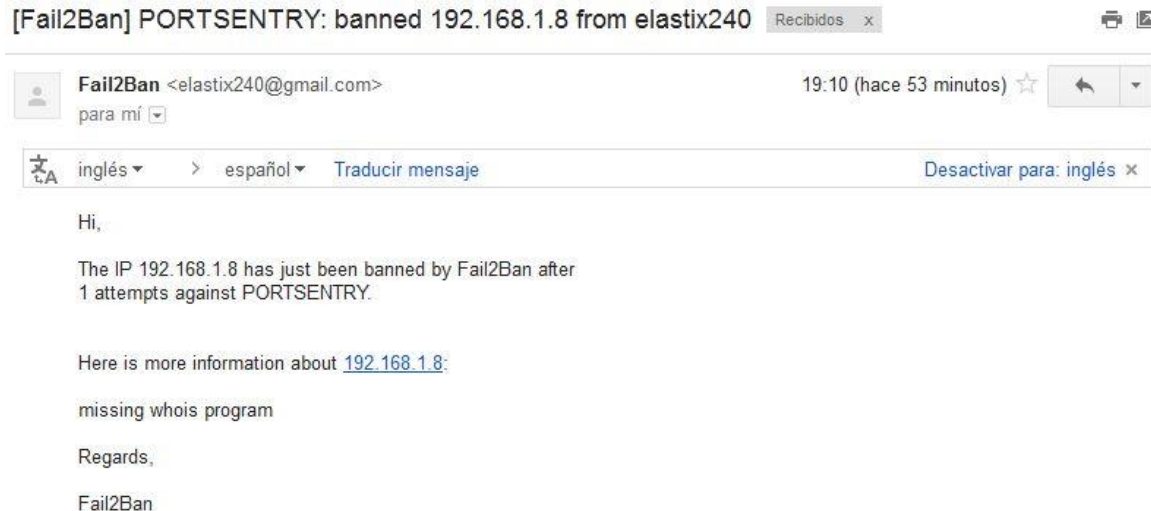
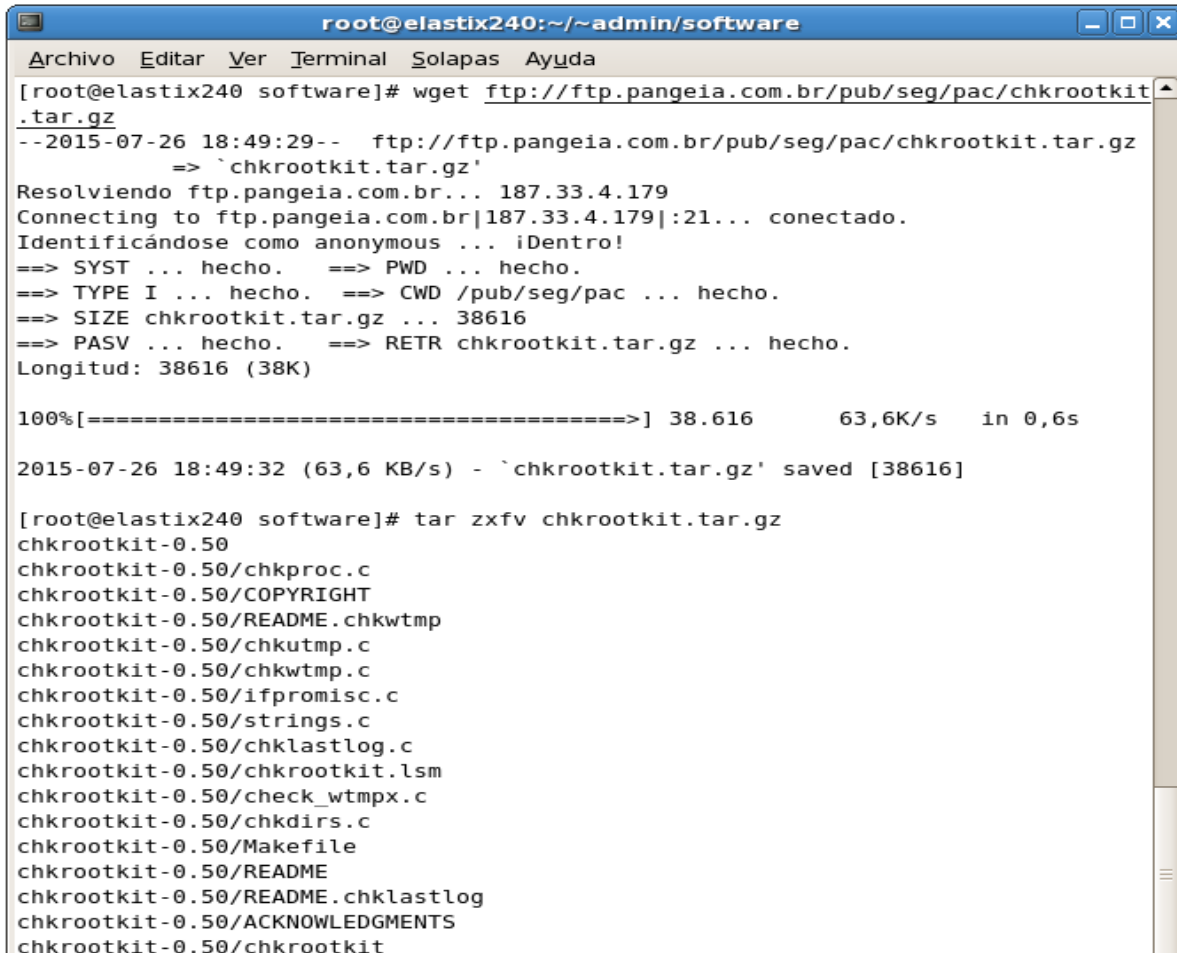


Figura 10.76: Correo Electrónico Enviado por Fail2ban como Resultado del Bloqueo de Dirección IP que intentó Ataque de Fuerza Bruta al Puerto 22 TCP

10.7.6 chkrootkit

Un rootkit es un tipo de software sigiloso, generalmente malicioso y diseñado para esconder la existencia de procesos o programas de los métodos normales de detección y que permitirá continuar con acceso privilegiado al computador. Se instala chkrootkit en el servidor para permitir localizar rootkits conocidos (ver Figura 10.77) y se programa para que todos los días a las 3 a.m. busque rootkits en el servidor (ver Figura 10.78) y envíe un

correo electrónico al administrador con los resultados. En caso de encontrar algún rootkit se detallará en el correo electrónico. Para ejecutar el chkrootkit de forma inmediata en el sistema se ejecuta el comando “./chkrootkit” dentro del directorio donde se encuentre instalado el chkrootkit (ver Figura 10.79).



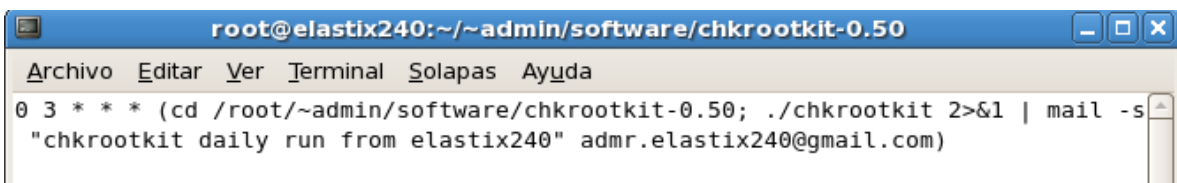
```
root@elastix240:~/~/admin/software
Archivo Editar Ver Terminal Solapas Ayuda
[root@elastix240 software]# wget ftp://ftp.pangeia.com.br/pub/seg/pac/chkrootkit
.tar.gz
--2015-07-26 18:49:29-- ftp://ftp.pangeia.com.br/pub/seg/pac/chkrootkit.tar.gz
=> `chkrootkit.tar.gz'
Resolviendo ftp.pangeia.com.br... 187.33.4.179
Connecting to ftp.pangeia.com.br|187.33.4.179|:21... conectado.
Identificándose como anonymous ... ¡Dentro!
==> SYST ... hecho. ==> PWD ... hecho.
==> TYPE I ... hecho. ==> CWD /pub/seg/pac ... hecho.
==> SIZE chkrootkit.tar.gz ... 38616
==> PASV ... hecho. ==> RETR chkrootkit.tar.gz ... hecho.
Longitud: 38616 (38K)

100%[=====] 38.616 63,6K/s in 0,6s

2015-07-26 18:49:32 (63,6 KB/s) - `chkrootkit.tar.gz' saved [38616]

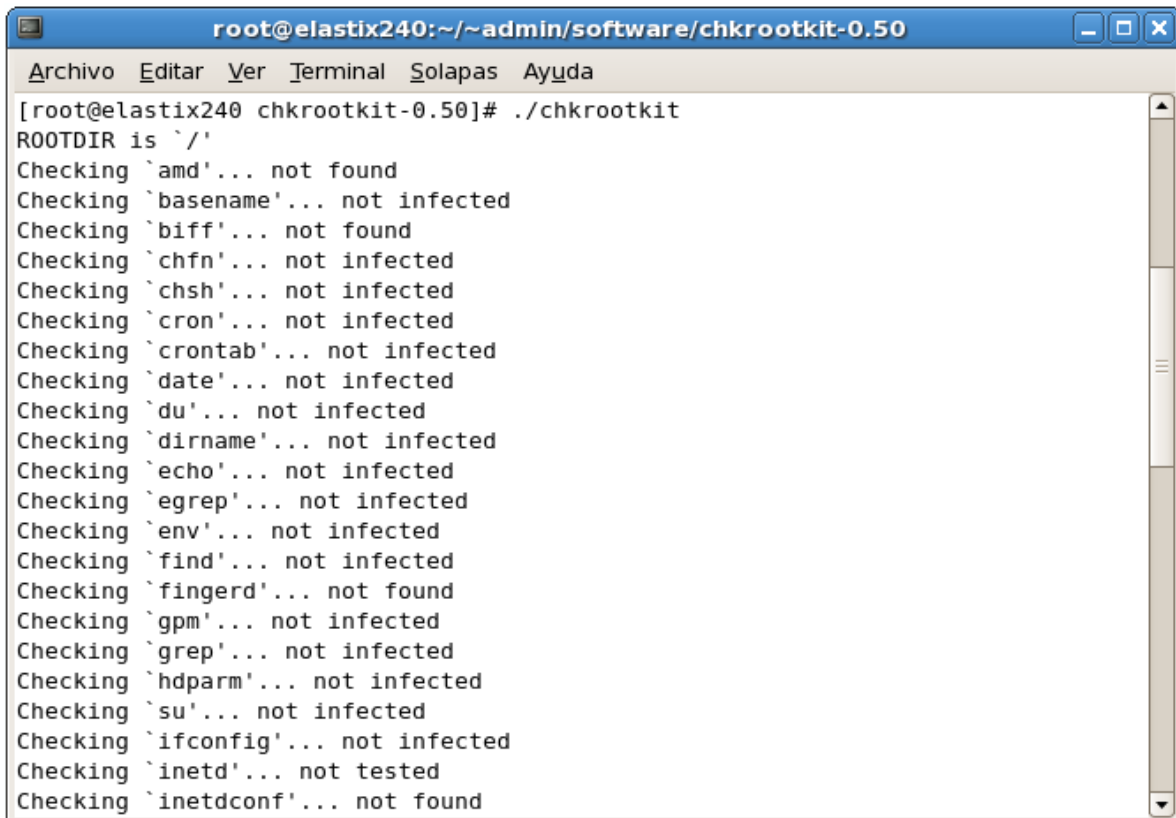
[root@elastix240 software]# tar xzfv chkrootkit.tar.gz
chkrootkit-0.50
chkrootkit-0.50/chkproc.c
chkrootkit-0.50/COPYRIGHT
chkrootkit-0.50/README.chkwtmp
chkrootkit-0.50/chkutmp.c
chkrootkit-0.50/chkwtmp.c
chkrootkit-0.50/ifpromisc.c
chkrootkit-0.50/strings.c
chkrootkit-0.50/chklastlog.c
chkrootkit-0.50/chkrootkit.lsm
chkrootkit-0.50/check_wtmpx.c
chkrootkit-0.50/chkdirs.c
chkrootkit-0.50/Makefile
chkrootkit-0.50/README
chkrootkit-0.50/README.chklastlog
chkrootkit-0.50/ACKNOWLEDGMENTS
chkrootkit-0.50/chkrootkit
```

Figura 10.77: Instalación de chkrootkit



```
root@elastix240:~/~/admin/software/chkrootkit-0.50
Archivo Editar Ver Terminal Solapas Ayuda
0 3 * * * (cd /root/~/admin/software/chkrootkit-0.50; ./chkrootkit 2>&1 | mail -s
"chkrootkit daily run from elastix240" admr.elastix240@gmail.com)
```

Figura 10.78: Programación de chkrootkit para Ejecutar Todos los Días a las 3 a.m.



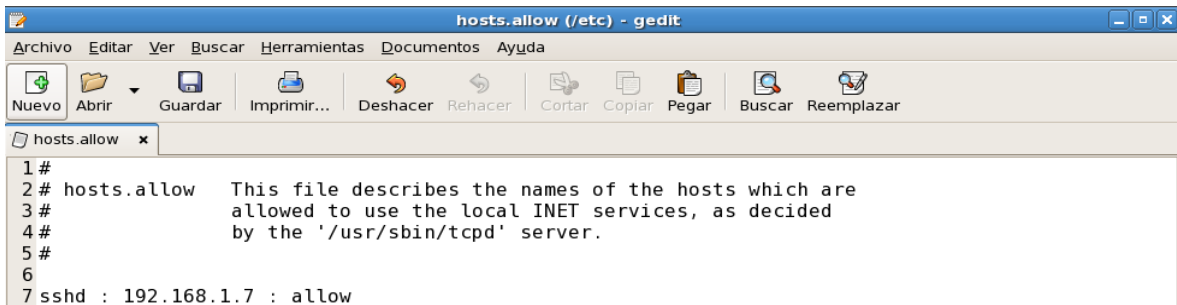
```
root@elastix240:~/~/admin/software/chkrootkit-0.50
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[root@elastix240 chkrootkit-0.50]# ./chkrootkit
ROOTDIR is '/'
Checking `amd'... not found
Checking `basename'... not infected
Checking `biff'... not found
Checking `chfn'... not infected
Checking `chsh'... not infected
Checking `cron'... not infected
Checking `crontab'... not infected
Checking `date'... not infected
Checking `du'... not infected
Checking `dirname'... not infected
Checking `echo'... not infected
Checking `egrep'... not infected
Checking `env'... not infected
Checking `find'... not infected
Checking `fingerd'... not found
Checking `gpm'... not infected
Checking `grep'... not infected
Checking `hdparm'... not infected
Checking `su'... not infected
Checking `ifconfig'... not infected
Checking `inetd'... not tested
Checking `inetdconf'... not found
```

Figura 10.79: Ejecución de chkrootkit

10.7.7 TCPWrapper

Es un sistema de ACL (Access Control List) que trabaja en terminales y se usa para filtrar el acceso de red a servicios de protocolos de Internet que se ejecutan en sistemas operativos tipo Unix como Linux o BSD. Permite que las direcciones IP, los nombres de terminales y/o respuestas de consultas de las terminales o subredes sean usadas como tokens para aplicar procesos de filtrado con el propósito de control de acceso. Por lo tanto, los TCPWrappers añaden una capa adicional de protección mediante la definición de cuáles hosts tienen permitido conectarse a los servicios de red especificados.

El paquete de TCPWrappers está instalado por defecto. Trabaja con los archivos “hosts.allow” y “hosts.deny” ubicados en el directorio “/etc”. Cuando un intento de conexión es hecho a un servicio, TCPWrapper consulta los archivos anteriormente mencionados para determinar si el cliente tiene permitido conectarse. En la mayoría de los casos, se utiliza el demonio syslog (syslogd) para escribir el nombre del host solicitante y el servicio solicitado a “/var/log/secure” o “/var/log/messages”. Si a un cliente se le permite conectarse, los TCPWrappers liberan el control de la conexión al servicio solicitado y no interfieren más con la comunicación entre el cliente y el servidor. Además del control de acceso y registro, los TCPWrappers pueden activar comandos para interactuar con el cliente antes de negar o liberar el control de la conexión al servicio solicitado. En la Figura 10.80 y la Figura 10.81, se muestran como filtrar el acceso a SSH con los archivos “hosts.allow” y “hosts.deny”.



```
1 #
2 # hosts.allow This file describes the names of the hosts which are
3 #             allowed to use the local INET services, as decided
4 #             by the '/usr/sbin/tcpd' server.
5 #
6 #
7 sshd : 192.168.1.7 : allow
```

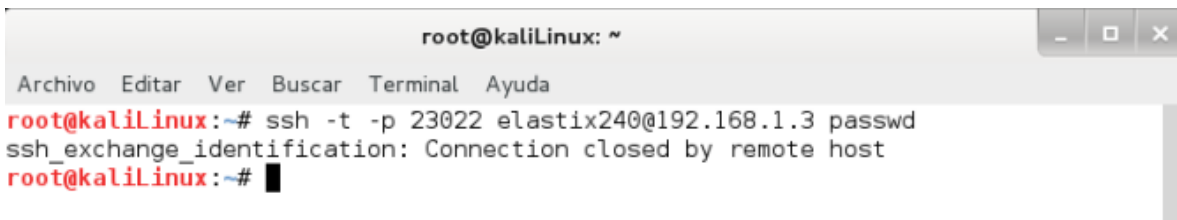
Figura 10.80: Configuración del Archivo "hosts.allow" para SSH



```
1 #
2 # hosts.deny This file describes the names of the hosts which are
3 #             *not* allowed to use the local INET services, as decided
4 #             by the '/usr/sbin/tcpd' server.
5 #
6 #
7 sshd : ALL : deny
```

Figura 10.81: Configuración del Archivo "hosts.deny" para SSH

Luego de configurar el sistema de ACL para SSH se procede a autenticar un usuario del servidor de comunicaciones unificadas. En la Figura 10.82 se muestra el mensaje de error cuando se intenta autenticar un usuario por SSH en el puerto 23022. A pesar de que se posee el usuario y contraseña correctos no se puede establecer la conexión debido a que la dirección IP donde se intenta llevar esta acción no se encuentra dentro de la lista de hosts permitidos. El sistema ACL se configuró para permitir la autenticación en SSH para determinadas IP y para denegar el acceso al resto de direcciones IP.



```
root@kaliLinux: ~
Archivo Editar Ver Buscar Terminal Ayuda
root@kaliLinux:~# ssh -t -p 23022 elastix240@192.168.1.3 passwd
ssh_exchange_identification: Connection closed by remote host
root@kaliLinux:~#
```

Figura 10.82: Conexión al Puerto SSH del Servidor VoIP utilizando el Bash de Kali Linux

10.7.8 Shellshock

Shellshock, también conocido como bashdoor, es el nombre de una familia de fallas de seguridad (6 CVEs) que afectan al Bash (Bourne-Again Shell) de Unix, un componente de software que interpreta órdenes en el sistema. La primera de ellas fue dada a conocer el 24 de septiembre de 2014. El peligro de Shellshock radica en que cualquier hacker podría controlar a distancia cualquier computador o sistema que utilice Bash.

Los CVEs (Common Vulnerabilities and Exposures) pertenecen a una lista de información registrada sobre vulnerabilidades conocidas de seguridad en donde cada referencia tiene un número de identificación único. Seis CVEs separados han sido asignados a Shellshock bajo los números de identificación:

- CVE-2014-6271

- CVE-2014-6277
- CVE-2014-6278
- CVE-2014-7169
- CVE-2014-7186
- CVE-2014-7186

Elastix corre sobre CentOS como sistema operativo y por lo tanto se debe verificar si es vulnerable a la familia de fallas de Shellshock. El servidor de comunicaciones unificadas es Elastix 2.4.0 basado en la distribución Linux CentOS 5.9 para la arquitectura Intel x86. En la Figura 10.83 se verifica con una sencilla prueba que el servidor es vulnerable a Shellshock. Para que el servidor ya no sea vulnerable a esta falla se debe actualizar el Bash con el comando “yum update bash”.

```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[root@elastix240 ~]# env check='Not vulnerable' x='()' { :; }; check='Vulnerable' bash -c 'echo $check'
Vulnerable

```

Figura 10.83: Prueba de Vulnerabilidad de Shellshock

Posterior a la actualización del Bash del servidor VoIP se realiza una sencilla prueba para verificar que el Bash ya no es vulnerable a la familia de fallas de seguridad reportadas. La Figura 10.84 muestra la ejecución del comando de prueba y su respuesta. La respuesta indica que el Bash ya no es vulnerable.

```

root@elastix240:~
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
[root@elastix240 ~]# env check='Not vulnerable' x='()' { :; }; check='Vulnerable' bash -c 'echo $check'
Not vulnerable

```

Figura 10.84: Prueba de la No Vulnerabilidad de Shellshock

10.7.9 SELinux

Security-Enhanced Linux (SELinux) un mecanismo de control de acceso avanzado incorporado en la mayoría de las distribuciones modernas de Linux. Fue desarrollado inicialmente por la Agencia de Seguridad Nacional de Estados Unidos para proteger los sistemas informáticos de la intrusión maliciosa y manipulación. Con el tiempo SELinux fue lanzado al dominio público y varias distribuciones ya lo han incorporado en su código.

SELinux utiliza una serie de reglas conocidas en conjunto como una política “policy” para autorizar o denegar operaciones. La gestión de permisos es completamente distinta a la de los sistemas Unix tradicionales. Los permisos de un proceso dependen de su contexto de seguridad. El contexto está definido por la identidad del usuario que ejecuta el proceso, el rol y el dominio que el usuario tenía en ese momento. Los permisos realmente dependen del dominio pero los roles controlan la transición entre dominios. Por último, las transiciones posibles entre roles dependen de la identidad.

10.7.10 Configuraciones en Elastix

Se debe eliminar la ruta por defecto "Route 9_outside" de las rutas salientes del Elastix debido a que permite realizar llamadas salientes. En la Figura 10.85 se observa la ruta saliente que se debe eliminar y la opción donde permite su eliminación.

Route Settings

Route Name: 9_outside

Route CID: Override Extension

Route Password:

Route Type: Emergency Intra-Company

Music On Hold?: default

Time Group: ---Permanent Route---

Route Position: ---No Change---

Figura 10.85: Ruta Saliente "Route 9_outside" con Opción a Eliminación

Se debe configurar listas de acceso (ACL) para el registro de las extensiones. No se debe aceptar peticiones de autenticación SIP desde cualquier dirección IP. Las listas de acceso en Elastix se configuran usando los campos de "permit" y "deny" dentro de la configuración de cada extensión. La Figura 10.86 muestra la configuración de una extensión, la regla deny=0.0.0.0/0.0.0.0 niega todo el tráfico y la regla permit=192.168.1.5 permite el registro sólo de esa dirección IP. En los campos "permit" y "deny" se pueden especificar segmentos de red. La Figura 10.87 muestra la regla "permit" asociada a un segmento de red.

dial	SIP/200
accountcode	<input type="text"/>
mailbox	200@device
vmexten	<input type="text"/>
deny	0.0.0.0/0.0.0.0
permit	192.168.1.5

Figura 10.86: ACL en Elastix para el Registro de Extensión con Dirección IP Específica

dial	SIP/200
accountcode	<input type="text"/>
mailbox	200@device
vmexten	<input type="text"/>
deny	0.0.0.0/0.0.0.0
permit	192.168.1.0/255.255.255.0

Figura 10.87: ACL en Elastix para el Registro de Extensión perteneciente a un Segmento de Red

Cambiar la contraseña del AMI (Asterisk Manager Interface) por una contraseña segura y limitar las conexiones con los campos “permit” y “deny”. El AMI es un modelo cliente/servidor sobre TCP que permite tener el control de la PBX de Asterisk, realizar llamadas, monitorear canales y colas, revisar el estado del buzón de correo y ejecutar comandos de Asterisk. Dejar el AMI con la configuración por defecto representa un riesgo para la central telefónica, en la Figura 10.88 se muestra el cambio de la configuración que se realiza en el archivo “/etc/asterisk/manager.conf”.

```

2 ; AMI - Asterisk Manager interface
3 ;
4 ; FreePBX needs this to be enabled. Note that if you enable it on a different IP, you need
5 ; to assure that this can't be reached from un-authorized hosts with the ACL settings
6 ; (permit/deny).
7 ; Also, remember to configure non-default port or IP-addresses in amportal.conf.
8 ; The AMI connection is used both by the portal and the operator's panel in FreePBX.
9 ;
10 ; FreePBX assumes an AMI connection to localhost:5038 by default.
11 ;
12 [general]
13 enabled = yes
14 port = 5038
15 bindaddr = 0.0.0.0
16 displayconnects=no ;only effects 1.6+
17
18 [admin]
19 secret = s3cr3t0**790
20 deny=0.0.0.0/0.0.0.0
21 permit=127.0.0.1/255.255.255.0
22 read =
23   system, call, log, verbose, command, agent, user, config, command, dtmf, reporting, cdr, dialplan, origina
24 write =
25   system, call, log, verbose, command, agent, user, config, command, dtmf, reporting, cdr, dialplan, origina

```

Figura 10.88: Configuraciones en “/etc/asterisk/manager.conf”

FreePBX es una GUI open source que controla y administra la PBX de Asterisk. Se encuentra incluida en Elastix. Si realmente no se necesita utilizar se recomienda mantenerlo inactivo. La Figura 10.89 muestra dónde deshabilitar FreePBX y la opción para cambio de contraseña de la base de datos y administración web por medio del GUI de Elastix. Se recomienda cambiar la contraseña por una segura.



Figura 10.89: Opciones Avanzadas del Módulo de Seguridad del GUI de Elastix

Si se cambia la contraseña del AMI, se debe modificar la contraseña almacenada en el archivo “/etc/amportal.conf” por la contraseña actual para el correcto funcionamiento de FreePBX (ver Figura 10.90). Luego de realizar cambios en el archivo “amportal.conf”, se debe ejecutar el comando “amportal restart” para que los cambios tengan efecto.

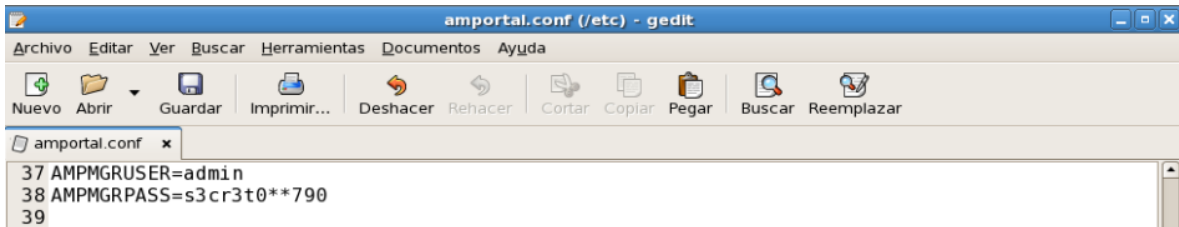


Figura 10.90: Configuraciones en “/etc/amportal.conf”

Se puede cambiar la contraseña de MySQL, AMI, FreePBX e interfaz web de Elastix con un solo comando al utilizar:

```
elastix-admin-passwords --change
```

En el archivo “/etc/asterisk/sip_general_custom.conf”, se debe establecer el valor del campo “allowguest” en “no” para no permitir usuarios no autenticados y “alwaysauthreject” en “yes” para que se rechacen las peticiones de autenticación fallidas utilizando extensiones válidas (ver Figura 10.91).

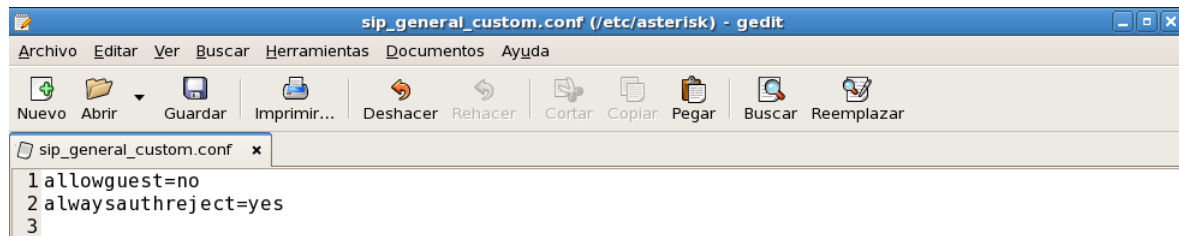


Figura 10.91: Configuraciones en “/etc/asterisk/sip_general_custom.conf”

Luego de realizar modificaciones a los archivos de configuración de Asterisk se debe recargar todos los módulos para que los cambios tengan efecto con el siguiente comando:

```
asterisk -rx "reload"
```

También se debe cambiar el contexto “from-internal” por contextos personalizados. Una extensión con el contexto “from-internal” le permite acceder a todas las rutas salientes definidas en Elastix. Se crean contextos para llamadas locales, nacionales y celulares por el PSTN y el ITSP. FreePBX permite crear fácilmente contextos personalizados. La Figura 10.92 muestra a la derecha el formulario de creación de un nuevo contexto y a la izquierda los contextos creados. La Figura 10.93 muestra fragmentos de la configuración del contexto “PSTN-Locales”. Se configura el plan de discado y se permite la ruta saliente “PSTN-Locales” y se niega el acceso a las demás rutas salientes.



Figura 10.92: Contextos Personalizados en FreePBX

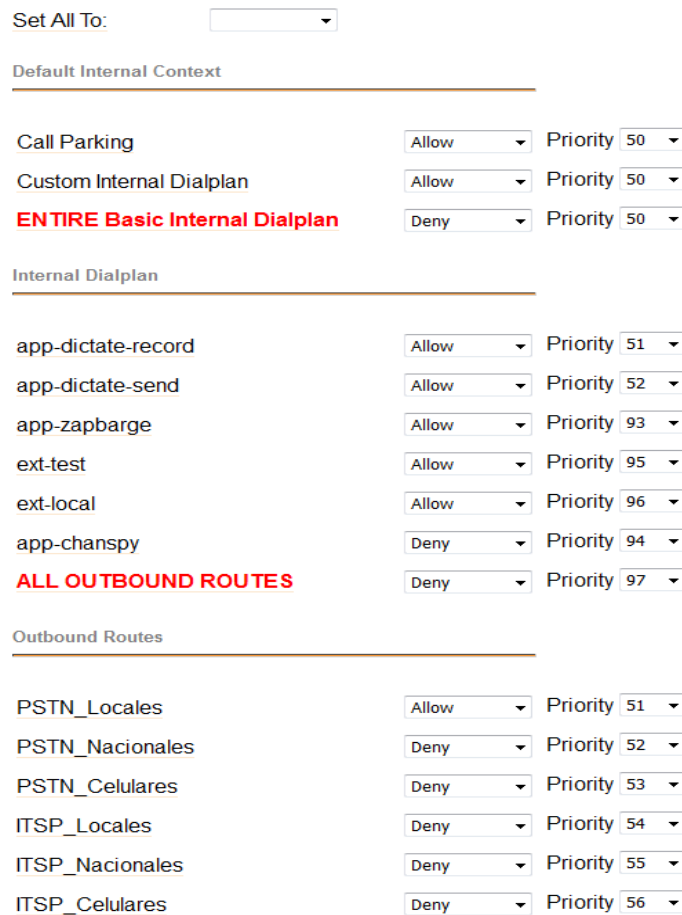


Figura 10.93: Configuración del Contexto Personalizado “PSTN-Locales”

10.7.11 Soluciones para Ataques de DoS y DDoS

El servidor de comunicaciones unificadas con Fail2ban puede detectar el envío masivo de peticiones sin autenticación, como por ejemplo las peticiones inviteflood. Esta acción ayuda en gran parte a contrarrestar el ataque cuando descarta (DROP) los paquetes enviados por el atacante y notifica por correo electrónico al administrador del Elastix cuando una dirección IP es bloqueada. El firewall permite filtrar los paquetes entrantes que no se encuentran autorizados permitiendo añadir una capa de seguridad ante ataques de DoS y DDoS. Sin embargo, Fail2ban y el firewall no solucionan por completo

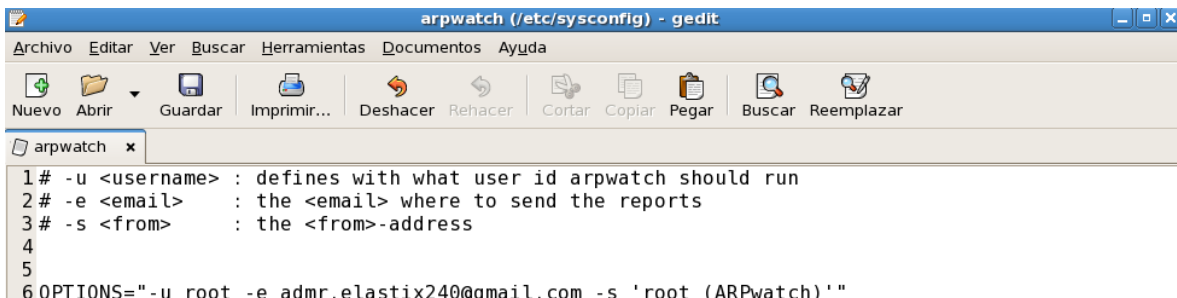
el problema de DoS y DDoS cuando el ataque es extremadamente excesivo, debido a que el servidor es incapaz de procesar tantas peticiones a la vez. El envío masivo de peticiones ocasiona que las extensiones pierdan el registro con el servidor y no puedan realizar llamadas. Los ataques de DoS y DDoS pueden ser ataques internos como externos y afectan de forma similar al servidor. Ante este problema se recomienda utilizar un firewall o router con un módulo de IPS y balanceadores de cargas o preferiblemente adquirir un equipo dedicado a detectar y mitigar ataques de DoS y DDoS, como por ejemplo el FortiDDoS de Fortinet. Existen soluciones que ofrecen servicios en la nube para proteger sistemas de DoS y DDoS, estos sistemas actúan como un proxy que protege y limpia el tráfico hacia el sistema. Lo más importante es que el servidor Elastix no quede expuesto al Internet sin ningún dispositivo que lo proteja.

10.7.12 Soluciones para ARP Spoofing y Ataques Relacionados

El ARP Spoofing intercepta el tráfico VoIP entre el servidor Elastix y una extensión dentro de la red interna haciendo creer a ambas víctimas que se están comunicando directamente. Muchos ataques se encuentran relacionados, como por ejemplo Man-in-the-Middle y Eavesdropping. La principal solución que existe para el ARP Spoofing es utilizar rutas estáticas en la tabla ARP del servidor Elastix. Esto permite invalidar los mensajes ARP provenientes de un atacante debido a que las direcciones IP se asocian a una dirección MAC y esta no cambiará en el tiempo. Esta es una solución simple y por lo general se aplica para asegurar que la puerta de enlace predeterminada sea realmente la de la red y no la de un atacante. Sin embargo, es una estrategia difícil de implementar si se posee una red con una gran cantidad de extensiones o dispositivos finales.

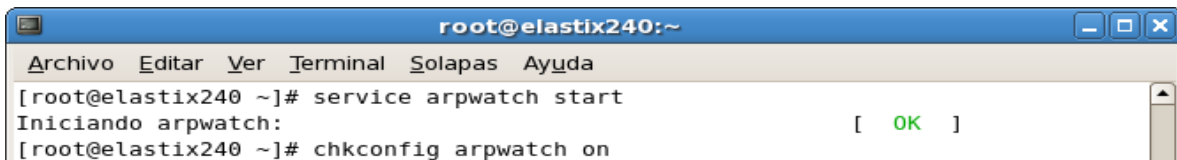
El DHCP snooping y DAI (Dynamic ARP Inspection) son una serie de técnicas que se aplican para asegurar la seguridad existente en una infraestructura DHCP. Estas técnicas mantienen un registro de las direcciones MAC que se encuentran conectadas en cada puerto y detecta inmediatamente si existe una suplantación. Varios fabricantes de equipos de red incorporan esta solución en sus equipos, como por ejemplo Cisco Systems.

Existen herramientas para monitorear el tráfico ARP, como por ejemplo ARPWatch. Esta herramienta se encuentra disponible para el sistema operativo CentOS pudiéndose integrar fácilmente al servidor Elastix para observar cambios sospechosos entre la correspondencia de direcciones IP con direcciones MAC. Para instalar ARPWatch, se utiliza el comando "yum install arpwatc". Para poder utilizar ARPWatch, se debe modificar el archivo de configuración "/etc/sysconfig/arpwatch" como se muestra en la Figura 10.94 para que envíe un reporte por correo electrónico al usuario root cuando una nueva conexión o cambio de conexión se produzca en la tabla ARP. Finalmente se inicia el servicio de ARPWatch y se configura para que inicie siempre en el arranque del servidor (ver Figura 10.95).



```
1 # -u <username> : defines with what user id arpwatc should run
2 # -e <email> : the <email> where to send the reports
3 # -s <from> : the <from>-address
4
5
6 OPTIONS="-u root -e admr.elastix24@gmail.com -s 'root (ARPWatc)'"
```

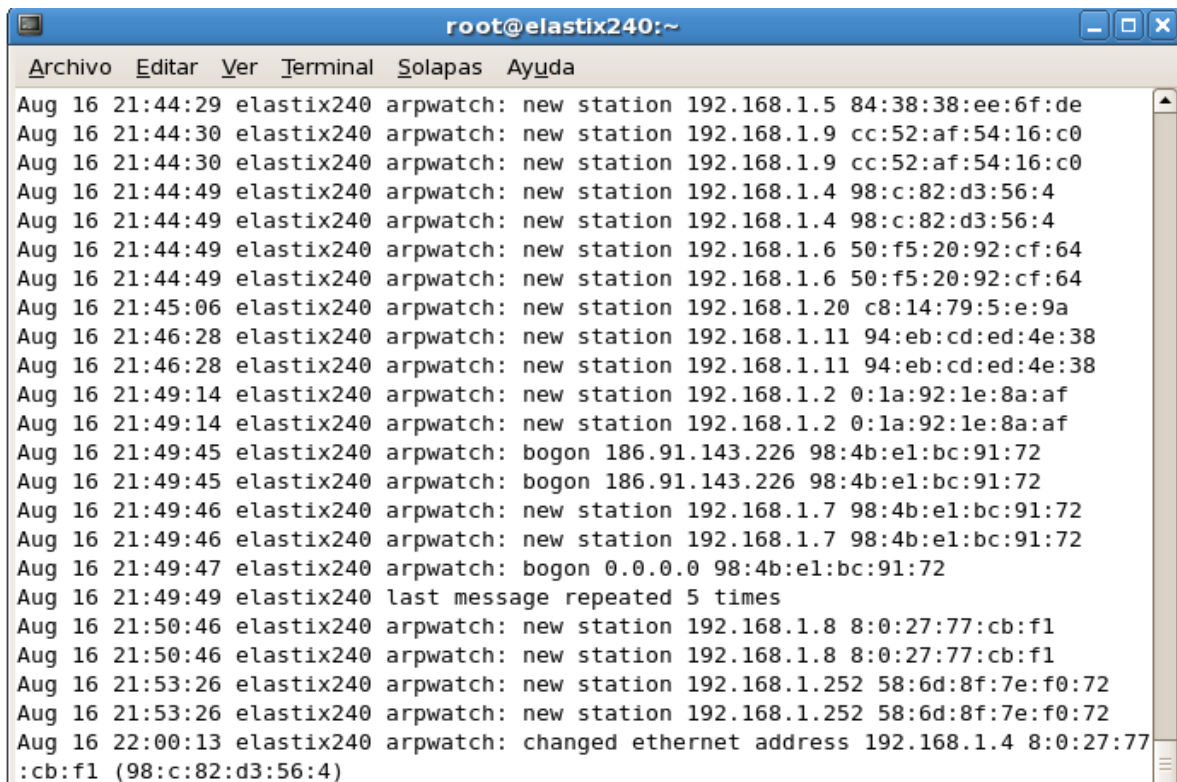
Figura 10.94: Archivo de Configuración de ARPWatch



```
root@elastix240:~  
Archivo Editar Ver Terminal Solapas Ayuda  
[root@elastix240 ~]# service arpwatc start  
Iniciando arpwatc: [ OK ]  
[root@elastix240 ~]# chkconfig arpwatc on
```

Figura 10.95: Iniciación del Servicio de ARPWatch y Configuración de Arranque

Cuando una nueva estación es detectada por ARPWatch se escribe en el archivo de registro “/var/log/messages” (ver Figura 10.96). En este archivo también se escribe cuando una estación cambia o continuamente cambia su dirección MAC. ARPWatch puede enviar estos registros en forma de notificación a una dirección de correo electrónico. En la Figura 10.97 se muestra el envío de correo electrónico cuando una nueva estación es detectada, en la Figura 10.98 se observa un correo electrónico cuando se hace cambio de una dirección MAC. Con el envío de estas notificaciones se tiene un mayor control de las estaciones que se conectan a la red y se pueden prevenir ataques hacia el servidor Elastix.



```
root@elastix240:~  
Archivo Editar Ver Terminal Solapas Ayuda  
Aug 16 21:44:29 elastix240 arpwatc: new station 192.168.1.5 84:38:38:ee:6f:de  
Aug 16 21:44:30 elastix240 arpwatc: new station 192.168.1.9 cc:52:af:54:16:c0  
Aug 16 21:44:30 elastix240 arpwatc: new station 192.168.1.9 cc:52:af:54:16:c0  
Aug 16 21:44:49 elastix240 arpwatc: new station 192.168.1.4 98:c:82:d3:56:4  
Aug 16 21:44:49 elastix240 arpwatc: new station 192.168.1.4 98:c:82:d3:56:4  
Aug 16 21:44:49 elastix240 arpwatc: new station 192.168.1.6 50:f5:20:92:cf:64  
Aug 16 21:44:49 elastix240 arpwatc: new station 192.168.1.6 50:f5:20:92:cf:64  
Aug 16 21:45:06 elastix240 arpwatc: new station 192.168.1.20 c8:14:79:5:e:9a  
Aug 16 21:46:28 elastix240 arpwatc: new station 192.168.1.11 94:eb:cd:ed:4e:38  
Aug 16 21:46:28 elastix240 arpwatc: new station 192.168.1.11 94:eb:cd:ed:4e:38  
Aug 16 21:49:14 elastix240 arpwatc: new station 192.168.1.2 0:1a:92:1e:8a:af  
Aug 16 21:49:14 elastix240 arpwatc: new station 192.168.1.2 0:1a:92:1e:8a:af  
Aug 16 21:49:45 elastix240 arpwatc: bogon 186.91.143.226 98:4b:e1:bc:91:72  
Aug 16 21:49:45 elastix240 arpwatc: bogon 186.91.143.226 98:4b:e1:bc:91:72  
Aug 16 21:49:46 elastix240 arpwatc: new station 192.168.1.7 98:4b:e1:bc:91:72  
Aug 16 21:49:46 elastix240 arpwatc: new station 192.168.1.7 98:4b:e1:bc:91:72  
Aug 16 21:49:47 elastix240 arpwatc: bogon 0.0.0.0 98:4b:e1:bc:91:72  
Aug 16 21:49:49 elastix240 last message repeated 5 times  
Aug 16 21:50:46 elastix240 arpwatc: new station 192.168.1.8 8:0:27:77:cb:f1  
Aug 16 21:50:46 elastix240 arpwatc: new station 192.168.1.8 8:0:27:77:cb:f1  
Aug 16 21:53:26 elastix240 arpwatc: new station 192.168.1.252 58:6d:8f:7e:f0:72  
Aug 16 21:53:26 elastix240 arpwatc: new station 192.168.1.252 58:6d:8f:7e:f0:72  
Aug 16 22:00:13 elastix240 arpwatc: changed ethernet address 192.168.1.4 8:0:27:77  
:cb:f1 (98:c:82:d3:56:4)
```

Figura 10.96: Registros de Nuevas Estaciones y Cambios de Direcciones MAC en las Estaciones

new station Recibidos x 📄 🖨️ 🗑️

ARPwatch 21:44 (hace 19 minutos) ☆
hostname: <unknown> ip address: 192.168.1.20 ethernet address: c8:14:79:5:e:9...

ARPwatch 21:44 (hace 19 minutos) ☆
hostname: <unknown> ip address: 192.168.1.3 ethernet address: 8:0:27:de:df:4f...

ARPwatch 21:44 (hace 19 minutos) ☆
timestamp: Sunday, August 16, 2015 21:44:29 -0430

ARPwatch 21:44 (hace 19 minutos) ☆
hostname: <unknown> ip address: 192.168.1.9 ethernet address: cc:52:af:54:16:...

ARPwatch 21:44 (hace 19 minutos) ☆
hostname: <unknown> ip address: 192.168.1.5 ethernet address: 84:38:38:ee:6f:de

ARPwatch 21:44 (hace 18 minutos) ☆
hostname: <unknown> ip address: 192.168.1.9 ethernet address: cc:52:af:54:16:...

ARPwatch 21:44 (hace 18 minutos) ☆
hostname: <unknown> ip address: 192.168.1.5 ethernet address: 84:38:38:ee:6f:...

ARPwatch <elastix240@gmail.com> 21:44 (hace 18 minutos) ☆ ↩️ ▾
para mí ▾

🌐 inglés ▾ > español ▾ [Traducir mensaje](#) [Desactivar para: inglés x](#)

Figura 10.97: Correo Electrónico Enviado por ARPWatch como Resultado del Descubrimiento de una Nueva Estación

changed ethernet address Recibidos x 📄 🖨️ 🗑️

ARPwatch <elastix240@gmail.com> 22:00 (hace 2 minutos) ☆ ↩️ ▾
para mí ▾

🌐 inglés ▾ > español ▾ [Traducir mensaje](#) [Desactivar para: inglés x](#)

hostname: <unknown>
ip address: 192.168.1.4
ethernet address: 8:0:27:77:cb:f1
ethernet vendor: CADMUS COMPUTER SYSTEMS
old ethernet address: 98:c:82:d3:56:4
old ethernet vendor: <unknown>
timestamp: Sunday, August 16, 2015 22:00:13 -0430
previous timestamp: Sunday, August 16, 2015 22:00:13 -0430
delta: 0 seconds

Figura 10.98: Correo Electrónico Enviado por ARPWatch como Resultado del Cambio de Dirección MAC en una Estación

10.7.13 OpenVPN

OpenVPN es un software de código abierto bajo licencia GPL que implementa una VPN (Virtual Private Network) para crear conexiones punto-a-punto seguras y que facilita el acceso remoto. Tiene dos modos de autenticación: (1) Static Key y (2) TLS. El modo Static Key usa una llave pre-compartida estática y el modo TLS utiliza SSL/TLS con certificados para la autenticación y el intercambio de llaves. Puede utilizarse para todo el tráfico de Internet, incluyendo tráfico web, correo electrónico, mensajería instantánea y VoIP. En VoIP se utiliza para encriptar y asegurar las conversaciones que pasan a través de Internet con las extensiones remotas. Se configura el servidor VPN en este caso OpenVPN en el servidor Elastix y se instala un cliente VPN en una de las extensiones remotas, como por ejemplo un teléfono IP o softphone. De esta forma OpenVPN ayuda a añadir una capa de seguridad contra los ataques de “Man-in-the-Middle”.

10.7.14 IDS/IPS

Un IDS (Intrusion Detection System) es un dispositivo o software para la detección de accesos no autorizados a un computador o una red, se encarga de monitorear los eventos que ocurren en un sistema informático buscando intentos de intrusión. Un IPS (Intrusion Prevention System) es un dispositivo o software que ejerce el control de acceso a una red para proteger los sistemas informáticos de ataques. Los IPSs presentan una ventaja respecto a los firewalls tradicionales por tomar decisiones de control de acceso basado en el contenido del tráfico en lugar de direcciones IP, puertos y tipo de protocolo. Un IPS puede prevenir ataques antes de que ocurran. Es importante tener el IDS/IPS fuera del servidor Elastix, es decir, los paquetes deben ser interceptados por un IDS/IPS antes de llegar al servidor y mejorando entonces su protección. Para ofrecer una robustez en la seguridad del servidor mediante el uso de un IDS/IPS, es necesario que este dispositivo se mantenga al día con las actualizaciones para garantizar su buen funcionamiento y prevenir futuros ataques.

10.7.15 Buenas Prácticas y Recomendaciones

A continuación se presentan algunas prácticas y recomendaciones para implementaciones VoIP que ayudarán a reducir los problemas de seguridad que se pueden presentar en estos sistemas:

- Tener una política de acceso físico al servidor adecuada.
- Mantener actualizado el sistema.
- Evitar utilizar puertos estándares.
- Utilizar contraseñas seguras para las entidades SIP. Usar una mezcla de letras minúsculas, mayúsculas, números y caracteres especiales con al menos 12 caracteres de longitud.
- Utilizar nombres de usuarios SIP diferentes a sus extensiones.
- No dejar los usuarios y claves por defecto.
- Cambiar contraseñas de forma periódica.
- No habilitar llamadas internacionales si no se utiliza.
- Denegar peticiones a los puertos 5060/4569 UDP desde el exterior si el sistema no posee usuarios SIP/IAX externos.
- Revisar periódicamente los registros del sistema.
- No permitir llamadas sin autenticar.

- Desactivar servicios que no son utilizados.
- Usar un firewall para filtrar las solicitudes entrantes.
- Instalar programas de detección de intrusos (IDS).
- Usar redes privadas virtuales (VPNs).
- Tener un control de la integridad de los directorios, archivos y ejecutables del sistema.
- Tener listas de acceso (ACL) para los registros de extensiones.
- Hacer DROP a paquetes procedentes de direcciones IP que no están dentro de la red de confianza.
- No instalar productos adicionales a menos que sea necesario.
- Utilizar VLANs que separen el tráfico de voz y datos.

11. Conclusiones y Trabajos Futuros

VoIP es otro ejemplo de cómo Internet está cambiando las comunicaciones mediante la reducción de los costos y la simplificación de infraestructuras. Sin embargo proteger las redes de VoIP no es una tarea fácil pero es esencial que VoIP se asiente sobre una infraestructura de red segura. Actualmente las comunicaciones son cada día más importantes y los servidores de comunicaciones, como Elastix, se encuentran expuestos a posibles ataques.

Un avance constante en la telefonía y en la VoIP hace que cada día crezcan el número de instalaciones de servidores de comunicaciones y como consecuencia aumenten las técnicas de ataques a estos servidores. Las herramientas actuales hacen que cualquier persona con un mínimo de conocimiento pueda realizar grandes daños en estos.

Muchas empresas buscan reducir costos utilizando los avances en comunicaciones pero con frecuencia no toman las precauciones de seguridad necesarias ya sea por desconocimiento o simplemente por optar por una experticia inadecuada. Esto se traduce en errores que generan grandes costos en comunicaciones para la empresa.

Es evidente que junto con los avances tecnológicos, las vulnerabilidades y ataques seguirán apareciendo. Sin embargo, los mecanismos esenciales de protección también se desarrollarán. El desafío siempre estará en el orden del conocimiento, el análisis y la aplicación de tal modo que se pueda determinar una solución para cada necesidad.

En este trabajo, se utilizaron herramientas de seguridad para realizar ataques a los diferentes escenarios propuestos y con el estudio de seguridad en redes VoIP se proponen soluciones para su mitigación. La idea principal de este TEG fue dar un conjunto de lineamientos o medidas que ayuden a proteger o mantener un esquema de seguridad robusto en las diferentes arquitecturas propuestas que implantan VoIP, sin olvidar que no hay un sistema seguro. Adicionalmente, los lineamientos buscan resolver los problemas de seguridad encontrados para un ahorro de recursos y servicios.

VoIP hereda los problemas de seguridad de los protocolos o sistemas en los cuales se apoya y es por ello que su seguridad no se debe limitar a esta tecnología, se debe asegurar el sistema operativo donde se implementa, los dispositivos e incluso la misma red. Con los lineamientos o medidas descritas, se pretende alcanzar un nivel de seguridad mayor y un correcto funcionamiento de las implementaciones VoIP en los escenarios propuestos. No son medidas definitivas debido a que los ataques y soluciones evolucionan con el paso del tiempo. En este trabajo, no solo se dieron medidas de seguridad para la tecnología VoIP, sino que también se presentaron soluciones para el sistema operativo, firewall, dispositivos finales y algunas recomendaciones adicionales para cumplir los 5 objetivos básicos de la seguridad: (1) confidencialidad, (2) integridad, (3) disponibilidad, (4) autenticación y (5) no repudio.

Como trabajos futuros, se propone implantar nuevas arquitecturas con soporte para VoIP debido a que la tecnología evoluciona cada día. También es posible utilizar la PBX de Asterisk junto con el protocolo IAX2 para el soporte VoIP, utilizar otras herramientas para generar ataques, por ejemplo, BackBox y realizar otros tipos de ataques.

Referencias Bibliográficas

- [1] J. Gómez and F. Gil. VoIP y Asterisk Redescubriendo la Telefonía. Alfaomega Grupo Editor. Primera Edición. Enero 2009.
- [2] R. Gutiérrez. Seguridad en VoIP: Ataques, Amenazas y Riesgos. Universidad de Valencia, España. Octubre 2008.
- [3] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261. Junio 2002.
- [4] V. Mendillo. Voz sobre IP y Telefonía Mediante SIP. Febrero 2012.
- [5] W. Stallings. Comunicaciones y Redes de Computadoras. Pearson. Séptima Edición. 2008.
- [6] M. Flores. Análisis de los Protocolos de Tiempo Real en Ethernet: RTP, RTCP y RTSP. Universidad de Málaga. 2006.
- [7] J. Postel. Internet Protocol. RFC 791. Septiembre 1981.
- [8] R. Stewart. Stream Control Transmission Protocol. RFC 4960. Septiembre 2007.
- [9] L. Ong and J. Yoakum. An Introduction to the Stream Control Transmission Protocol (SCTP). RFC 3286. Mayo 2002.
- [10] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550. Julio 2003.
- [11] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman. The Secure Real-time Transport Protocol (SRTP). RFC 3711. Marzo 2004.
- [12] H. Schulzrinne, A. Rao, and R. Lanphier. Real Time Streaming Protocol (RTSP). RFC 2326. Abril 1998.
- [13] M. Handley, V. Jacobson, and C. Perkins. SDP: Session Description Protocol. RFC 4566. Julio 2006.
- [14] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP). RFC 2205. Septiembre 1997.
- [15] M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg. SIP: Session Initiation Protocol. RFC 2543. Marzo 1999.
- [16] P. Resnick. Internet Message Format. RFC 2822. Abril 2001.
- [17] E. Landívar. Comunicaciones Unificadas con Elastix Volumen 1. 2008.
- [18] E. Landívar. Comunicaciones Unificadas con Elastix Volumen 2. 2009.
- [19] B. Caswell and M. Poor. Snort 2.1 Intrusion Detection. Syngress. Segunda Edición. 2004.
- [20] R. Kuhn, T. Walsh and S. Fries. Security Considerations for Voice Over IP Systems. National Institute of Standards and Technology. Enero 2005.
- [21] J. Oliva. Seguridad en Implementaciones de Voz Sobre IP. Abril 2014.
- [22] R. Martin. Seguridad en Servidores CentOS con Elastix. 2010.
- [23] S. Cornu. Asegurando Elastix. Primera Edición. Agosto 2011.
- [24] R. Del Valle and M. Herrera. Seguridad del Protocolo SIP en la VoIP. 2009.

- [25] J. Arkko, V. Torvinen, G. Camarillo, A. Niemi, and T. Haukka. Security Mechanism Agreement for the Session Initiation Protocol (SIP). RFC 3329. Enero 2003.
- [26] National Institute of Standards and Technology. Advanced Encryption Standard (AES). FIPS 197. Noviembre 2001.
- [27] National Institute of Standards and Technology. Data Encryption Standard (DES). FIPS 46. Octubre 1999.
- [28] National Institute of Standards and Technology. Triple Data Encryption Standard (TDES). NIST Special Publication 800-67. Enero 2012.
- [29] EC-Council. Penetration Testing: Procedures and Methodologies. 2011.
- [30] R. Beggs. Mastering Kali Linux for Advanced Penetration Testing. Junio 2014.