Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación

SEGURIDAD EN VoIP :
APLICACIÓN DE SEÑUELOS

TRABAJO FIN DE MÁSTER

Elena Krasheninnikova

2013

Universidad Politécnica de Madrid
Escuela Técnica Superior de Ingenieros de Telecomunicación

Máster Universitario en
Ingeniería de Redes y Servicios Telemáticos

TRABAJO FIN DE MÁSTER

SEGURIDAD EN VoIP:
APLICACIÓN DE SEÑUELOS

Autor
Elena Krasheninnikova

Director
Enrique Vázquez Gallo

Departamento de Ingeniería de Sistemas Telemáticos

2013

Resumen

El objetivo del presente trabajo es llevar a cabo un estudio de los aspectos de seguridad en las redes de VoIP con el enfoque de señuelos o honeypots.

Lo que se pretende es estudiar varios honeypots existentes a día de hoy que puedan ser utilizados en las redes VoIP y conocer sus ventajas e inconvenientes.

Para comprender en todo su alcance cada uno de los aspectos a estudiar, se necesita previamente revisar el panorama general relativo a la seguridad en las redes de voz sobre IP.

Antes de ver el panorama de la seguridad, hace falta conocer la tecnología y su funcionamiento. De esta manera primero se describirá la arquitectura de la red VoIP basada en el protocolo SIP y los protocolos utilizados en esta tecnología a día de hoy.

A continuación, entrando ya en el área de la seguridad, hace falta detallar y clasificar los ataques y las vulnerabilidades en las redes VoIP. Estas son múltiples y varían en diferentes niveles de la tecnología.

El servicio de telefonía, independientemente de la tecnología con que se ofrezca, debe cumplir requisitos de seguridad básicos para los usuarios tales como confidencialidad, autenticación, integridad y disponibilidad. Aunque los requisitos son los mismos tanto si el servicio se presta por conmutación de circuitos como de paquetes, la naturaleza de cada tecnología afecta a la forma de cumplirlos.

Se recogerán también varios protocolos de seguridad aplicables en redes de voz sobre IP basadas en el protocolo SIP. Se trata de protocolos generales que implementan mecanismos de gestión de claves, cifrado, etc. y que pueden utilizarse tanto en voz sobre IP como en otros escenarios.

Luego se estudiarán los mecanismos de detección de intrusiones con el enfoque de honeypots. Como se ha mencionado ya anteriormente el objetivo es estudiar varios honeypots que se utilizan en las redes VoIP, ver su usabilidad y mencionar sus pros y contras.

Abstract

The aim of this work is to study the safety aspects of VoIP networks with a focus on honeypots.

Here several honeypots existing today that can be used in VoIP networks are studied and their advantages and disadvantages are presented.

To understand the full scope of each of the aspects that are to be studied, first it is necessary to explore the general concept of security in VoIP networks.

Before studying the security concept, knowing the technology and its operating principals is essential. Thus VoIP network architecture based on SIP protocol and the protocols used in this technology today are described first.

Then, touching upon the security issues, it is necessary to detail and classify the attacks and vulnerabilities in VoIP networks. These are numerous and vary at different levels of technology.

Telephone service, regardless of the technology offered, must meet basic security requirements for users such as confidentiality, authentication, integrity and availability. Although the requirements are the same whether the service is provided as a circuit-switched or packet-switched, the nature of each technology affects meeting these requirements.

Various security protocols applicable in VoIP networks based on SIP protocol are also presented. These are general protocols that implement key management mechanisms, encryption, etc. and can be used both in voice over IP and in other scenarios.

Then the mechanisms of intrusion detection are studied using honeypots approach. As already mentioned above, the objective is to study several honeypots used in VoIP networks, analyze their usability and highlight their advantages and disadvantages.

Índice general

Índice de figuras

Índice de tablas

Siglas

AAA	Authentication, Authorization and Accounting
AES	Advanced Encryption Standard
AIB	Authenticated Identity Body
AIX	Advanced Interactive eXecutive
ARP	Address Resolution Protocol
COPS	Common Open Policy Service
DDoS	Distributed Denial of Service
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Denial of Service
DTLS	Datagram Transport Layer Security
DTMF	Dual-Tone Multifrequency
EPS	Encapsulating Security Payload
ETSI	European Telecommunications Standards Institute
ETSI	European Telecommunications Standers Institute
GSM	Global System for Mobile Communications
HTTP	Hyper Text Transfer Protocol
ICMP	Internet Control Message Protocol
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IMS	IP Multimedia Subsystem

IP	Internet Protocol
IPS	Intrusion Prevention System
IPsec	Internet Protocol Security
ISUP	Integrated Services User Part
ITU-T	International Telecommunication Union - Telecommunication sector
MAC	Media Access Control
MG	Media Gateway
MGC	Media Gateway Controller
MIKEY	Multimedia Internet Keying
MMUSIC	Multiparty Multimedia Session Control
NGN	Next Generation Network
OSA	Open Service Access
RADIUS	Remote Authentication Dial In User Service
RFC	Request For Comments
RSA	Remote Supervisor Adapter
RTCP	Real Time Control Protocol
RTCP	Real Time Control Protocol
RTP	Real-Time Transport Protocol
S/MIME	Secure/Multipurpose Internet Mail Extensions
SCCP	Skinny Call Control Protocol
SCTP	Stream Control Transmission Protocol
SCTP	Stream Control Transmission Protocol
SDP	Session Description Protocol
SG	Signaling Gateway
SHA	Secure Hash Algorithm

SIGTRAN	Signaling Transport
SIP	Session Initiation Protocol
SNMP	Simple Network Management Protocol
SPIT	Spam over Internet Telephony
SQL	Structured Query Language
SRTP	Secure Real Time Protocol
SS7	Sistema de Señalización nº 7
SSL	Secure Sockets Layer
TCP	Transmission Control Protocol
TFTP	Trivial File Transfer Protocol
TISPAN	Telecommunications and Internet converged Services and Protocols for Advanced Networking
TLS	Transport Layer Security
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
VoIP	Voice over IP
XCAP	Extensible mark-up language (XML) Configuration Access Protocol
XML	Extensible Mark-up Language

1 Introducción

En los últimos años las empresas de telecomunicación han presenciado grandes evoluciones en cuanto a tecnologías, los cuales han impresionado tanto a las organizaciones como a las personas, de manera que se han percibido cambios rápidos en Internet y las aplicaciones basadas en IP. Se ha notado que los servicios y tráficos de voz son aplicaciones que mayor ventaja tomarán de IP.

Estas herramientas son generalmente referidas como Voz sobre IP (VoIP), la cual conjuga dos mundos históricamente separados: la transmisión de voz y la de datos. Se trata de transportar la señal previamente convertida a datos, entre dos puntos distantes. Esto posibilitaría utilizar las redes de datos para efectuar las llamadas telefónicas, además trasmitir cualquier tipo de información. VoIP suministra beneficios a los portadores y clientes quienes dependen en IP, los cuales son: ahorro de costos, estándares, redes integradas para voz y datos. Luego se implementaron algunos protocolos como SIP, H.323 entre otros, con el fin de mejorar los servicios de esta técnica de comunicación.

Hoy en día, sin embargo, VoIP como cualquier otra tecnología de telecomunicación se enfrenta a muchas amenazas de seguridad. A medida que crece su popularidad aumentan las preocupaciones por la seguridad de las comunicaciones. VoIP es una tecnología que ha de apoyarse necesariamente muchas otras capas y protocolos ya existentes de las redes de datos. Por eso en cierto modo VoIP va a heredar ciertos problemas de las capas y protocolos ya existentes, siendo algunas de las amenazas más importantes de VoIP problemas clásicos de seguridad que afectan al mundo de las redes de datos. Por supuesto, existen también multitud de ataques específicos de VoIP.

Existen numerosos métodos propuestos para detectar ataques contra ordenadores y redes, que pueden implementarse en sistemas de detección de intrusiones (IDS) y sistemas de prevención de intrusiones (IPS). El IPS puede bloquear el tráfico por sí mismo o actuando sobre un cortafuegos para cambiar su configuración. Un señuelo o honeypot puede verse como un mecanismo para engañar a los atacantes y estudiar su comportamiento, o también como un mecanismo de respuesta que se despliega para desviar determinados ataques.

El objetivo del presente trabajo es llevar a cabo un completo estudio de los aspectos de seguridad en las redes de VoIP con el enfoque de honeypots a esta tecnología. Primero se describirá la arquitectura de la red basada en el protocolo SIP y se

presentarán otros protocolos utilizados. A continuación se detallarán los ataques y las vulnerabilidades en las redes VoIP. En la siguiente sección se presentarán los requisitos y los protocolos de seguridad en VoIP. Luego se estudiarán los mecanismos de detección de intrusiones con el enfoque de honeypots. El objetivo es estudiar varios honeypots existentes al día de hoy que puedan ser utilizados en las redes VoIP y conocer sus ventajas e inconvenientes.

2 Arquitectura de la red VoIP basada en SIP

2.1 Protocolo de Inicio de Sesión (SIP)

El protocolo SIP (Session Initiation Protocol) fue creado por el IETF y se trata de un protocolo de nivel de aplicación que permite establecer, finalizar y gestionar sesiones multimedia. Las sesiones multimedia incluyen telefonía a través de Internet, conferencias y otras aplicaciones similares que involucren multimedia, tanto audio como vídeo y/o datos. Entre las características principales se deben destacar que el protocolo SIP soporta conferencia múltiple, es independiente del protocolo de transporte, y que es un protocolo de tipo texto. Además SIP soporta numerosos servicios, tales como la presencia de mensajería instantánea, servicios complementarios de telefonía, etc. [1].

Se debe destacar que el protocolo SIP es sólo un protocolo de señalización. Una vez la sesión ha sido establecida, los participantes intercambiarán sus datos de audio/vídeo a través de otro protocolo como, por ejemplo, el protocolo Real-Time Transport Protocol (RTP). Por tanto es habitual que el protocolo SIP actúe en colaboración con otros protocolos. Además se debe de puntualizar que SIP no asegura calidad de servicios, ya que no realiza ningún tipo de reserva del medio de transmisión; se trata de un protocolo de control de llamada y no de control del medio.

Al día de hoy el protocolo SIP, junto con los otros protocolos relacionados con SIP, constituyen la arquitectura dominante del servicio VoIP.

Las funciones básicas de este protocolo son:

- Determinar la ubicación de los usuarios.
- Determinar la ubicación de servidores SIP para facilitar la búsqueda de otro usuario.
- Establecer una sesión de datos mediante intercambio de mensajes de tipo oferta/respuesta
- Modificar una sesión existente usando intercambio de mensajes de tipo oferta/respuesta
- Expresar las capacidades y características de los agentes de usuario (los usuarios finales)

• Averiguar el estado, capacidades y disponibilidad de otro usuario agente.

• Peticiones futuras de actualizaciones sobre el estado y capacidad de otro usuario agente.

• Intercambio de información de señalización de una llamada

• Intercambio de mensajes cortos con otros UA

Como ya se ha dicho, SIP es un protocolo para establecer sesiones multimedia entre dos o más participantes. Las arquitecturas implicadas en un sistema de VoIP basado en SIP se describen brevemente en 2.2. El protocolo SIP se encarga precisamente de definir la forma en que se realizan las transacciones entre elementos para establecer, modificar o terminar una sesión entre los agentes de usuario. Cada una de estas transacciones para el intercambio de mensajes entre elementos de la arquitectura VoIP está formada por una petición seguida por una o varias respuestas.

2.1.1 Mensajes SIP

Los mensajes SIP pueden ser tanto de tipo petición, dirigidos de quién inicia la llamada (llamante) al que se esta llamando (llamado), como de tipo respuesta, dirigidos en este caso del llamado al llamante. Estos últimos mensajes se denominan mensajes de estado. Cada mensaje, independientemente de si se trata de respuesta o petición, contiene una línea de comienzo y a continuación una o más cabeceras opcionales seguidas por el cuerpo del mensaje.

```
message = request-line | status-line (start-line) *message-header CRLF [message-body]
```

En la primera línea "start-line" se observa el campo "request-line", si se trata de una petición aparecerá este campo, mientras que si el mensaje es de respuesta se tendrá "response-line" que indicará el éxito o el fracaso para la petición recibida.

La parte de las cabeceras de los mensajes (message-header) proveerá información adicional mediante diferentes campos. Por ejemplo el campo "Retry-after" indica cuando una petición debería ser realizada de nuevo. Otro ejemplo puede ser el campo "Subject" que indicará el asunto de la llamada, con lo que el cliente al que se le llama podría aceptar o rechazar la llamada dependiendo de dicha cabecera. Los campos que pueden aparecer en los mensajes y su descripción se explican de forma detallada en el documento [3].

El cuerpo del mensaje describirá el tipo de sesión que va a ser establecida, incluyendo información acerca de los datos intercambiados. Además, en el cuerpo del mensaje, también se puede comunicar la estructura mediante la que se desea realizar una llamada de voz, es decir, la codificación específica que quiere ser utilizada. Se debe de tener en cuenta que SIP no especifica la estructura o contenido del cuerpo del mensaje, de esto se encargarán otros protocolos como por ejemplo el SDP (2.3.1 Protocolo de Descripción de Sesiones (SDP)). El cuerpo del mensaje será examinado solamente en los dos extremos finales, SIP lo lleva de una parte a otra sin importarle lo que haya dentro.

Petición SIP

Siguiendo con la estructura anterior presentada de un mensaje SIP, el mensaje de petición vendrá definido en el esquema presentado por la línea "request-line". En esta línea son definidos los siguientes campos:

• Method: Identificará el método de la petición específica, por ejemplo INVITE, ACK, etc.

• EQUEST-URI: La dirección de la entidad que envía la petición

• Versión SIP: La versión SIP que se utiliza, por ejemplo: SIP/2.0

De esta forma se podría definir el siguiente esquema:

```
request-line = method SP Request-URI SP SIP-Version CRLF
```

En el esquema anterior se debe de tener en cuenta que SP representa un espacio en blanco. A continuación se presenta un ejemplo real de la línea de request:

```
request-line = INVITE sip:500@ekiga.net SIP/2.0
```

En el ejemplo se tiene como método "INVITE", como dirección URI "sip:500@ekiga.net" y el último campo, la versión es "SIP/2.0".

SIP define seis tipos básicos de mensajes de peticiones, llamados métodos:

• INVITE. Inicia una sesión incluyendo información acerca del tipo de datos a ser intercambiados. También ofrece capacidad para iniciar una llamada de multiconferencia.

• ACK. Se utiliza para confirmar que la respuesta final del extremo al que se le llama ha sido recibida.

• OPTIONS. Pregunta al servidor por sus capacidades. Es utilizado, por ejemplo, para saber qué tipo de datos soporta un agente o para determinar cómo debe responder el agente de usuario si se le envía una petición INVITE.

• BYE. Este método indica el final de una sesión y puede ser usado por cualquiera de los extremos.

• CANCEL. Se usa para terminar una petición pendiente. Por ejemplo, un uso que se le da es para el caso de que se mande una petición INVITE pero no se reciba la respuesta en un tiempo dado.

• REGISTER. Usado para registrarse y logarse en un servidor SIP. Un cliente se puede registrar con múltiples servidores. En el caso de poseer varios registros, la llamada puede ser enviada a todos los registros de destino del usuario. Con este procedimiento se puede activar el servicio "one-number", en el cual un usuario publica un sólo número de teléfono, pero cuando se le llama, le suena el teléfono de la oficina, el de casa y el teléfono conectado al wireless, por ejemplo.

Los métodos anteriores son los básicos para el funcionamiento de SIP. Sin embargo, hay otros mensajes de peticiones que buscan añadir funcionalidades extras al protocolo y que son especificados en otros RFCs o en borradores. Por ejemplo, el método SIP INFO que está especificado en el RFC 6086 [3]. En el estándar oficial del protocolo SIP fue propuesto en 1999, este método fue propuesto en Octubre de 2000 como una extensión. El método SIP INFO es un medio para transferir información durante las sesiones en curso, como por ejemplo:

• Transferencia de dígitos Dual-Tone Multifrequency (DTMF).

• Transferencia de información de balance de cuenta.

• Transferencia de la señal de información de una *midcall* generada en otra red y pasada a la red IP a través de una *gateway*.

Respuesta SIP (SIP Response)

Todas las peticiones, a excepción ACK, requieren una respuesta. Esta respuesta puede ser inicialmente una respuesta provisional, pero finalmente es necesaria la confirmación mediante una respuesta definitiva.

La primera línea de estos mensajes es la de estado. Esta línea contiene el código de estado, el cual es un número de tres dígitos que indica los resultados de la solicitud previa. Este resultado será descrito por una frase contenida en esta primera línea. Cada código de respuesta tiene una frase por defecto, pero a menudo esa frase es modificada para proveer más detalles sobre el resultado de una petición. El cliente *software*

interpretará el código de la respuesta y actuará según ésta. La frase asociada al código de respuesta tendrá sentido de cara al usuario con el fin de ayudarle a entender la respuesta. La sintaxis de esta línea es como sigue:

```
status-line = SIP-version SP status-code SP reason-phrase CRLF
```

A continuación se presenta un ejemplo real de la línea de status:

```
Status-Line: SIP/2.0 200 OK
```

El código de estado (RFC 3261,[4]) toma valores entre 100 y 699. El primer dígito indica la clase de la respuesta. Además todos los códigos entre X00 y X99 pertenecen a la misma clase. Las clases permitidas por la versión 2.0 de SIP son:

• 1XX: Provisional. Petición recibida, se continua con el proceso de petición (por ejemplo, 181 indica que la llamada está siendo transmitida).

• 2XX: Éxito. La acción se ha recibido exitosamente, se ha entendido y aceptado. Por ejemplo, el código 200 que la petición ha sido entendida y transmitida. En el caso de un INVITE la respuesta 200 es usada para indicar que se ha aceptado la llamada.

• 3XX: Redirección. Por ejemplo, el código 302 indicará que la llamada a realizar no está disponible en la dirección usada en la petición y que la petición debería de ser redirigida a la nueva dirección incluida con la respuesta.

• 4XX: Error de cliente. La petición tiene una mala sintaxis o no cumple con los requisitos de este servidor. Por ejemplo, 401 indica que el cliente no está autorizado a realizar la petición.

• 5XX: Error de servidor. Por ejemplo, indica que el servidor no soporta la versión de SIP especificada en la petición.

• 6XX: Fallo global. La petición no pudo ser completada por ningún servidor. Por ejemplo, 604 indica que el usuario al que se le llama no existe en ningún sitio.

Todas las respuestas, excepto para la clase 1XX, son consideradas finales y deberían ser reconocidas con un mensaje de tipo ACK por el cliente que inició la llamada mediante un INVITE. Las respuestas de tipo 1XX son provisionales y no necesitan ser reconocidas (no tienen por qué ser contestadas con un mensaje ACK).

2.1.2 SIP y URIs

Las direcciones en SIP son conocidas como SIP URIs (Uniform Resource Indicators). Las URIs son un tipo general de dirección usadas en Internet. Contiene información sobre el recurso o el servidor y provee información de cómo contactar con el servidor. Como las sesiones SIP pueden incluir tanto datos multimedia como voz y además pueden interactuar con redes de circuitos conmutados, SIP permite el uso de una porción de la dirección para un número de teléfono. Por ejemplo, se podría direccionar como: sip:3344556789@telco.net. Para dar más información se puede proveer a la URI con otros suplementos como por ejemplo, sip:3344556789@telco.net; user=phone. De esta forma se indicará que se debe de realizar una llamada a un número de teléfono. SIP y SIPS siguen los esquemas del RFC 3986 [5].

2.1.3 Establecimiento de la llamada SIP

A continuación se muestra un ejemplo sencillo del establecimiento de una llamada entre dos usuarios finales (ver Figura 1: Ejemplo del establecimiento de la llamada SIP.). En el ejemplo se establece la llamada entre los dos usuarios directamente, sin servidores de por medio. En la práctica esta situación no será habitual, ya que entre los usuarios finales habrá diferentes elementos de la arquitectura VoIP como proxies, servidores registrar, etc. (ver 2.2 Arquitectura de la red VoIP basada en SIP). No obstante, este ejemplo es válido para ilustrar el funcionamiento del protocolo. Para el establecimiento de la llamada, suponiendo que se realiza directamente, se seguirían los siguientes pasos:

1. INVITE. El extremo que quiere iniciar la conversación envía el mensaje de petición INVITE al otro extremo.

2. RING. Informa al cliente (usuario que ha iniciado la llamada) que la llamada está teniendo lugar (está a la cola en el servidor). Es el equivalente a cuando realizamos una llamada y oímos un tono que nos indica que el teléfono donde llamamos está sonando. En el destino se originará una alerta.

3. OK. El destino acepta la llamada generando con ello un mensaje OK que será enviado al origen. Equiparándolo nuevamente con una llamada telefónica convencional este mensaje equivaldría al descolgar del auricular de la persona a la que se está realizando la llamada.

4. ACK. El origen envía un mensaje de confirmación de que sabe que el destino ha aceptado la llamada.

5. Sesión VoIP. En este punto se da el intercambio de datos (ej. paquetes de voz), será soportado por otro protocolo, como por ejemplo, por el protocolo RTP (Real Time Protocol).

6. BYE. Cuando una de las partes cuelga, envía el mensaje BYE. Aunque en el ejemplo el mensaje lo envía el cliente, se debe tener en cuenta que podría haber sido enviado por el servidor.

7. OK. La parte que recibe el mensaje de BYE envía el mensaje OK para confirmar dicha recepción. En este punto la llamada acaba.

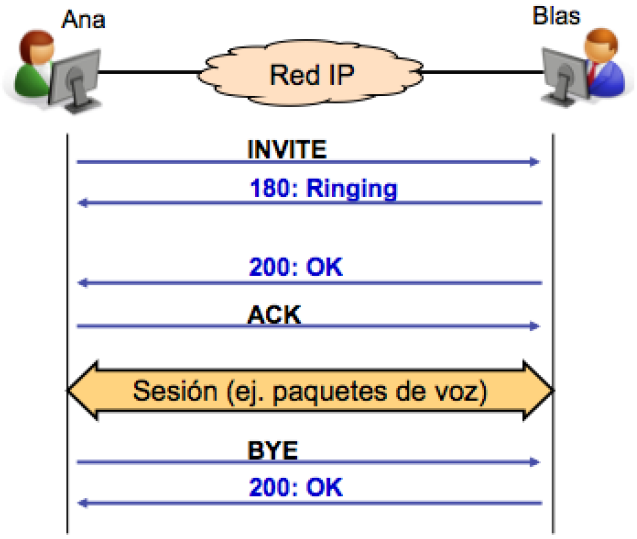

Figura 1. Ejemplo del establecimiento de la llamada SIP

2.1.4 Ventajas frente a otros protocolos

La señalización del protocolo SIP es muy simple en comparación con el número de mensajes que se utilizan para establecer la llamada en otros protocolos. Por ejemplo, se puede comprobar que el protocolo SIP es más rápido que H.323, especialmente si H.323 utiliza señalización Gatekeeper-routed2 en lugar del procedimiento Fast-Connect3.

El protocolo SIP permite que con los mensajes de petición y respuesta pueda ser incluida información no estándar. De esta forma se puede permitir a los usuarios y a las máquinas tomar decisiones sobre el funcionamiento de la llamada invocando diferentes servicios. Por ejemplo, si se realiza una llamada a un usuario que no está

disponible en ese momento, se remitiría esta información a quien realizó la llamada, pero además, se podría incluir información acerca de la hora en la que se encontrará disponible el usuario.

Por otra parte, hay que destacar que en el protocolo SIP son los clientes los que tienen el control de las características, no el operador de red.

Finalmente, otra ventaja es que SIP es un protocolo basado en texto, similar al protocolo HTTP, por lo que los programas realizados para parsear http pueden ser adaptados fácilmente para trabajar con SIP. Sin embargo, esto presenta la desventaja de que los mensajes consumirán mayor ancho de banda comparado con los protocoles que envían los mensajes directamente en binario.

2.2 Arquitectura de la red VoIP basada en SIP

Como ya se ha indicado, el Protocolo de Inicio de Sesiones (SIP) es el protocolo básico de la arquitectura de voz sobre IP considerada. SIP se encarga de los procedimientos de señalización de abonado (intercambio de mensajes de control entre los terminales utilizados para acceder al servicio VoIP y equipos de red) y de señalización de red (intercambio de mensajes de control interno entre equipos de red) necesarios para iniciar, mantener y terminar sesiones entre usuarios del servicio.

Estas sesiones pueden considerarse el equivalente a las llamadas telefónicas que se establecen en el servicio telefónico tradicional sobre redes de conmutación de circuitos, aunque con algunas diferencias sustanciales: el protocolo SIP no se limita a llamadas de voz, ya que está diseñado para el establecimiento de sesiones multimedia entre usuarios y la información que envían los usuarios durante la sesión (voz, vídeo, texto, etc.) se transporta sobre paquetes IP, no sobre circuitos.

Por otra parte, SIP se encarga también de funciones que son necesarias para localizar a un usuario en la red IP a partir de un identificador, por ejemplo, su número de teléfono o URI (Uniform Resource Identifier).

La figura siguiente muestra la arquitectura genérica de la red VoIP basada en SIP, con sus principales elementos: agentes de usuario, pasarelas, servidores proxy, redirect y registro de SIP, servidores de aplicaciones. A continuación se describe brevemente cada uno de ellos.

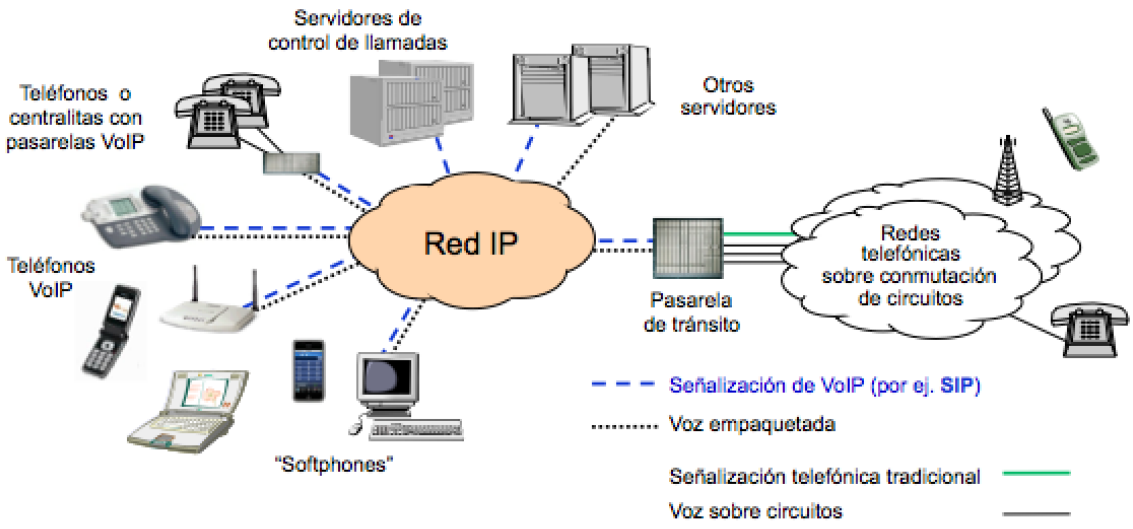

Figura 2. Arquitectura de la red VoIP basada en SIP

Agente de usuario SIP

El agente de usuario es el software SIP que procesa la señalización SIP en el terminal de usuario. Funciona como cliente cuando hace peticiones de inicio de sesión (llamadas salientes) y como servidor cuando las recibe (llamadas entrantes). Los terminales de usuario pueden ser programas (denominados "softphones") que funcionan en un ordenador de propósito general (un PC fijo o portátil, una agenda electrónica, etc.), teléfonos IP compatibles con SIP o incluso teléfonos convencionales conectados a través de pasarelas SIP.

Pasarela SIP

Una pasarela SIP actúa como un agente de usuario que convierte la señalización SIP en la señalización propia del bucle de abonado que espera un teléfono convencional y viceversa.

También pueden usarse pasarelas SIP para la interconexión entre la red IP sobre la que se presta el servicio VoIP y las redes telefónicas públicas, fijas o móviles. En este caso, la pasarela se denomina normalmente pasarela de tránsito y se encarga de hacer la conversión entre la señalización SIP por el lado de la red IP y la señalización de red que utiliza la red telefónica que hay detrás de la pasarela. La señalización de red telefónica está normalizada por la ITU-T en el conjunto de recomendaciones que definen el Sistema de Señalización nº 7 (SS7) [6]. Dentro del SS7, el protocolo de señalización "equivalente" a SIP se denomina Parte de Usuario de Servicios Integrados (ISUP). El SS7 define también protocolos de niveles bajos encargados de transportar los mensajes de señalización en la red telefónica, que serían el equivalente al protocolo IP

más el protocolo de transporte (por ejemplo, UDP) que se utilice para llevar los mensajes SIP.

A diferencia de una pasarela SIP de acceso encargada de conectar un solo teléfono, o unos cuantos, a la red de VoIP, una pasarela de tránsito puede ser un equipo de alta capacidad que deba manejar muchas llamadas simultáneas entre terminales VoIP y teléfonos fijos o móviles convencionales. Se han normalizado varios protocolos de control adicionales que permiten dividir las funciones de una pasarela de tránsito entre varios equipos separados conectados entre sí. En general, pueden considerarse tres tipos de equipos que formen parte de una pasarela de tránsito: controlador de pasarelas, pasarela de señalización y pasarela de medios. Ver figura.

El controlador se encarga de la función básica de conversión entre señalización SIP y señalización ISUP. La pasarela de señalización se encarga de hacer el interfaz con la red SS7 y de enviar los mensajes ISUP hasta el controlador.

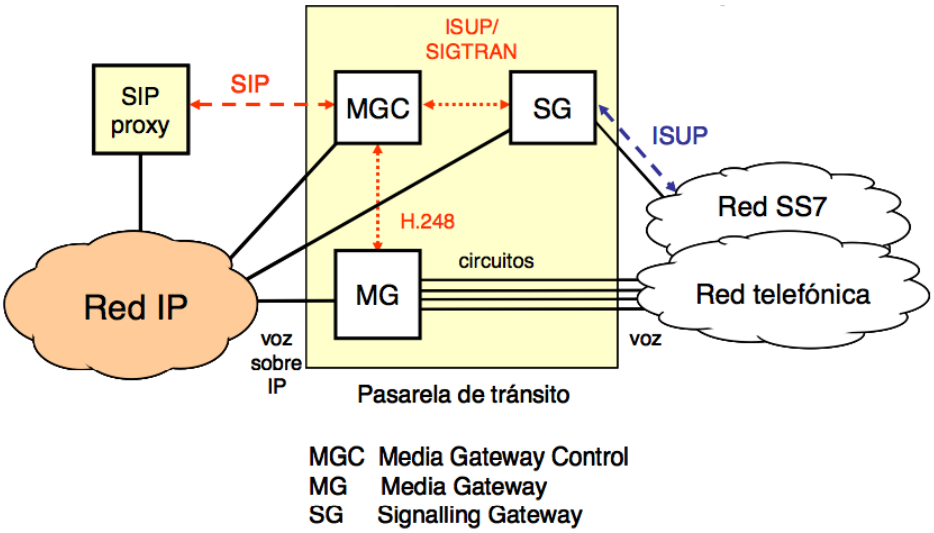

Figura 3. Elementos de una pasarela SIP de tránsito

Para transportar ISUP sobre la red IP desde la pasarela al controlador y viceversa se utilizan protocolos normalizados por el grupo SIGTRAN (Signaling Transport) del IETF. Por último, la pasarela de medios se encarga de terminar los circuitos de interconexión con la red telefónica y de las funciones de empaquetado y desempaquetado de voz (incluyendo los codec necesarios) sobre IP. El protocolo H.248/MEGACO, normalizado por ITU-T e IETF, permite a la pasarela de medios notificar eventos al controlador y a éste enviar las órdenes oportunas a la pasarela, por ejemplo, cada vez que se establece o libera una llamada. Ver detalles de estos protocolos en apartados posteriores.

Mediante pasarelas como la representada en la figura anterior, se pueden interconectar las redes de voz sobre IP con las redes telefónicas fijas o móviles,

utilizando coordinadamente los protocolos de control SIP, H.248 y SIGTRAN. No se consideran escenarios de tránsito de llamadas sobre IP, como, por ejemplo, el de la figura siguiente, en los que ambos lados de la comunicación son terminales conectados a la red telefónica y no se utiliza SIP.

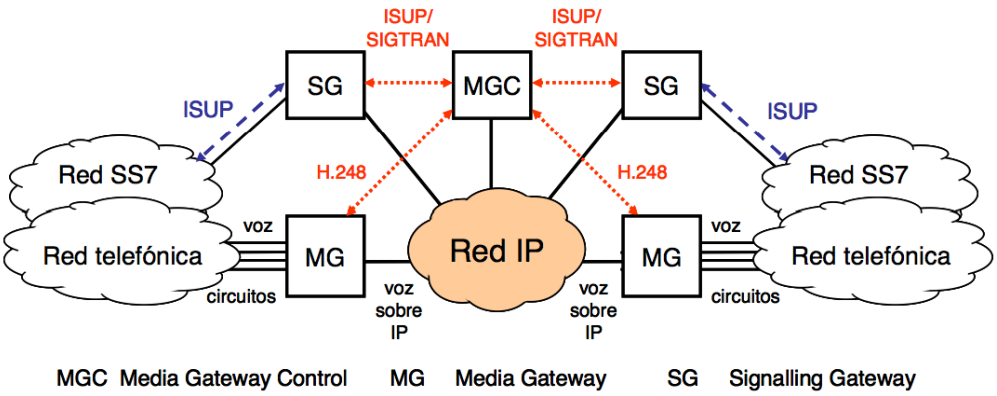

Figura 4. Escenario de tránsito de llamadas sobre IP sin SIP

Proxy SIP

El proxy SIP es un tipo de servidor intermedio, cuya misión es resolver, a partir de un número de teléfono o identificador, la dirección IP del usuario, con el que se quiere iniciar una sesión, así como controlar las fases de inicio y finalización de la sesión por motivos de contabilidad y facturación, en su caso. Un proxy SIP puede reenviar las peticiones de inicio de sesión a diversos agentes de usuario que pueda tener registrados un mismo usuario (por ejemplo, un terminal móvil, un PC y un teléfono IP fijo) de modo que todos "suenen" y al aceptar la sesión en alguno de ellos dejen de sonar los demás. La resolución de la dirección IP del usuario puede ejecutarse mediante distintos métodos, incluyendo consultas a bases de datos con correspondencias preestablecidas, accesos al DNS o delegando la resolución en otros proxies.

SIP Redirect

Este otro tipo de servidor intermedio SIP tiene la labor de responder a la resolución de nombres y la ubicación del usuario, proporcionando al agente de usuario la información acerca de la dirección del servidor requerido, de modo que pueda contactar con él.

Registro SIP

El registro proporciona el servicio de información de ubicación, recibiendo información de un agente de usuario (por ejemplo, dirección actual) para almacenarla y proporcionarla a otros agentes de usuario.

Otros servidores

Un servidor proxy puede comunicarse con uno o varios servidores de aplicaciones intercambiando mensajes SIP con ellos para prestar servicios asociados a una llamada. Por ejemplo, un usuario puede pedir a la red que todas las llamadas que cumplan ciertas condiciones ("criterios de filtrado") reciban un servicio determinado. La comunicación del proxy puede ser directa con cada servidor o a través de un elemento intermediario que haga funciones de "orquestación de servicios" o "service brokering".

Por otra parte, algunos servicios pueden estar implementados en el mismo servidor de aplicaciones SIP, mientras que otros pueden estar implementados en servidores externos que no usan SIP, por ejemplo, puntos de control de servicios de Red Inteligente o servidores de aplicaciones OSA (Open Service Access) [8]. En este caso, el servidor de aplicaciones SIP hace de pasarela entre el protocolo SIP por un lado y el protocolo de acceso al servidor externo por otro.

Aunque normalmente los servidores solo intervienen en el intercambio de mensajes de señalización en el plano de control, mientras que la voz empaquetada se envía directamente entre los terminales IP de los usuarios (o, en su caso, entre un terminal y una pasarela), en algunos casos puede ser necesario que el servidor procese flujos de paquetes de voz, por ejemplo, en un servicio de reconocimiento de voz y respuesta automática. Además, en este caso el servidor puede descomponerse en dos elementos separados: un controlador que maneja de la señalización SIP y un procesador de medios que maneja los flujos de voz empaquetada. Entre ambos se puede utilizar H.248 como protocolo de control. (Esta descomposición es similar a la de una pasarela SIP de tránsito, ver punto correspondiente, salvo porque aquí no hace falta la pasarela de señalización telefónica, ni los protocolos SIGTRAN).

Por último, pueden usarse servidores para guardar información de los usuarios del servicio de voz sobre IP. Estos servidores de información se consultan y actualizan con protocolos diferentes de SIP, por ejemplo, DIAMETER, o bien XCAP (XML Configuration Access Protocol) sobre HTTP en el caso de información almacenada en formato XML (Extensible Mark-up Language).

2.3 Otros protocolos en la arquitectura VoIP

En esta sección se presentan los protocolos básicos que se utilizan, junto con el protocolo SIP en la arquitectura de servicios VoIP considerada. Los aspectos de seguridad se detallan posteriormente.

2.3.1 Session Description Protocol (SDP)

Para describir las características de las sesiones que maneja, el protocolo SIP utiliza un subconjunto de SDP. SDP es un formato textual para describir sesiones multimedia, definido por el grupo MMUSIC (Multiparty Multimedia Session Control) del IETF. Ver RFC4566. Por ejemplo, el mensaje de petición INVITE típicamente contiene un cuerpo en formato SDP que especifica las direcciones IP y números de puerto a utilizar para enviar y recibir la voz o los medios asociados a la sesión, los codificadores a utilizar, el protocolo de transporte en el plano de usuario, etc.

En principio SDP considera el transporte de medios sobre protocolos no orientados a conexión como RTP y UDP. Adicionalmente, la RFC4145 extiende SDP para poder especificar el transporte de medios orientado a conexión sobre TCP.

2.3.2 Real-Time Transport Protocol (RTP)

El protocolo RTP, definido por el grupo AVT (Audio/Video Transport) del IETF, ver RFC3550, se utiliza habitualmente para la transmisión en tiempo real de audio y vídeo sobre UDP e IP. RTP incluye información sobre el formato de la carga útil que transporta (por ejemplo, códec utilizado, número de octetos/paquete), así como números de secuencia y marcas de tiempo que pueden servir a la aplicación para compensar los efectos del transporte imperfecto sobre IP. RTP carece de mecanismos para reservar recursos o garantizar una calidad de servicio mínima.

Opcionalmente, el protocolo RTCP (Real Time Control Protocol) puede usarse en paralelo con el flujo de paquetes RTP para enviar mensajes de control con datos de calidad de servicio observada (cantidad de paquetes enviados y recibidos, pérdidas, fluctuaciones de retardo,...).

Secure RTP, ver RFC3711, permite cifrado y autenticación de los mensajes de RTP y RTCP. Ver detalles posteriormente.

2.3.3 H.248/MEGACO

A finales de los años 90 se hicieron varias propuestas sobre protocolos de control de pasarelas que cristalizaron en un protocolo común definido conjuntamente por el grupo de trabajo MEGACO (Media Gateway Control) del IETF y el grupo de estudio 16 del ITU-T. El grupo MEGACO publicó la RFC3015 y la RFC3525 en 2000 y 2003 respectivamente. Por su parte, ITU-T publicó el mismo protocolo en sus recomendaciones H.248 y H.248.1 en 2000 y 2002. En los últimos años, el ITU-T se ha hecho cargo del desarrollo del protocolo, publicando una nueva versión de H.248.1 en 2005 y más recientemente en 2013 [7]. En febrero de 2008, el IETF clasificó la RFC3525 como histórica (ver RFC5125).

H.248 es un protocolo basado en transacciones entre un controlador y una o varias pasarelas de medios entre la red de voz sobre IP y la red de voz sobre conmutación de circuitos. El controlador se encarga de las funciones de proceso de llamadas (señalización, encaminamiento), envía órdenes a las pasarelas (crear conexión, poner/quitar tono, códec a usar,...) y recibe notificaciones de eventos desde las pasarelas.

Por su parte, la pasarela de medios se encarga del proceso de la voz en el plano de usuario (empaquetado/desempaquetado, cancelación de eco, compensación de fluctuaciones de retardo,...) bajo órdenes del controlador, así como de la detección de eventos básicos (colgar/descolgar, marcación de dígitos,...) y su notificación al controlador.

En H.248 el lado IP y el lado de la red telefónica de la pasarela se modelan mediante el concepto de "terminaciones". En el lado IP las terminaciones corresponden a puertos UDP asociados a flujos RTP y en el de la red telefónica, a circuitos de voz. Los mensajes MEGACO transportan los comandos necesarios para crear terminaciones, asociarlas en contextos o modificar su estado según progresan las llamadas, y las correspondientes respuestas indicando el resultado de cada comando. Los mensajes de solicitud y respuesta se pueden codificar en forma de texto o en binario y se transportan sobre UDP, TCP o SCTP. Como protocolo de seguridad se usa IPsec. Ver detalles posteriormente.

2.3.4 Signalling Transport (SIGTRAN)

El grupo SIGTRAN del IETF ha definido una arquitectura de protocolos para transportar sobre redes IP los protocolos de señalización utilizados en las redes telefónicas, por ejemplo, el protocolo SS7 ISUP de señalización entre conmutadores de circuitos o el protocolo Q.931 de señalización digital de abonado. En una red de VoIP, SIGTRAN se usa normalmente entre los controladores de pasarelas y las pasarelas de señalización. (SIGTRAN admite otros usos no necesariamente relacionados con el transporte de voz sobre IP. Por ejemplo, un operador de red telefónica fija o móvil puede usar SIGTRAN para transportar mensajes de señalización SS7 sobre IP, aunque la voz se siga transportando sobre circuitos).

Los elementos básicos de la arquitectura de protocolos SIGTRAN, ver RFC2719, se representan en la figura siguiente y se describen a continuación.

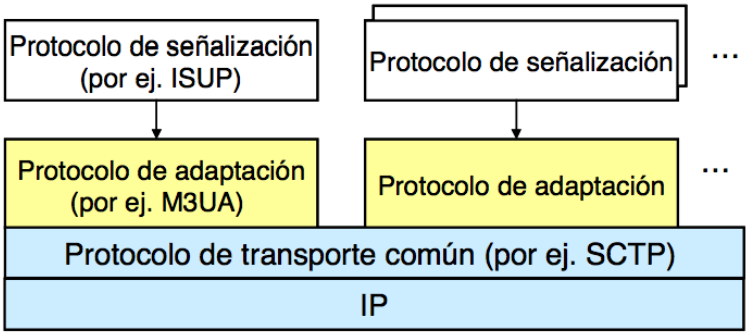

Figura 5. Protocolos de transporte de señalización sobre IP (SIGTRAN)

Stream Control Transmission Protocol (SCTP)

Al igual que TCP, SCTP, ver RFC4960, es un protocolo orientado a conexión y ofrece transporte fiable sobre IP retransmitiendo los paquetes perdidos. Además SCTP ofrece varias funciones adicionales, entre las que cabe destacar la protección ante fallos en la red IP mediante "multihoming". Cada extremo de una conexión SCTP, llamada "asociación", puede tener asociadas varias direcciones IP, de forma que si una dirección falla, se usa otra sin interrumpir la asociación SCTP.

SCTP incluye algunos procedimientos de protección frente a ataques de denegación de servicio, pero carece de mecanismos específicos para la seguridad de los mensajes en tránsito, ver RFC3788. Estos pueden implementarse por encima de SCTP, caso de TLS (Transport Layer Security) sobre SCTP, ver RFC3436, o bien por debajo, caso de SCTP sobre IPsec, ver RFC3554. Los detalles de TLS e IPsec se describen más adelante.

Recientemente, la RFC5062 ha descrito una serie de ataques conocidos sobre SCTP y formas de contrarrestarlos utilizando procedimientos publicados en la RFC4460 y posteriormente incluidos en la nueva definición del protocolo recogida en la RFC4960.

Protocolos de adaptación de usuario (UAs)

Ofrecen al protocolo de señalización transportado el mismo interfaz de servicio que en su entorno original en la red telefónica, de manera que pueda llevarse sobre IP sin necesidad de ningún cambio en el protocolo transportado. Hay varios protocolos UA definidos, incluyendo: M3UA (SS7 MTP3-User Adaptation layer, RFC4666), M2UA (SS7 MTP2-User Adaptation layer, RFC3331) e IUA (ISDN Q.921-User Adaptation layer, RFC4233). La capa UA incluye mecanismos de protección ante fallos en los nodos conectados a la red IP. Para ello se pueden definir varios procesos activos y de reserva residentes en máquinas diferentes manejados por el UA. Si un proceso falla, los mensajes de señalización se envían a otro de reserva capaz de tratar los mismos mensajes. Este mecanismo, junto con la protección ante fallos en la red que ofrece SCTP, sirve para aumentar la disponibilidad del transporte de señalización sobre IP.

La RFC4166 describe el uso de los diferentes protocolos que componen SIGTRAN para transportar señalización telefónica en diversos escenarios.

2.3.5 Otros protocolos

Además de los protocolos básicos de señalización (SIP/SDP, H.248, SIGTRAN) y de transporte en el plano de usuario (RTP), una red de voz sobre IP puede utilizar otros protocolos para funciones adicionales relacionadas con políticas de control de acceso, gestión de datos de los servicios, tarificación, etc. Para ello, pueden usarse protocolos como COPS (Common Open Policy Service), ver RFC2748, DIAMETER, ver RFC3588 y RFC4740, y XCAP (XML Configuration Access Protocol), RFC4825. La descripción de protocolos de seguridad para VoIP viene más adelante.

3 Ataques y vulnerabilidades en las redes VoIP

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 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 problemas clásicos de seguridad que afectan al mundo de las redes de datos. Por supuesto, existen también multitud de ataques específicos de VoIP como se describe más adelante.

3.1 Clasificación de ataques

Figura 6. Capas de seguridad de la información en las redes VoIP

Como se ve en la Figura 7, la seguridad de VoIP se construye sobre muchas otras capas tradicionales de seguridad de la información.

En la tabla siguiente se detallan algunos de los puntos débiles y ataques que afectan a cada una de las capas. Aunque posteriormente se analizaran muchos de ellos en profundidad algunos ataques que pueden afectar directamente o indirectamente a la telefonía VoIP no serán explicados al ser problemas comunes a cualquier otra red de datos o al alejarse demasiado de la temática del trabajo.

Capa	Ataques y vulnerabilidades
Políticas y Procedimientos	Contraseñas débiles. Ej.: Contraseña del VoiceMail; Mala política de privilegios; Accesos permisivos a datos comprometidos.
Seguridad Física	Acceso físico a dispositivos sensibles. Ej.: Acceso físico a un gatekeeper; Reinicio de máquinas; Denegaciones de servicio.
Seguridad de Red	DDoS; ICMP unreacheable; SYN floods; Gran variedad de floods.
Seguridad en los Servicios	SQL injections; Denegación en DHCP; DoS.
Seguridad en el Sistema Operativo	Buffer overflows; Gusanos y virus; Malas configuraciones.
Seguridad en las aplicaciones y protocolos de VoIP	Fraudes; SPIT (SPAM); Vishing (Phising); Fuzzing; Floods (INVITE, REGISTER, etc.); Secuestro de sesiones (Hijacking); Interceptación (Eavesdroping); Redirección de llamadas (CALL redirection); Reproducción de llamadas (CALL replay).

Se pude apreciar algunos de estos ataques tendrán como objetivo el robo de información confidencial y algunos otros degradar la calidad de servicio o anularla por completo (DoS). Para el atacante puede ser interesante no solo el contenido de una conversación (que puede llegar a ser altamente confidencial), sino también la información y los datos de la propia llamada, que utilizados de forma maliciosa permitirán al atacante realizar registros de las llamadas entrantes o salientes, configurar y redirigir llamadas, grabar datos, utilizar información para bombardear con SPAM, interceptar y secuestrar llamadas, reproducir conversaciones, llevar a cabo robo de identidad e incluso realizar llamadas gratuitas a casi cualquier lugar del mundo. Los dispositivos de la red, los servidores, sus sistemas operativos, los

protocolos con los que trabajan y prácticamente todo elemento que integre la infraestructura VoIP podrá ser susceptible de sufrir un ataque.

Durante los siguientes apartados se va a intentar detallar cuales son las amenazas más significativas que afectan a la telefonía sobre redes IP. Como ya se ha comentado la mayoría los riesgos son inherentes de las capas sobre las que se apoya la tecnología VoIP por lo que muchos de los ataques se basarán en técnicas bien conocidas. Se mostraran, también, ciertas vulnerabilidades que afectan específicamente a las redes VoIP y a sus protocolos.

Las amenazas de las redes de telefonía IP se pueden clasificar en las siguientes categorías:

- Accesos desautorizados y fraudes.
- Ataques de denegación de servicio
- Ataques a los dispositivos
- Vulnerabilidades de la red subyacente.
- Enumeración y descubrimiento.
- Ataques a nivel de aplicación.

3.2 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 que, si son comprometidos, pueden ser utilizados para realizar fraudes. El costo de usar fraudulentamente esos datos VoIP a nivel empresarial pueden ser devastadores. El acceso a los datos telefónicos (de facturación, registros, datos de cuentas, etc.) pueden ser usados con fines fraudulentos.

Una de las mas importantes amenazas de las redes VoIP, son los fraudes consecuencia de un acceso desautorizado a una red legal VoIP (por ejemplo, haber obtenido anteriormente datos de cuentas). Una vez se ha obtenido el acceso, usuarios desautorizados realizan llamadas de larga distancia, en muchos casos incluso internacionales. Principalmente ocurren en entornos empresariales. El control y el registro estricto de las llamadas puede paliar el problema

A modo de curiosidad cabe señalar que las técnicas utilizadas por estos individuos son descendientes de las que utilizaban los famosos "phreakers" en las antiguas líneas telefónicas.

3.3 Ataques de denegación de servicio

Los ataques de denegación de servicio 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 construidos 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.

Llegan a ser especialmente dañinos los llamados DDoS o ataques de denegación distribuidos. Son ataques DoS simples, pero realizados desde múltiples computadores de forma coordinada. Las redes y sistemas VoIP son especialmente vulnerables a los DDoS por diversas razones:

- La primera, y quizás más importante, es la dependencia y la necesidad de garantías en la calidad de servicio, que hacen que las redes IP, donde se mantengan llamadas telefónicas, tengan una tolerancia mucho menor a problemas de rendimiento.
- Otra razón es que 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.

La relación de VoIP y los ataques distribuidos de DoS viene reflejada en el siguiente párrafo:

Recientemente investigadores de la Universidad de Cambrige y del Massachussets Institute of Technology (MIT) han determinado que las aplicaciones de voz sobre IP, como puede ser Skype, pueden ser una herramienta ideal para dar cobertura y lanzar ataques de denegación de servicio distribuidos. El descubrimiento de algún fallo en la aplicación o en su protocolo podría dejar al descubierto miles de ordenadores que

serían potencialmente secuestrados por los atacantes para realizar un ataque mayor contra algún servicio de Internet.

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, GOODBYE o ICMP Port Unreacheable, con el objetivo de desconectar ciertos usuarios de sus respectivas llamadas o evitar que se produzcan no permitiendo la correcta configuración inicial de la llamada (señalización).

Hay que destacar también que en algunas situaciones VoIP será vulnerable a ataques de fragmentación IP o envío de resets TCP, que conllevarán la prematura finalización de la llamada.

3.4 Ataques a los dispositivos

Muchos de los ataques realizador 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, sin olvidar los teléfonos IP, serán potencialmente objetivos a explotar por parte de un intruso.

Hay que tener en cuenta que los dispositivos VoIP son tan vulnerables como lo es el sistema operativo o el firmware que ejecutan. Son muy frecuentes los ataques de *fuzzing* con paquetes malformados que provocan cuelgues o reboots en los dispositivos cuando procesan dicho paquete. Otros ataques de denegación de servicio llamados "*flooders*" tienen como objetivo los servicios y puertos abiertos de los dispositivos VoIP.

Otro aspecto que hace muchas veces de los dispositivos un punto débil dentro de la red son configuraciones incorrectas. A menudo los dispositivos VoIP trabajan con sus configuraciones por defecto y presentan gran variedad de puertos abiertos. Los servicios por defecto corren en dichos puertos y pueden ser vulnerables a ataques de DoS, desbordamientos de buffer o cualquier otro ataque que pueden resultar en el compromiso del dispositivo VoIP.

El intruso a la hora de penetrar en la red tendrá en cuenta estos aspectos e intentará explotarlos. Buscará puertos por defecto y servicios innecesarios, comprobará passwords comunes o los que usa por defecto el dispositivo, etc.

No hay que olvidarse de los dispositivos VoIP que utiliza el usuario directamente: los teléfonos. A pesar de ser dispositivos más pequeños, obviamente son igual de vulnerables que cualquier otro servidor de la red, y el resultado de comprometer uno de ellos puede llegar a ser igual de negativo.

A modo de ejemplo se detalla una vulnerabilidad de que afectó al teléfono IP Linksys SPA-921 v1.0 y que provocaba una denegación de servicio en el mismo.

Modelo: Linksys SPA-921
Version: 1.0.0
Tipo vulnerabilidad: DoS
Explicación:
1) La petición de una URL larga al servidor http del dispositivo provoca que el teléfono se reinicie.
2) Un nombre de usuario o un password demasiado largo en la autenticación http provoca que el teléfono se reinicie.
Modo de explotarlo: Trivial

3.5 Vulnerabilidades de la red subyacente

Una de las principales debilidades de la tecnología VoIP es apoyarse sobre una red potencialmente insegura como son las redes IP. Gran cantidad de ataques hacia las infraestructuras IP van a afectar irremediablemente a la telefonía. Ataques de denegación de servicio, inundación de paquetes o cualquier otro tipo de ataque que intente limitar la disponibilidad de la red suponen un gran problema para la telefonía IP tal y como hemos visto anteriormente. Además VoIP será vulnerable a ataques a bajo nivel, como el secuestro de sesiones, interceptación, fragmentación IP, paquetes IP malformados y spoofing.

Uno de los mayores problemas sea quizás la interceptación o ***eavesdropping***. Traducido literalmente como "escuchar secretamente", 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, estamos hablando de la interceptación de las conversaciones VoIP por parte de individuos que no participan en la conversación.

El ***eavesdropping*** en VoIP presenta pequeñas diferencias frente la interceptación de datos en las redes tradicionales. El impacto de esta técnica es más que evidente, interceptando comunicaciones es posible obtener toda clase 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.

Las formas de conseguir interceptar una comunicación pueden llegar a ser tan triviales como esnifar el tráfico de la red, si los datos no van cifrados. Existen excelentes sniffers como *ethereal/wireshark* que permitirán capturar todo el tráfico de tu segmento de la red. Por lo contrario, lo normal es que nos encontramos dentro de redes conmutadas por lo que para esnifar el tráfico que no vaya dirigido a nuestro equipo serán necesarias otras técnicas más elaboradas, como realizar un "*Main in the Midle*" utilizando *Envenenamiento ARP*. Entre las herramientas que se pueden utilizar se encuentra el programa *ettercap*, *Cain & Abel*, la suite de herramientas para Linux *Dsniff* y *vomit* (Voice over misconfigured Internet telephones) por citar algunos ejemplos.

Hay que señalar también la creciente utilización de redes inalámbricas, que supone en muchos casos una vía más a explotar por parte del intruso. Redes Wifi mal configuradas junto con una infraestructura de red insegura pueden facilitar el trabajo del intruso a la hora de acceder a la red VoIP para lanzar sus ataques.

3.6 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 victima. 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* y 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. 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.

Un ejemplo de ataque de enumeración podría ser utilizar la fuerza bruta contra servidores VoIP para obtener una lista de extensiones telefónicas válidas. Información

que sería extremadamente útil para lanzar otros ataques como inundaciones INVITE o secuestro de registro.

Durante esta subsección se explicarán algunas técnicas de enumeración y descubrimiento de objetos, así como la obtención de información sensible que atacante podría utilizar a su favor.

3.6.1 Footprinting

Se conoce como *footprinting* el proceso de acumulación de información de un entorno de red espeífico, usualmente con el propósito de buscar formas de introducirse en el entorno.

La herramienta básica para esta etapa del reconocimiento será el Google. Las búsquedas se centrarán entorno a la web de la empresa y en su dominio. Se intentarán encontrar perfiles o direcciones de contacto, correos y teléfonos. Estos datos ofrecerán información al hacker para poder realizar ataques de suplantación de identidad y/o ingeniería social. El contacto del servicio técnico también puede resultar útil para extraer algún tipo de información. Otro tipo de información interesante pueden ser las ofertas de trabajo o los perfiles de personal que busca la empresa. Pueden dar información acerca de la estructura de la organización y de la tecnología que emplea.

La mayoría de dispositivos VoIP corren algún servicio web de administración remota. Es posible encontrarlos con Google:

Algunas otras búsquedas interesantes se resumen en la siguiente tabla:

Asterisk Management Portal	*intitle:asterisk.management.portal web-access*
Cisco Phones	*inurl:"NetworkConfiguration" cisco*
Cisco CallManager	*inurl:"ccmuser/logon.asp"*
D-Link Phones	*intitle:"D-Link DPH" "web login setting"*
Grandstream Phones	*intitle:"Grandstream Device Configuration" password*
Linksys (Sipura) Phones	*intitle:" SPA Configuration"*
Polycom Soundpoint Phones	*intitle:"SoundPoint IP Configuration"*
Snom Phones	*"(e.g. 0114930398330)" snom*

Otras técnicas se centran en intentar localizar a través de Google extensiones, para después realizar llamadas a los Voicemail y estudiar la grabación. El objetivo es obtener el fabricante del servidor, que seguramente sea el mismo para el resto de dispositivos VoIP de la red.

3.6.2 Escaneando

A partir de la dirección de red de la victima, se pretende obtener un listado de direcciones IP y servicios activos en la red. La mejor forma es escaneando la red con las

herramientas adecuadas. Quizás el mejor escáner de puertos existente hoy por hoy sea NMAP (http://insecure.org/namp) que ofrece muchas más posibilidades que un simple escáner de puertos.

Entre todas las funcionalidades de NMAP existe una que se destaca especialmente. Y es la identificación del sistema operativo de la máquina escaneada a partir de información que obtiene NMAP, como los puertos abiertos que presenta, tipos de servicios, y huellas identificativas de la pila TCP/IP.

En el caso concreto NMAP tiene la mejor base de datos de huellas para identificar dispositivos VoIP. A continuación se presenta un ejemplo de cómo lo hace:

```
nmap -O -P0 192.168.1.1-254
Starting Nmap 4.01 ( http://www.insecure.org/nmap/ ) at 2006-02-20 01:03 CST
Interesting ports on 192.168.1.21:
(The 1671 ports scanned but not shown below are in state: filtered)
PORT   STATE SERVICE
23/tcp open  telnet
MAC Address: 00:0F:34:11:80:45 (Cisco Systems)
Device type: VoIP phone
Running: Cisco embedded
OS details: Cisco IP phone (POS3-04-3-00, PC030301)
Interesting ports on 192.168.1.23:
(The 1671 ports scanned but not shown below are in state: closed)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 00:15:62:86:BA:3E (Cisco Systems)
Device type: VoIP phone|VoIP adapter
Running: Cisco embedded
OS details: Cisco VoIP Phone 7905/7912 or ATA 186 Analog Telephone Adapter
Interesting ports on 192.168.1.24:
(The 1671 ports scanned but not shown below are in state: closed)
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 00:0E:08:DA:DA:17 (Sipura Technology)
Device type: VoIP adapter
Running: Sipura embedded
OS details: Sipura SPA-841/1000/2000/3000 POTS<->VoIP gateway
```

Vemos resaltado en negrita los detalles del sistema operativo. Una vez ha escaneado los puertos de la máquina y ha obtenido información suficiente, es capaz de identificar de una forma suficiente fiable (al menos mejor que ninguna otra herramienta) el sistema del que se trata.

A la hora de escáner la red objetivo para identificar sistemas VoIP se debería tener en cuenta que los dispositivos SIP usualmente responden a los puertos 5060-5061 tanto en UDP como en TCP. En cambio los dispositivos de Cisco que utilicen el protocolo SCCP abrían los puertos 2000-2001 TCP.

3.6.3 Enumeración

La enumeración es una técnica que tiene por objetivo obtener información sensible que el intruso podría utilizar para basar sus ataques posteriores.

La primera información a obtener es el tipo de servicio que esta corriendo en un determinado puerto, esta identificación ya la realiza correctamente herramientas como NMAP, pero se podrían hacer manualmente conectado al puerto .En el siguiente ejemplo se conecta a un servidor SIP utilizando la herramienta Netcat bien conocida como la navaja suiza:

```
[root@attacker]# nc 192.168.1.104 5060
  OPTIONS sip:test@192.168.1.104 SIP/2.0
  Via: SIP/2.0/TCP 192.168.1.120;branch=4ivBcVj5ZnPYgb
  To: alice <sip:test@192.168.1.104>
  Content-Length: 0


  SIP/2.0 404 Not Found
  Via: SIP/2.0/TCP 192.168.1.120;branch=4ivBcVj5ZnPYgb;received=192.168.1.103
  To: alice <sip:test@192.168.1.104>;tag=b27e1a1d33761e85846fc98f5f3a7e58.0503
  Server: Sip EXpress router (0.9.6 (i386/linux))
  Content-Length: 0
  Warning: 392 192.168.1.104:5060 "Noisy feedback tells:  pid=29801 req_src_ip=192.168.1.120
       req_src_port=32773 in_uri=sip:test@192.168.1.104 out_uri=sip:test@192.168.1.104 via_cnt==1"
```

Al conectar al puerto especificado manualmente se envía una petición OPTIONS genérica al servidor, para poder estudiar su respuesta. En ella podemos observar que nos muestra información clara sobre el tipo de dispositivo que se trata.

Algunas otras herramientas que automatizan este proceso son:

- **Smap**: Permite identificar dispositivos SIP.
- **Sivus**: Un escáner de vulnerabilidades para SIP. Permite entre otra cosa generar peticiones SIP.
- **Nessus**: Uno de los mejores escáneres de vulnerabilidades. Permite además identificar los servicios y sistemas.
- **VoIPAudit**: Otro escáner VoIP y de vulnerabilidades.

Para poder realizar la mayoría de ataques, el intruso deberá conocer nombres de usuario y extensiones telefónicas correctas. Existen diversos métodos para recabar ese tipo de información. Una de las técnicas es utilizando las operaciones de registros de usuario. Cuando un usuario pretende registrarse envía una petición REGISTER al servidor de registro y este le responde con un 200 OK si todo va bien o con un mensaje 4xx si ha habido algún error, el usuario no existe o no tiene los credenciales de autenticación adecuados. Dependiendo del software la respuesta del servidor de

registro contra una petición de REGISTER de un usuario existente y no existente puede ser diferente en el sentido de que, si un usuario existe puede que conteste con un mensaje 401, ya que le falte autenticarse, pero si el usuario no existe, responderá directamente con un mensaje 403 Forbidden. Esta diferencia en la respuesta puede ser utilizada para enumerar y obtener un listado de usuarios válidos de la red VoIP.

Un método similar al anterior consiste en utilizar mensajes INVITE para enumerar posibles usuarios de la red. Algunos servidores responderán con un mensaje 401 cuando se intenta llamar a un usuario inexistente. El gran problema de este método es que cuando se acierte y se encuentre un usuario correcto, se estará realizando una llamada y el teléfono del usuario en cuestión sonará y quedará registrada la llamada.

Quizás el método más silencioso para enumerar usuarios es el que utiliza peticiones OPTION. Las peticiones OPTION se utilizan para determinar, por ejemplo, que codecs soporta un determinado UA. El servidor contestará con un 200 OK si el usuario existe y un 404 Not Found si no reconoce el usuario.

Algunas de las herramientas que automatizan todo este proceso, utilizando diccionarios o fuerza bruta con mensajes REGISTER, INVITE o OPTION son: Sipsak y Sipscan.

Dentro de la plataforma VoIP coexisten gran cantidad de servicios que se podrían aprovechar para obtener información. Algunos de ellos son el DHCP y DNS pero nos concentraremos en algunas técnicas contra el servicio TFTP y el protocolo SNMP.

La mayoría de dispositivos telefónicos utilizan el protocolo TFTP para manejar sus ficheros de configuración. Normalmente cada vez que un dispositivo VoIP se conecta, intenta obtener su configuración del servidor TFTP. El problema es que el servicio TFTP es un servicio altamente inseguro, problema que se agrava con el hecho de que en la configuración de los dispositivos se podrá encontrar todo tipo de información valiosa: extensiones, usuarios, contraseñas, estructura de la red, servidores, etc. Por lo que los servidores TFTP de configuración se convierten en un objetivo claro para comprometer la red VoIP.

La premisa en TFTP es que si se puede averiguar el nombre del fichero de configuración, lo puedes descargar. Muchos dispositivos utilizan nombre por defecto públicamente conocidos, por lo tanto si se identifica el dispositivo puede resultar trivial obtener su configuración del servidor TFTP. Por ejemplo, en los dispositivos CISCO el nombre del archivo de configuración mantiene relación con su dirección MAC.

Evidentemente el primer paso debería ser localizar el servidor TFTP en la red. Se puede utilizar un escáner como NMAP buscando direcciones con el puerto 69 UDP abierto.

Una vez localizado el servidor TFTP, el intruso intentará descargar los ficheros de configuración y, como ya ha quedado demostrado, la única dificultad que se le presenta es adivinar el nombre de los ficheros. Existen herramientas como Tftpbrute (que utilizan listados de palabras y diccionarios para atacar el servidor TFTP y descargarse ficheros de configuración). También es posible realizar todo el trabajo manualmente, ya que existen diversas listas que relacionan modelo/fabricante con el nombre por defecto de su archivo de configuración.

El protocolo SNMP (Simple Network Management Protocol) que se presenta activo en muchos de los dispositivos VoIP es otro de los protocolos vulnerables de los que se puede obtener gran cantidad de información.

Los pasos serían los siguientes:

1) Buscar dispositivos con soporte SNMP. Usualmente tendrán el puerto 162 UDP. Se pueden utilizar herramientas como NMAP o SolarWindos SNMPSweep.

2) Sivno se conoce el OID del dispositivo utilizar la SolarWind MIB para encontrarlo.

3) Con la herramienta Snmpwalk y el OID del dispositivo es posible listar la mayoría de aspectos de su configuración.

3.7 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 ellos su propio riesgo de segundad. 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 y algunos otros que se explicarán a continuación utilizando el protocolo SIP como base.

3.7.1 Autenticación en VoIP

En toda comunicación, servicio o transmisión de datos existe la necesidad de demostrar que los clientes son quien dicen ser. En VoIP la autenticación requiere que los dos dispositivos que se van a comunicar se autentiquen uno al otro antes de que se produzca cualquier intercambio de información. Esta autenticación mutua está basada en algún tipo de secreto compartido que es conocido a priori por los dos.

El protocolo SIP utiliza la autenticación *digest* para comprobar la identidad de sus clientes. La autenticación *digest* fue originalmente diseñada para el protocolo HTTP, y se trata de un mecanismo bastante simple, basado en hashes que evita que se envíe la contraseña de los usuarios en texto claro.

Cuando el servidor quiere autenticar un usuario genera un desafío *digest* que envía al usuario. Un ejemplo de desafío podría ser:

```
Digest realm="iptel.org", qop="auth,auth-int",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093", opaque="",
algorithm=MD5
```

Destacar que `nonce` es la cadena que genera como desafío utilizando el algoritmo MD5 de algún otro dato.

Después de recibir el desaío el UA pedirá al usuario el nombre y la contraseña (si no están presentes en la configuración del dispositivo) y a partir de ellos y del desafío enviado por el servidor generará una respuesta *digest* como la siguiente:

```
Digest username="jan", realm="iptel.org",
nonce="dcd98b7102dd2f0e8b11d0f600bfb0c093",
uri="sip:iptel.org", qop=auth, nc=00000001, cnonce="0a4f113b",
response="6629fae49393a05397450978507c4ef1", opaque=""
```

De una forma similar el campo `response` contendrá la respuesta generada por el UA. Cabe destacar el significado del `URI` que indica la dirección SIP a la que se quiere acceder y el `conoce` que es una cadena utilizada por el cliente y el servidor que ofrece cierta protección de integridad al mensaje.

Cuando recibe la respuesta del cliente, el servidor realiza exactamente los mismos pasos. Generando una respuesta *digest* a partir del desafío y del *password* del usuario que tiene almacenado en su configuración. Si el hash generado coincide con la respuesta del cliente, el usuario acaba de autenticarse demostrando ser quien dice ser.

Cuando el servidor SIP recibe alguna petición SIP, comprueba si en el mensaje se encuentran las credenciales que autentiquen al usuario, en caso contrario, generará un mensaje de error 401 Unauthorized al cliente incluyen el desafío *digest* para iniciar el proceso de autenticación.

El siguiente ejemplo muestra un mensaje REGISTER que contiene las credenciales *digest*.

```
REGISTER sip:iptel.org SIP/2.0.
Via: SIP/2.0/UDP 195.37.78.121:5060.
From: sip:jan@iptel.org.
To: sip:jan@iptel.org.
Call-ID: 003094c3-bcfea44f-40bdf830-2a557714@195.37.78.121.
CSeq: 102 REGISTER.
User-Agent: CSCO/4.
Contact: <sip:jan@195.37.78.121:5060>.
Authorization: Digest username="jan",realm="iptel.org",
uri="sip:iptel.org",response="dab81127b9a7169ed57aa4a6ca146184",
nonce="3f9fc0f9619dd1a712b27723398303ea436e839a",algorithm=md5.
Content-Length: 0.
Expires: 10.
```

A continuación se describen los métodos y las herramientas para romper esa autenticación y crackear los hashes *digest* con el fin de obtener el *password* de un usuario y poder utilizar la identidad de la víctima de forma maliciosa.

Entre las herramientas se encuentra SIPCrack , que como su nombre indica, crackea las contraseñas del protocolo SIP en Linux. Contiene dos programas *sipdump* para esnifar los hashes de la autenticación y *sipcrack* para crackear los logins capturados.

Fácilmente se puede descargar de las siguientes direcciones: página oficial http://www.codito.de o PacketStorm http://packetstormsecurity.org.

En caso de encontrarnos una vez más en redes conmutadas puede que se utilicen herramientas como *ettercap* para realizar la técnica de "*man in the middle*" y poder esnifar el tráfico necesario.

El programa *sipdump* actúa a modo de *sniffer*, analizando el tráfico y extrayendo autenticaciones SIP que encuentre.

```
# ./sipdump -i eth0 -d captura.dump
SIPdump 0.1 ( MaJoMu | www.remote-exploit.org )

* Using dev 'eth0' for sniffing
* Starting to sniff with filter 'tcp or udp'
```

Sipdump puede también analizar una captura realizada de algún otro *sniffer* como *tcpdump*. Localiza los paquetes SIP dentro de la captura, los decodifica y extrae los logins que encuentre.

```
# ./sipdump -f capturaSIP.pcap -d fichdump
SIPdump 0.1       ( MaJoMu | www.remote-exploit.org )

* Using tcpdump data file 'capturaSIP.pcap' for sniffing
* Starting to sniff with filter 'tcp or udp'
* Adding 192.168.0.35:50451 <-> 192.168.0.1:50195 to monitor
list...id
0
* New traffic on monitored connection 0 (192.168.0.35 ->
192.168.0.1)
* Found challenge response (192.168.0.35:50451 <->
192.168.0.1:50195)
* Wrote sniffed login 192.168.0.35 -> 192.168.0.1 (User: '200')
to
dump file
* Exiting, sniffed 1 logins
*  Adding 192.168.1.35:50451 <-> 192.168.1.100:50195 to monitor
list...id 0
*  New traffic on monitored connection 0 (192.168.1.35 ->
192.168.1.100)
*  Found challenge response (192.168.1.35:50451 <->
192.168.1.100:50195)
*  Wrote sniffed login 192.168.1.35 -> 192.168.1.100 (User:
'100') to dump file
```

Como es normal, el éxito de este ataque dependerá de lo bueno y preciso que sea el diccionario que se utilice.

Los ataques de fuerza bruta se encargan de probar todas las palabras generadas por todas las combinaciones posibles de cierto grupo de carácteres. Se demuestra uno de los crackeadores más famosos de la historia: *John the Ripeer*, el cual se puede descargar de la pagian oficial: http://www.openwall.com/john. Con el *John the Ripper* se genera un diccionario con todas las posibles combinaciones de cierto grupo de caracteres que se le indique.

Otra herramienta que sin duda merece la pena comentar para el crackeo de contraseñas es *Cain*. Permite realizar todo el proceso de captura de tráfico, envenenamiento ARP, decodificación de protocolos y crackeo de hash por diccionario y fuerza bruta.

3.7.2 Manipulación de la señalización

A continuación se detallan algunos de los ataques que se pueden conseguir capturando y manipulando los mensajes de señalización previos al establecimiento de la llamada.

Suplantación de identidad en el registro

El registro de usuarios es la primera comunicación que se estable en el entorno VoIP entre el usuario y el servidor de registro. Necesariamente esta comunicación debe de realizarse de forma segura, ya que en caso contrario no hay garantías de que el usuario registrado sea quien dice ser durante todo el resto de la sesión. A través de los mensajes REGISTER, los agentes de usuario SIP informan al servidor de su localización actual de manera que el servidor sepa a dónde tiene que enviar peticiones posteriores. Si un servidor no autentica las peticiones REQUEST cualquiera puede registrar cualquier contacto para cualquier usuario, y por lo tanto secuestrar su identidad y sus llamadas.

Cuando un Proxy recibe la petición para procesar la llamada (INVITE), el servidor realiza una búsqueda para identificar donde puede ser encontrado el destinatario . En la figura podemos observar un mensaje de respuestas del servidor de registro a una petición de búsqueda de un Proxy Server.

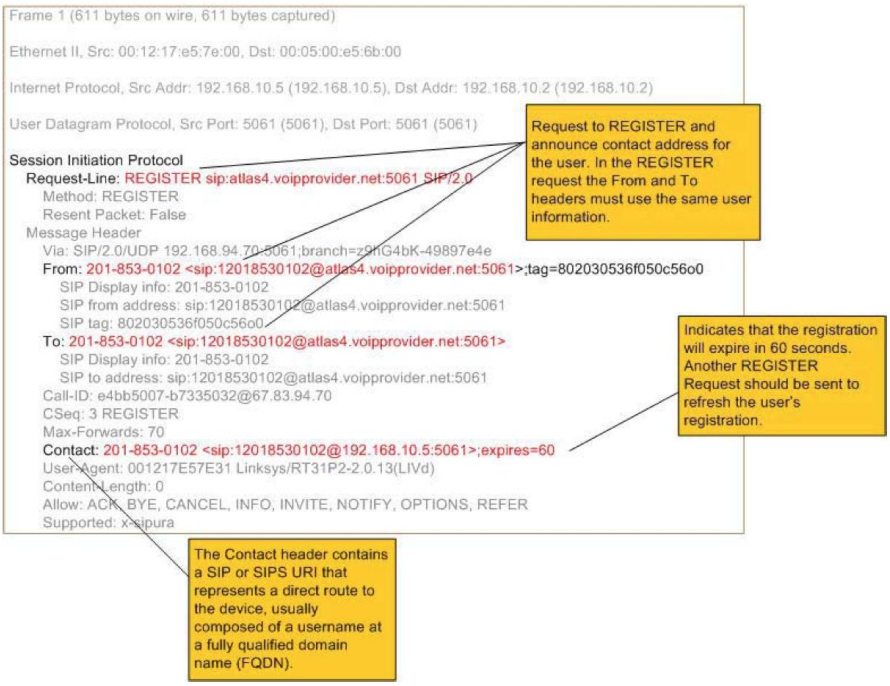

Figura 7. Mensaje de respuestas del servidor de registro a una petición de búsqueda de un Proxy Server

El mensaje REGISTER contiene el campo en la cabecera `Contact`: que indica la dirección IP del hardware o software VoIP del usuario destino. En el caso del ejemplo, el usuario puede ser localizado en el número de teléfono 201-853-0102 o a través de la IP 192.168.94.70. El Proxy redirige la petición INVITE hacia esta dirección IP.

En la figura siguiente se muestra una versión modificada de una petición REGISTER que es enviada por el atacante. En esta petición todo los parámetros de la cabecera son iguales, excepto por el campo `contact` que se ha modificado para escribir la IP del atacante. Para generar la petición se ha utilizado la herramienta SiVus:

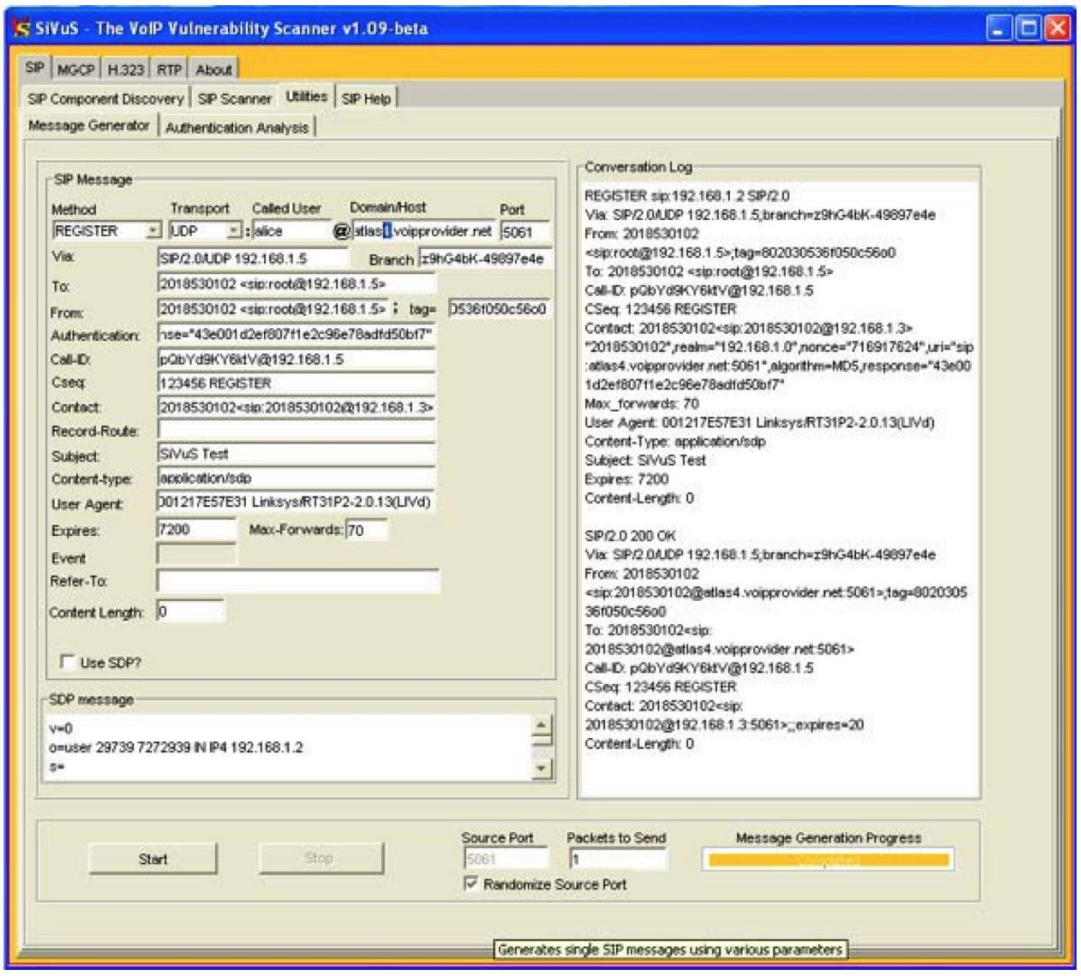

Figura 8. Petición REGISTER a través de la herramienta SiVus

El ataque funciona de la manera siguiente:

1. Deshabilitando el registro legítimo del usuario.
2. Enviando el mensaje REGISTER con la IP del atacante.
3. En el servidor de registro queda registrado el usuario Bob, pero con la dirección IP del atacante.
4. Cuando recibe la llamada, el servidor Proxy consulta la dirección del destinatario Bob, pero obtendrá la dirección IP del atacante.

5. 5. El ataque ha tendido éxito. El intruso ha suplantado la identidad de Bob y mientras mantenga el registro, todas las llamadas dirigidas a Bob llegarán a su teléfono IP.

Ver Figura 10.

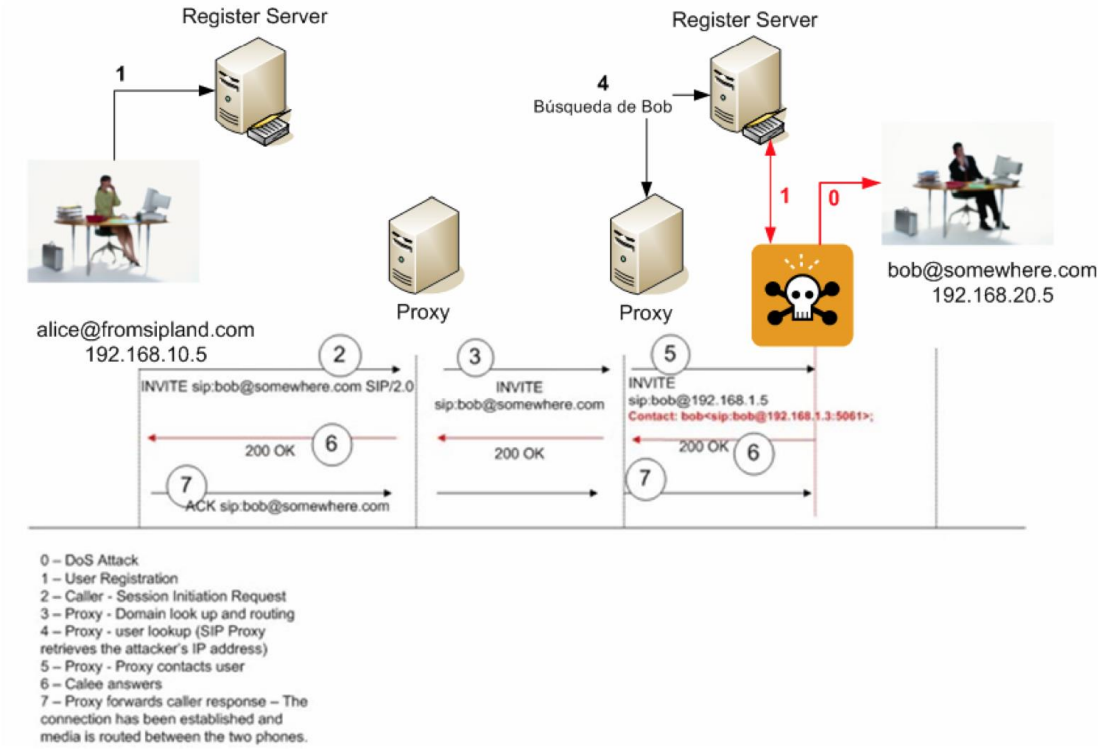

Figura 9. Suplantación de identidad en el registro

Este ataque es posible llevarlo a cabo por el hecho de que los mensajes de señalización se envían en texto plano, lo que permite al intruso capturarlos, modificarlos y retransmitirlos como él quiera.

A partir de la técnica de secuestro de registro se puede realizar alguna variante del ataque. En el caso anterior se evitaba que el destinatario legítimo recibiera la llamada, pero en algunos casos se puede conseguir realizar un ataque de "*Man in the middle*" a nivel de red. De esta forma el destinatario legítimo recibirá la llamada y el atacante actúa a modo de servidor Proxy. Se trataría entonces de un ejemplo claro de *eavesdropping*.

Además de la potente herramienta SiVus existen un conjunto de tres herramientas para manipular los aspectos del registro de usuarios en SIP:

Registration Hijacker:
http://www.hackingexposedvoip.com/tools/reghijacker.tar.gz

Registration Eraser:
http://www.hackingexposedvoip.com/tools/erase_registrations.tar.gz

Registration Adder:
http://www.hackingexposedvoip.com/tools/add_registrations.tar.gz

Desregistrar Usuarios

El desregistro de usuarios legítimos es una necesidad para conseguir suplantar su identidad como hemos visto en el ejemplo anterior. Básicamente el intruso podrá conseguirlo de las siguientes formas:

- Realizando un ataque de DoS al usuario.
- Generando una condición de carrera en la que el atacante envía repetidamente peticiones REGISTER en un corto espacio de tiempo con el objetivo de superponerse a la petición de registro legítimo del usuario.
- Desregistrando el usuario con mensajes REGISTER.

El intruso puede ser capaz de desregistrar fácilmente un usuario, enviando al servidor de registro una petición REGISTER (simulando ser la víctima) con el siguiente campo "`Contact: *`" y valor del atributo "`Expires`" a cero. Esta petición eliminará cualquier otro registro de la dirección del usuario (especificada en el campo "`To`" de la cabecera).

El atacante deberá realizar este envío periódicamente para evitar el re-registro del usuario legítimo o en su defecto provocarle una ataque DoS para evitar que vuelva a registrarse al menos por el tiempo que necesite para realizar el secuestro de la llamada.

Desconexión de Usuarios

El hecho de que muchos de los protocolos se utilizan sin encriptación alguna y de que los mensajes no se autentican de forma adecuada, es trivial para un intruso desconectar a los usuarios de sus llamadas enviando mensajes BYE con la identidad falsificada simulando ser el usuario del otro lado de la línea.

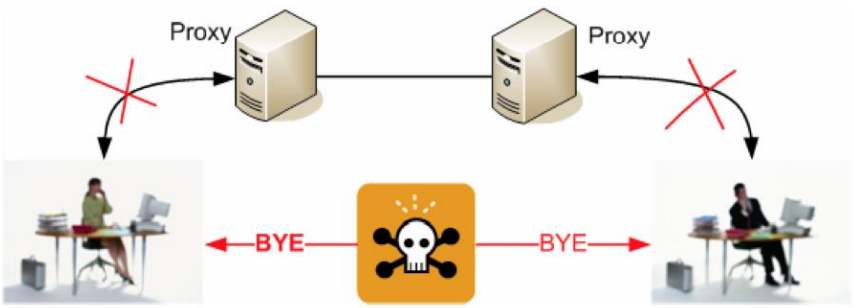

Figura 10. Desconexión de usuario

Se puede realizar un ataque similar utilizando mensajes CANCEL, pero sólo afectan cuando se está estableciendo la llamada, es decir, antes de que el destinatario descuelgue el teléfono.

Otro tipo de ataques consistirían en utilizar mensajes ICMP "port unreachalbe", menajes RESET del protocolo SCCP, o HANGUP para AIX.

Existen algunas herramientas para automatizar este tipo de ataques:

- *Teardown* – Injector de mensajes SIP,
- *sip-kill* – Inyecta mensajes BYE válidos en una sesión existente,
- *sip-proxykill* – Técnica similar, pero el objetivo son los servidores proxys.

Redirección de llamadas

La redirección de llamadas suele ser otro de los ataques comunes en las redes VoIP. Existen diferentes métodos que van desde comprometer los servidores o el call manager de la red para que redirijan las llamadas donde el intruso quiera, hasta las técnicas ya mostradas de suplantación de identidad en el registro, "*man in the middle*", etc.

Otra posibilidad es utilizar una herramienta como *RedirectPoison* que escucha la señalización SIP hasta encontrar una petición INVITE y responder rápidamente con un mensaje SIP de redirección, causando que el sistema envíe un nuevo INVITE a la localización especificado por el atacante.

Otro modo de redirección el flujo de datos se consigue con las herramientas como *sip-redirectrtp* y *rtpproxy*. Se basan en utilizar mensajes la cabecera SDP para cambiar la ruta de los paquetes RTP y dirigirlas a un *rtproxy*, que a su vez serán reenviados donde el intruso quiera.

3.7.3 Manipulación de la transmisión

Eavesdropping

La técnica de la interceptación de la comunicación o *eavesdropping* ya ha sido explicada por lo que en este caso veremos un ejemplo práctico de cómo se captura la señalización y el flujo de una llamada para después poder reproducir el contenido de la misma.

Los pasos para capturar y decodificar los paquetes de voz interceptados son realmente sencillos. En el primer ejemplo se utiliza un sniffer como *ethereal*. En algunos casos para poder esnifar el tráfico en redes conmutadas pueden ser necesarias técnicas

como el envenenamiento ARP, que básicamente consiste en realizar un "*man in the middle*" utilizando tramas ARP spoofeadas. Herramientas como *ettercap* y *arpspoof* también se utilizan en este caso.

- Capturar y decodificar los paquetes RTP. Esnifar el tráfico de la comunicación con el *ethereal*, este sniffer permite además interpretar los paquetes UDP indicándole que son del protocolo RTP.
- Seleccionar la opción "Analizar Sesión". Permite seleccionar un flujo de datos y analizarlo ya no como paquetes individuales sino común flujo continuo de datos.
- Salvar a un fichero de audio, para reproducirlo posteriormente. Ethereal permite analizar los datos RTP y salvarlos como un fichero de audio.

Se puede automatizar aun más el proceso si se utiliza la herramienta *Cain*. Que además de ser un buen sniffer, puede realizar infinidad de funciones y ataques. En el ejemplo que nos ocupa, con el propio *Cain*, se puede esnifar la comunicación VoIP, utiliza el envenenamiento ARP si fuera necesario, y además permita decodificar y reproducir los datos de voz capturados, todo en un mismo programa.

Además existen otras herramientas como: *Oreka*, *Orktrack* y *Orkweb*, *Voipong*, *Angst* y *Vomit*.

Inserción de Audio

En las llamadas VoIP la transmisión del flujo de datos se realiza por razones de sencillez y eficiencia sobre el protocolo UDP. Desgraciadamente UDP es un protocolo que no da garantías en la entrega de sus mensajes y no mantiene ningún tipo de información de estado o conexión. Por lo que a priori la inserción de paquetes UDP extraños dentro de un flujo legítimo puede llegar a ser trivial.

Encapsulado en UDP se encuentra el protocolo RTP que transporta verdaderamente los datos de voz. RTP tampoco lleva un control exhaustivo sobre el flujo de datos relegando las funciones de recuento de paquetes y calidad de servicio al protocolo RTCP. El único método que tiene RTP para controlar tramas perdidas y reordenar las que le llega es el campo número de secuencia de la cabecera.

En esta situación ¿Qué ocurriría, si a un dispositivo le llegan dos tramas UDP con el mismo número de secuencia (y diferentes datos)? ¿Descartaría la última que llega por estar repetida? ¿Y si la última es la trama legítima? En caso contrario ¿Sobrescribirían los datos de la segunda a la primera al reordenar y reensamblar?. Es evidente que la forma de manejar estas situaciones dependerán mucho del dispositivo o de la implementación del software, pero en cualquiera de los dos casos el atacante podría

realizar ataques de inserción de paquetes dentro de un flujo RTP consiguiendo insertar de forma exitosa audio en una conversación telefónica. Incluso se ha comprobado que contra algunos dispositivos es suficiente bombardear con paquetes UDP, para que esto se inserten en la conversación.

En este caso también existen algunas herramientas con las que se puede realizar este tipo de ataque:

- *RTP InsertSound* : Es capaz de insertar un archivo wav en una conversación activa que este esnifando.
- *RTP MixSound*: Muy parecida a la anterior pero mezcla el sonido insertado con el real de la conversación.

3.7.4 Fuzzing

Los ataques de *fuzzing* o también conocidos como testeo funcional del protocolo, es uno de los mejores métodos para encontrar errores y agujeros de seguridad. Consiste en crear paquetes o peticiones especialmente malformadas para ir más allá de las especificaciones del protocolo. El objetivo es comprobar como manejan los dispositivos, las aplicaciones o el propio sistema operativo que implementa el protocolo, estas situaciones anómalas que desgraciadamente no se han tenido en cuenta en la implementación y casi siempre terminan en un error, denegación de servicio o en alguna vulnerabilidad más grave.

Gracias a la técnica de *fuzzing* se han llegado a encontrar gran cantidad de ataques de DoS y buffer overflows en los productos que implementan los protocolos SIP y H.323.

Un ejemplo sencillo podría ser el siguiente:

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP 10.1.1.1:5060;branch=77ef4c2312983.1
Via: SIP/2.0/UDP 10.1.3.3:5060
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@10.1.3.3
CSeq: 314159 INVITE
Contact: <sip:alice@10.1.3.3>
Content-Type: application/sdp
Contact-Length: 142
(Carga SDP no mostrada)

INVITE sip:bob@biloxi.com SIP/2.0
Via:aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
aaaaaaaaaa
Via: SIP/2.0/UDP 10.1.3.3:5060
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@10.1.3.3
CSeq: 314159 INVITE
Contact: <sip:alice@10.1.3.3>
Content-Type: application/sdp
Contact-Length: 142
```

Como vemos el objetivo es provocar un desbordamiento de buffer y la consiguiente denegación de servicio en el dispositivo que procese la petición.

Existe una herramienta llamada PROTOS que se encarga de automatizar este tipo de ataques contra diversos protocolos como SIP, HTTP y SNMP. Otras herramientas son: *Ohrwurm*, *Fuzzy Packet*, *Asteroid*.

3.7.5 Ataques DoS

Las redes VoIP siguen siendo vulnerables a los tradicionales ataques de DoS como pueden ser los SYN flood, UDP flood etc. Las aplicaciones VoIP escuchan en ciertos puertos determinados, es posible atacar esos servicios causando un ataque DoS.

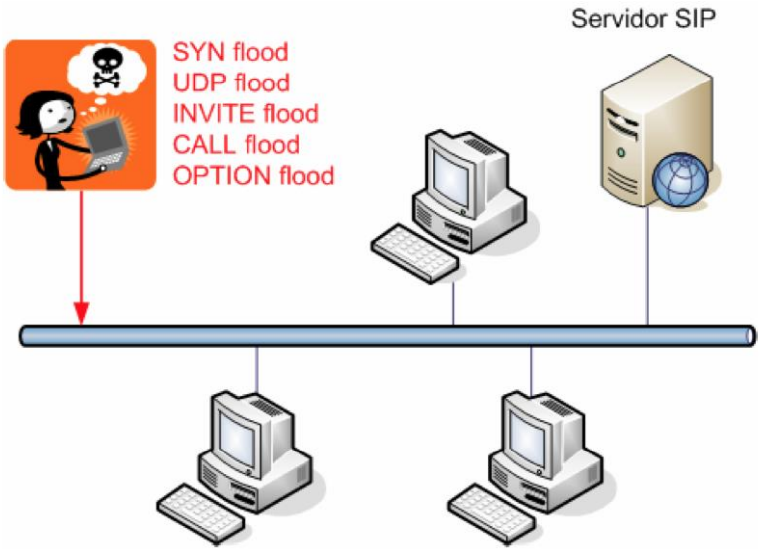

Figura 11. Ataque DoS

Existe gran cantidad de flooders disponibles en la red, se puede descargar y testear el *UDP flooder* de las siguiente dirección :

http://www.hackingexposedvoip.com/tools/udpflood.tar.gz.

O en cambio se puede utilizar algún generador de paquetes convencional como *Scapy*:

http://www.secdev.org/projects/scapy/.

VoIP presenta cierta dependencia del protocolo DNS por la necesidad de resolver los nombres de dominio. Un ataque a los servidores DNS de la red podría derivar en una denegación de servicio de la red VoIP. Una herramienta interesante para testear el servicio de resolución de nombres es *DNS Auditing tool* que se encuentra disponible en la dirección web : http://www.packetfactory.net/projects/dnsa.

Existen también ataques específicos de protocolos como SIP. El caso más común es intentar atacar a las capacidades de los servidores hasta conseguir que dejen de prestar el servicio.

En el siguiente ejemplo se puede ver como una inundación de peticiones INVITE a toda la red VoIP, falseando la identidad de llamante, provocaría que los teléfonos que no estén en uso comenzaran a sonar y si la inundación continua terminaría por colapsar las líneas y los servidores.

Otros ataques de inundación similares se pueden reproducir también con mensajes REGISTER, OPTIONS y CALL.

Algunas herramientas son *INVITE Flooder* y *RTP Flooder* .

Otro tipo de ataques son los llamados de "smurf" o de amplificación, consiste en identificar los procesos de red que responden con paquetes mucho mayores a los de la petición. De este modo, si el atacante falsifica la dirección origen, emitiendo paquetes pequeños e datos, las respuestas a esas peticiones serán mucho mayores en cuanto a tamaño y le llegaran a la víctima, con el único objetivo de realizar una denegación de servicio.

En general existe gran cantidad métodos diversos para sobrecargar la red y los servidores con el fin de conseguir una denegación de servicio. Problema que se agrava con el hecho que en una infraestructura IP pueden coexistir gran cantidad de protocolos (SIP, CMMS, H.225, H.245, RAS, MGCP, TGCP, NC, H.284, Megaco, SKINNY, SCCP, Q.931+, SIGTRAN, ISTOP, SS7, RUDP, RADIUS, COPS, RTP, RTCP) y dispositivos cada uno de ellos vulnerables de una forma diferente.

3.7.6 Ingeniería social

SPIT: Spam over Internet Telephony

El SPAM es uno de los problemas más graves en las comunicaciones hoy en día, y la telefonía IP tampoco se escapa. Recibe el nombre de SPIT (Spam over Internet Telephony).

A pesar de que hoy por hoy no es una práctica demasiado extendida y no se han registrados demasiados casos, las redes VoIP son inherentemente vulnerables al envío de "mensajes de voz basura". Siendo el impacto en la red VoIP mucho mayor que el SPAM tradicional.

Se prevé que esta tendencia de realizar llamadas y llenar los voicemail de los usuarios con mensajes pregrabados crecerá durante los próximos años a medida que se generalice el uso de telefonía por IP.

Vishing: Voip Phishing

Al igual que ocurría con el SPAM las amenazas de *phishing* suponen un gran problema para el correo electrónico. Las denuncias por robo de información confidencial de forma fraudulenta están a la orden del día y exactamente las mismas técnicas son aplicables a la plataforma VoIP. Gracias a la telefonía IP un intruso puede realizar llamadas desde cualquier lugar del mundo al teléfono IP de un empleado de la empresa y con técnicas de ingeniería social y mostrando la identidad falsa o suplantando otra conocida por la victima, obtener información confidencial, datos

personales, números de cuenta o cualquier otro tipo de información. Las opciones son prácticamente ilimitadas y al igual que el SPIT es posible que el número de incidentes de este tipo se disparen en los próximos años.

4 Seguridad en las redes VoIP

4.1 Requisitos de seguridad

El servicio de telefonía, independientemente de la tecnología con que se ofrezca, debe cumplir requisitos de seguridad básicos para los usuarios tales como confidencialidad, autenticación, integridad y disponibilidad. Aunque los requisitos son los mismos tanto si el servicio se presta por conmutación de circuitos como de paquetes, la naturaleza de cada tecnología afecta a la forma de cumplirlos.

4.1.1 Confidencialidad

El contenido de una conversación debe ser conocido únicamente por los interlocutores, sin que terceros puedan acceder al mismo. Aunque este sea el aspecto más evidente, no hay que olvidar la confidencialidad de los propios datos de la llamada (número llamante, número llamado, horas de inicio y finalización, etc.), que deben permanecer desconocidos para terceros.

En las redes de conmutación de circuitos, vulnerar la confidencialidad de una conversación requiere tener acceso físico al circuito que utiliza en algún punto del mismo ("pinchar la línea"). Sin embargo, en una red IP es factible acceder a los paquetes de voz en tránsito de forma remota, por lo que deben aplicarse técnicas de cifrado para proteger la confidencialidad. En los tramos más vulnerables de las redes de conmutación de circuitos, por ejemplo, en el interfaz de acceso inalámbrico de las redes móviles como GSM, se usa también cifrado.

Un requisito relacionado es el de *privacidad*, que incluye el control sobre qué datos personales de un usuario pueden revelarse o no a otro u otros usuarios, incluso aunque participen en la comunicación.

4.1.2 Autenticación

La autenticación es un requisito de seguridad consistente en que los participantes de una comunicación pueden estar seguros de la identidad del resto de participantes, evitando posibles suplantaciones. Ha de mencionar que éste es un requisito que no siempre se cumple en la red telefónica de conmutación de circuitos, por ejemplo, cuando se habla con una persona desconocida. Con interlocutores conocidos de antemano puede reconocerse a la otra persona por su voz, si la calidad de la

comunicación es suficiente para ello, pero este mecanismo no basta. Para asegurar la autenticación se pueden utilizar, por ejemplo, contraseñas, técnicas basadas en secretos compartidos o en claves públicas, o incluso características biométricas de los usuarios.

La autenticación, es decir, la verificación de la identidad, se completa, desde el punto de vista del proveedor del servicio, con comprobar la autorización para el uso de un determinado recurso o servicio y, en tercer lugar, con la contabilidad necesaria para cobrar al usuario como corresponda. Estas tres funciones se denominan AAA (Authentication, Authorization and Accounting). La prevención de fraudes por usos no autorizados de servicios, suplantaciones de usuarios legítimos, etc. Es obviamente un requisito de seguridad clave para el operador, aunque no afecte directamente a la seguridad de cada comunicación.

Otro requisito de seguridad ligado a la autenticación, que puede ser relevante en ciertos casos, es el conocido como *no repudio*, consistente en evitar que un interlocutor niegue luego haber participado en la comunicación.

4.1.3 Integridad

Una vez establecida la comunicación, hay que asegurar que el contenido generado por cada participante llegue sin modificaciones (maliciosas o accidentales) a los demás. En comunicaciones de voz es relativamente difícil modificar una conversación sin que el oyente lo perciba, aunque hay ataques específicos que afectan a este requisito. En cambio, en comunicaciones textuales (mensajería instantánea, correo electrónico, etc.) la modificación es mucho más fácil. Existen técnicas criptográficas que permiten detectar posibles cambios en un mensaje en tránsito y proteger así su integridad.

4.1.4 Disponibilidad

El requisito de disponibilidad consiste en que un usuario legítimo del servicio pueda acceder a él en todo momento. Al margen del mal funcionamiento de terminales, enlaces o nodos de la red debidos a ataques físicos o causas accidentales, la disponibilidad puede comprometerse generando una demanda elevada que llegue a saturar la capacidad de algún elemento de la red necesario para la comunicación o la disminuya sensiblemente.

Por ejemplo, en una red de conmutación de circuitos se podía generar de forma maliciosa un gran número de llamadas que ocupen todas las líneas disponibles. En las redes IP los ataques a la disponibilidad generando un gran número de paquetes, peticiones a un servidor, etc. son un problema grave y los denominados ataques de

denegación de servicio (DoS) están a la orden del día. El problema puede mitigarse en parte mediante técnicas de control que sólo permitan el acceso al servicio (a la red) a los usuarios legítimos, técnicas de limitación de tráfico, protección de los terminales para impedir que puedan utilizarse de forma remota para generar tráfico malicioso.

El servicio de voz es particularmente sensible a la degradación de parámetros de calidad de servicio como el retardo y la variación de retardo, que puede provocarse con ataques de denegación de servicio. En este sentido, el requisito de disponibilidad implica no sólo que los usuarios legítimos puedan acceder al servicio, sino también que el servicio ofrezca una calidad suficiente en todo momento.

Adicionalmente, los requisitos de calidad de servicio de la voz deben tenerse en cuenta a la hora de implementar mecanismos de seguridad para evitar que estos introduzcan un retardo excesivo. [9][10][11].

En general, puede decirse que las redes de voz sobre IP presentan mayores problemas para cumplir los requisitos de seguridad que las redes telefónicas tradicionales.

4.2 Protocolos de seguridad para VoIP

En este apartado se recogen varios protocolos de seguridad aplicables en redes de voz sobre IP basadas en el protocolo SIP:

- HTTP (Hyper Text Transport Protocol) Digest, para autenticación de agentes de usuario en el proxy SIP al solicitar un inicio de sesión.
- TLS (Transport Layer Security), para permitir confidencialidad e integridad en el intercambio de información entre el agente de usuario y el proxy, así como entre proxies (seguridad salto a salto).
- S/MIME (Secure Multipurpose Internet Mail Extensions), para permitir confidencialidad e integridad extremo a extremo de los mensajes SIP.
- IPsec (Internet Protocol Security), para proporcionar autenticación, confidencialidad e integridad a nivel IP.
- SRTP (Secure Real Time Protocol), para proporcionar autenticación, confidencialidad e integridad en el flujo de audio intercambiado entre los extremos.
- DIAMETER, para proporcionar control de acceso y otros aspectos relacionados con AAA.

Se trata de protocolos generales que implementan mecanismos de gestión de claves, cifrado, etc. y que pueden utilizarse tanto en voz sobre IP como en otros escenarios. En

el caso de voz sobre IP, se usan para proteger el transporte de señalización (mensajes SIP, mensajes de señalización telefónica sobre SCTP, mensajes de control de pasarelas) y el transporte de la voz (en paquetes RTP). En algunos casos, su utilización en VoIP puede imponer requisitos específicos, por ejemplo el uso de determinados algoritmos criptográficos.

4.2.1 HTTP Digest

El protocolo SIP proporciona un mecanismo de autenticación de usuarios tomado de HTTP (Hyper Text Transport Protocol) mediante la utilización de las cabeceras WWW Authenticate, Authorization, Proxy-Authenticate y Proxy- Authorization. Este mecanismo, denominado HTTP digest, ver RFC2617, se basa en secreto compartido y en técnicas de reto/respuesta para poder comprobar el conocimiento del secreto compartido sin tener que enviarlo explícitamente.

La figura siguiente muestra un ejemplo esquemático de autenticación del llamante durante el establecimiento de una llamada de voz sobre IP con SIP. La respuesta 407 contiene el reto o "pregunta" que la red formula al cliente. Este devuelve un ACK, calcula la "respuesta" correcta usando su clave secreta, conocida sólo por él y por la red, y la devuelve en un nuevo INVITE. El servidor proxy comprueba que la respuesta es la esperada antes de continuar con la llamada. La autenticación puede hacerse también directamente entre el servidor del usuario llamado y el cliente llamante.

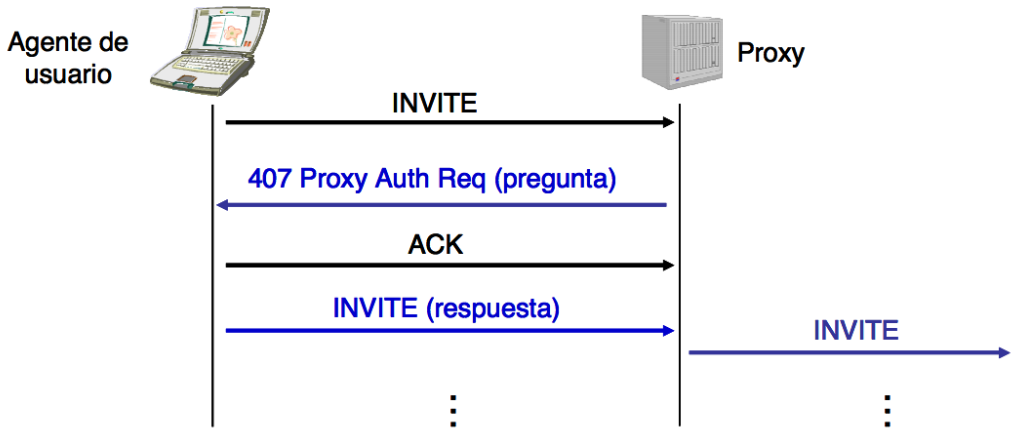

Figura 12. Ejemplo de autenticación con SIP

Este procedimiento de reto/respuesta usando una clave secreta compartida de antemano entre el usuario y la red procedimiento es conceptualmente el mismo que el utilizado en otras redes, por ejemplo GSM. Se trata de un procedimiento relativamente sencillo y robusto frente a ataques de denegación de servicio, ya que los servidores que ofrecen retos pueden permanecer sin almacenar información de estado hasta que la

nueva petición llegue, de modo que no se bloquean recursos durante largos periodos de tiempo.

Sin embargo, la autenticación descrita aplica solo a la comunicación entre un agente de usuario y el primer servidor (o entre dos agentes de usuario), pero no a la comunicación de un servidor al siguiente en los tramos intermedios de la llamada, ni al tramo final desde el servidor destino al agente de usuario del otro extremo. Para estos tramos la autenticación debe hacerse en niveles por debajo de SIP, por ejemplo, mediante TLS o IPsec.

La RFC3261 define la utilización de HTTP Digest en SIP, así como la de otros mecanismos como TLS y S/MIME que se detallan en subsecciones siguientes.

4.2.2 Transport Layer Security (TLS)

Al igual que el mecanismo anterior de autenticación, tomado de HTTP, para resolver el problema de la confidencialidad y la integridad puede recurrirse nuevamente a soluciones ya presentes en servicios de Internet. El protocolo TLS (Transport Layer Security) del IETF, basado en SSL (Secure Sockets Layer) estándar de facto diseñado en los años 90 para las comunicaciones web seguras, ofrece autenticación, integridad y confidencialidad sobre conexiones TCP.

La autenticación se basa en la validación de certificados de clave pública del servidor y, opcionalmente, del cliente, aunque dicha validación no es hecha directamente por TLS, sino que se considera responsabilidad del protocolo situado por encima de TLS. La confidencialidad e integridad de la información se protege mediante algoritmos de cifrado simétrico (con una clave de sesión que se calcula tras la fase de autenticación) y funciones resumen.

El nivel de seguridad que ofrece TLS es muy variable dependiendo de los algoritmos criptográficos y longitudes que se utilicen. Por ejemplo, es obligatorio implementar los algoritmos RSA (firma digital), 3DES (cifrado simétrico) y SHA1 ("hash" o resumen) que ofrecen un nivel de seguridad elevado, ver detalles en RFC4346, pero se admiten múltiples combinaciones negociables entre los interlocutores que pueden tener un nivel de seguridad menor.

En VoIP, TLS puede utilizarse como una de las alternativas de transporte de los mensajes de señalización SIP entre los diferentes elementos de la arquitectura: agentes de usuario, registros, servidores proxy, etc. Cuando transporta SIP, la RFC3261 especifica que TLS debe implementar el algoritmo de cifrado AES, además de 3DES u otros.

Adicionalmente, en algunos casos puede interesar el transporte seguro de medios en el plano de usuario usando TLS. Para ello, se utiliza la extensión de SDP definida en RFC4572. (El transporte de medios orientado a conexión directamente sobre TCP estaba ya contemplado en SDP según la RFC4145).

En principio, la aplicación de TLS en SIP es salto a salto, no extremo a extremo entre los usuarios, ya que los proxies y elementos intermedios de la arquitectura SIP han de tener acceso a los mensajes de señalización en claro para poder desarrollar sus funciones. El usuario llamante (el que envía la petición de sesión), puede establecer una comunicación TLS con el proxy de SIP, pero en principio no tendría garantías de que el resto de saltos van a ser seguros. Para ello se han definido URIs de SIP seguros (SIPS), al estilo de HTTPS, garantizando así que todos los saltos dentro de la red se harán utilizando TLS, al menos hasta llegar al proxy del dominio destino del usuario llamado.

Como se ha dicho, TLS se deriva del protocolo SSL propuesto por Netscape (SSL versión 3 de 1996). TLS versión 1.0 se publicó en 1999 (RFC2246), seguida por varias RFCs sobre uso de TLS en el periodo1999-2005. En 2006 se definió TLS versión 1.1 (RFC4346) y extensiones (RFC4366). TLS 1.2 fue definido en 2008 (RFC5246), en esta nueva versión la combinación MD5-SHA-1 en la función pseudoaleatoria (PRF) y en el mensaje fue reemplazada por SHA-256. incluirá, entre otros, cambios en algoritmos de resumen como consecuencia de los ataques publicados en los últimos años contra los algoritmos MD5 y SHA1. TLS 1.2 fue después redefinido en el RFC6176 de marzo de 2011 redactando su retro compatibilidad con SSL y TLS para que dichas sesiones jamás negocien el uso de SSL versión 2.0.

TLS, al igual que SSL, fue definido inicialmente para funcionar sobre un protocolo de transporte fiable como TCP. Posteriormente, la RFC3436 definió el uso de TLS sobre Stream Control Transmission Protocol (SCTP), que también es orientado a conexión y fiable, aunque sin poder aprovechar algunas funciones nuevas del protocolo SCTP de las que carece TCP.

Más recientemente, se ha definido Datagram Transport Layer Security (DTLS) versión 1.0, similar a TLS pero capaz de funcionar sobre transporte no orientado a conexión (UDP), ver RFC4347, o bien sobre SCTP pero sin las limitaciones de TLS sobre SCTP.

Como resumen, la figura siguiente presenta varias opciones para transporte del protocolo SIP, usando seguridad a nivel de transporte, con TLS o DTLS, o bien seguridad a nivel de red con IPsec.

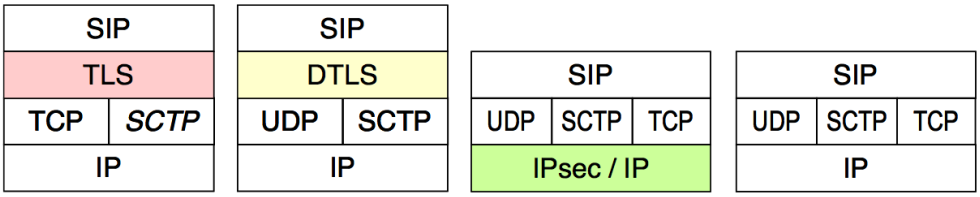

Figura 13. Opciones para transporte del protocolo SIP

4.2.3 Secure Multipurpose Internet Mail Extensions (S/MIME)

En la subsección anterior se ha mencionado que los proxies necesitan acceder al contenido de los mensajes de señalización SIP, impidiendo el uso de una solución completa de seguridad extremo a extremo. Esta necesidad se limita solo a algunos datos que figuran en la cabecera SIP, por ejemplo los relativos al destino de la llamada y los proxies intermedios. Otros campos incluidos en los mensajes SIP intercambiados a través de proxies entre el origen y el destino contienen datos que sólo incumben a los extremos de la comunicación.

Con el fin de proteger selectivamente los mensajes, manteniendo confidenciales los datos entre los extremos pero permitiendo al tiempo que los proxies accedan a los campos que necesitan, se utiliza otra solución de seguridad proveniente del mundo Internet, en este caso S/MIME. MIME (Multipurpose Internet Mail Extensions) es un esquema de codificación diseñado con el fin de poder adjuntar a los correos electrónicos informaciones de distinto tipo de forma homogénea, pero sirve igualmente para codificar los cuerpos de los mensajes SIP. S/MIME (Secure MIME), ver RFC3850 y RFC3851, añade la capacidad de poder firmar y cifrar partes de un mensaje para proteger su integridad y confidencialidad.

En el caso de SIP, S/MIME aparece como un mecanismo opcional, que puede aplicarse al cuerpo de los mensajes solamente o también a su cabecera, según se especifica en la definición de SIP (RFC3261). Adicionalmente la RFC3853 modifica los algoritmos a implementar en S/MIME cuando se usa con SIP, estableciendo el uso de RSA, AES y SHA1 (es decir, los mismos que en el caso de TLS con SIP).

El mecanismo que propone RFC3261 para proteger la cabecera de los mensajes SIP con S/MIME se basa en hacer una copia de la cabecera que se firma digitalmente y se añade al cuerpo del mensaje SIP. De esta forma los mensajes llevan dos copias de la cabecera: la original y la copia firmada que va en el cuerpo. Este método, conocido como "túnel SIP", crea problemas debido a que algunos campos de la cabecera original pueden ser legítimamente modificados por servidores intermedios, por lo que el

receptor de un mensaje debe distinguir estos cambios en la cabecera de otros que pueden tener origen malicioso.

Como mejora, la RFC3893 define un mecanismo denominado Authenticated Identity Body (AIB) que permite proteger selectivamente solo algunos campos de la cabecera.

Por otra parte, la RFC4189 define un mecanismo de seguridad intermedio entre el salto a salto (caso de TLS) y el extremo a extremo (caso de S/MIME) que se denomina seguridad "end to middle". Este mecanismo es útil cuando en el recorrido que sigue un mensaje SIP intervienen servidores con diferentes niveles de confianza para el usuario. Por ejemplo, puede haber servidores a los que se les muestra solo la información necesaria para encaminar los mensajes y otros servidores a los que se muestra más información.

4.2.4 IP Security (IPsec)

IPsec es un conjunto de protocolos cuya función es asegurar las comunicaciones sobre el Protocolo de Internet (IP) autenticando y/o cifrando cada paquete IP en un flujo de datos. IPsec también incluye protocolos para el establecimiento de claves de cifrado.

IPsec proporciona funciones de autenticación, integridad y confidencialidad en la capa IP, para lo cuál se basa en tres procedimientos:

- AH (Authentication Header), para ofrecer el servicio de autenticación e integridad de paquetes (RFC4302).
- ESP (Encapsulating Security Payload), para proporcionar confidencialidad y opcionalmente servicio de autenticación e integridad (RFC4303).
- IKE (Internet Key Exchange), utilizado para establecer una "asociación de seguridad" que define el conjunto de parámetros de seguridad que se van a usar en la comunicación entre dos entidades IPsec: funciones de seguridad a utilizar, algoritmos, claves, tiempo de vida de la asociación (RFC4306).

IPsec puede utilizarse en modo transporte o en modo túnel, según se indique al establecer la asociación de seguridad. En el primero, IPsec se implementa en los extremos de la comunicación. Los paquetes llevan la cabecera IP original con las direcciones origen y destino, seguida de una cabecera ESP, el contenido cifrado del paquete y una cola con datos de autenticación. En modo túnel, IPsec se aplica entre dos puntos intermedios de la comunicación, entre los cuales se establece un tramo de comunicación segura. En este caso, el paquete original completo, con cabecera IP y

contenido, va cifrado y rodeado por la cabecera ESP y los datos de autenticación, todo ello precedido por una nueva cabecera IP que especifica como direcciones origen y destino los puntos intermedios que delimitan el túnel seguro. IPsec es un protocolo general utilizable para el transporte seguro de todo tipo de paquetes. En el caso de VoIP, puede usarse para transportar SIP, SCTP y MEGACO/H.248, como se ha comentado en apartados anteriores, o incluso paquetes de voz RTP, ver apartado siguiente.

4.2.5 Secure Real Time Protocol (SRTP)

Secure RTP, ver RFC3711, permite proteger los mensajes de los protocolos RTP (Real Time Protocol) y RTCP (Real Time Control Protocol) presentados previamente. En particular, SRTP protege:

- la confidencialidad del cuerpo de los paquetes RTP (que a su vez contiene la información de usuario, por ejemplo, voz) y de los paquetes RTCP,
- la integridad de los paquetes RTP y RTCP completos, proporcionando además protección frente a ataques de reenvío.

SRTP no cifra las cabeceras de los paquetes RTP, con lo cual no impide la aplicación de técnicas de compresión de cabeceras que son muy útiles para reducir la sobrecarga que sufren los paquetes de voz, especialmente sensibles a este problema debido a que cada paquete transporta un número pequeño de octetos de voz. Si se considera que la información de las cabeceras debe ser confidencial, en lugar de SRTP se puede usar RTP con otros protocolos de seguridad, por ejemplo IPsec.

Al igual que IPsec, SRTP requiere el establecimiento de un "contexto criptográfico" entre las entidades que se comunican y para ello utiliza un procedimiento de gestión de claves externo. Una posibilidad es el protocolo MIKEY (Multimedia Internet KEYing), descrito en la RFC3830. MIKEY es un protocolo de gestión de claves con baja latencia y diseñado específicamente para ser usado en escenarios de comunicación multimedia entre dos usuarios o entre un grupo reducido de ellos que usan SRTP.

Otra opción es utilizar el nuevo atributo de SDP denominado "crypto", ver RFC4568, definido para negociar parámetros criptográficos para flujos de medios en general, y en particular para el caso de SRTP.

4.2.6 DIAMETER

El protocolo DIAMETER fue desarrollado por el IETF para proporcionar un entorno de funciones AAA (Authentication, Authorization and Accounting) para aplicaciones

de acceso remoto a redes o movilidad IP, mejorando en los aspectos de rendimiento, seguridad, escalabilidad y flexibilidad frente al actualmente extendido RADIUS (Remote Authentication Dial-In User Service). La primera versión de Diameter Base Protocol se publicó en RFC3588 en 2003 y fue después redefinida en el RFC6733 de octubre de 2012. DIAMETER también ofrece ventajas frente a otros protocolos que pueden utilizarse para control de acceso, como por ejemplo COPS (Common Open Policy Service). COPS es una solución sencilla, pero que carece de la flexibilidad y extensibilidad de DIAMETER.

El protocolo DIAMETER proporciona las siguientes facilidades:

- Autenticación de usuarios y negociación de capacidades
- Entrega de pares atributo-valor (AVPs)
- Notificación de errores
- Extensibilidad, mediante la incorporación de nuevos comandos y pares atributo-valor
- Servicios básicos de tarificación

Las sesiones típicas en DIAMETER consisten en el intercambio de pares atributo-valor entre los clientes y servidores. Parte de estos atributos son propios de DIAMETER, pero otros pueden ser definidos por la propia aplicación para realizar el control de acceso en función de otros parámetros o condiciones, de ahí las ventajas en cuanto a extensibilidad del protocolo.

La aplicación de DIAMETER con SIP está descrita en la RFC4740. Mediante un cliente DIAMETER un servidor SIP puede acceder a un servidor DIAMETER para obtener información que permita autenticar a los usuarios y autorizar el uso de recursos.

Los mensajes DIAMETER se transportan sobre TCP o SCTP, utilizando TLS o IPsec como protocolo de seguridad.

4.3 Mecanismos de detección de intrusiones

Existen numerosos métodos propuestos para detectar ataques contra ordenadores y redes, que pueden implementarse en sistemas de detección de intrusiones (IDS) y sistemas de prevención de intrusiones (IPS) [30]. Algunos autores usan también la denominación "sistemas de respuesta ante intrusiones". Estos sistemas suelen clasificarse en dos tipos, según analicen el comportamiento de un ordenador (host based) o el tráfico en una red (network based). Vamos a ver con más detalle el segundo tipo y, en particular, en el caso de redes VoIP.

El objetivo de los IDS es detectar los ataques de seguridad y generar las correspondientes alertas, mientras que los IPS añaden la posibilidad de tomar acciones de respuesta que eviten el ataque o minimicen sus efectos, por ejemplo bloquear el tráfico considerado atacante. Además de IDS e IPS, existen otros sistemas de seguridad que pueden utilizarse, como cortafuegos (firewalls) y señuelos (honeypots o honeynets). Un cortafuegos se configura de antemano para dejar pasar ciertos tipos de tráfico y bloquear otros. En cambio, un IPS deja pasar todo el tráfico y solo bloquea aquel que en un momento dado se considera atacante. El IPS puede bloquear el tráfico por sí mismo o actuando sobre un cortafuegos para cambiar su configuración. Un señuelo o honeypot puede verse como un mecanismo para engañar a los atacantes y estudiar su comportamiento, o también como un mecanismo de respuesta que se despliega para desviar determinados ataques. Honeypots en VoIP se describen en la subsección siguiente.

4.3.1 Honeypots en VoIP

En la terminología de informática un honeypot es una trampa para detectar, desviar, o de alguna manera contrarrestar intentos de uso no autorizado de la información del sistema. Por lo general, se compone de un ordenador, datos, o una página web que parece ser parte de una red, pero en realidad está aislado y controlado, y que se construye para que parezca contener cierta información o un recurso que podría interesar a un atacante .

Sistemas honeypot son servidores o sistemas señuelos destinados a recopilar información acerca de un atacante o un intruso. Es importante recordar que los honeypots siendo sólo un nivel o un sistema adicional, no sustituyen a otros sistemas tradicionales de seguridad en Internet.

Honeypots se pueden configurar dentro, fuera o en la zona desmilitarizada (DMZ) (Fig. 15.) de firewall o incluso en todos los lugares a pesar de que más a menudo se despliegan en el interior de firewall con fines de control [12]. En cierto sentido, no son más que una variación de sistemas de detección de intrusos (IDS), a pesar de que están más centrados en la recopilación de información y el engaño.

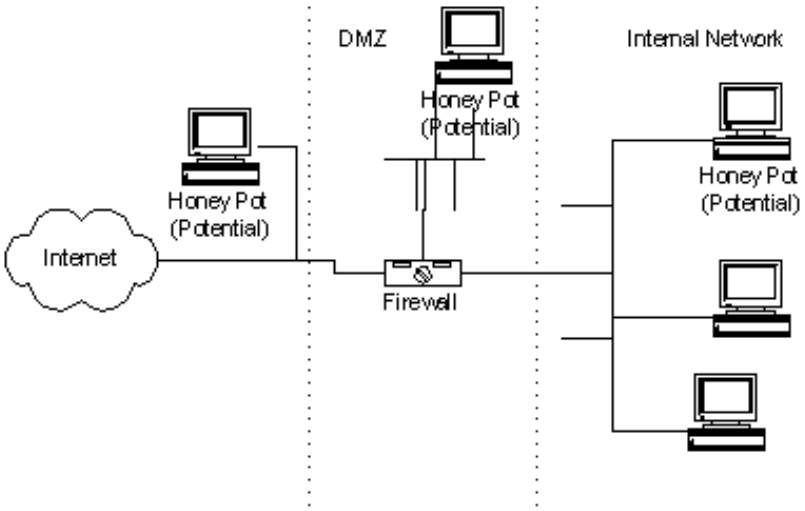

Figura 14. Ejemplo de sistema honeypot

Un sistema honeypot se hace como una presa más fácil para los intrusos que los sistemas reales de producción, pero con menores modificaciones del sistema de modo que su actividad puede estar registrada y rastreada. Se supone que una vez que un intruso irrumpe en un sistema, va a volver más. Durante estas visitas posteriores, honeypots almacenan información adicional y monitorean intentos adicionales de seguridad, archivos o sistemas de acceso.

Hay dos razones populares o los objetivos detrás de la creación de un honeypot:

1. Estudia cómo intrusos prueban y tratan de obtener acceso a los sistemas. La idea general es que, dado que se mantiene un registro de las actividades del intruso, podemos aprender sus métodos de ataque para proteger mejor nuestros sistemas de producción reales.

2. Recopila la información necesaria para ayudar en la detención o el enjuiciamiento de los intrusos. Este tipo de información, si es necesario, puede ser proporcionada a los funcionarios encargados de hacer cumplir la ley con los detalles necesarios para procesar al atacante.

La línea de pensamiento común en la creación de sistemas honeypot es que es aceptable el uso de la mentira o el engaño cuando se trata de intrusos. Esto significa que a la hora de crear un honeypot hay que tener en cuenta ciertos objetivos:

1. El sistema honeypot debe parecer lo más genérico posible. Debería parecer al intruso potencial como si el sistema no hubiera sido modificado, si no, él puede desconectarse antes de que la información necesaria haya sido recopilada.

2. Tenemos que tener cuidado en el hecho de qué tráfico el intruso puede enviar de vuelta a Internet, ya que no queremos convertirnos en un punto de lanzamiento para nuevos ataques contra otras entidades de Internet. Es una de las razones por que honeypots se establecen a menudo dentro de los cortafuegos.
3. Tendremos que hacer nuestro honeypot como una web interesante, colocando alguna información interesante o hacer que parezca un servidor importante para un intruso. Es necesario para el servidor de honeypot que parezca legítimo para que los intrusos puedan pasar bastante tiempo investigando el sistema para que la mayor cantidad posible de información se recopile.

4.3.2 Clasificación de honeypots

Los honeypots pueden clasificarse en función de su despliegue y en función de su nivel de participación. Basado en el despliegue, honeypots pueden ser clasificados como: *honeypots de producción* y *honeypots de producción*.

Honeypots de producción son fáciles de usar. Capturan sólo información limitada y son utilizados principalmente por empresas o corporaciones. Honeypots de producción se colocan dentro de la red de producción con otros servidores de producción por una organización para mejorar su estado general de la seguridad. Por lo general, para honeypots de producción se utilizan honeypots de baja interacción. Dan menos información sobre los ataques o atacantes que honeypots de investigación.

Honeypots de investigación se implementan para recopilar información sobre motivos y tácticas del atacante. Estos honeypots no son utilizados por una organización específica, sino que se utilizan para investigar las amenazas que enfrentan las organizaciones y para aprender a protegerse mejor contra estas amenazas. Honeypots de investigación son complejos de implementar y mantener, capturan información extensa. Se utilizan sobre todo por organizaciones de investigación, militares o del gobierno.

En base a los criterios de diseño, honeypots pueden ser clasificados como:

- honeypots puros,
- honeypots de alta interacción,
- honeypots de baja interacción.

Honeypots puros son los sistemas de producción plenos. Las actividades del atacante se monitorean usando un casual tap que se instala en el enlace de honeypot a la red. Ningún otro software debe ser instalado. A pesar de que un honeypot puro es

útil, el sigilo de los mecanismos de defensa puede ser garantizada por un mecanismo más controlado.

De acuerdo con investigaciones recientes en la tecnología de honeypots de alta interacción, mediante del empleo de máquinas virtuales, varios honeypots pueden ser alojados en una única máquina física [13]. Por lo tanto, incluso si el honeypot está comprometido, puede ser restaurado más rápidamente. En general, honeypots de alta interacción proporcionan más seguridad por ser difíciles de detectar, pero son muy caros de mantener. Si las máquinas virtuales no están disponibles, un honeypot se debe mantener por cada equipo físico, lo que puede ser bastante caro.

Honeypots de baja interacción simulan sólo los servicios solicitados con frecuencia por los atacantes. Dado que consumen relativamente pocos recursos, varias máquinas virtuales pueden ser fácilmente alojadas en un único sistema físico. Estos sistemas virtuales tienen un tiempo de respuesta corto y son mucho menos complejos en comparación con honeypots de alta interacción.

También hay un tipo de honeypots llamados honeypots cliente o honeyclients [14]. Honeypots cliente son mecanismos activos de seguridad que buscan servidores maliciosos que atacan a los clientes. El honeypot cliente se camufla como cliente e interactúa con el servidor para examinar si se produce un ataque. A menudo el enfoque de honeyclients es en navegadores web, pero cualquier cliente que interactúa con los servidores puede ser parte de un honeyclient.

Un honeyclient se compone de tres componentes. El primer componente, un queuer, es responsable de crear una lista de servidores que el cliente visita. Esta lista se puede crear, por ejemplo, a través de rastreo. El segundo componente es el propio cliente, que es capaz de hacer solicitudes a los servidores identificados por el queuer. Después de la interacción con el servidor que ha tenido lugar, el tercer componente, un motor de análisis, es responsable de determinar si el ataque ha tenido lugar en el honeyclient.

Además de estos componentes, los honeyclients suelen estar equipados con algún tipo de estrategia de contención para prevenir ataques con éxito que se propaguen más allá del honeypot cliente. Esto se logra generalmente mediante el uso de cortafuegos y sandboxes de máquina virtual.

Honeyclients como honeypots tradicionales se clasifican principalmente por su nivel de interacción: alta o baja. También hay enfoques nuevos híbridos que denotan el uso de ambas técnicas de detección de alta y baja interacción.

4.3.3 Arquitectura de honeypot

Como ya se ha mencionado hay muchos tipos diferentes de honeypots y cada uno tiene sus propios métodos. Sin embargo, típicamente un honeypot consta de seis componentes principales, que se describen a continuación. El esquema de honeypot que representa estos componentes y sus relaciones se puede ver en la figura 16.

- El agente honeypot. Es el núcleo de la arquitectura de honeypot y la parte inteligente de la aplicación. Es responsable de aceptar llamadas entrantes y de investigar posibles ataques.

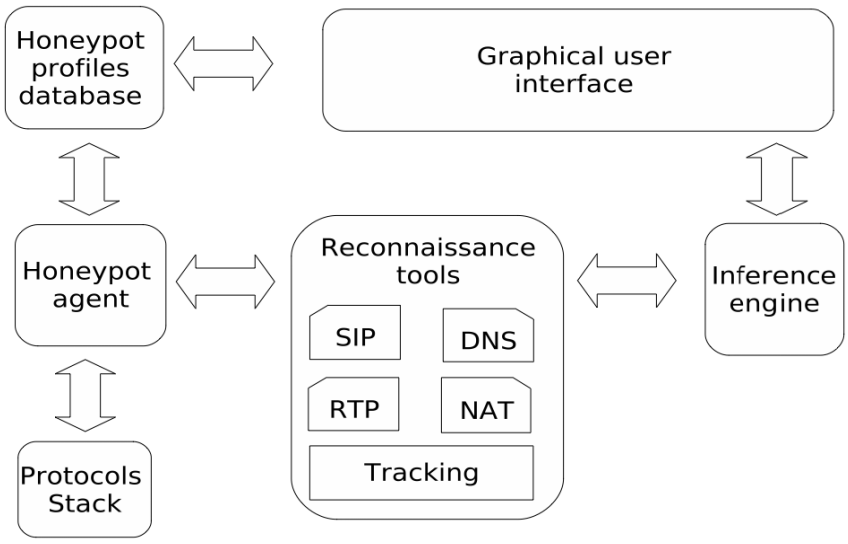

Figura 15. Arquitectura de honeypot

- La pila de protocolos (SIP, SDP, RTP). Se construyen sobre la base de estándares de protocolos y son responsables de construir y analizar los mensajes, así como la transmisión y recepción de los paquetes a través de la capa de transporte.

- La base de datos de perfiles de honeypot. La base de datos contiene varios archivos de configuración y por lo tanto permite al administrador elegir un perfil que sea adecuado a sus necesidades en lugar de construir por sí mismo. La ventaja de este componente es, en primer lugar, establecer el honeypot en su entorno, y en segundo lugar, controlar su comportamiento. Los perfiles en la base de datos pueden estar lejos de la zona de honeypot. Pueden ser bots que evalúan la eficacia o la seguridad del dominio.

- Las herramientas de reconocimiento. Se utilizan en el procedimiento de investigación para comprobar los mensajes recibidos.

- El motor de inferencia. Este componente es capaz de interpretar de forma automática los resultados de la investigación a través de métricas especiales y el modelo de Bayes que a menudo se elige por su capacidad de adaptación, escalabilidad y rendimiento en tiempo real.

- La interfaz gráfica de usuario (Graphical User Interface, GUI). Permite al administrador elegir y configurar un perfil honeypot, así como visualizar los rastros, las alertas y las estadísticas.

Un perfil está formado por dos conjuntos independientes de parámetros. Los parámetros del entorno se utilizan para configurar el honeypot en su entorno. Se parecen a los parámetros de configuración del soft-phone. Determinan dos tipos de configuración:

1. Host y configuraciones de la identidad de honeypot, tales como la dirección IP, puertos de protocolo que se utilizan y el contacto o la lista de contactos.

2. Dependencias de la red que incluyen el servidor de registro, el servidor DNS, SIP proxy y el servidor de correo de voz. Los parámetros de comportamiento permiten controlar el comportamiento de honeypot. El honeypot puede estar en uno de dos modos de funcionamiento, tales como determinado o aleatorio. El modo determinado se basa en configuraciones fijos, mientras que el modo aleatorio se basa en distribuciones aleatorias matemáticas.

La fase de preparación es el proceso de transformación del archivo de configuración de comportamiento en una máquina de estado ejecutable y que está completamente automatizada. Todos los operadores y los eventos se precodifican en clases apropiadas y se documentan para que puedan ser utilizados fácilmente. Una forma más amigable para configurar diferentes requisitos de comportamiento puede ser a través de la interfaz gráfica. La GUI pide al administrador rellenar un formulario de revisión y proporciona un archivo adecuado para la etapa preparatoria.

El procedimiento de investigación de un mensaje INVITE recibido se añade en la máquina honeypot en la cláusula Idle como una sentencia INVITE. Las direcciones IP y puertos de protocolo que participan en el inicio de sesión forman una lista de los interrogatorios. Para cada dirección IP o nombre de host en la lista antes citada el procedimiento de investigación tiene como objetivo evaluar los siguientes campos:

- Determinación de la propiedad y la información del mundo real a través de registros de Internet. En la base de datos de honeypot WHOIS asigna a cada entrada un coeficiente de confianza entre 0 y 1.

- Comprobación de la información de DNS a través de DNS, DNS inversa, DNS SRV o solicitudes ENUM, según el caso.
- Detección de números autónomos donde residen la dirección IP o host. En la base de datos de honeypot un AS asigna a cada entrada un coeficiente de confianza entre 0 y 1.
- Determinación de la ubicación geográfica de IP o host. En la base de datos de honeypot se asigna a cada ciudad un coeficiente de confianza entre 0 y 1.
- Averiguar si un puerto (SIP o RTP) proclamado ser abierto es realmente abierto.
- Obtención de la huella digital que normalmente se encuentra en la cabecera User-Agent. Sin embargo, el dispositivo del llamante puede tomar huellas digitales activamente a través de una secuencia de solicitudes OPTIONS. A cada dispositivo se asigna un coeficiente de confianza entre 0 y 1.
- Verificando la frecuencia con la que IP o host ha sido visto por el honeypot. La frecuencia máxima es 1 y se alcanza si el IP o host ha sido visto 10 veces o más.

Además, el procedimiento de investigación permite trazar rutas IP a partir de honeypot y llegar a diferentes hosts y servidores proxy mencionados en INVITE. Los resultados del procedimiento de investigación se llevan hacia el motor de inferencia.

El mero hecho de recibir un mensaje en el honeypot es un objeto de sospecha. Al mismo tiempo esto podría ser el resultado de un error no intencional de un usuario inocente.

4.3.4 Artemisa

En este subsección la arquitectura de honeypot y sus características se describen en un ejemplo de código abierto Artemisa honeypot. Artemisa simula un dispositivo de punto extremo activo. Un honeypot Artemisa se puede implementar en cualquier infraestructura de VoIP que utiliza el protocolo SIP [15]. En esta infraestructura hace el papel de un teléfono SIP regular [16] (fig. 17).

El honeypot puede ejecutarse en una máquina virtual o física. Se conecta al proxy SIP camuflado de una cuenta de usuario real con el fin de establecer un mejor enmascaramiento contra los atacantes potenciales.

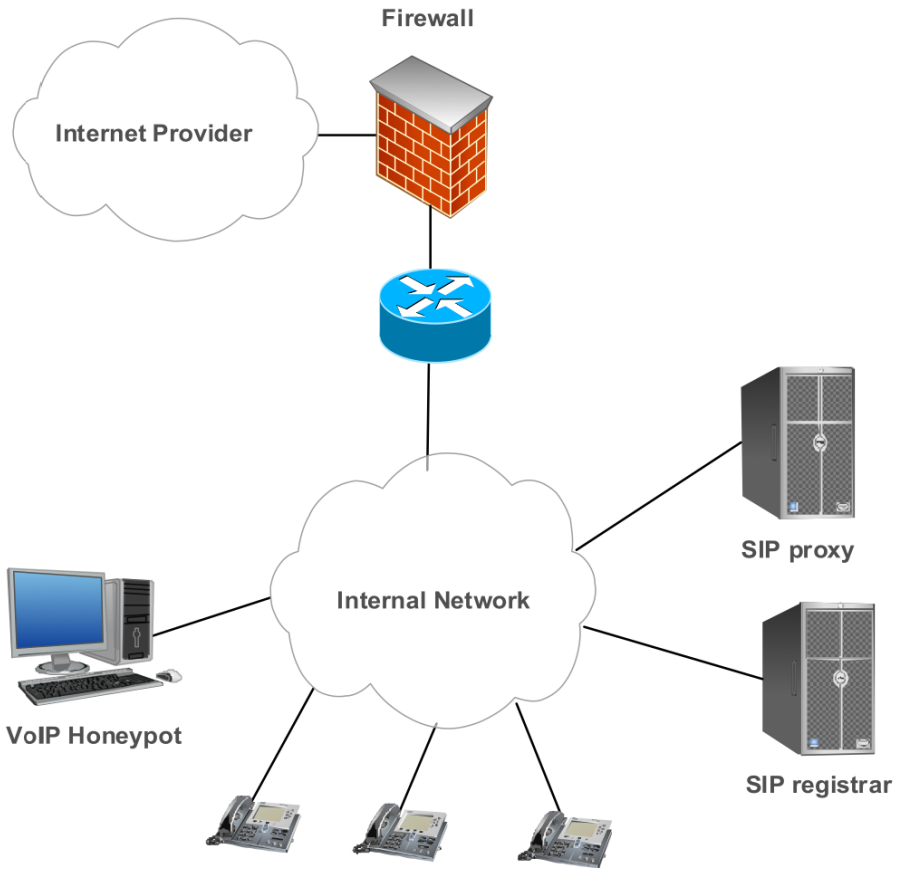

Figura 16. Topología de VoIP con honeypot

Artemisa se compone de varios módulos agrupados [15]. Estos módulos y sus relaciones se puede ver en la figura. 18:

1. Módulo Tcpdump controla la colección de tráfico de la red prima en la máquina de honeypot utilizando la herramienta Tcpdump y el módulo de grabación de llamadas, que utiliza la biblioteca PIMedia para salvaguardar los flujos de audio recibidos en formatos adecuados.

2. PJSIP User Agent, PJSIP y PJMEDIA stacks registran extensiones virtuales, llamadas contestadas, responden a las peticiones SIP y anuncian las huellas virtuales. También controlan otros componentes basados en el modo de comportamiento.

3. Herramientas activas. La recolección de información para recopilar información complementaria sobre el origen de los mensajes recibidos.

4. Clasificación basada en reglas es responsable de la interpretación de los datos obtenidos por los instrumentos de reconocimiento y también genera conclusiones sobre la base de reglas a partir de un árbol de decisión.

5. Reglas de huellas digitales se aplican cuando se realiza una operación para comprobar las huellas digitales. Una regla de toma de huellas digitales busca una expresión regular en la cabecera de SIP especificada o atributo, o en todo el mensaje.

6. Correlación basada en reglas se aplica cuando se utilizan varios mensajes SIP para inferir una conclusión. Estas reglas son especialmente necesarias para detectar las inundaciones (flooding), la exploración (scanning) y la serie SPIT de eventos.

7. Armadura de flooding protege honeypot de ser inundado por solicitudes SIP. Se hace asegurando que el honeypot procesa un número limitado de solicitudes en un período determinado de tiempo. Para este objetivo se utilizan las reglas de correlación.

8. Los scripts de respuesta son ejecutados por el honeypot para reaccionar a un ataque detectado bloqueándolo o mitigándolo.

9. Hay dos tipos de alertas que se utilizan: alerta de mensaje que contiene todas las conclusiones que se deducen de un mensaje SIP y alerta compuesta que contiene información sobre la lista de mensajes SIP correlacionados. Las alertas se proporcionan a la interfaz de línea de comandos y se anotan en el formato de texto y HTML. El honeypot también se puede configurar para que envíe alertas al administrador por correo electrónico.

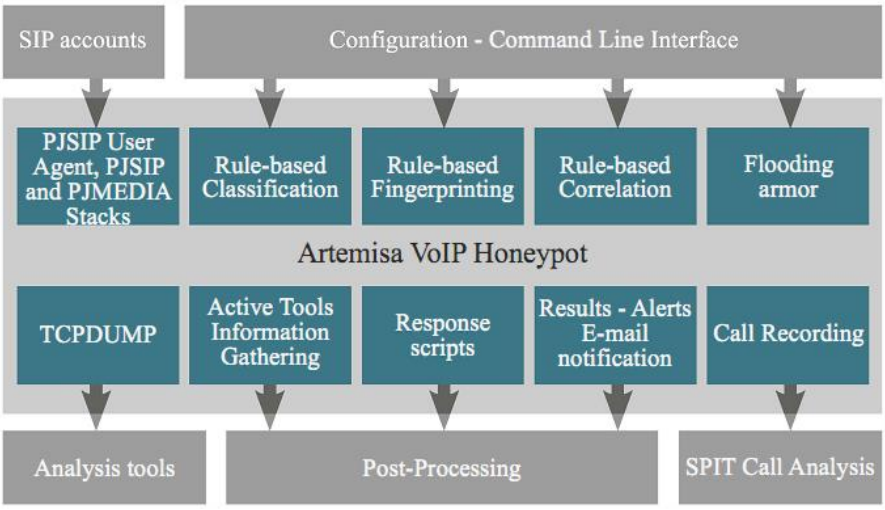

Figura 17. Mapa de módulos de Artemisa

Una vez que cualquier extensión de Artemisa recibe una llamada, el honeypot responde a la llamada como si fuera un simple usuario. Al mismo tiempo empieza el

análisis de mensajes entrantes SIP. Artemisa clasifica la llamada y guarda el resultado en una nueva revisión por el administrador de seguridad.

El mensaje se clasifica en los siguientes pasos:

1. En primer lugar, Artemisa revisa huellas de herramientas de ataques conocidos. Si el atacante utiliza alguna herramienta de hacking conocida, la huella de esta herramienta puede revelar fácilmente intenciones maliciosas.

2. A continuación, se comprueban los nombres de dominio y los puertos abiertos SIP en la parte del atacante. También se comprueban URIs solicitados y mensajes de ACK recibidos del usuario. Entonces, finalmente, Artemisa comprueba el flujo RTP recibido si éste ha sido establecido.

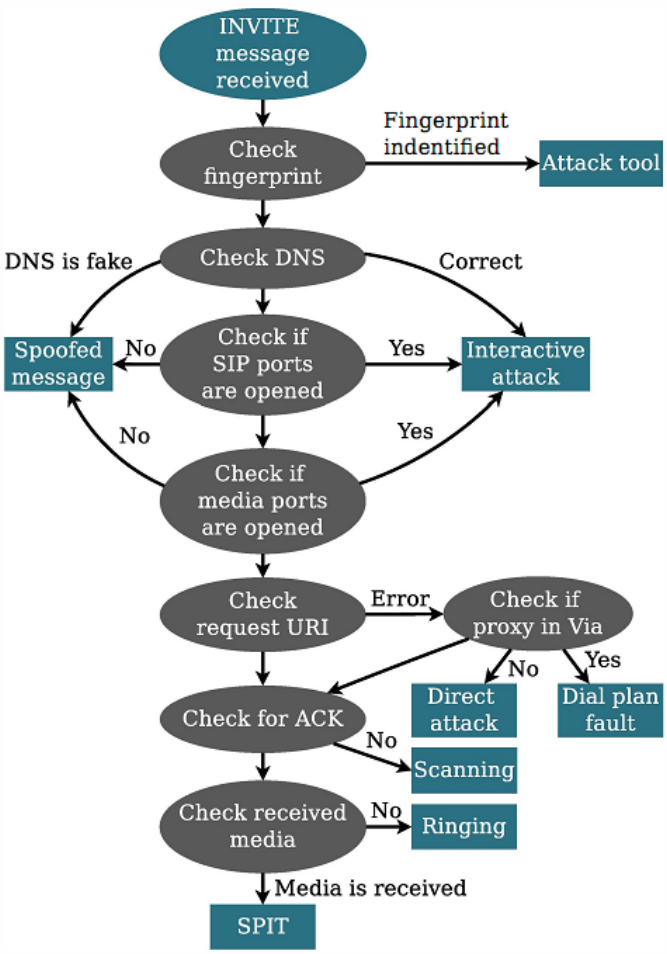

Figura 18. Árbol de algoritmo para el recibo de un mensaje INVITE

Esta secuencia de los procedimientos ayuda a Artemisa a clasificar la llamada tal como se representa en la figura 19. El resultado se muestra entonces en una consola. Los resultados se pueden guardar en una carpeta predefinida o pueden ser enviados

como una notificación por e-mail. En la figura 20 se muestra in ejemplo del fichero con los resultados. Una vez que la llamada ha sido examinada se ejecuta una serie de bash scripts. Estos scripts se ejecutan con argumentos predefinidos. Artemisa puede poner en marcha algunas medidas para enfrentarse a los ataques.

```
... output ommitted ...

| | Category: Interactive attack

+ Checking if media port is opened...
|
| No RTP info delivered.
|
| Category: Spoofed message

... output ommitted ...

+ The message is classified as:
| Attack tool
| Spoofed message
| Interactive attack
| Dial plan fault
| Scanning
| Ringing

************************************* Correlation **************

Artemisa concludes that the arrived message is likely to be:

* The attack was created employing the tool SIPVicious.
* A flooding attack.

... output ommitted ...
```

Figura 19. Ejemplo del fichero con resultados de Artemisa

Artemisa puede funcionar en tres modos distintos: pasivo, activo o agresivo dependiendo de la configuración de su archivo de configuración.

En el modo pasivo Artemisa sólo toma las llamadas entrantes y las contesta. Usando el modo activo es posible tener la misma funcionalidad que en el modo pasivo. Además, Artemisa comienza a examinar los mensajes entrantes SIP.

El modo agresivo ataca al intruso con su propio script. Por lo general, Artemisa se utiliza en el modo activo de manera que se analizan todos los mensajes SIP enrutados al honeypot.

4.3.5 Kippo

El segundo honeypot estudiado se basa en otros conceptos. No es orientado a VoIP como Artemisa. Kippo es una herramienta honeypot para la emulación del servicio de Secure Shell (SSH). Está diseñada principalmente para interactuar con los ataques de

fuerza bruta y almacenar las bitácoras. Está inspirada, pero no basada, en otra herramienta llamada Kojoney. En caso de que un atacante tuviera acceso al honeypot, es capaz de almacenar todas las herramientas maliciosas descargadas para un análisis posterior.

Cuando alguien intenta conectarse a un servidor con un honeypot en él, la aplicación twistd redirige a este usuario a honeypot. Esto sucede cuando la dirección IP del usuario no se incluye en la lista de direcciones IP permitidas. Una vez establecida la conexión con el honeypot, el atacante debe introducir correctamente el nombre de usuario y la contraseña. El nombre de usuario más utilizado es *root* y la contraseña más utilizada es la combinación de números 123456, Tabla 1 enumera las 10 contraseñas más utilizadas.

Password	count
	28146
123456	17625
password	6325
1234	5663
12345	5501
123	5342
1qa2ws3ed	5278
a	5121
test	4743
qwerty	4601

Tabla 20. Las contraseñas más utilizadas

Kippo registra cada intento de conexión. Cuando la combinación introducida es válida, el intruso tiene acceso a un sistema de archivos falsos. Cada comando introducido en el honeypot se registra y el comportamiento típico de un comando en particular es emulado (sólo para los comandos más comunes). Si el usuario intenta descargar un archivo desde Internet, Kippo guarda este archivo en una carpeta segura para examinarlo con más detalle después.

Todos los registros realizados por Kippo se guardan en una base de datos MySQL que facilita el análisis posterior.

4.3.6 Dionaea

Dos honeypots mencionados anteriormente son orientados a los servicios individuales. Dionaea pertenece a un honeypot orientado a multi-servicio que puede simular muchos servicios a la vez. Normalmente es sólo información general de servicios múltiples, pero Dionaea es capaz de recoger datos sobre las interacciones con un buen número de protocolos, incluyendo SMB (Microsoft's printers, files, serial ports sharing protocol), HTTP, FTP, TFTP, MSSQL (Microsoft SQL server), protocolos SIP. Los atacantes abusan de estos protocolos en la mayoría de los casos. Dionaea también tiene capacidad de guardar el contenido malicioso, si es necesario, pero como contrario a Kippo también puede emular código a partir de estos archivos.

Al mismo tiempo, Dionaea trabaja de una manera diferente de Artemisa. No hay necesidad de conexión a un servidor de VoIP externo (o producción). Simplemente espera a cualquier mensaje SIP y trata de responderle. Es compatible con todas las solicitudes SIP, como REGISTER, INVITE, ACK, CANCEL, BYE, OPTIONS, etc. Dionaea soporta múltiples sesiones SIP y flujos de audio RTP (datos de secuencia se pueden grabar). Para una mejor simulación de un sistema de telefonía IP real, es posible configurar diferentes agentes miméticos de teléfono del usuario con nombre de usuario personalizado, combinaciones de contraseñas. Todo el tráfico se controla, y los registros se guardan en archivos de texto plano y en la base de datos SQLite.

4.3.7 Pruebas de usabilidad de honeypots

Artemisa

A continuación se describen los resultados de pruebas de tres tipos de honeypots mencionados anteriormente realizados por autores del trabajo [17].

Para probar la utilidad del honeypot Artemisa los autores han preparado algunas pruebas en topología de las pruebas como se representa en la figura 21. Se ha construido una red sencilla de VoIP con Asterisk como PBX y un poco de hardware de punto final y los teléfonos de software. El honeypot ha sido instalado en una máquina dentro de la red con cinco extensiones. Estas extensiones se ejecutan en el modo activo. Dado que se están desarrollando propio IPS (Sistema de Prevención de Intrusiones), han optado por probar el honeypot bajo escenarios de prueba similar al sistema IPS.

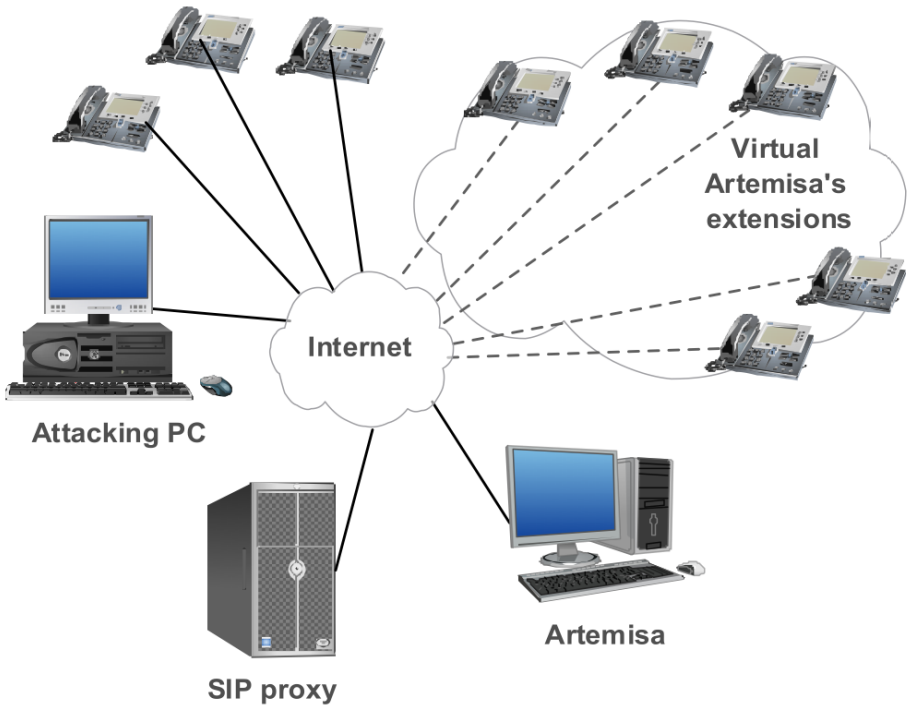

Figura 21. Topología de pruebas con Artemisa

En primer lugar, hay que comenzar a escanear toda la red desde el punto de vista de un intruso típico. Muchas aplicaciones se pueden utilizar para este propósito. Se utilizan dos de estas aplicaciones: *nmap* y *SIPVicious*.

Ambas aplicaciones han producido información útil. Sin embargo, ni *nmap* ni *svmap* han sido detectados por el honeypot. En caso de *svmap* había información sobre el mensaje SIP entrante, pero este mensaje no ha sido analizado. Ningún resultado se ha creado después de que escanear la red. Este comportamiento ha sido bastante sorprendente, ya que por lo general, cada ataque se inicia mediante el escaneo de la red. Artemisa debe tener en cuenta este tipo de situaciones. Con *svmap* sabemos sobre el usuario-agente que se ejecuta en la dirección IP del honeypot, como se indica a continuación.

```
158.196.244.241:5060 | Twinkle/1.4.2 |
T-Com Speedport W500V / Firmware v1.37
MxSF/v3.2.6.26
```

Con esta información empiezan a buscar extensiones que se ejecutan en la red de prueba. Escaneo directo del servidor proxy SIP no ha sido detectado por el honeypot, pero cuando usan la herramienta *svwar* directamente en contra de la dirección IP en la que se está ejecutando un honeypot, se obtiene información acerca de todas las extensiones activas. Esta exploración ha sido reconocida por el honeypot y un archivo

de resultado adecuado se ha creado. Artemisa concluye correctamente que los mensajes recibidos provienen de un escáner *SIPVicious*. Por otro lado, sabemos por la salida *SIPVicious* que estas extensiones no se comportan como clientes normales. Esto puede suscitar una mayor cautela en el lado del intruso.

El objetivo de otros ataques es inundar el dispositivo de cliente con varios tipos de mensajes SIP. Utilizando algunos de estos ataques, el intruso puede realizar un ataque DoS en un grupo cerrado de dispositivos de punto final [18], [19]. Para este tipo de ataque se ha utilizado una serie de herramientas, entre ellas *udpflood*, *rtpflood*, *inviteflood* y *sipp* [18], [20].

Cada una de estas aplicaciones puede lanzar un ataque DoS simple. Como Artemisa es un mero honeypot de VoIP, sólo se detectan ataques que utilizan el protocolo SIP. En consecuencia, sólo se han detectado ataques de inundación de *inviteflood* y *sipp* [21]. En caso de *inviteflood*, la aplicación ha sido reconocida con éxito gracias a su conocida huella digital.

La aplicación *sipp* no está diseñada para las pruebas de hackong o de penetración, pero esta funcionalidad se puede conseguir fácilmente. Se han utilizado escenarios de llamadas específicos con un impacto similar como en las herramientas de inundaciones anteriormente mencionadas. Usando *sipp* se puede generar un gran número de mensajes SIP que se detectan inmediatamente como un ataque de inundación por el honeypot. En esta situación, todo el honeypot ha dejado de responder en breve y no se ha registrado ningún resultado sobre el ataque en absoluto, 250 mensajes SIP por segundo han causado esta situación. Cuando han utilizado las tasas de envío menores, el ataque ha sido reconocido y el archivo de salida se ha creado con éxito.

La identificación de una llamada SPIT es una de las características más importantes de honeypot. Se ha utilizado la aplicación llamada SPITFILE para la simulación de estas llamadas [22]. SPITFILE es una herramienta de penetración SIP de código abierto que se desarrolló a fines de investigación y desarrollo de nuevos algoritmos de protección contra el spam en telefonía IP. SPITFFILE pone mucho énfasis en la sencillez de uso y la generación de ataques SPIT. SPITFILE fue programado en Python usando wxPython GUI y el objetivo de la aplicación es generar llamadas telefónicas y reproducir un mensaje de voz pre-grabada. Los autores han aprobado la aplicación SIPp que se centra en las pruebas y simulación de llamadas SIP en la infraestructura de VoIP. SIPp es una herramienta de pruebas de código abierto o generador de tráfico para el protocolo SIP y puede leer archivos XML personalizados de escenarios que describen desde muy simples hasta complejos flujos de llamadas y también envían el tráfico de comunicación a través de RTP [23], [24]. SPITFILE implementa una interfaz

gráfica para SIPp y trabaja con los ready-made .xml diagramas. Por lo tanto, la simulación de un ataque SPIT es mucho más simple.

Su control es muy intuitivo. Los valores solicitados se presentan en los campos pertinentes y el ataque SPIT se pone en marcha pulsando el botón de ENVIAR. SPITFILE está disponible tanto para Linux como para MS Windows. SPITFILE puede generar spam en dos modos.

- Modo Directo, genera SPIT en el teléfono IP directamente sin utilizar un proxy SIP.
- Modo Proxy, genera SPIT a través de SIP Proxy y luego se puede ejecutar en contra de todo lo que está disponible detrás del proxy, con la participación no sólo teléfonos IP, sino también teléfonos fijos. El menú del Modo Proxy se representa en la figura. 22.

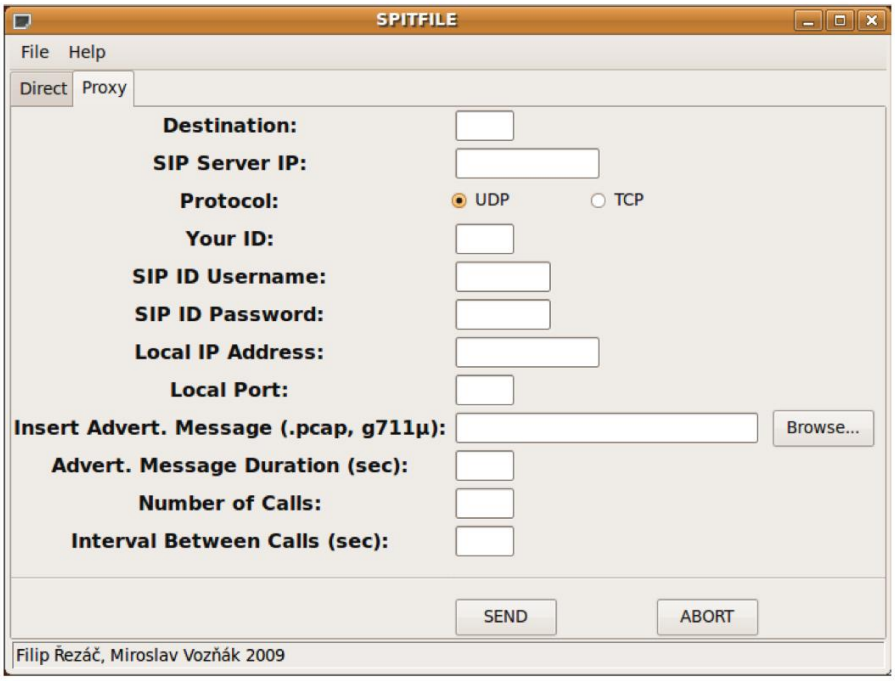

Figura 22. SPITFILE interfaz en Modo Proxy

Con esta herramienta, se puede fácilmente generar llamadas arbitrarias. Todas estas llamadas han sido detectadas con éxito por Artemisa y los archivos de salida apropiados se han generado.

Por fin, han tratado de hacer una llamada a las extensiones honeypot con el hardware y el software teléfono. Las llamadas se han marcado como una exploración y un ataque ringing en ambos casos. Así que parece que Artemisa evalúa casi todos los mensajes SIP enviados a sus extensiones como algún tipo de ataque.

Los resultados de los ataques detectados se almacenan en results/directory dentro de la carpeta de Artemisa. El archivo de salida es en formato de texto plano y en formato html, ambos contienen los mismos datos. Todas las funcionalidades antes mencionadas se refieren a honeypots que se ejecutan en el modo activo. El modo agresivo parece más interesante con su capacidad para contraatacar al intruso. Artemisa contiene tres secuencias de comandos bash para detener las actividades maliciosas. Sin embargo, un vistazo de cerca a estos scripts es sorprendente.

Vamos a empezar con el último *on_spit.sh* script. Dentro de esta script sólo hay un comentario. Este comentario puede activar una regla de firewall, pero el comando no está incluido. Este script es totalmente inútil a menos que reescribirlo. Incluso scripts restantes no contienen nada, sólo comentarios en su interior. Una condición simple (comentada) está incluida en script *on_scanning.sh*, que ejecuta un script python para bloquear el escaneado por parte de la aplicación *SIPVicious* (pero sólo esta aplicación en particular).

El script *on_flood.sh* tiene un comando comentado que añade en IPtables reglas para la dirección IP y el puerto. Se determinan mediante el uso de un parámetro. Esta solución no es mala, pero si se quiere bloquear algo de tráfico, existe la posibilidad de que un ataque falso se bloquea, por lo que también debe ser incluido algún mecanismo de recuperación automática. Esto puede ser fácilmente resuelto mediante la adición de otro script. Este script eliminará la regla después de un cierto intervalo. El principal problema en el bloqueo de tráfico usando IPtables es que el comando aplica la regla en una máquina local. Sin embargo, sólo bloquea las consecuencias en honeypot, y no en el firewall principal que protege toda la infraestructura. Esta característica hace que el modo agresivo sea inútil contra cualquier tipo de ataque.

Los autores mencionan que usar el honeypot en el modo pasivo es inútil, porque Artemisa sólo responde a la llamada sin mayor análisis y sin resultados guardados en un archivo.

Análisis de datos de Kippo

Se utiliza Kippo honeypot para analizar el tráfico SSH en una red real con siete sensores activos de monitoreo. El honeypot fue activo y recoleccionó datos por un mes. Durante este período, se observaron 873.342 intentos de conexión.

Sólo una pequeña parte de estas conexiones se realizó correctamente. La Tabla 1 enumera diez combinaciones de contraseñas más utilizadas que permiten conexiones.

Como podemos ver en la tabla, la contraseña 123456 fue utilizada 17.625 veces. El nombre de usuario correcto *root* sólo se utilizó en 2.551 casos. Estos 2.551 casos

condujeron a una conexión a un sistema de ficheros falso. Un intruso entonces normalmente sube algún tipo de script que se debe utilizar para un ataque de denegación de servicio en un servidor externo.

Otra información interesante adquirida de honeypot es la ubicación aproximada del origen de la conexión del atacante como se representa en la figura 23.

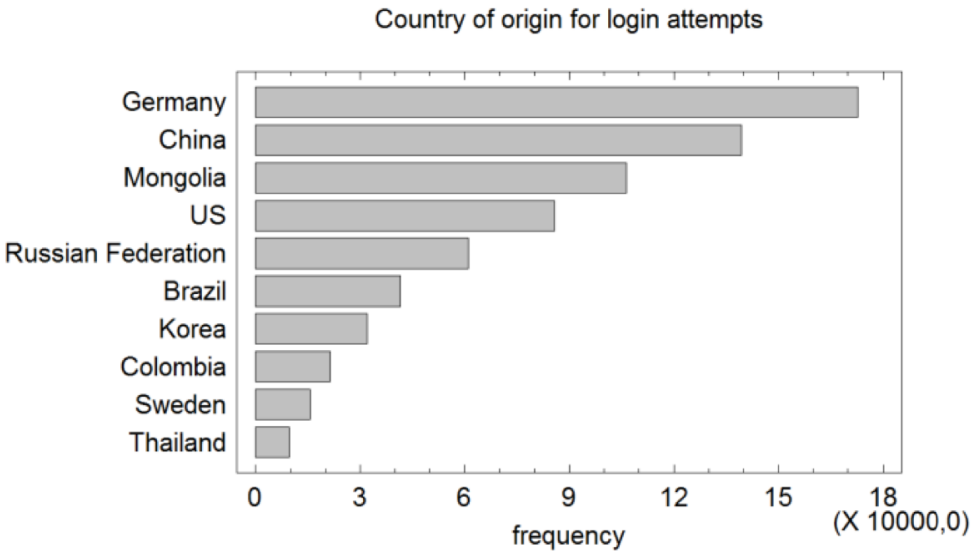

Figura 23. País de origen en el registro de los intentos

La figura 23 muestra sólo los primeros diez posiciones. Los ataques de Alemania representan 25,21% de los intentos de conexión totales. China se ubicó en el segundo lugar con 20,33% y Mongolia en el tercer lugar con 15,52%. Casi el 75% de los intentos de conexión se hace de uno de los primeros cuatro países.

Como se mencionó anteriormente, tenemos siete sensores diferentes. Con cerca de un millón de intentos, cada sensor se probó cada 23 segundos.

Análisis de datos de Dionaea

Dionaea estuvo monitoreando el tráfico malicioso durante 18 días. El número de ataques no es tan alto como en el caso de Kippo pero suficientemente alto.

La mayoría de los intentos de ataque provienen de Israel con dirección IP 37.8.54.135. La segunda IP más utilizada fue originada en Alemania y la tercera en Rusia.

SIP message	Groups	Count	Ratio
ACK	40	303	7,575
BYE	4	4	1,000
CANCEL	1	11	11
INVITE	18	85	4,722
OPTIONS	76	76	1,000
REGISTER	28	1745	62,321

Tabla 2. Datos recopilados sobre mensajes SIP

Dionaea proporciona también información adicional sobre los ataques como utilización de mensajes SIP, información de cabeceras SIP, estadísticas de SDP y RTP. Es interesante el comportamiento típico de ataques basados en envío de mensajes SIP. Todos los ataques se producen en secuencias típicas. La Tabla 2 muestra los datos recopilados sobre mensajes SIP. En la columna grupos es el número de mensajes SIP agrupados por diferentes conexiones. Una conexión es una sola sesión de SIP proxy emulada por honeypot. Ratio simplemente ilustra el número promedio de mensajes en cada grupo de conexión.

La mayoría de los ataques observados se pueden dividir en dos grupos. El primer grupo representa varios tipos de PBX de exploración y sondeo. Los atacantes envían el mensaje OPTION y esperan una respuesta, o simplemente tratan de hacer una llamada con la cancelación inmediata.

Otro grupo representa ataques de inundación.

Por fin los autores han observado varios ataques que no podían ser simplemente clasificados en grupos.

Los autores del artículo analizaron estos tres honeypots antes de crear una aplicación de honeypots en la nube para las redes VoIP. Después del análisis realizado decidieron dejar sólo Kippo y Dionaea. Ver [17].

Se puede concluir que Dionaea es el honeypot más adecuado para los casos de múltiples servicios o si no se sabe qué protocolo utiliza el atacante. Artemisa sigue siendo el más clásico de honeypots y con los modos pasivo y agresivo proporciona una solución equilibrada para los atacantes de seguimiento. Kippo no es orientado a VoIP en comparación con los demás y es muy limitado en algunos aspectos, lo que también hace que sea menos adecuado para el uso de la seguridad de VoIP, aunque pueda mostrar buenos resultados en tareas pequeñas.

4.3.8 Detección de anomalías en VoIP

En general, las técnicas de detección de intrusiones pueden clasificarse en dos categorías:

- Detección basada en firmas (signature detection o misuse detection)
- Detección basada en anomalías

Estas dos clases también se conocen, respectivamente, como detección basada en conocimiento y detección basada en comportamiento.

La detección de ataques mediante firmas consiste en buscar patrones de actividad maliciosa definidos de antemano y que corresponden a ataques conocidos. En cambio, la detección de anomalías busca desviaciones del comportamiento considerado normal de una red o un sistema y que pueden ser debidos a ataques maliciosos, incluyendo nuevos ataques que todavía no han sido caracterizados y que, por tanto, no podrían detectarse mediante firmas. Frente a esta ventaja, los sistemas de detección de anomalías presentan como inconveniente un número potencialmente elevado de falsos positivos (alarmas debidas a desviaciones del comportamiento considerado normal que, sin embargo, no corresponden a ataques). Otros inconvenientes a considerar son lentitud en el análisis de tráfico, dificultad de adaptación a cambios en la red monitorizada y falta de métodos de evaluación y de referencias para comparar unos sistemas con otros.

Dentro de los sistemas de detección de anomalías se puede establecer una clasificación adicional en función del método que se utilice para definir cuál es el comportamiento normal de la red. Una alternativa es que el comportamiento normal se deduzca automáticamente a partir de observaciones de la red, utilizando diversas técnicas. Otra opción es que el comportamiento normal sea especificado manualmente por un experto, por ejemplo, utilizando algún lenguaje de descripción formal (autómatas extendidos, etc.). Ver ejemplos en el diagrama siguiente.

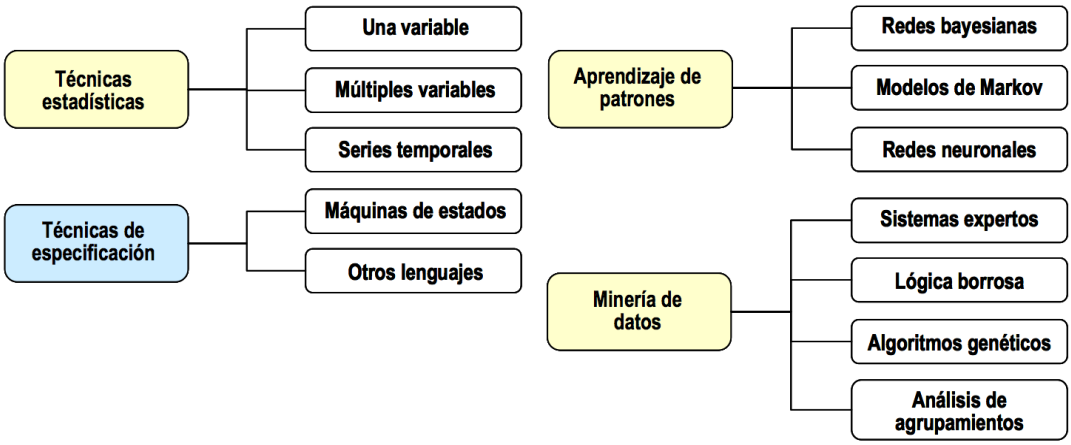

Figura 21. Técnicas de caracterización del comportamiento normal y detección de anomalías

Existen varios estudios recientes que analizan los elementos y funciones principales de los sistemas de detección de anomalías (parametrización, entrenamiento, modelado, detección), que comparan las prestaciones de diferentes algoritmos de detección o que los clasifican según diversos criterios, dando lugar a las correspondientes propuestas de taxonomías para dichos sistemas. Se puede consultar, por ejemplo, [25][26][27] y [28] sobre IDS/IPS (que puede abreviarse a IDPS) en general y [29][30] y [31] sobre sistemas basados en anomalías.

En [Vuong04] se presentan algunas consideraciones generales sobre la aplicación de los sistemas de detección de intrusiones a redes de Voz sobre IP que utilizan protocolos como H.323, SIP, MEGACO, SIGTRAN, IPsec, etc.

En cuanto a la aplicación de métodos específicos de detección de intrusiones en VoIP, se pueden citar varios trabajos recientes. Por ejemplo, en [32] se describe un método estadístico de caracterización del tráfico normal de VoIP basado en medidas de varianza sobre intervalos de tiempo de varios minutos que permite detectar sobrecargas, cambios anómalos del nivel de carga y presencia de tráfico no VoIP.

[33] describe un mecanismo de detección de intrusiones para SIP basado en reglas de inferencia Bayesiana, aplicadas a partir de un conjunto de eventos observables, por ejemplo, tasa de peticiones SIP recibidas, tasa de respuestas de error (identificadas en SIP por códigos numéricos entre 300 y 699), tasa de errores en la decodificación de mensajes SIP recibidos, número de direcciones destino diferentes, número de diálogos en espera y número de puertos RTP abiertos. A partir de estas observaciones sobre el tráfico existente en la red, se infiere un vector de probabilidades de que la red esté en estado normal o bajo determinados tipos de actividades maliciosas como: denegación de servicio, escaneado, SPIT y etc.

En [34] se propone una combinación de técnicas de detección basada en firmas y basada en anomalías, junto con otros mecanismos como señuelos (honeypots) con el fin de dar respuesta a los diferentes tipos de ataques que amenazan las redes de VoIP.

El uso de especificaciones del comportamiento normal para detectar anomalías es un enfoque relativamente nuevo [35]. (En lugar de incluir la especificación manual del comportamiento normal como un caso particular de detección de anomalías, criterio que se sigue en este documento, algunos autores presentan los métodos basados en especificaciones como una tercera clase de IDS, separada de la detección mediante firmas y la detección de anomalías basada en técnicas automáticas de caracterización del comportamiento normal).

La especificación del protocolo o protocolos cuyo comportamiento normal se quiere caracterizar puede hacerse, por ejemplo, mediante máquinas de estados o autómatas extendidos. No toda desviación del comportamiento que marca la máquina de estados del protocolo tiene necesariamente que deberse a ataques de seguridad, pues podría deberse a errores de protocolo, mensajes perdidos o desordenados en la red u otras causas no maliciosas. Por ello junto con los autómatas se definen contadores y umbrales para determinar cuando la desviación observada es suficientemente significativa para generar una alerta de seguridad.

[36] propone un sistema de detección de intrusiones que usa información cruzada sobre el estado de diferentes protocolos de VoIP y la compara con una serie de reglas para detectar ataques. [37] aplica la detección basada en especificaciones para la protección ante ataques de denegación de servicio a servidores de redes VoIP basadas en protocolos H.323.

En [38] se propone un sistema para detectar ataques de denegación de servicio en redes SIP causados por inundación de peticiones o envío de mensajes SIP inválidos. La detección se basa en una especificación del comportamiento normal del protocolo mediante autómatas. El sistema controla la evolución del estado de las entidades SIP en función de los eventos del protocolo (paquetes enviados, recibidos, temporizadores) y siguiendo los cuatro tipos de transacciones definidos en el estándar del protocolo SIP: transacción INVITE en el lado cliente, INVITE en el lado servidor, no INVITE en el lado cliente y no INVITE en el servidor.

Para detectar situaciones anómalas se fijan cuatro umbrales:

1. Número de transacciones erróneas por segundo (por ejemplo, por recepción de mensajes inesperados en determinado estado).

2. Número máximo de paquetes por transacción (por ejemplo para evitar inundación del servidor con un número excesivo de retransmisiones).

3. Número de transacciones permitidas por nodo (para evitar ataques de inundación con transacciones correctas).

4. Número de respuestas SIP de error (respuestas en el rango 300-699) por segundo.

[39] sigue también el enfoque de detección de intrusiones mediante especificación del comportamiento de los protocolos de VoIP con autómatas extendidos. Este trabajo no solo especifica cada protocolo por separado, sino que tiene en cuenta las interacciones entre diferentes protocolos de VoIP, por ejemplo entre SIP en el plano de control y RTP en el de usuario, para mejorar la detección de ataques. Por ejemplo, el ataque consistente en que un tercero envía un mensaje de señalización SIP BYE para forzar la liberación de una llamada en curso entre dos usuarios puede detectarse porque tras la recepción del BYE se siguen recibiendo paquetes RTP de voz del interlocutor.

En [40] se comparan dos de los sistemas de detección basados en especificaciones para VoIP citados más arriba, SCIDIVE [36] y vIDS [39], con otros anteriores propuestos para otros protocolos, como NetSTAT [41].

[42] propone otro sistema de detección de intrusiones para VoIP basado en especificaciones con máquinas de estados de SIP y RTP, y hace una evaluación de prestaciones utilizando la herramienta de simulación OPNET Modeler (www.opnet.com). Para la evaluación usa cuatro ataques contra SIP o contra RTP: inundación de peticiones INVITE, denegación de servicio mediante envío de peticiones BYE, redirección maliciosa de llamadas mediante re-INVITE e inserción de paquetes espurios en el flujo RTP.

5 Conclusiones

Los resultados principales del presente trabajo son:

1. Ha sido descrita la arquitectura típica de la red de voz sobre IP basada en el protocolo SIP. Se ha presentado su funcionamiento y se han estudiado los protocolos utilizados en la arquitectura VoIP, tales como SDP, RTP, H.248/MEGACO, SIGTRAN y otros.

2. Se han detallado las amenazas más significativas que afectan a la telefonía sobre redes IP. La mayoría de los riesgos son inherentes de las capas sobre las que se apoya la tecnología VoIP por lo que muchos de los ataques se basan en técnicas bien conocidas. Por supuesto, también se han descrito ciertas vulnerabilidades que afectan específicamente a las redes VoIP y a sus protocolos. Las amenazas de las redes de telefonía IP han sido clasificadas en las siguientes categorías:

   - Accesos desautorizados y fraudes.
   - Ataques de denegación de servicio.
   - Ataques a los dispositivos.
   - Vulnerabilidades de la red subyacente.
   - Enumeración y descubrimiento.
   - Ataques a nivel de aplicación.

3. Se ha recogido una serie de protocolos de seguridad aplicables en redes de voz sobre IP basadas en el protocolo SIP, tales como HTTP Digest, TLS, S/MIME, IPsec, SRTP y DIAMETER.

4. Dentro del ámbito de mecanismos de detección y prevención de intrusiones ha sido estudiada la aplicación de honeypots. Se pretendía estudiar varios honeypots existentes a día de hoy que puedan ser utilizados en las redes VoIP y conocer sus ventajas e inconvenientes. Entre gran variedad de honeypots he podido encontrar información sólo sobre tres que se utilizan para las redes VoIP: Artemisa, Kippo y Dionaea. Además Kippo no es orientado a VoIP, pero puede ser utilizado en conjunto con otros honeypots para lograr más eficiencia.

   Se puede concluir que Dionaea es el honeypot más adecuado para los casos de múltiples servicios o si no se sabe qué protocolo utiliza el atacante. Artemisa sigue siendo el más clásico de honeypots y con los modos pasivo y agresivo proporciona una solución equilibrada para hacer seguimiento de los atacantes.

Los futuros trabajos en esta área consistirían en el despliegue de la red de voz sobre IP y la realización de las propias pruebas de los honeypots destacados.

Bibliografía

1. J. Dauphin, R. Geldwerth, S Znaty "SIP: Session Initation Protocol". EFORT, 2005. http://www.efort.com/media_pdf/SIP_ESP.pdf
2. Proyecto CENIT Segur@, "Tecnologías de monitorización, detección y respuesta para intrusiones y otros elementos de fraude y estudio legal asociado, 2008.
3. C. Holmberg, E. Burger, H. Kaplan "Session Initiation Protocol (SIP) INFO Method and Package Framework", RFC 6086. Enero 2011. http://www.ietf.org/rfc/rfc6086.txt
4. J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, E. Schooler "SIP: Session Initiation Protocol", RFC 3261. Junio 2002. http://www.ietf.org/rfc/rfc3261.txt
5. T. Berners-Lee, R. Fielding, L. Masinter "Uniform Resource Identifier (URI): Generic Syntax", RFC 3986. Enero 2005. http://www.ietf.org/rfc/rfc3986.txt
6. ITU-T Rec. Q.700: "Introduction to Signalling System Nº. 7". Marzo 1993. http://www.itu.int/rec/T-REC-Q.700/en
7. ITU-T Rec. H.248.1: "Gateway control protocol version 3". Marzo 2013. http://www.itu.int/rec/T-REC-H.248.1/en
8. D. O'Neill, U. Mulligan "Parlay/OSA: open APIs for service development". 3GPP, 2006.
9. L. Kilmartin, M. K. Ranganathan "Performance Implications of Securing Session Initiation Protocol based VoIP Networks". Computer Communications, vol. 26, num. 6. Abril 2003.
10. T.J. Walsh, R. Kuhn "Challenges in Securing Voice over IP". IEEE Security and Privacy, 2005.
11. E. Cha, H. Choi, S. Cho "Evaluation of Security Protocols for the Session Initiation Protocol". 16th International Conference on Computer Communications and Networks, ICCCN, 2007.
12. P. Diebold, A. Hess, G. Schafer "A VoIP Honeypot Architecture for Detecting and Analyzing Unknown Network Attacks". 14th Kommunikation in Vertrilten Systemen 2005 (KiVS05) Kaiserslautern, Alemania. Febrero 2005.
13. C. Valli "An Analysis of Malfeasant Activity Directed at a VoIP Honeypot". Proceedings of the 8th Australian Digital Forensics Conference, Edith Cowan University, Perth Western Australia. Noviembre 2010.

14. M. Gruber, F. Fankhauser, S. Taber, C. Schanes, T. Grechenig "Security Status of VoIP Based on the Observation of Real-World Attacks on a Honeynet". IEEE International Conference on Privacy, Security, Risk, and Trust, and IEEE International Conference on Social Computing, 2011.

15. R. do Carmo, M. Nassar, O. Festor "Artemisa: an Open-Source Honeypot Back-End to Support Security in VoIP Domains". 12th IFIP/IEEE International Symposium on Integrated Network Management, 2011.

16. M. Voznak, J. Safarik, L. Macura, F. Rezac "Malicious Behavior in Voice over IP Infrastructure". Recent Researches in Communications and Computers, 2012.

17. M. Voznak, J. Safarik, F. Rezac "Threat Prevention and Intrusion Detection in VoIP Infrastructures", International Journal of Mathematics and Computers in Simulation, vol. 7, num. 1, 2013.

18. D. Endler, M. Collier "Hacking Exposed VoIP". McGraw-Hill Osborne Media, 2009.

19. M. Voznak, J. Safarik "DoS attacks targeting SIP server and improvements of robustness". International Journal of Mathematics and Computers in Simulation, vol. 6, num. 1, 2012.

20. D. Sisalem, J. Kuthan, T.S. Elhert, F. Fraunhofer "Denial of Service Attacks Targeting SIP VoIP Infrastructure: Attack Scenarios and Prevention Mechanisms". IEEE Network, 2006.

21. M. Voznak, F. Rezac "Threats to voice over IP communications systems". WSEAS Transactions on Computers, vol. 9, num. 11. Noviembre 2010.

22. M. Voznak, F. Rezac "VoIP SPAM and a defense against this type of threat". 14th WSEAS International Conference on Communications, 2010.

23. M. Voznak, J. Rozhon "SIP infrastructure performance testing". 9th WSEAS International Conference on Telecommunications and Informatics, 2010.

24. M. Voznak, J. Rozhon "Methodology for SIP infrastructure performance testing". WSEAS Transactions on Computers, vol. 9, num. 9. Septiembre 2010.

25. P. Kabiri, A. Ghorbani "Research on Intrusion Detection and Response: A Survey". International Journal of Network Security, vol. 1, num. 2. Septiembre 2005.

26. T.S. Sobh "Wired and wireless intrusion detection system: Classifications, good characteristics and state-of-the-art". Computer Standards & Interfaces, vol. 28, 2006.

27. P. Gamper "Towards automated exploit signature generation using honeypots". Swiss Federal Institute of Technology Zurich, Master Thesis, 2007.

28. N. Stakhanova, S. Basu, J. Wong "A taxonomy of intrusion response systems". International Journal on Information and Computer Security, vol. 1, num. 1-2, 2007.

29. A. Lazarevic, A. Ozgur, L. Ertoz, J. Srivastava, V. Kumar "A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection". SIAM International Conference on Data Mining, San Francisco. Mayo 2003.

30. P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, E. Vázquez "Anomaly-based Network Intrusion Detection: Techniques, Systems and Challenges". Computers & Security, vol. 28, num. 1-2, Elsevier, ISSN 0167-4048. Febrero-Marzo 2009. http://www.sciencedirect.com/science/article/pii/S0167404808000692

31. A. Patcha, J.M. Park "An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends". Computer Networks, 2007.

32. H.H. Fengg, O.M. Kolesnikov, P. Fogla, W. Lee, W. Gong "Anomaly detection using call stack information". IEEE Symposium on Security and Privacy, 2003.

33. M. Nassar, R. State, O. Festor "Intrusion detection mechanisms for VoIP applications". 3rd Annual VoIP Security Workshop, Berlín. Junio 2006.

34. M. Nassar, S. Niccolini, R. State, T. Ewald "Holistic VoIP Intrusion Detection and Prevention System". IPTComm 2007, Nueva York. Julio 2007.

35. R. Sekar, A. Gupta, J. Frullo, T. Shanbhag, A. Tiwari, H. Yang, S. Zhou "Specification-based anomaly detection: A new approach for detecting network intrusions". ACM Computer and Communication Security Conference (CCS), Washington DC. Noviembre 2002.

36. Y. Wu, S. Bagchi, S. Garg, N. Singh, T. Tsai "SCIDIVE: A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments". International Conference on Dependable Systems and Networks. Junio 2004.

37. P. Truong, D. Nieh, M. Moh "Specification-based Intrusion Detection for H.323-based Voice over IP". 5th IEEE International Symposium on Signal Processing and Information Technology, Atenas. Diciembre 2005.

38. E. Chen "Detecting DoS Attacks on SIP Systems". 1st IEEE workshop on VoIP Management and Security: VoIP MaSe, Vancouver, Canadá. Abril 2006.

39. H. Sengar, D. Wijesekera, H. Wang, S. Jajodiay "VoIP Intrusion Detection Through Interacting Protocol State Machines". International Conference on Dependable Systems and Networks, Philadelphia. Junio 2006.

40. B. Barry, H. Chan "A Hybrid, Stateful and Cross-Protocol Intrusion Detection

System for Converged Applications". Lecture Notes in Computer Science (LNCS), vol. 4804. 2nd International Symposium on Information Security, Vilamoura, Algarve, Portugal. Noviembre 2007.

41. G. Vigna, R.A. Kemmerer "NetSTAT: A Network-based Intrusion Detection Approach". 14th Annual Computer Security Conference, Scottsdale, Arizona. Diciembre 1998.

42. T. Phit, K. Abe "Protocol Specification-based Intrusion Detection System for VoIP". Informe técnico del IEICE (Institute of Electronics, Information and Communication Engineers). Febrero 2008. http://almond.cs.uec.ac.jp/papers/pdf/2008/thyda_IN.pdf