

ALMA MATER STUDIORUM - UNIVERSITÀ DI BOLOGNA
SEDE DI CESENA
SECONDA FACOLTÀ DI INGEGNERIA CON SEDE A CESENA
CORSO DI LAUREA MAGISTRALE IN
INGEGNERIA INFORMATICA

TITOLO DELLA TESI:

**STUDIO ED ANALISI DELLA SICUREZZA DEL
PROTOCOLLO SIP NELLE RETI VOICE OVER IP**

Tesi in:

LABORATORIO DI RETI DI TELECOMUNICAZIONI LM

Relatore:
Prof. WALTER CERRONI

Presentata da:
PAOLO PIAGNANI

Correlatore:
Prof. ALDO CAMPI

Sessione III
Anno Accademico 2011/2012

PAROLE CHIAVE

Voice-over-IP

Sicurezza Reti

SIP

Classi di Vulnerabilità

Traffico VoIP

Acronimi:

AOR - Address-of-Record

B2BUA - Back-to-Back User Agent

DoS - Denial of Service

HJ - Hijack

IDS - Intrusion Detection System

IETF - Internet Engineering Task Force

IM - Instant Message

IP - Internet Protocol

IPSec - Internet Protocol Security

MIME - Multipurpose Internet Mail Extensions

MitM - Man-in-the-Middle

NAT - Network Address Translation

NGN - Next-Generation Network

QoS - Quality of Service

PSTN - Public Switched Telephone Network

RFC - Request For Comments

RTP - Real-Time Traffic Protocol

SDP - Session Description Protocol

SIP - Session Initiation Protocol

SIPS - SIP Secure

TLS - Transport Layer Security

UA - User Agent

URI - Uniform Resource Indicators

VoIP - Voice over IP

VoIPSA - Voice over IP Security Alliance

Abstract

Nell'era di Internet e della digitalizzazione, anche la telefonia ha avuto la possibilità di evolversi, e grazie alle tecnologie Voice-over-IP è stato possibile realizzare servizi di comunicazione avanzata su reti di dati. Anche se la comunicazione vocale è l'aspetto chiave di questi sistemi, le reti VoIP supportano altri tipi di servizi, tra cui video, messaggistica istantanea, condivisione di file, ecc.

Il successo di questa nuova tipologia di rete è dovuto ad una migliore flessibilità rispetto ai vecchi sistemi analogici, grazie ad architetture aperte e implementazioni a livello software, e soprattutto ad un minor costo legato alle apparecchiature ed ai collegamenti utilizzati, ed ai nuovi modelli di business e di consumo sempre più orientati allo sfruttamento della connettività a banda larga.

Tuttavia, l'implementazione dei sistemi VoIP rappresenta anche un grado di complessità maggiore in termini di architetture di rete, di protocolli, e di implementazione, e con questo ne segue un incremento delle possibili vulnerabilità. Una falla nella sicurezza in questi sistemi può portare a disservizi e violazione della privacy per gli utenti con conseguenti ripercussioni economiche per i relativi gestori.

La tesi analizza la sicurezza delle reti VoIP concentrandosi sul protocollo che sta alla base dei servizi multimediali, il protocollo SIP. SIP è un protocollo di livello applicativo realizzato per creare, modificare e terminare delle sessioni multimediali tra due o più utenti. Dopo un'introduzione alle generalità del protocollo, vengono esaminate le classi di vulnerabilità delle reti VoIP e gli attacchi a SIP, e vengono presentate alcune contromisure attuabili. Viene mostrato un esempio di come vengano attuati alcuni dei principali attacchi a SIP tramite l'utilizzo di appositi strumenti. L'elaborato conclude con alcune considerazioni sulle minacce al protocollo e sugli obiettivi futuri che la comunità scientifica dovrebbe perseguire.

Indice

Indice	VII
Capitolo 1	- 1 -
1. Introduzione.....	- 1 -
1.1. La Rete Telefonica Generale.....	- 1 -
1.2. Il passaggio a Voice-over-IP.....	- 2 -
1.3. Panoramica della tesi	- 3 -
Capitolo 2	- 5 -
2. Session Initiation Protocol.....	- 5 -
2.1. Generalità del Protocollo	- 6 -
2.2. Entità.....	- 8 -
2.3. Struttura di un Messaggio.....	- 10 -
2.4. Metodi.....	- 14 -
2.5. Protocolli Correlati	- 16 -
2.6. Considerazioni	- 17 -
Capitolo 3	- 19 -
3. Classi di Vulnerabilità VOIP	- 19 -
3.1. La Tassonomia VoIPSA	- 19 -
3.2. Social Threats	- 21 -
3.3. Eavesdropping	- 22 -

3.4. Interception and Modification	- 23 -
3.5. Denial of Service Threats	- 25 -
3.6. Considerazioni	- 28 -
Capitolo 4	- 29 -
4. Attacchi a SIP	- 29 -
4.1. Eavesdropping, Interception e Modification	- 29 -
4.2. Denial of Service	- 33 -
4.3. Social Threats	- 35 -
4.4. Considerazioni	- 37 -
Capitolo 5	- 39 -
5. Dimostrazione	- 39 -
5.1. Analizzatore di traffico: Wireshark	- 39 -
5.2. Generatore di traffico: SIPp.....	- 40 -
5.3. Dettagli della Dimostrazione	- 42 -
5.4. DoS: Faked Call Teardown.....	- 43 -
5.5. DoS: Call Hijacking.....	- 44 -
5.6. Man-in-the-Middle	- 46 -
5.7. Considerazioni	- 48 -
Capitolo 6	- 51 -
6. Contromisure	- 51 -
6.1. Difese del protocollo SIP	- 52 -

6.2. Difese della Rete VoIP	- 58 -
6.3. Considerazioni	- 62 -
Capitolo 7	- 65 -
7. Conclusioni.....	- 65 -
7.1. La Ricerca sulla Sicurezza nelle Reti VoIP	- 66 -
7.2. Considerazioni Finali.....	- 70 -
Bibliografia.....	- 73 -
Ringraziamenti	- 75 -

Capitolo 1

1. Introduzione

Le reti Voice-over-IP stanno guadagnando un'importanza crescente nel tempo. Le nuove tecnologie a banda larga offrono flessibilità e soprattutto una riduzione dei costi di implementazione e gestione, giustificando la crescente transizione dalla rete telefonica generale (Public Switched Telephone Network in inglese) alle Reti di Nuova Generazione (Next-Generation Network).

1.1. La Rete Telefonica Generale

Nell'era dei computer e della digitalizzazione, le reti VoIP rappresentano il naturale passaggio dall'offerta di servizi vocali via rete telefonica generale (una rete a commutazione di circuito), all'offerta degli stessi (ed ulteriori) servizi via rete a commutazione di pacchetto, quale la rete Internet. Questa transizione non è tuttavia immediata: il dominio che le reti PSTN hanno avuto nel corso degli ultimi decenni, assieme alle difficoltà nell'implementare le nuove tecnologie digitali, non hanno sicuramente facilitato il passaggio, specialmente quando si compara la qualità di una chiamata VoIP ad una via PSTN. L'efficacia di quest'ultima nella comunicazione vocale era dovuta alla rete a commutazione di circuito, e ad un sistema di indirizzamento fortemente legato alla locazione fisica. Ovvero, quando viene creata una chiamata su rete telefonica, la chiamata ha una quantità di banda dedicata costante sul proprio percorso, e dunque dispone sempre di un collegamento sufficiente per mantenere la telefonata, garantendo una latenza estremamente bassa dovuta solo alla propagazione fisica del segnale. Inoltre, il

protocollo di comunicazione è estremamente semplice da implementare, in quanto legato totalmente alla locazione fisica.

1.2. Il passaggio a Voice-over-IP

Le limitazioni delle PSTN vennero fuori con l'aumento di popolarità dei telefoni cellulari, quando sorse l'esigenza di associare ad un apparecchio telefonico non più la locazione dell'utente (che dovevano invece essere indipendenti), quanto la sua identità. Ed è qui che iniziarono ad entrare in gioco i servizi di telefonia via Internet. Nonostante fosse inizialmente difficile garantire una bassa latenza di banda a causa delle elaborazioni da eseguire sui singoli pacchetti che incapsulavano il traffico, con il miglioramento delle apparecchiature e dei collegamenti si è giunti a livelli accettabili di qualità del servizio (QoS). Sono stati però i decrescenti costi di implementazione e gestione delle reti VoIP rispetto a quelle PSTN il motivo principale che ha spinto una parte sempre maggiore delle imprese e delle pubbliche amministrazioni a passare a questi nuovi tipi di infrastrutture e servizi.

Per VoIP si intende una classe di prodotti e strumenti che consentano l'esecuzione di servizi di comunicazione avanzata tramite delle reti di dati. Anche se la comunicazione vocale è l'aspetto chiave di queste tecnologie, lo spettro dei servizi include anche video, condivisione di file, editing collaborativo, ecc. In una rete VoIP, due o più dispositivi hanno la possibilità di trasmettere e ricevere del traffico real-time che permette a due o più utenti di comunicare, appunto, in tempo reale in una sessione multimediale, e sfruttando come mezzo di comunicazione proprio una rete a commutazione di pacchetto basata sul protocollo IP (Internet Protocol). I fornitori di servizi di comunicazione stanno passando dal vecchio tipo di rete telefonica generale ad un nuovo modello di infrastruttura, detta di nuova generazione (NGN): una rete a pacchetto in grado di fornire agli utenti servizi di telecomunicazione grazie all'utilizzo di bande larghe, tecnologie abilitate alla qualità del servizio (QoS), e soprattutto dove le funzioni relative ai servizi sono indipendenti dalle relative tecnologie di trasporto sottostanti, favorendo così anche la mobilità.

Un'architettura VoIP è generalmente suddivisa in due parti principali: una fase di segnalazione, dove la sessione viene inizializzata e potenzialmente negoziata o terminata, ed una fase di trasferimento di media, dove i dispositivi finali si scambiano traffico multimediale. Vi sono diversi protocolli di segnalazione, tra cui SIP, un protocollo di inizializzazione di sessione standardizzato da IETF, e che ultimamente ha guadagnato molta popolarità. Questo protocollo si occupa sia di nozioni astratte quali indirizzamento e denominazione degli utenti, sia di funzioni concrete quali negoziazione di parametri, controllo di accesso, proxying, e billing. SIP ha guadagnato popolarità grazie ad una delle standardizzazioni di rete NGN, ovvero IMS, in cui SIP è appunto il protocollo di segnalazione. IP Multimedia Subsystem è la parte delle reti VoIP mobili 3G responsabile per fornire servizi multimediali (voce, video, e altri servizi dati).

Con il crescente numero di implementazioni di reti VoIP che vanno a sostituire le vecchie architetture di comunicazione, è fondamentale che la sicurezza di questi nuovi sistemi cresca in parallelo alla complessità. Un'unica infrastruttura di rete in grado di offrire un'elevata quantità e flessibilità di servizi trasportando diversi tipi di traffico merita una grande attenzione vista la diversità di tecnologie utilizzate. Una breccia nella sicurezza di uno dei suoi componenti può arrivare a compromettere l'intera rete, causando potenziali interruzioni del servizio e violazione della privacy, a danni dunque sia degli utenti che dei gestori.

1.3. Panoramica della tesi

La tesi propone, nel Capitolo 2, un'introduzione alle meccaniche del protocollo SIP, prerequisito necessario per comprendere il resto del documento. Si presenta quindi il protocollo di segnalazione in questione, descrivendone le entità coinvolte, i metodi che esse utilizzano per coordinarsi fra loro, e la struttura dei messaggi che incapsulano tali metodi. Nel Capitolo 3, si parla della tassonomia VoIPSA, che definisce le minacce alle reti VoIP, raggruppandole per categorie. Nel Capitolo 4, si spiega come avvengono gli attacchi a SIP, individuandoli all'interno delle classi di vulnerabilità precedentemente descritte, e descrivendone

l'attuazione a livello di meccaniche del protocollo. Nel Capitolo 5, si illustrano alcuni esempi di attacchi proposti, realizzati tramite l'uso di un analizzatore di traffico e di un generatore di messaggi SIP. Nel Capitolo 6 si riportano alcune tecniche e meccanismi di prevenzione proposti nel corso degli anni e tuttora utilizzati, a livello di protocollo SIP e a livello di rete. Si propone infine un esame di quelle che sono le effettive vulnerabilità rilevate nelle reti VoIP e come siano correlate con il protocollo di segnalazione utilizzato.

Capitolo 2

2. Session Initiation Protocol

Ci sono molte applicazioni su Internet che richiedono la creazione e gestione di una sessione, dove per sessione si intende uno scambio di dati tra due o più partecipanti. Come preannunciato, i servizi VoIP necessitano proprio di un protocollo che sia in grado di localizzare gli utenti e stabilire fra loro una sessione di comunicazione multimediale, dove gli utenti possano spostarsi fra gli endpoint, utilizzando più di un identificativo. SIP consente la creazione di un'infrastruttura di host di rete (chiamati server proxy) ai quali gli User Agent (UA) possono inviare registrazioni, inviti, ed altri tipi di richieste. E' un protocollo versatile, general-purpose, atto a creare, modificare, e terminare sessioni che operino in modo indipendente sia protocolli di trasporto sottostanti che dal tipo di sessione che si vuole creare.

In seguito viene presentato il protocollo di inizializzazione di sessione SIP, illustrandone le entità coinvolte ed i servizi offerti, ed il modo in cui avviene la comunicazione a livello intrinseco, analizzando i campi dell'intestazione di un messaggio SIP e i metodi di comunicazione disponibili. Infine, si accennano altri protocolli rilevanti con i quali SIP è strettamente collegato. Per la descrizione completa e formale di SIP, con tutte le estensioni e gli aggiornamenti continui applicati finora, si veda la documentazione RFC consultabile nel sito dell'IETF [1].

2.1. Generalità del Protocollo

SIP è un protocollo di strato applicativo, basato su righe di testo come HTTP, utilizza la codifica UTF-8, e di per sé non offre alcun tipo di servizio, ma piuttosto offre le primitive che possono essere usate per implementare diversi servizi. Quello che SIP offre sono quelle funzionalità tipiche della telefonia via Internet come trasferimento di chiamata, audio e video conferenze, dove gli utenti sono identificati con degli indirizzi, simili a quelli utilizzati per le email.

Essendo un protocollo basato su testo in chiaro, SIP presenta facilitazioni in fase di implementazione e troubleshooting, ma è fortemente vulnerabile a letture indesiderate dei parametri della sessione e a conseguenti offensive da parte di malintenzionati.

A differenza della rete telefonica generale, SIP distribuisce le funzionalità di processamento di una chiamata direttamente agli endpoint, basandosi su una logica di rete distribuita di tipo Peer-to-Peer, con l'aiuto di server proxy intermediari.

Il protocollo supporta cinque aspetti per stabilire e terminare sessioni multimediali:

- **User location:** la locazione dell'utente. Determinazione dell'endpoint che deve essere utilizzato per la comunicazione. Un'alterazione impropria dell'utente può generare un pericolo per la sicurezza.
- **User availability:** la disponibilità dell'utente. Determinazione della volontà della parte chiamata di partecipare alla conversazione, quindi è necessaria la definizione di meccanismi che impediscano l'impersonificazione di un utente.
- **User capabilities:** le capacità dell'utente. Decisione del tipo di media e i relativi parametri da utilizzare. La conoscenza di una terza parte malevola di queste capacità può portare all'invio di media indesiderati.

- **Session setup:** stabilimento dei parametri di sessione ad entrambi i capi della chiamata. Durante la fase di setup di una sessione, vengono scambiate le informazioni critiche per il corretto funzionamento della segnalazione. E' durante questa fase che possono avvenire attacchi di grosso impatto.
- **Session management:** la gestione della sessione. Possibilità di trasferire o annullare le sessioni, modificarne i parametri ed invocare i servizi. Risulta facile pensare che un malintenzionato possa intromettersi in una conversazione già stabilita e modificarne i parametri a proprio favore (o semplicemente a sfavore di uno dei partecipanti).

SIP è composto da transazioni fondate sul modello di richiesta-risposta (Request-Reply), ed è indipendente dagli strati più bassi del protocollo di trasporto. Un attaccante che intercetti una richiesta può raccogliere i parametri necessari per rispondere al fine di generare un attacco. La figura 1 mostra la configurazione più semplice del setup di un dialogo, quello che viene chiamato Sip trapezoid.

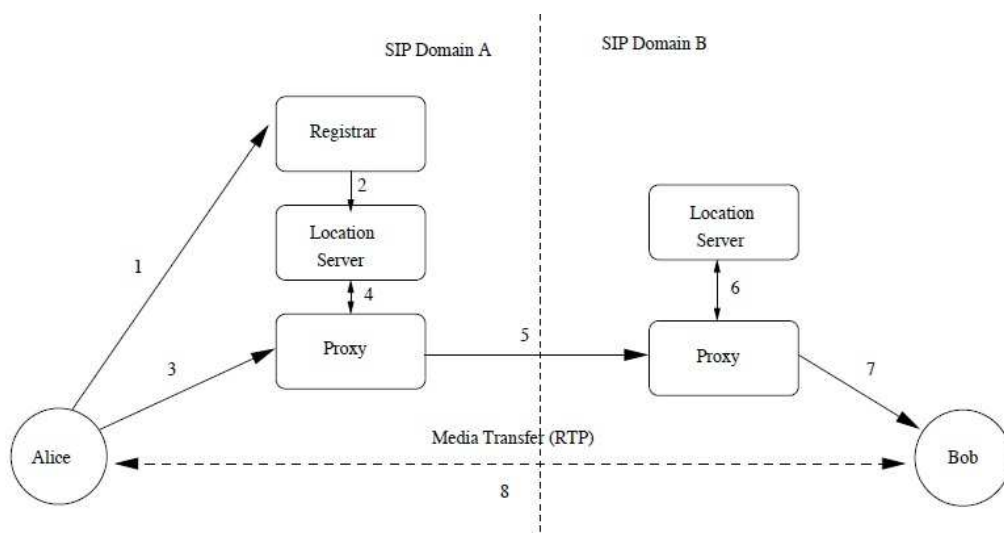


Fig. 1: Trapezoide SIP [2]

In un'ipotetica situazione, un utente Alice si registra con il registrar server del proprio dominio, domainA, che salva le informazioni nel location server. Alice ottiene il proprio Address-of-Record *sip:alice@domainA.com*, e desidera iniziare una chiamata con un utente Bob. Alice conosce l'indirizzo SIP di Bob, bob@domainB.com, non sa come instradare l'invito per conto proprio a Bob, ma è configurato in modo da inviare tutto il traffico in uscita attraverso il proxy SIP della propria azienda. Tale server proxy riconosce che il dominio di Bob è diverso da quello dell'azienda di Alice, e si occupa di instradare la richiesta di chiamata al proxy server della rete di Bob. Quest'ultimo, tramite le proprie informazioni sulla locazione di Bob, provvede ad inoltrargli la richiesta di Alice. La risposta di Bob segue il percorso a ritroso. Dopodichè, se Bob ha accettato la richiesta, la comunicazione può avvenire direttamente fra i due utenti Alice e Bob (salvo diverse politiche di rete che forzino tutto il traffico tramite i proxy). La call setup è quindi assimilabile ad un trapezoide.

2.2. Entità

Una sessione viene creata da due o più endpoint, detti User Agent (UA), che si distinguono tra client (UAC) e server (UAS), che rispettivamente iniziano una transazione SIP tramite un messaggio di richiesta, oppure gestiscono i messaggi SIP in arrivo. Oltre a queste entità, ve ne sono altre intermedie che aiutano gli UA durante la segnalazione. Un attacco che coinvolga una di queste entità può avere forti ripercussioni sul servizio offerto dalla rete:

- **Proxy server:** processano autonomamente sia le richieste che le risposte e le instradano. Possono o meno mantenere lo stato della chiamata (stateful o stateless proxy rispettivamente) a seconda del controllo che si vuole esercitare sullo stato della comunicazione. Sostanzialmente, uno stateless proxy si limita ad inoltrare i messaggi che riceve, non garantendo nessuna funzione aggiuntiva ma risultando molto performante. Uno stateful proxy invece offre controllo sulla sessione e sull'instradamento del traffico. Un utente

malintenzionato potrebbe decidere sia di effettuare un attacco ad un server proxy con lo scopo di metterlo fuori uso e quindi impedire agli UA di stabilire una sessione fra loro, oppure potrebbe spacciarsi per un proxy stateless e monitorare segretamente il traffico o addirittura impersonare un proxy stateful, con il potere di modificare pesantemente i parametri della sessione.

- **Back-to-Back User Agent (B2BUA):** entità logica che riceve una richiesta e la processa come se fosse uno UAS. Per determinare come debba rispondere ad una richiesta, si comporta come uno UAC e genera richieste. A differenza di un proxy stateful, non si limita a mantenere lo stato del dialogo fra gli UA partecipanti ad una sessione, ma crea un dialogo separato con ciascuno di essi, e deve partecipare a tutte le richieste inviate nei dialoghi che ha stabilito. Inutile dire che una terza parte malevola che si spaccia per un B2BUA assuma il controllo totale della sessione fra i partecipanti originali, con la possibilità di modificare tutto il traffico da esso passante a proprio piacimento. Questo tipo di attacco è tra i più difficili da neutralizzare ed è conosciuto come Man-in-the-Middle (letteralmente, uomo-nel-mezzo).
- **Registrar Server:** accetta le richieste REGISTER. Riceve informazioni di registrazione dagli utenti, ne estrae le informazioni riguardanti la loro locazione (indirizzo IP, porta, e nome utente) e le mantiene in un location database. Tipicamente, è collocato assieme ad un proxy o ad un redirect server e può offrire servizi di locazione. Il servizio di registrazione permette ad uno UA di far sapere al proxy o redirect server a quale indirizzo o indirizzi può essere raggiunto. Uno UA può anche utilizzarlo per installare delle particolari opzioni di gestione delle chiamate direttamente sul server. Un attaccante può spacciarsi per un registrar server per negare o ridirigere una richiesta di registrazione di uno UA, impedendogli rispettivamente di accedere al servizio da lui richiesto o di forzarlo ad utilizzare un server esterno malevolo.
- **Location Server:** è utilizzato da un proxy o redirect server per ottenere informazioni riguardo ad una possibile locazione di un'entità chiamante. Può essere collocato assieme ad un server SIP. Interrogazioni non autorizzate ad

un location server possono fornire dati utili ad un malintenzionato per generare un attacco contro gli UA registrati nel relativo dominio.

- **Redirect Server:** riceve una richiesta e risponde con un messaggio contenente la locazione corrente di un particolare utente, consultando il location database. Un attaccante può spacciarsi per un redirect server per ridirigere le chiamate verso una terza parte esterna al dominio dell'utente.

Uno UA ha associato quello che viene chiamato Address-of-Record (AOR), definito formalmente come un SIP o SIPS URI che punta ad un dominio con un location service che può mappare l'URI con un altro URI dove l'utente potrebbe essere disponibile (example: "sip:alice@domain.com"). Sostanzialmente, è l'indirizzo pubblico dello UA, molto simile come formato ad un indirizzo email.

2.3. Struttura di un Messaggio

SIP è un protocollo basato su linee di testo, basato sul modello di transazione di richiesta/risposta. Ciascuna transazione consiste in una richiesta che invoca un particolare metodo sul server, ed almeno una risposta. Ciascun messaggio SIP è composto dunque da una first line, da un header, e dal body. La struttura tipica, nell'esempio di una richiesta, è mostrata in figura [2]. La maggior parte degli attacchi al protocollo avviene proprio origliando i parametri dell'header di una richiesta o eventualmente anche di una risposta, andando a modificare in maniera non autorizzata la segnalazione e il corretto setup della sessione.

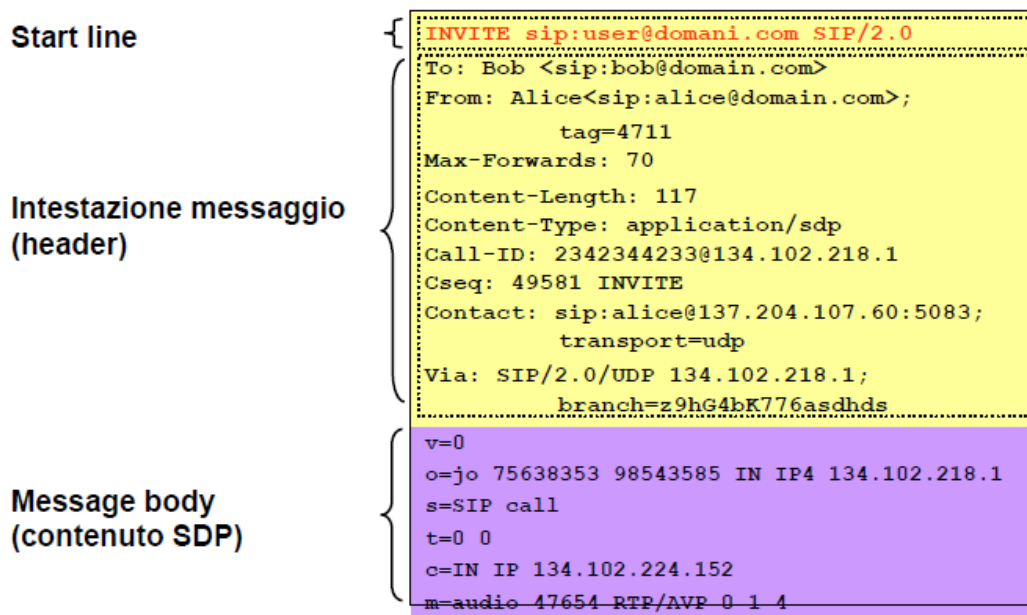


Fig. 2: Un esempio di una richiesta SIP [3]

La first line, la prima riga del messaggio, ne definisce il tipo di richiesta o di risposta, per esempio un INVITE nel primo caso, o un 200OK nel secondo. Una richiesta serve ad iniziare una transazione, oppure a dare un qualche tipo di informazione al destinatario. Le risposte confermano o rigettano la richiesta e chiudono la transazione se non sono provvisorie. Più precisamente quindi, la first line contiene il metodo, la Request-URI, e la versione del protocollo SIP in uso, enunciati in quest'ordine nelle richieste, ed in ordine inverso nelle risposte.

L'header contiene i campi di intestazione, ovvero attributi che forniscono informazioni aggiuntive relative al messaggio e soprattutto alla sessione da stabilire o da rinegoziare, a seconda del tipo di metodo invocato. Un messaggio di INVITE, ad esempio, include un identificativo univoco per la chiamata, l'indirizzo sorgente e di destinazione, e ulteriori informazioni sul tipo di sessione che si vuole stabilire. In seguito è riportata una lista di campi header:

- **Via:** contiene l'indirizzo al quale il chiamante si aspetta di ricevere la risposta alla propria richiesta, ed un **branch parameter** che identifica la transazione corrente. Tipicamente è l'indirizzo del chiamante oppure del proxy SIP della rete di appartenenza. Una risposta valida alla richiesta

corrente necessita dello stesso branch parameter. Inserendo come campo Via il proprio indirizzo o quello del proxy della propria rete, una terza parte potrebbe ridirigere tutte le future risposte dei partecipanti, facendole transitare per percorsi non legittimi.

- **To:** contiene il nome utente ed il SIP URI del destinatario verso cui la richiesta è originariamente diretta.
- **From:** è l'indirizzo del chiamante, cioè il suo nome utente ed il proprio SIP URI. Oscurando questo campo, un utente potrebbe generare una richiesta anonima.
- **Call-ID:** è l'identificatore univoco della chiamata, generato dalla combinazione di una stringa casuale e del nome host o indirizzo IP dell'apparecchio chiamante. La combinazione del To tag, From tag, e della Call-ID definiscono in maniera completa una relazione peer-to-peer fra due endpoint, ovvero un dialogo. Per potersi intromettere in maniera valida in una conversazione, l'attaccante deve generare richieste o risposte utilizzando la stessa Call-ID del dialogo originale.
- **CSeq:** contiene un intero ed il nome del metodo della richiesta della transazione corrente. Il sequence number viene incrementato ad ogni nuova richiesta interna allo stesso dialogo, ed è un normale numero di sequenza. Un attaccante deve rispondere ad una richiesta con lo stesso CSeq o generare una nuova richiesta con il successivo CSeq del dialogo corrente per infiltrarsi nella sessione.
- **Contact:** contiene l'AOR che rappresenta una route diretta per contattare il chiamante, quindi dove il chiamato deve inviare future richieste, a differenza del campo Via che indica dove il chiamato dovrà inviare invece future risposte. Un malintenzionato può inserire il proprio AOR o l'indirizzo di un server di terze parti per ridirigere una richiesta di uno UA.
- **Max-Forwards:** limita il numero di salti che una richiesta può fare per arrivare a destinazione prima di venire eliminata. E' un intero che viene

decrementato ad ogni salto. Modificando questo valore, un attaccante può porre termine al passaggio di un messaggio.

- **Content-Type:** è una descrizione del body del messaggio. Può essere usato da un attaccante per aggiungere un messaggio testuale per l'utente senza bisogno di modificare il body, spesso a fini di SPAM.
- **Content-Length:** è un contatore di byte del body del messaggio, utilizzato per farne un match con l'effettiva lunghezza. Una modifica a questo parametro permette ad un malintenzionato di danneggiare la sessione o potenzialmente anche il sistema a causa di un mismatch.
- **Allow:** è la lista dei metodi supportati dallo UA. Modificando questo campo un attaccante può apportare forti restrizioni ad una sessione, negando ad esempio l'uso di metodi di sicurezza.

I campi sopra elencati sono gli attributi principali di un messaggio header SIP, ma ve ne sono molti altri opzionali che verranno presentati in seguito se necessario. Una nota rilevante riguarda i campi From, To, Call-ID e CSeq di una risposta che coincidono con quelli della relativa richiesta, ed i valori del campo Via che rimangono gli stessi e nello stesso ordine. Vi sono altri campi, come WWW-Authenticate e Authentication che verranno descritti in seguito, nel capitolo riguardante le contromisure agli attacchi.

I dettagli della sessione come i tipi di media, codec, ecc. non sono descritti utilizzando SIP. Bensì, il body di un messaggio SIP contiene una descrizione della sessione formattata utilizzando un altro protocollo, come il Session Description Protocol (SDP). Il messaggio SDP è trasportato nel messaggio SIP in un modo analogo al quale un documento viene allegato ad un'email.

2.4. Metodi

I metodi SIP sono dunque suddivisi in richieste e risposte, distinguibili dalla first line dove viene definito anche il metodo invocato. I principali tipi di richieste sono i seguenti (i metodi di richiesta non citati sono relativi alle estensioni di SIP):

- **INVITE:** metodo principale, indica che uno UAS viene invitato da uno UAC a partecipare ad una sessione. Nello stabilire una chiamata, è il primo metodo invocato. Un attaccante può generare richieste di INVITE malevole per generare vari attacchi.
- **ACK:** conferma che lo UAC ha ricevuto una risposta finale dello UAS ad una richiesta di INVITE. E' richiesto per completare con successo l'instaurazione di una sessione visto che, a causa della natura asincrona degli inviti, SIP si basa sul modello 3-way hand-shaking.
- **CANCEL:** cancella una qualsiasi richiesta pendente, cioè prima che il destinatario abbia risposto con una risposta di tipo definitivo. Un invio non autorizzato di una CANCEL può impedire il setup di una sessione.
- **BYE:** termina una chiamata. Può essere inviato da qualunque dei chiamanti. Un invio non autorizzato di una CANCEL può impedire il setup di una sessione. Un invio non autorizzato di un BYE può terminare prematuramente una sessione.
- **REGISTER:** registra nel server l'indirizzo nel campo To dell'header del messaggio, cioè informazioni riguardo la locazione dell'utente che ha generato la richiesta. Bloccare o falsificare richieste di REGISTER può portare a disservizi.

Un metodo INVITE inviato durante la fase di negoziazione della sessione è detto RE-INVITE, e serve per cambiare i parametri di comunicazione, come indirizzi o porte, oppure codec e media stream da utilizzare. Si realizza inviando

una nuova richiesta di INVITE nello stesso dialogo che ha stabilito la sessione. Un malintenzionato potrebbe usare un RE-INVITE per modificare una sessione già esistente a proprio favore.

Le risposte SIP si distinguono invece secondo un criterio di numerazione, che serve a classificarne la tipologia. Il nome di una risposta è caratterizzato da un numero a tre cifre, di cui la prima ne definisce appunto la categoria:

- **1xx: Provisional.** Indica che la richiesta è stata ricevuta e sta venendo processata. Lo UA chiamante deve interrompere la ritrasmissione della richiesta. Raramente vengono utilizzate per generare un'offensiva.
- **2xx: Success.** La richiesta è stata ricevuta, elaborata con successo, ed accettata.
- **3xx: Redirection.** Servono ulteriori operazioni per completare la richiesta (ad esempio, il destinatario della chiamata si trova ad un altro indirizzo). Utilizzate per generare alcuni tipi di attacco dove il malintenzionato intende ridirigere la chiamata verso una terza parte non valida o fraudolenta (incluso sé stesso).
- **4xx: Client Error.** La richiesta contiene una sintassi errata o non può essere soddisfatta dal server, e non può essere dunque processata. Può implicare una ritrasmissione (ad esempio con una richiesta di autenticazione da parte del mittente della richiesta). Falsi errori possono essere usati da un attaccante mascherato da server per impedire l'accesso di uno UA ad un servizio qualunque, ma risposte legittime di errore vengono usate regolarmente per richiedere parametri aggiuntivi ad uno UA proprio al fine di evitare certi tipi di attacchi. Ad esempio, una risposta 407 Proxy Authentication Required viene inviata da un proxy server a seguito di una richiesta di uno UA per richiedere informazioni aggiuntive che ne verifichino la vera identità.
- **5xx: Server Error.** Il server non è riuscito a soddisfare una richiesta apparentemente valida. Un attaccante che impersoni un server può inviare

false risposte di errore per impedire ad un utente di accedere ai servizi di rete.

- **6xx: Global Failure.** Nessun server è in grado di soddisfare la richiesta. Stesse considerazioni del caso precedente.

2.5. Protocolli Correlati

SIP si basa su altri protocolli ben conosciuti per creare la sessione dati ed assicurare una comunicazione relativamente sicura. I principali protocolli sono:

- **SDP:** E' il protocollo utilizzato per descrivere il body di un messaggio SIP. E' il meccanismo utilizzato dagli endpoint per negoziare i parametri della sessione dati. SDP permette al chiamante di pubblicizzare quali protocolli di comunicazione (es. Speex, GSM) che esso è in grado di utilizzare, protocolli che sono fondamentali per codificare voce, video, e altri dati multimediali. Il modello che SIP utilizza con SDP è quello di offerta/risposta, ovvero dove colui che ha iniziato la sessione offre al ricevente tutte quelle opzioni desiderate, e la risposta di quest'ultimo determina i parametri specifici della conversazione scegliendo un sottoinsieme di opzioni offerte dal chiamante.
- **RTP:** I pacchetti RTP sono contenitori per i dati della sessione, come voce o video in una sessione SIP. Funzioni importanti sono time-stamping e data sequencing per correggere il jitter di latenza (la variazione del ritardo sulla comunicazione). I pacchetti RTP possono essere trasportati tramite protocollo TCP o UDP, ma essendo un problema maggiore i ritardi che la perdita di pacchetti in una comunicazione real-time, solitamente il trasporto avviene tramite UDP.
- **URI:** E' il meccanismo usato da SIP per descrivere le locazioni degli utenti e dei server. Gli URI solitamente consistono in uno schema SIP o SIPS, nome utente o del server, e nome dell'host quale dominio o indirizzo IP.

- **TLS:** E' un protocollo creato per fornire una comunicazione privata via Internet, grazie all'utilizzo di autenticazione tramite crittografia con chiave pubblica. L'uso di questo protocollo in SIP è limitato per via del metodo multi-hop col quale le richieste SIP viaggiano da uno UA all'altro, che non garantisce segretezza da endpoint a endpoint. Inoltre, usare TLS da UA a UA con segretezza garantita non è possibile a meno che uno degli UA abbia un certificato con una "common trust chain" con l'altro UA. TLS è molto utile, tuttavia, nell'autenticare i server, e verrà visto in maniera più dettagliata in seguito nel capitolo sulle contromisure.

Avendo dunque una panoramica del protocollo SIP, delle sue entità e delle loro interazioni, si possono andare ad analizzare le classi di vulnerabilità di una rete che utilizzi il protocollo SIP come strumento di segnalazione, andando a capire quale sia l'importanza di SIP nelle falle di sicurezza di questo sistema.

2.6. Considerazioni

La potenza di SIP risiede nel fatto che è un protocollo agile, ovvero indipendente dal tipo di sessione che si vuole stabilire, o dal tipo di media usato, che lo rende un'ottima scelta quando si vogliono integrare in una rete servizi che offrano diversi tipi di media.

Essendo basato su righe di testo, i messaggi SIP sono facili da leggere ed interpretare, semplificando il monitoraggio del corretto funzionamento della rete per quanto riguarda la segnalazione. Tuttavia, processare messaggi di testo è un'operazione più onerosa per le apparecchiature di rete, come i router ad esempio, in quanto devono essere tradotti in una forma ad esse comprensibile.

Qualsiasi protocollo di comunicazione, per definizione, implementa delle meccaniche che permettano uno scambio di informazioni, nonché una qualche

forma di sincronizzazione. Come spesso accade, uno o più aspetti di un sistema possono essere usati per andare oltre a quello che dovrebbe essere il corretto e, soprattutto, voluto funzionamento. Il protocollo SIP non fa eccezione: le meccaniche che regolano la comunicazione fra due o più entità SIP possono essere usate per compiere degli attacchi al protocollo che impattano sul corretto funzionamento del sistema e sull'esperienza d'uso degli utenti.

La descrizione di SIP presentata in questa sezione serve a dare un panoramica generale al lettore riguardo le entità coinvolte e le funzionalità che permettono loro di comunicare. Gli attacchi a SIP avvengono proprio sfruttando la conoscenza delle sue dinamiche: conoscendo le meccaniche di un metodo, si può andare ad usare o addirittura a modificare i campi dell'intestazione di un messaggio SIP, per generare un comportamento corretto ma indesiderato (da parte della rete) della segnalazione.

Prima di andare a studiare nel dettaglio quali siano precisamente gli attacchi conosciuti e come vengano attuati a livello di protocollo SIP, si fornisce nel prossimo capitolo una visione di quali siano le classi di vulnerabilità di una rete VoIP generica, dando maggiore importanza a quelle che racchiudono proprio gli attacchi alla segnalazione.

Capitolo 3

3. Classi di Vulnerabilità VOIP

Mantenere sicura una rete VoIP ed i suoi utenti richiede un'analisi attenta dei suoi componenti e di come questi interagiscano fra loro, così da poter monitorare le entità coinvolte e controllarne i loro comportamenti. A tal scopo, è importante comprendere quali siano le classi di vulnerabilità, da quali parti del sistema possano essere generate, e come poterle prevenire.

3.1. La Tassonomia VoIPSA

Una definizione delle minacce alla sicurezza VoIP è stata realizzata da VoIPSA. Voice over IP Security Alliance (VoIPSA) è un'organizzazione non-profit indipendente composta da venditori di servizi VoIP e sicurezza, ed organizzazioni ed individui con un interesse nella sicurezza dei protocolli e delle installazioni VoIP, che ha definito una tassonomia delle minacce alla sicurezza VoIP [4] al fine di fornire un documento informativo di pubblico dominio.

Gli elementi chiave di questa tassonomia sono:

- **Social threats:** attacchi diretti a persone. Ad esempio, bug, configurazioni, e cattive interazioni fra procolli in sistemi VoIP possono facilitare attacchi che riportino una falsa identità degli attaccanti agli altri utenti della rete. Questo tipo di azioni può portare a conseguenti attacchi come phishing, furto del servizio, o contratti non desiderati.

- **Eavesdropping, interception, and modification threats:** tutte le situazioni dove un attaccante può, violando la legge e senza autorizzazione dalle giuste parti, ascoltare la segnalazione o il contenuto di una sessione VoIP, e persino modificare aspetti della sessione senza venire individuato. Esempi di questi attacchi sono ad esempio il reinstradamento di chiamata e l'intercettazione di una sessione RTP non cifrata.
- **Denial of service threats:** minacce che hanno le potenzialità di negare l'accesso ai servizi VoIP da parte degli utenti. Può essere particolarmente problematico in caso di emergenze, o quando un attacco DoS colpisce tutte le capacità di comunicazione di un utente o dell'intera rete, ad esempio non solo il traffico voce ma anche quello dati che passa per la stessa rete. Gli attacchi DoS possono essere specifici a VoIP (sfruttando difetti nel setup di una chiamata) oppure agnostici a VoIP (generando un generico attacco di traffic flooding). Possono infine riguardare attacchi ad apparecchiature e collegamenti fisici.
- **Service abuse threats:** attacchi che coprono l'uso improprio di servizi VoIP, specialmente (ma non esclusivamente) in quelle situazioni dove tali servizi sono offerti in maniera commerciale. Tali esempi includono toll fraud e billing avoidance, cioè dove l'attaccante usa un servizio senza pagare o facendo pagare un altro utente al suo posto.
- **Physical access threats:** minacce che si riferiscono ad un accesso inappropriato o non autorizzato ad apparecchiature VoIP, o allo strato fisico di una rete (secondo la pila ISO/OSI).
- **Interruption of service threats:** attacchi che si riferiscono a problemi non intenzionali che possono comunque portare dei servizi VoIP ad essere inutilizzabili o inaccessibili, come ad esempio perdita di corrente per cause naturali, o semplicemente problemi di prestazioni che possono degradare la qualità delle chiamate.

La tesi concentra l'attenzione su quelle classi di vulnerabilità che coinvolgono direttamente il protocollo SIP, cioè le prime tre. Queste classi coprono tutti quegli attacchi che possono compromettere la sicura fase di segnalazione delle sessioni stabilite tra gli utenti della rete ed il corretto utilizzo dei servizi offerti.

3.2. Social Threats

Gli attacchi pericolosi a livello sociale ricadono nell'ambito di quella che viene definita *ingegneria sociale*, ovvero dove si induce con l'inganno un utente ad eseguire azioni che recheranno un danno a lui o al sistema (banalmente detta "truffa").

Un attacco a persone può avvenire tramite **misrepresentation** (travisamento in italiano), dove l'attaccante, utilizzando una falsa identità presenta all'utente delle informazioni fuorvianti, che lo inducono a fornire dati sensibili o ad utilizzare servizi non sicuri. L'esempio chiave di questo tipo di minacce è conosciuto come phishing, una delle truffe che ha accompagnato la crescita dell'utilizzo delle caselle di posta elettronica. Spacciandosi per uno dei contatti dell'utente o per un'organizzazione dove egli è iscritto, un malintenzionato invia alla vittima un'email con contenuti che la inducano ad accedere ad un sito esterno per utilizzarne i servizi o ad aprire un allegato dannoso, tramite i quali l'attaccante può raccogliere dati sensibili della vittima quali ad esempio nome account, password, e numero di carta di credito per avere poi accesso ai suoi dati bancari. Questo tipo di attacco prende anche un'altra forma, dove il malintenzionato si presenta all'utente come un'autorità quali le forze dell'ordine, presentando una parvenza di autenticità nel messaggio riportando alcuni dati sensibili della vittima, al fine di indurla a fornirne altri più importanti per chi compie l'attacco.

Un'altra categoria di attacchi sociali comprende quelli che vengono definiti **unwanted contact** (o contatti indesiderati). Una volta ottenuto l'indirizzo o il contatto di una vittima, un attaccante può inviarle senza il suo consenso messaggi, chiamate, e altri tipi di media, contenenti una qualsiasi forma di materiale che

rappresenti un comportamento offensivo alla persona, minatorio, o comunque perseguibile per legge, in maniera anche ripetitiva.

Infine, vi sono anche attacchi sociali che danneggiano i fornitori del servizio, come i **service thefts** (furto dei servizio) VoIP. Un malintenzionato trae un beneficio economico ai danni del fornitore cancellando senza permesso o alterando le fatture riguardanti i servizi da lui utilizzati, oppure bypassando direttamente la fase di fatturazione.

3.3. Eavesdropping

Attacchi di tipo eavesdropping (origliare in italiano) descrivono un metodo con il quale un malintenzionato è in grado di monitorare l'intera sessione di segnalazione o di scambio dati tra due o più endpoint di una rete VoIP, ma senza alterarne il traffico. Si limita quindi ad ascoltare in segreto le conversazioni.

Un attaccante dunque può effettuare diverse operazioni, tutte non autorizzate, mentre origlia il traffico passante:

- **Call Pattern Tracking:** il traffico proveniente da o diretto a un qualsiasi nodo della rete viene analizzato al fine di scoprire l'identità delle entità coinvolte e le abitudini nell'utilizzo dei servizi della rete (ad esempio, gli orari o i giorni in cui un utente telefona con una certa regolarità, i servizi che utilizza, ecc). Un'analisi di questo genere porta tendenzialmente a conseguenti attacchi di tipo sociale come il phishing.
- **Traffic Capture:** il traffico catturato viene registrato, vengono cioè memorizzati nel dispositivo attaccante i pacchetti, il relativo contenuto, e le informazioni riguardanti il loro percorso sulla rete (ad esempio, data e ora di una chiamata).
- **Number Harvesting:** il traffico catturato consiste nella raccolta di informazioni contenute nell'header dei pacchetti dove sono indicati indirizzi

IP, email, numeri di telefono, e altri tipi di identificativi degli utenti della rete, al fine di utilizzarli in seguito per altri tipi di attacchi.

- **Conversation Reconstruction:** il traffico audio viene registrato al fine di estrarne informazioni e contenuti rilevanti al fine di riprodurli successivamente (potenzialmente modificandolo in seguito) in attacchi detti *replay attacks*.
- **Voicemail Reconstruction:** viene catturato e riprodotto il traffico riguardante messaggi vocali.
- **Fax Reconstruction:** viene catturato e riprodotto il traffico riguardante messaggi fax.
- **Video Reconstruction:** viene catturato e riprodotto il traffico riguardante qualsiasi media di tipo visivo.
- **Text Reconstruction:** viene catturato e riprodotto il traffico riguardante messaggi testuali.

Gli attacchi di eavesdropping costituiscono solitamente una prima fase necessaria per effettuare tipi di attacchi più complessi e più dannosi. Vengono utilizzati quindi inizialmente per raccogliere dei dati rilevanti sulle vittime.

3.4. Interception and Modification

Attacchi di intercettazione e modifica descrivono un metodo secondo il quale un attaccante possa vedere l'intera fase di segnalazione e stream dati tra due endpoint, e possa anche modificare il traffico come se fosse un intermediario in una conversazione (sono dunque un'estensione dell'eavesdropping). I vari tipi di attacchi, ovviamente eseguiti tutti senza autorizzazione, sono i seguenti:

- **Call Black Holing:** consiste nel creare un "buco nero" nella chiamata, ovvero di impedire di inoltrare uno o più elementi essenziali in un protocollo VoIP, quale SIP ad esempio, al fine di impedire la creazione, modifica o semplice proseguimento di una sessione di chiamata, con il risultato di terminarla prematuramente, bloccando il passaggio di alcuni messaggi o modificandone dei campi essenziali nell'intestazione.
- **Call Rerouting:** si tratta di ridirigere un indirizzo IP o un altro elemento essenziale di un protocollo VoIP con l'effetto di divergere la comunicazione da nodi intermediari autorizzati, convergendo invece su altri nodi esterni alla rete. E' bene precisare che, se una procedura di call rerouting è autorizzata all'interno della rete, può essere usata invece come meccanismo difensivo contro attacchi o come abilitazione ad altri servizi.
- **Conversation Alteration:** si tratta di modificare qualsiasi elemento o porzione di informazione audio, video o testuale di una comunicazione, inclusi l'identità, lo stato o le informazioni di presenza di un utente.
- **Conversation Degrading:** consiste nel ridirigere o modificare il traffico al fine di deteriorare la QoS all'interno della rete, di limitare o rendere frustrante la comunicazione.
- **Conversation Impersonation and Hijacking:** è la modifica, cancellazione, aggiunta, o sostituzione o qualsiasi altro tipo di modifica di una porzione di una qualsiasi comunicazione che ne alteri il contenuto o l'identità, la presenza o lo stato di una delle parti coinvolte, applicata a qualsiasi tipo di conversazione e media.
- **False Caller Identification:** è la semplice segnalazione di una falsa identità o presenza di un'entità interna alla rete.

3.5. Denial of Service Threats

La classe di vulnerabilità riguardante le negazioni dei servizi è quella che comprende il maggior numero di attacchi. Un'impedimento ad utilizzare un servizio della rete può essere effettuato in diversi modi: la rete può venire congestionata da un esubero di richieste che ne satura e potenzialmente danneggia le apparecchiature che le devono gestire, siano esse richieste valide o richieste malformate che ne causano comportamenti anomali, oppure le richieste dell'utente possono essere intercettate e reindirizzate verso altre entità che non offrono i servizi della rete a cui un utente è connesso.

3.5.1. Request Flooding

Comprende tutti quegli attacchi che mirano ad opprimere un bersaglio con un numero elevato di richieste, siano esse valide o meno. Il modus operandi ed i risultati sono i seguenti:

- **User Call Flooding:** l'attaccante invia un numero elevato di valide richieste. Il dispositivo endpoint è in grado di processare le richieste, ma le sessioni dell'utente vengono ripetutamente interrotte.
- **User Call Flooding Overflowing to Other Devices:** è come il caso precedente, con l'aggiunta però che alcune chiamate possono sovraccaricare altre risorse intermedie, quali ad esempio server mail o gateway per le chiamate.
- **Endpoint Request Flooding:** viene inviato un elevato numero di richieste valide o meno di setup della chiamata (ad esempio, dei messaggi SIP INVITE) che possono causare un crash o un riavvio all'apparecchio endpoint.
- **Endpoint Request Flooding after Call Setup:** è analogo al caso precedente, con la differenza che i messaggi inviati sono di controllo della chiamata (es. SIP RE-INVITE), che causano un crash o un riavvio

all'apparecchio endpoint, e possono finire per terminare le sessioni esistenti.

- **Call Controller Flooding:** invio di un numero eccessivo di messaggi di call setup (es. SIP INVITE) che possono causare il crash o il riavvio di un call controller quale un server proxy, che può lasciare un'intero insieme di endpoint senza la possibilità di iniziare o ricevere chiamate.
- **Request Looping:** come il caso precedente, ma vengono utilizzati due endpoint di due domini differenti per congestionare le risorse del server proxy, sfruttando un ciclo o una spirale in una implementazione impropria del server.
- **Directory Service Flooding:** invio di un numero elevato di interrogazioni ad un server (es. DNS server) che ne causa il crash o il riavvio.

3.5.2. Malformed Requests and Messages

In questo caso, vengono inviati dei messaggi di richiesta non validi, che causano il crash di endpoint o entità intermedie. Essendo le specifiche dei messaggi di controllo in molte implementazioni deliberatamente open-ended, al fine di permettere l'aggiunta di ulteriori capacità nel tempo, si ha che non si riesce a testarne l'implementazione per un processamento corretto di tutti i messaggi validi o per riconoscere accuratamente quelli invalidi. Di conseguenza, messaggi validi ma troppo complessi rischiano di venire scartati, o ancora peggio, messaggi non validi sufficientemente complessi ma ben architettati possono essere accettati da un proxy o da un endpoint, con la conseguente causa di un processo auto-distruttivo nel comportamento di questi elementi, solitamente un overflow del buffer dei messaggi ed un conseguente crash o riavvio.

3.5.3. QoS Abuse and Spoofed Messages

QoS abuse si è un caso di DoS in cui un attaccante viola la QoS negoziata durante il setup della chiamata, come usare un codec media differente da quello prestabilito.

Spoofed Messages sono la conseguenza di messaggi intercettati da un attaccante, il quale procede poi ad iniettarne di fasulli nel percorso della segnalazione. Riuscendo a farli riconoscere come veri dal destinatario o da un proxy intermediario, porta a diversi risultati:

- **Faked Call Teardown Message:** interruzione del servizio causando la fine prematura di una sessione, negando quindi l'utilizzo corretto dei servizi per la vittima. Nel caso di SIP ad esempio, l'invio di un messaggio BYE valido termina la relativa chiamata prematuramente.
- **Faked Response:** invio di una risposta di errore ad una richiesta di chiamata verso un altro utente, negando quindi l'inizializzazione della sessione alla vittima.

3.5.4. Call Hijacking

Una volta compromessa la sicurezza, un sistema è suscettibile a quegli attacchi che mirano a rubare informazioni scambiate durante sessioni tra endpoint VoIP e la rete. Il furto di una chiamata dunque avviene quando le transazioni VoIP sono in mano ad un attaccante. Questo porta quindi non solo ad un semplice DoS, ma anche ad un potenziale utilizzo di un servizio esterno. Alcuni esempio sono:

- **Registration Hijacking:** impedisce alle vittime di registrarsi presso il registration server desiderato, facendole registrare invece in un location database predisposto dall'attaccante che permette a quest'ultimo di ridirigerne e controllarne le future sessioni che le vittime cercheranno di stabilire.

- **Media Session Hijacking:** l'attaccante "ruba" una sessione multimediale, imponendosi come nuovo endpoint destinatario dei dati, e negando quindi una corretta comunicazione alla vittima chiamante e impedendo alla vittima chiamata di ricevere dati.
- **Server Masquerading:** l'attaccante impersona un server VoIP, convincendo la vittima ad inviare a lui richieste di comunicazione, impedendo quindi all'endpoint di utilizzare i servizi.

3.6. Considerazioni

Rispetto alle altre classi di vulnerabilità che riguardano i servizi VoIP, il denial of service sembra essere di forte impatto non solo per i singoli utenti, ma per il sistema VoIP in generale, cioè l'intera rete. A tal proposito, attacchi di questo tipo meriterebbero una forte attenzione da parte di amministratori e gestori dei servizi, senza ovviamente sottovalutare l'importanza della prevenzione di qualsiasi tipo di attacco possa essere apportato alla rete.

Nel capitolo seguente si analizzano queste tre classi di vulnerabilità appena viste, guardando come questi attacchi siano attuabili modificando i messaggi SIP, e riportandone degli esempi.

Capitolo 4

4. Attacchi a SIP

Ipotizzando di avere a che fare con una rete VoIP senza difese, un attacco a SIP richiede giusto l'utilizzo di un apposito software ed una conoscenza più o meno approfondita delle meccaniche del protocollo a seconda dello strumento scelto. Considerando la facilità di reperire sulla rete Internet un programma del genere con tanto di relativa guida all'uso, il rischio di subire un attacco da parte di malintenzionati non è più trascurabile. Non si pensi solo ad attaccanti professionisti, ma anche a quelli che, nella cultura hacker, vengono sminuiti col soprannome di *script kiddies*, ossia persone, spesso adolescenti, con scarse competenze informatiche che utilizzano script o programmi sviluppati da altri per danneggiare reti o sistemi informatici.

Si propone un'analisi dei principali tipi di attacco a SIP identificando ciascuno in una delle categorie della tassonomia VoIPSA, studiandone le relative informazioni chiave fornite dalle meccaniche di comunicazione del protocollo.

4.1. Eavesdropping, Interception e Modification

Per effettuare un eavesdropping, basta catturare il traffico VoIP scambiato fra due utenti di due reti qualsiasi. L'intercettazione può avvenire "via cavo", cioè collegandosi fisicamente ad un apparato di rete che inoltri i pacchetti della chiamata bersaglio, oppure ancora più facilmente sniffando ad esempio un segnale radio di un router wireless.

Di seguito, è riportato un esempio [5] delle informazioni esposte in una generica richiesta di INVITE SIP:

INVITE sip:123456789@192.168.100.100:5060 SIP/2.0

Via: SIP/2.0/UDP 10.10.200.200:5060;branch=z9hG4bK38475930491

From: Alice <sip:987654321@10.10.200.200:5060>;tag=192837

To: Bob <sip:123456789@192.168.100.100>

Call-Id: 987654321-0001

CSeq: 1 INVITE

Contact: <sip:987654321@10.10.200.200>

Expires: 1200

Max-Forwards: 70

Content-Type: application/sdp

Content-Length: 143

Session Description Protocol Version (v): = 0

Owner/Creator, Session Id (o): 2 2 2 IN IP4 10.10.200.200

Session Name (s): Session SDP

Connection Information (c): IN IP4 10.10.200,200

Media Description, name and address (m): audio 9876 RTP/AVP 0 8 18

Media Attribute (a): rtpmap:0 PCMU/8000

Media Attribute (a): rtpmap:8 PCMA/8000

Media Attribute (a): rtpmap:18 G729a/8000

Un attaccante può estrapolare una serie di informazioni dall'header SIP del messaggio relative alla segnalazione, e dal body relative al protocollo SDP. Ad esempio:

- L'indirizzo IP del SIP proxy server è 192.168.100.100, e la porta d'ascolto è la 5060;
- Il protocollo di trasporto è UDP senza alcuna cifratura;
- Il numero del softphone del chiamante Alice è 987654321, che effettua una chiamata ad uno UA Bob, con numero 123456789.
- L'indirizzo IP del telefono di Alice è 10.10.200.200
- Il media gateway accetta codec G.729a

Un attaccante che riesca a catturare del traffico SIP da una qualsiasi richiesta o risposta può estrarre tutte le informazioni che gli servono per poter attuare un piano d'attacco al protocollo.

Una volta sniffato il traffico, per poterlo modificare e reinserire in maniera valida nella conversazione, si utilizzano i valori di alcuni campi dell'header SIP relativi alla chiamata origliata. Si effettua quello che viene chiamato conosciuto come un generico attacco **Man-in-the-Middle**.

Si supponga che una terza parte non autorizzata, chiamata Attacker, performi un attacco di eavesdropping, interception e modification di una sessione già inizializzata tra due ipotetici UA, Alice e Bob. Lo scopo di Attacker è quello di rinegoziare la chiamata in questione stabilendo due dialoghi distinti, uno con Alice e uno con Bob, spacciandosi per la rispettiva controparte, cosicchè al termine della rinegoziazione Attacker si sarà intromesso nella chiamata senza che ne Alice ne Bob se ne siano accorti. L'attaccante avrà quindi un ruolo di B2BUA, con il potere di modificare pesantemente la comunicazione tra le due vittime, mascherando la propria identità con quella della controparte della rispettiva vittima.

L'attaccante ha bisogno dei valori di alcuni campi dell'header SIP contenuti nei messaggi scambiati tra i due UA vittime durante la fase di inizializzazione di sessione. Ottenuto il necessario, l'attaccante genera una richiesta di **RE-INVITE** verso Alice e verso Bob. Nel caso di Alice (per Bob il caso è analogo), la First-line conterrà il nome del metodo INVITE, la Request-URI di Alice, e la versione di SIP utilizzata. I campi nell'intestazione del messaggio saranno:

- **Via:** l'indirizzo tramite il quale l'attaccante vuole ricevere future risposte da parte della vittima.
- **From (sniffed):** dovendosi spacciare per Bob, l'attaccante prende il suo AOR ed il From Tag che lo identifica come peer valido per il dialogo corrente.
- **To (sniffed):** inserisce l'AOR di Alice ed il To Tag che lo identifica come peer valido per il dialogo corrente.
- **Call-ID (sniffed):** usa l'identificatore della chiamata corrente.
- **CSeq (sniffed):** la richiesta di INVITE deve utilizzare il sequence number dell'ultima transazione, incrementato di uno.
- **Contact:** inserisce l'AOR dell'attaccante, dove quest'ultimo vuole ricevere le future richieste da Alice.

In questo modo, l'attaccante si è infiltrato con successo nel mezzo della comunicazione tra Alice e Bob, compromettendola con il potere di modificare pesantemente i messaggi che essi si scambiano, inclusi i parametri SDP ed il contenuto effettivo del body.

4.2. Denial of Service

Come già visto, vi è una notevole varietà di tipologie di attacchi DoS: è bene dunque procedere con ordine ed analizzare una categoria alla volta.

- **Flooding:** l'attaccante "inonda" di richieste un proxy SIP o uno UA bersaglio, generando un numero molto elevato di messaggi che l'apparato di rete vittima dovrà processare. Un attacco del genere ha un'implementazione estremamente basilare: si tratta di semplici e lecite richieste di **INVITE** o di **REGISTER**, a seconda del tipo di flooding che si vuole effettuare. Se non ci si preoccupa nemmeno di mascherare la propria identità, basta conoscere la **Request-URI** dello UA vittima nel caso di una INVITE, o del dominio nel caso di una REGISTER, e si può creare una richiesta valida. Il resto dei valori dei campi header come From sono conosciuti dall'attaccante, o automaticamente generati come Call-ID o CSeq. Dato che generare delle richieste SIP richiede un minor impegno computazionale rispetto al doverle gestire, se si pensa addirittura alla possibilità di effettuare un attacco DoS distribuito dove vi sono non uno ma più dispositivi che bombardano un componente SIP di una rete, è facile intuire che tale componente rischia di avere seri problemi di gestione del traffico che possono risultare in saturazione della banda ed un potenziale crash.

- **Malformed requests:** richieste deformi sono messaggi SIP con una sintassi errata. Alcuni esempi [5] sono i seguenti sottolineati:

First-line: INVITE aaaaaaaaaa sip:Bob@domainB.com SIP/2.0

From:::::::::::: Alice <sip:Alice@domainA.com>;tag=9fxced76sl

Un parametro aggiuntivo nella first-line o un errore di punteggiatura come una ripetizione di un carattere separatore sono errori di formato che il parser SIP può non essere in grado di rilevare e gestire correttamente a seconda dell'implementazione adottata, entrando in uno stato confusionale

detto **fuzzed**, e risultando in comportamenti anomali: cicli infiniti di parsing, buffer overflow, inabilità di gestire futuri messaggi, crash del sistema. Anche in questo caso, è sufficiente conoscere il Request-URI della vittima per effettuare l'attacco.

- **Spoofed Messages:** in questo tipo di attacchi DoS, è necessario che l'attaccante sia in grado di intercettare le richieste provenienti dalla vittima o ad essa dirette, ed effettuare l'offensiva in maniera reattiva. In un attacco con vittime due generici UA Alice e Bob, l'attaccante ha origliato la richiesta di INVITE di Alice: se effettua un attacco Faked Response invia un generico messaggio d'errore, come una risposta client failure 4xx (es. 403 Forbidden) se si spaccia per uno UA o una server failure 5xx (500 Server Internal Error) se si spaccia per un server, oppure se usa un attacco Faked Call Teardown Message può attendere che si concluda la transazione di INVITE inviata da Alice verso Bob ed inviare successivamente un messaggio di BYE ad entrambi, terminando effettivamente la chiamata in modo lecito (anche se indesiderato). Le considerazioni sui valori dell'header SIP necessari per generare delle valide richieste di BYE o risposte di errore sono analoghe a quelle viste per un attacco di intercettazione e modifica del traffico SIP, con alcune piccole differenze:
- **Faked Response:** Per generare una risposta di errore valida, non è necessario inserire un campo Contact, dato che non ci si aspetta di ricevere future richieste da parte della vittima. Il **CSeq** da utilizzare è lo stesso della richiesta di INVITE da declinare. Tuttavia, l'attaccante deve intercettare la risposta 200 OK dello UAS, perchè ha bisogno del **To Tag** che lo autentica come Bob.
- **Faked Call Teardown Message:** Non necessita del campo Contact.
- **Call Hijacking:** Un furto di una chiamata si basa sul medesimo principio di una Faked Response, dove però tale risposta non è una risposta di errore, bensì una risposta Redirect 3xx. L'attaccante cerca di ridirigere il flusso della chiamata verso sè stesso o verso un proxy server da lui scelto

esterno alla rete della vittima. Come per gli ultimi due attacchi DoS visti, l'attaccante deve poter intercettare la richiesta di INVITE dello UA mittente e rispondergli prontamente con una risposta 301 Moved Permanently o 302 Moved Temporarily per sostituire uno UA, oppure con un 305 Use Proxy per sostituire un proxy server. Supponendo dunque che uno UAC Alice invii una richiesta di INVITE ad uno UAS Bob, l'attaccante risponderà ad Alice con un messaggio, ad esempio, di redirect 301 Moved Permanently, specificando nel campo **Contact** il nuovo indirizzo a cui contattare Bob (che sarà magari l'AOR dell'attaccante). Alice, dopo aver risposto con un messaggio di acknowledgement ACK, ripeterà la richiesta di INVITE, questa volta verso il falso indirizzo di Bob.

4.3. Social Threats

Attacchi a SIP di tipo sociale divergono in qualche maniera dalle altre offensive più "tecniche", in termini di intenzione e metodologia. Il fulcro dell'attacco è quello di manipolare il contesto sociale tra i membri di una comunicazione cosicchè l'attaccante possa spacciarsi per un'entità fidata e propinare false informazioni alle vittime. Questo genere di attacchi segue solitamente una fase iniziale di sniffing del traffico della vittima tramite eavesdropping, non solo per raccogliere parametri utili come gli indirizzi o i numeri che ha contattato, ma anche per fare del *data mining*, cioè una raccolta dati che evidenzia le abitudini dell'utente quali orario di determinate chiamate, terminologia utilizzata, interessi sociali, ecc. Ottenere dati del genere aiuta l'attaccante ad effettuare un'offensiva più convincente e di conseguenza più efficace nei confronti del bersaglio, al fine di ottenere dati sensibili normalmente non reperibili tramite sniffing del traffico (numero carta d'identità, conto corrente, credenziali d'accesso, ecc.).

Una volta che un malintenzionato ha raccolto una quantità di dati sufficienti, può effettuare uno dei seguenti attacchi:

- **Misrepresentation:** Fa parte solitamente della prima fase di un attacco più complesso. L'attaccante presenta la propria identità con informazioni false inserite in una richiesta, quali un AOR nel campo **From**, spacciandosi ad esempio per uno UA che la vittima ha precedentemente contattato.
- **Call Spam-over-IP (SPIT):** un attacco che ricade nella categoria **unwanted contact**. L'attaccante *spammer*, dopo aver raccolto una serie di **Request-URI** dalle chiamate di una o più vittime, genera delle nuove richieste di INVITE verso ciascun contatto acquisito, con lo scopo di iniziare una comunicazione video o audio con uno dei destinatari. Se uno di questi dovesse effettivamente rispondere, lo spammer procede inviando il proprio messaggio di truffa (audio o video) via real-time media.
- **Instant Messaging Spam (SPIM):** è molto simile al precedente attacco illustrato. L'attacco non avviene utilizzando l'ingegneria sociale nel body del messaggio, ma si attua direttamente nell'intestazione SIP o in forma testuale nel body del messaggio.

Una tipica richiesta SPIM in SIP è comunemente di tipo **MESSAGE**: questo messaggio contiene nel proprio body un campo **Line-based text data: text/plain** contenente a sua volta una stringa di testo destinata alla lettura dell'utente vittima (es. Line-based text data: text/plain "Tired of your current service? Check the best one at www.bestVoIPservice.com"). Un'altra possibile richiesta è un **INVITE** con un campo **Subject** molto grande: tale campo contiene solitamente una stringa di testo con un messaggio, diretto all'interpretazione dell'utente vittima e non dello stack SIP, contenente un link pubblicizzato dall'attaccante che quest'ultimo cerca di indurre la vittima ad aprire (es. Subject: Hi, looking for a good VoIP service? Try at www.external-VoIP-service.com). A differenza di un'email, un IM è più invasivo: quest'ultimo può presentarsi automaticamente all'utente tramite pop-up, mentre le email devono essere deliberatamente selezionate ed aperte.

4.4. Considerazioni

L'aspetto in comune che presentano gli attacchi è che implicano una conoscenza preacquisita della vittima. Che l'informazione necessaria a generare un'offensiva sia un semplice identificativo o indirizzo, oppure un'insieme di valori di più campi, nel caso di un attacco al protocollo SIP queste informazioni possono essere lette direttamente dall'intestazione SIP del messaggio. Un malintenzionato in grado di monitorare le attività di uno UA SIP in una rete VoIP, analizzandone il relativo traffico può effettuare un atto dannoso nei confronti della vittima e/o della rete di cui essa fa parte.

Conoscendo il Request-URI di uno UA, un attaccante può inviargli richieste di tipo **sociale** non desiderate, contenenti nel campo Subject dell'intestazione SIP o nel body del messaggio testi offensivi o a fine pubblicitario, oppure richieste di apertura di una sessione multimediale con contenuti analoghi di tipo audio o video. Maggiore l'impegno nello studio delle abitudini dell'utente vittima tramite data mining, più alte le probabilità di convincerlo a rispondere positivamente al messaggio pubblicitario.

Impostando il valore del campo From con un SIP-URI uguale a quello di un contatto dello UA vittima, un malintenzionato può rubarne l'identità per iniziare una chiamata con essa spacciandosi per un contatto conosciuto.

Riuscendo poi ad intercettare e a generare messaggi reattivamente in tempo reale, un attaccante può **interporsi** nella comunicazione fra due o più parti e **modificarne** il contenuto multimediale, oppure registrarlo e riproporlo in seguito. In alternativa, può ridirigere a sé stesso o ad una terza parte una richiesta di INVITE o REGISTER, oppure rifiutare tali richieste al posto del destinatario o di un server intermediario, o troncando prematuramente con un messaggio di BYE la conversazione iniziata. In tutti questi casi, impedisce alla vittima di utilizzare i servizi offerti dalla rete, incluso quello di chiamare un altro UA, creando quindi dei **denial of service**.

Nel prossimo capitolo, si mostra la realizzazione di alcuni esempi di attacco a SIP utilizzando un semplice analizzatore di traffico ed un generatore di messaggi SIP.

Capitolo 5

5. Dimostrazione

Si può controllare il traffico dati e voce passante interamente all'interno della propria rete privata, ma se i pacchetti vengono inoltrati nella rete Internet (scenario più che plausibile) diventano potenzialmente vulnerabili ad attacchi. Senza alcun tipo di meccanismo difensivo, sono sufficienti giusto due tipi di strumenti per compiere un assalto al protocollo SIP: un analizzatore ed un generatore di traffico. Il primo serve per *sniffare* (catturare) il traffico di una rete passante per la macchina su cui questo programma è installato allo scopo di raccogliere dati rilevanti per la segnalazione, mentre il secondo genera del nuovo traffico utilizzando i parametri rilevati dall'analizzatore.

5.1. Analizzatore di traffico: Wireshark

Wireshark è un analizzatore di pacchetti, open-source e disponibile gratuitamente in rete [6], e largamente utilizzato sia in ambito amatoriale che professionale. Il suo utilizzo principale è quello di troubleshooting e analisi del comportamento delle reti, e per sviluppare programmi e protocolli di comunicazione. E' utilizzabile sia tramite la comoda interfaccia grafica, o in alternativa tramite una versione basata su terminale chiamata TShark. Implementa una conoscenza dei principali protocolli di comunicazione, ad eccezione di alcuni proprietari come quello usato da Skype, rendendo possibile una visualizzazione dei campi di ciascun pacchetto e del relativo significato, divisi per ciascun protocollo. La cattura e la visione del traffico passante per le interfacce di rete (Ethernet, IEEE 802.11, loopback) della macchina locale può essere fatta tramite

una connessione di rete in tempo reale oppure letta da un file dove sono registrati pacchetti già catturati, opportunamente filtrati, ad esempio, per protocollo d'interesse.

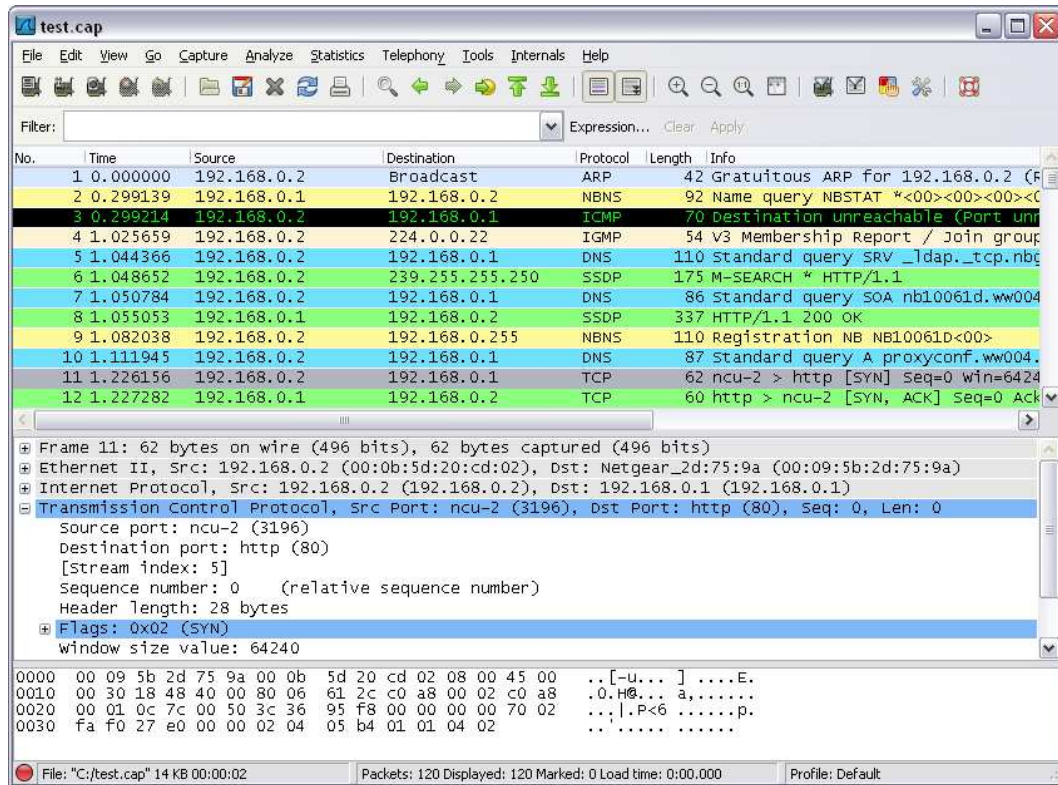


Fig. 3: Una schermata di Wireshark dopo una cattura [8].

Utilizzando Wireshark è quindi possibile catturare, visualizzare, ed analizzare il traffico SIP intercettato in una rete VoIP, in particolare i campi della segnalazione.

5.2. Generatore di traffico: SIPp

SIPp è un generatore di traffico per il protocollo SIP, open-source e reperibile gratuitamente sulla rete Internet [7]. Viene principalmente utilizzato a fini di stress test e test di prestazioni su reti VoIP che utilizzano SIP come protocollo di segnalazione. Include una serie di scenari di base detti *SipStone*, come una semplice creazione e chiusura di una sessione tra due UA, ma

soprattutto consente di creare degli scenari di chiamata personalizzati più complessi tramite la scrittura di file XML con appropriata formattazione, con la possibilità di estrarre ed iniettare i valori dei campi di un qualsiasi protocollo.

```

ocadmin@vista:~/sipp
----- Scenario Screen ----- [1-4]: Change Screen --
Call-rate(length)  Port    Total-time  Total-calls  Remote-host
    10 cps(0 ms)   5061      4.01 s      40           127.0.0.1:5060(UDP)

10 new calls during 1.000 s period      16 ms scheduler resolution
0 concurrent calls (limit 30)           Peak was 1 calls, after 0 s
0 out-of-call msg (discarded)
1 open sockets

          Messages  Retrans  Timeout  Unexpected-Msg
INVITE ----->      40         0         0
  100 <-----      0         0         0
  180 <-----      40         0         0
  200 <----- E-RTD  40         0         0
  ACK ----->      40         0
    [    0 ms]
  BYE ----->      40         0
  200 <-----      40         0

----- [+-|*|/]: Adjust rate ---- [q]: Soft exit ---- [p]: Pause traffic -----

```

Fig. 4: Un'esecuzione di SIPp che mostra la call flow [7].

SIPp è un programma eseguibile da riga di comando, sprovvisto di interfaccia grafica ma con una completa visualizzazione dinamica del flusso delle chiamate effettuate e della relativa reportistica. Ciascuna istanza di SIPp viene lanciata separatamente, in quanto ogni processo è in ascolto su un indirizzo di rete univoco, richiamando la configurazione definita nel relativo file XML, con la possibilità di settare ulteriori parametri della chiamata, quali frequenza del numero di chiamate, valore dell'interfaccia di rete, ecc.

SIPp può funzionare sia come uno UAC sia come uno UAS, può cioè rispettivamente prendere l'iniziativa per avviare una chiamata SIP inviando delle richieste, oppure può rimanere in ascolto su una certa interfaccia di rete e rispondere ad un certo tipo di richiesta con delle risposte predefinite, con la possibilità di utilizzare un branching condizionale per eseguire uno scenario in maniera non lineare.

Con SIPp si possono dunque creare e modellare delle sessioni basate sul protocollo SIP che coinvolgono ignare terze parti andando a *sniffare* e raccogliere i parametri necessari dall'intestazione dei messaggi SIP utilizzando Wireshark.

5.3. Dettagli della Dimostrazione

Gli esempi di attacchi al protocollo SIP sono stati realizzati utilizzando i seguenti strumenti e modalità:


- Uno UAC **Alice**, e uno UAS **Bob**, che effettuano uno scambio di messaggi durante la prima transaction, quella di call setup.
- Per semplicità, non vi sono server di alcun tipo fra gli UA.
- Uno UA **Attacker** che effettua l'attacco.
- Tutti gli UA sono implementati localmente sull'interfaccia **localhost** della stessa macchina. Alice, Bob, e Attacker sono rispettivamente sulle porte **5062, 5063, 5064**.
- Alice e Bob utilizzano **PJSIP**, uno stack SIP che implementa già le meccaniche del protocollo.
- Gli scenari di chiamata sono catturati interamente da un'istanza di **Wireshark**, così da monitorare l'intero flusso di chiamata, incluso l'attacco.
- L'Attacker cattura lo scambio iniziale di messaggi tramite un'istanza di **tshark**, salvando i dettagli rilevanti al protocollo SIP in un file dati **.csv**, dove i valori dei campi sono separati dal carattere ";".
- L'Attacker reagisce alla cattura di un determinato tipo di richiesta, a seconda del tipo di attacco, e fa partire l'offensiva avviando un'istanza di SIPp specificando il relativo scenario **.xml** e il relativo file **.csv** della cattura dal quale iniettare i valori mancanti nello scenario.

5.4. DoS: Faked Call Teardown

L'attaccante intercetta i messaggi durante la three-way handshake che apre la chiamata tra Alice e Bob. Una volta conclusa l'inizializzazione della chiamata con l'ACK diretto ad Alice, egli provvede ad inviare un messaggio di BYE al mittente della richiesta di INVITE, cioè Alice, terminando la sessione. Può scegliere se inviare il messaggio di BYE anche a Bob, ma è una scelta opzionale.

Nel messaggio di BYE diretto ad Alice che l'Attacker genera, il campo From contiene l'AOR di Bob (affinchè Alice creda di rispondere a lui, e non all'Attacker), la stessa Call-ID degli altri messaggi del dialogo, gli stessi Tag del campo To e From in ordine inverso rispetto ai messaggi della transaction iniziale (dato che chi invia la richiesta di BYE sarebbe Bob verso Alice, il campo From contiene l'AOR di Bob e il campo To quello di Alice), e un CSeq number per il metodo BYE incrementato di uno. L'attaccante riceve così regolarmente il 200

```
Request: INVITE sip:bob@127.0.0.1:5063, with session description
Status: 100 Trying
Status: 200 OK, with session description
Request: ACK sip:bob@127.0.0.1:5063
```

 **Attacco Faked Call Teardown**

```
Request: INVITE sip:bob@127.0.0.1:5063, with session description
Status: 100 Trying
Status: 200 OK, with session description
Request: ACK sip:bob@127.0.0.1:5063
Request: BYE sip:alice@127.0.0.1:5062
Status: 200 OK
```

OK in risposta per chiudere correttamente la chiamata.

Fig. 5: L'evoluzione dello scenario vista tramite Wireshark, prima e dopo l'attacco.

```

⊕ Status-Line: SIP/2.0 200 OK
⊖ Message Header
    Call-ID: uCZ.xYsn7N8bGL-kbfuY6hyyHEUMlCJT
    ⊕ From: <sip:alice@127.0.0.1>;tag=P5YALPhkZIR9vxgU-ct8w9DY0bfcciyM
    ⊕ To: <sip:bob@127.0.0.1>;tag=AdDbcwVzc5AL62193uRHpCgm7b3zne14
    ⊕ CSeq: 3625 INVITE

⊕ Request-Line: BYE sip:alice@127.0.0.1:5062 SIP/2.0
⊖ Message Header
    ⊕ From: sip:bob@127.0.0.1;tag=AdDbcwVzc5AL62193uRHpCgm7b3zne14
    ⊕ To: sip:alice@127.0.0.1;tag=P5YALPhkZIR9vxgU-ct8w9DY0bfcciyM
    Call-ID: uCZ.xYsn7N8bGL-kbfuY6hyyHEUMlCJT
    ⊕ CSeq: 3626 BYE

```

Fig. 6: La risposta 200 OK e il BYE dell'attaccante (solo i campi rilevanti).

```

----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
10.0(0 ms)/1.000s  5064    0.10 s      1            127.0.0.1:5062(UDP)

Call limit reached (-m 1), 0.000 s period  0 ms scheduler resolution
0 calls (limit 30)                          Peak was 1 calls, after 0 s
0 Running, 2 Paused, 0 Woken up
0 dead call msg (discarded)                 0 out-of-call msg (discarded)
1 open sockets

          Messages  Retrans  Timeout  Unexpected-Msg
[ NOP ]
BYE ----->      1          0         0
200 <----- E-RTD1 1          0         0
----- Test Terminated -----

```

Fig. 7: La call flow di SIPp dell'attaccante.

5.5. DoS: Call Hijacking

L'attaccante intercetta la richiesta di INVITE di Alice destinata a Bob, e risponde prontamente con una risposta 301 Moved Permanently, specificando nel campo Contact l'indirizzo al quale contattare Bob (in realtà è l'indirizzo dell'Attacker). La seguente richiesta di Alice sarà un nuovo INVITE, questa volta però destinato direttamente all'attaccante (il Request-URI dell'Attacker compare nella Request-Line della nuova richiesta).


```

Request: INVITE sip:bob@127.0.0.1:5063, with session description
Status: 100 Trying
Status: 301 Moved Permanently
Request: ACK sip:bob@127.0.0.1:5063
Status: 100 Trying
Request: INVITE sip:attacker@127.0.0.1:5064, with session description
Status: 180 Ringing
Status: 200 OK, with session description
Request: ACK sip:attacker@127.0.0.1:5064;transport=UDP

```

Fig. 8: La call flow mostra l'attacco e la conseguenza.

```

Session Initiation Protocol
  Request-Line: INVITE sip:bob@127.0.0.1:5063 SIP/2.0
  Message Header
    Via: SIP/2.0/UDP 127.0.0.1:5062;rport;branch=z9hg4bKPj7hytB1s11Tmm2nxyB19CjihQEY8mYSNV
    Max-Forwards: 70
    From: sip:alice@127.0.0.1;tag=HCm9yXD0yKCbBh2FLKh9w4dPyX-TrxcE
    To: sip:bob@127.0.0.1
    Contact: <sip:alice@127.0.0.1:5062;ob>
    Call-ID: QRb8FGVU-9X8MXHwbXa5xJ.zJxkc6J73
    CSeq: 9344 INVITE

Session Initiation Protocol
  Status-Line: SIP/2.0 301 Moved Permanently
  Message Header
    Via: SIP/2.0/UDP 127.0.0.1:5062;branch=z9hg4bKPj7hytB1s11Tmm2nxyB19CjihQEY8mYSNV
    From: sip:alice@127.0.0.1;tag=HCm9yXD0yKCbBh2FLKh9w4dPyX-TrxcE
    To: sip:bob@127.0.0.1;tag=1
    Call-ID: QRb8FGVU-9X8MXHwbXa5xJ.zJxkc6J73
    CSeq: 9344 INVITE
    Contact: sip:attacker@127.0.0.1:5064

Session Initiation Protocol
  Request-Line: INVITE sip:attacker@127.0.0.1:5064 SIP/2.0
  Message Header
    Via: SIP/2.0/UDP 127.0.0.1:5062;rport;branch=z9hg4bKPj1ajnt0miecf56kcZhm2lq8-k1st5fjf1
    Max-Forwards: 70
    From: sip:alice@127.0.0.1;tag=HCm9yXD0yKCbBh2FLKh9w4dPyX-TrxcE
    To: sip:bob@127.0.0.1
    Contact: <sip:alice@127.0.0.1:5062;ob>
    Call-ID: QRb8FGVU-9X8MXHwbXa5xJ.zJxkc6J73
    CSeq: 9345 INVITE

```

Fig. 9: La richiesta di INVITE iniziale, la risposta 301, e l'INVITE seguente.

```

----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
10.0(0 ms)/1.000s  5064    9.87 s      1  127.0.0.1:5062(UDP)

Call limit reached (-m 1), 0.859 s period 1 ms scheduler resolution
1 calls (limit 120)                        Peak was 1 calls, after 0 s
0 Running, 3 Paused, 2 Woken up
0 dead call msg (discarded)                0 out-of-call msg (discarded)
3 open sockets

          Messages  Retrans  Timeout  Unexpected-Msg
301 ----->          1          0          0          0
ACK <----- E-RTD1  0          0          0          0
INVITE <-----          1          0          0          0
180 ----->          1          0          0          0
200 ----->          1          0          0          0
ACK <----- E-RTD1  1          0          0          0
BYE <-----          0          0          0          0
200 ----->          0          0          0          0
Pause [ 4000ms]      0          0          0          0
----- Test Terminated -----

```

Fig. 10: La call flow generata da SIPp

5.6. Man-in-the-Middle

L'attaccante si interpone nella comunicazione tra Alice e Bob: dopo aver sniffato la transazione di three-way handshake, ha tutti i dati necessari per fungere da B2BUA invisibile tra i due UA. L'Attacker rinegozia la chiamata, quindi invia una richiesta di RE-INVITE sia ad Alice che a Bob, con la stessa Call-ID della chiamata originale, un branch parameter diverso da quello dell'INVITE precedente ed un numero CSeq incrementato di uno (una RE-INVITE è a tutti gli effetti una nuova transaction). Un valore diverso lo inserisce invece nel campo Contact, specificando il proprio indirizzo come Contact-URI al posto di quello specificato in origine, e come campo Via inserisce il proprio indirizzo IP (con porta). Così facendo, qualsiasi richiesta futura da parte di una delle vittime verso l'altro endpoint verrà invece inviata allo UA Attacker.

```

Request: INVITE sip:bob@127.0.0.1:5063, with session description
Status: 100 Trying
Status: 200 OK, with session description
Request: ACK sip:bob@127.0.0.1:5063
Request: INVITE sip:alice@127.0.0.1:5062, in-dialog, with session description
Status: 200 OK, with session description
Request: ACK sip:alice@127.0.0.1:5062
Request: INVITE sip:bob@127.0.0.1:5063, in-dialog, with session description
Status: 200 OK, with session description
Request: ACK sip:bob@127.0.0.1:5063

```

Fig. 11: La call flow della chiamata prima e durante l'attacco

```

⊕ User Datagram Protocol, Src Port: ca-1 (5064), Dst Port: na-localise (5062)
⊖ Session Initiation Protocol
  ⊕ Request-Line: INVITE sip:alice@127.0.0.1:5062 SIP/2.0
  ⊖ Message Header
    ⊕ Via: SIP/2.0/UDP 127.0.0.1:5064;branch=z9hG4bK-4373-1-1
    ⊕ From: sip:bob@127.0.0.1;tag=TqW0ensRt0NO8ZCjwk5JD5wzpz1uPsto
    ⊕ To: sip:alice@127.0.0.1;tag=k1UZgf0IKkAnAPPnQra5G10KMC7Z-rs
    Call-ID: n1wJTCJ8e2UQku15KdcuzybBcT7b24mI
    ⊕ CSeq: 26596 INVITE
    ⊕ Contact: sip:attacker@127.0.0.1:5064

```

Fig. 12: La struttura del messaggio di RE-INVITE mandato ad Alice dall'Attacker

```

----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length) Port Total-time Total-calls Remote-host
0.1(0 ms)/1.000s 5064 13.79 s 1 127.0.0.1:5062(UDP)

Call limit reached (-m 1), 0.779 s period 1 ms scheduler resolution
1 calls (limit 3) Peak was 1 calls, after 10 s
0 Running, 4 Paused, 2 Woken up
0 dead call msg (discarded) 0 out-of-call msg (discarded)
3 open sockets

Messages Retrans Timeout Unexpected-Msg
[ NOP ]
INVITE -----> 1 0 0
200 <----- E-RTD1 1 0 0
ACK -----> 1 0
Pause [ 10.0s] 1 0
----- Test Terminated -----

```

Fig. 13: L'attacco di una istanza di SIPp (sulla porta 5064) allo UA Alice

```

----- Scenario Screen ----- [1-9]: Change Screen --
Call-rate(length)  Port  Total-time  Total-calls  Remote-host
0.1(0 ms)/1.000s  5065    14.19 s    1            127.0.0.1:5063(UDP)

Call limit reached (-m 1), 0.173 s period 1 ms scheduler resolution
1 calls (limit 3)                          Peak was 1 calls, after 10 s
0 Running, 4 Paused, 1 Woken up
0 dead call msg (discarded)                0 out-of-call msg (discarded)
3 open sockets

                                     Messages  Retrans  Timeout  Unexpected-Msg
      [ NOP ]
INVITE ----->                        1         0         0         0
  200 <----- E-RTD1 1                 0         0         0
  ACK ----->                        1         0         0         0
  Pause [ 10.0s]                       1         0         0         0
----- Test Terminated -----

```

Fig. 14: L'attacco di una istanza di SIPp (sulla porta 5065) allo UA Bob

5.7. Considerazioni

Gli esempi proposti mostrano le dinamiche, a livello di protocollo, con cui un attaccante può modificare a proprio piacimento il flusso di una chiamata. La scelta di mostrare questi tre casi invece al posto di altri è stata dettata dalla logic che ciascun attacco visto racchiude:

- **Man-in-the-Middle** è l'esempio calzante della classe di vulnerabilità Eavesdropping, Interception, and Modification. Il traffico viene origliato ed intercettato dall'attaccante che si interpone nella comunicazione fra due UA, al fine appunto di modificare la sessione. L'esempio mostra un caso molto generico, ma molto completo nella sua modifica dei valori dei campi della segnalazione, di Man-in-the-Middle: solitamente l'attacco, conosciuto come Man-in-the-Middle RTP, prosegue con una modifica da parte dell'attaccante di una parte o della totalità della sessione RTP, cioè del traffico media real-time. SIPp purtroppo mal si presta ad una realizzazione così completa: esso è stato pensato e realizzato per essere un generatore di traffico sul piano della segnalazione, e per tale motivo c'è un supporto limitato sul piano dei media, cioè RTP. Anche senza l'estensione ad un attacco RTP, questo attacco Man-in-the-Middle comporta un effetto

di **Black Holing**, ovvero tutto il traffico inviato da uno dei due UA vittima viene in realtà mandato all'attaccante che non lo inoltra, generando dunque una sorta di "buco nero" nella comunicazione.

- **Faked Call Teardown:** è un attacco DoS che richiede un precedente sniffing dell'ultima richiesta/risposta a **dialogo stabilito**, inviata da uno UA della chiamata che si vuol terminare.
- **Call Hijacking:** questo attacco DoS reagisce ad una prima richiesta di INVITE, agendo quindi a **dialogo non stabilito**.

Altri tipi di attacchi, come quelli di tipo sociale, incapsulano logiche a livello di modifica dei campi della segnalazione molto simili agli esempi riportati: un attacco Man-in-the-Middle, ad esempio, include già di per sé un furto di identità. Al di fuori delle meccaniche di segnalazione, gli attacchi di tipo sociale si concentrano maggiormente su una fase di raccolta di informazioni su un utente (inteso come persona fisica), di data mining e di costruzione di un messaggio testuale che dev'essere interpretato dall'utente e non da uno stack SIP. Uno studio di tecniche di ingegneria sociale sarebbe più opportuno per approfondire le dinamiche di questo attacco.

Wireshark fornisce un'accurata analisi del traffico presentando una chiara scomposizione del contenuto dei pacchetti. Alcuni protocolli proprietari, quali appunto quello utilizzato da Skype, non è supportato e quindi non è utilizzabile per essere elaborato da un generatore di attacco, ma per quanto riguarda SIP, l'analisi del traffico risulta dettagliata ad un livello più che accettabile.

SIPp è un generatore di traffico, creato per effettuare stress test su reti VoIP, e quindi ben si presta a tipologie di attacco di tipo DoS o SPIM o Spam in generale, ovvero offensive basate sulla generazione di un alto numero di richieste SIP. Le meccaniche a livello di segnalazione per realizzare questi attacchi sono già implementate negli esempi visti.

La facilità per un malintenzionato di accedere agilmente ai dettagli sul traffico degli utenti di una rete e di modificare il flusso delle relative chiamate intervenendo sulla segnalazione SIP è dunque concreto. Tecniche di difesa a livello di segnalazione e di prevenzione a livello di rete sono discusse nel successivo capitolo.

Capitolo 6

6. Contromisure

SIP è un protocollo basato sul testo, e se da un lato questa caratteristica lo rende un protocollo intuitivo e pratico per il troubleshooting in fase di implementazione e funzionamento all'interno di una rete VoIP, dall'altro lo rende tale anche per chi desidera sfruttarlo con intenzioni malevole.

Una domanda importante da porsi per capire come poter prevenire e reagire ad attacchi al protocollo SIP è la seguente: *cosa si può fare per proteggere la segnalazione SIP, sia a livello di protocollo che a livello di rete?*

Si può dire generalmente che, essendo basate su reti IP, le applicazioni VoIP ereditino le stesse minacce di tali reti, quali un qualsiasi tipo di attacco DoS, sniffing, spoofing, e Man-in-the-Middle. Per quanto concerne SIP, la natura del protocollo a livello applicativo lo rende vulnerabile a quegli specifici attacchi SIP analizzati in precedenza, quali messaggi di ridirezione o terminazione della chiamata (rispettivamente hijacking call e faked call teardown), ecc.

Nel testo che segue, si va ad esaminare prima le soluzioni attuabili direttamente al protocollo SIP, ovvero come proteggere l'accesso e la modifica ai messaggi, e poi quelle implementabili invece a livello di rete, proteggendo ad esempio la rete da accessi o applicazioni indesiderate.

6.1. Difese del protocollo SIP

Analizzando le categorie degli attacchi a SIP si evincono tre proprietà che devono essere garantite per proteggere la corretta e sicura segnalazione tra gli utenti:

1. **Autenticazione:** garantire l'autenticità degli utenti, rilevando i furti d'identità.
2. **Integrità:** garantire che il messaggio ricevuto da uno UA sia effettivamente quello spedito in origine.
3. **Confidenzialità:** evitare che terze parti siano in ascolto sulla chiamata tra due o più UA della rete, e che possano estrarne il contenuto.

Anche se vi sono alcune estensioni proposte quali SIP Secure, le specifiche base di SIP non prevedono meccanismi difensivi, ma è consigliato l'utilizzo di altri metodi di sicurezza già in uso. Essendo un protocollo basato su testo e basato su un modello richiesta/risposta come HTTP, si possono attuare le stesse contromisure di protezione.

SIP supporta tre forme complementari di cifratura per proteggere la privacy degli utenti:

- **end-to-end** per il body del messaggio e alcuni campi sensibili dell'header;
- **hop-by-hop** per prevenire eavesdropping e nascondere gli UA coinvolti;
- **hop-by-hop** dei valori Via per nascondere il percorso della chiamata.

End-to-end significa che una richiesta o una risposta SIP viene cifrata da uno UA e decifrata direttamente dallo UA destinatario. Questo ovviamente non è possibile: cifrare l'intero header SIP implica cifrare anche alcuni campi come il **To**, che specifica il destinatario del messaggio, **Via** che indica il cammino dell'inoltro delle risposte, e **Contact** per le funzioni di reindirizzamento, che devono essere letti e processati velocemente dai proxy SIP per una corretta

segnalazione. E' necessario che si completi la protezione utilizzando una cifratura hop-by-hop, dove viene cifrato l'intero messaggio volta per volta, così da proteggerne il contenuto dall'eavesdropping e dallo sniffing.

E' possibile cifrare tuttavia gli altri dei campi importanti dell'intestazione SIP, come il campo **From**, che indica la sorgente di una richiesta, dato che le informazioni per rispondere ed inviare richieste future sono definite rispettivamente nei campi Via e Contact, anche se alcuni proxy potrebbero rispondere con un messaggio 401 se richiedono il campo From in chiaro. Il campo **Authorization** deve rimanere in chiaro se contiene una firma digitale dato che tale firma è generata solo dopo la cifratura, ma può essere cifrata se contiene un'autenticazione *basic* o *digest*, spiegate nel testo successivo.

6.1.1. SIP Authentication

L'autenticazione SIP è ereditata dal noto meccanismo di **HTTP Authentication**, che si basa su un modello *challenge/response*: significa che un server SIP che implementi questo meccanismo risponde ad una richiesta SIP con un challenge, ovvero con una risposta di errore che solleciti il mittente a riprovare con una nuova richiesta contenente dei dati mancanti, ovvero quelli di autenticazione.

Il funzionamento è il seguente: uno UAC invia un messaggio SIP, come un INVITE o REGISTER, ed il server, che richiede un'autenticazione, risponde con una risposta **407 Proxy Authentication Required** (nel caso di un server proxy), oppure **401 unauthorized** (nel caso di un altro tipo di server), che nell'header incapsula un campo aggiuntivo, **WWW-Authenticate**, che contiene appunto il challenge al quale lo UAC deve rispondere se vuole autenticarsi. Il client risponde quindi generando una nuova richiesta, equivalente alla precedente ma con un campo aggiuntivo nell'header, **Authenticate**, contenente le credenziali richieste. Con questo sistema, un server SIP riesce ad autenticare i messaggi scambiati con uno UA o con un altro server.

Immagine HTTP Authentication call flow

HTTP 1.0 Basic Authentication richiede la trasmissione di uno username e della relativa password nell'header del messaggio. Tuttavia, le password testuali sono inviate in chiaro, e quindi facilmente intercettabili e leggibili da un semplice analizzatore di traffico, vanificando lo sforzo di segretezza.

HTTP 1.1 Digest Authentication ha lo stesso funzionamento ma migliora decisamente l'aspetto di sicurezza utilizzando una crittografia con funzione di hashing MD5 per cifrare la firma prima di inviarla nel messaggio. Il campo WWW-Authenticate del challenge del server contiene tre valori: l'algoritmo di Digest da utilizzare (MD5), un valore *realm* contenente il nome del dominio presso il quale autenticarsi (ogni dominio ha il proprio insieme di username e relative password), ed un valore *nonce* casuale generato dal server da usare nella cifratura con la funzione di hashing, al fine di evitare futuri attacchi *replay*.

Se da un lato HTTP Digest Authentication risolve il problema di un server della rete di autenticare altri server o client, permane purtroppo il problema inverso: un client non ha modo di sapere se il server sia affidabile o meno, perchè questo meccanismo non prevede una challenge da parte dello UAC. Soprattutto, HTTP Digest **non supporta l'integrità e la confidenzialità** del traffico scambiato, e questo lo rende una soluzione incompleta alla sicurezza, che dovrebbe garantire tutte e tre le proprietà elencate in precedenza. La segnalazione rimane dunque sempre vulnerabile ad attacchi di tipo Man-in-the-Middle, dove l'attaccante può spacciarsi per un proxy server della rete. Inoltre, anche se le credenziali non sono trasmesse in chiaro, rimangono comunque ricavabili tramite forza bruta in modalità offline.

6.1.2. IP Security

SIP è un protocollo applicativo, ed opera quindi al livello più alto della pila di rete ISO/OSI. Un modo per proteggerlo è quello di proteggerne gli strati sottostanti.

IPSec si occupa di applicare una protezione al protocollo IP, cioè a livello tre, quello di rete. L'obiettivo è quello di proteggere i pacchetti IP che incapsulano la segnalazione del protocollo SIP da attacchi di tipo spoofing, hijacking, ecc. garantendo confidenzialità, integrità, e autenticazione. IPSec è uno schema di sicurezza end-to-end, basato su una relazione di fiducia reciproca tra le entità coinvolte, che può essere usato per proteggere il flusso di dati tra una coppia di host, di gateways tra due reti, o tra un gateway di rete ed un host. Gli strumenti principali che IPSec fornisce sono:

- **Authentication Headers (AH):** sono usati per garantire integrità dei pacchetti IP in maniera indipendente dalla connessione, in modo hop-by-hop. Garantiscono l'autenticazione dei datagrammi IP e proteggono contro attacchi replay.
- **Encapsulating Security Payloads (ESP):** garantiscono **confidenzialità**, **autenticazione** dell'origine dei dati, **integrità**, e protezione da attacchi replay.
- **Security Association:** fornisce algoritmi e dati necessari per eseguire le operazioni di AH o ESP.
- **Key Management Protocol:** serve per distribuire le chiavi pubbliche di autenticazione necessarie ad inizializzare associazioni sicure in IPSec.

Dato che ciascun server proxy sul percorso di una comunicazione deve essere in grado di accedere, sia in lettura che in scrittura, all'header SIP, i meccanismi AH e ESP devono essere applicati su una base hop-by-hop.

Cifrando l'intero pacchetto IP, l'utilizzo di IPSec non necessita di ulteriori forme di sicurezza di strato più alto. Tuttavia, se da un lato IPSec presenta numerosi vantaggi in termini di sicurezza, permettendo di garantire le tre proprietà precedentemente elencate, il suo utilizzo può causare un considerevole overhead

dovuto all'incapsulamento dei pacchetti con IPSec-ESP, senza considerare la maggior elaborazione da parte dei server che li processano. Inoltre, molti client SIP non implementano ancora questo protocollo, e per tale ragione IPSec può proteggere efficacemente solo il traffico tra i corrispondenti server di rete. Infine, le specifiche SIP non suggeriscono l'utilizzo di alcun framework per la gestione delle chiavi, che è un requisito per la sicurezza di IPSec.

6.1.3. Transport Layer Security

Un ulteriore modo per proteggere il protocollo SIP è quello di lavorare ancora ad un livello più basso dello strato applicativo, ma questa volta sopra al livello di rete: lo strato di trasporto.

TLS, come il suo predecessore Secure Sockets Layer, garantisce privacy fra due applicazioni che comunicano in rete, impenendo dunque un eavesdropping o intercettazione della comunicazione. Fornisce un meccanismo di autenticazione, per le entità della rete, **TLS Handshake Protocol**, ed un meccanismo di cifratura, **TLS Record Protocol**. Il protocollo Handshake permette l'autenticazione reciproca fra un client ed un server, tramite uno scambio di certificati, e la negoziazione dell'algoritmo di cifratura e chiavi crittografiche prima di iniziare lo scambio dei dati.

Il protocollo TLS ha molti dei vantaggi di IPSec, ma a differenza di quest'ultimo non presuppone che ci sia alcuna relazione di fiducia fra i membri della comunicazione, e produce un overhead significativamente inferiore. Il problema di TLS, a differenza di IPSec, rimane però quello di essere orientato alla connessione e dover usare TCP come protocollo di trasporto, rischiando di aumentare il carico computazionale per i server proxy se vi sono troppe connessioni TCP aperte. Come per IPSec, molti client SIP ancora non implementano questo protocollo. Rimane quindi il rischio che un messaggio scambiato fra due entità venga intercettato in uno dei nodi che non implementa TLS, non riuscendo quindi a garantire la sicurezza intesa end-to-end. Sono entrambi infatti protocolli di sicurezza hop-by-hop.

6.1.4. IP Secure

SIPS è un meccanismo di sicurezza end-to-end, equivalente a HTTPS, atto a proteggere le conversazioni SIP tramite autenticazione e cifratura dei messaggi. E' stato inserito come estensione di SIP nel RFC 3261 dell'IETF.

SIPS utilizza uno schema di indirizzamento avente la stessa struttura di SIP, con l'unica eccezione nel cambio della parola chiave SIP in SIPS. Un esempio di indirizzo sicuro è *sips:alice@domainA.com*. Un SIPS-URI è uno speciale tipo di indirizzo che garantisce una sicurezza a livello di trasporto tra tutti gli hop di una conversazione SIP: un UAC che specifica una richiesta SIPS non fa altro che generare una normale richiesta SIP, utilizzando nella Request-Line un indirizzo SIPS invece che SIP, e tale richiesta verrà processata da altri client e server utilizzando TLS in ciascun hop.

SIPS presenta dunque i vantaggi e svantaggi TLS, non potendo alla fine garantire una vera sicurezza end-to-end se non tutti i nodi intermedi nella comunicazione implementano questo protocollo.

6.1.5. Secure MIME

MIME è uno standard Internet che estende il formato di una email al fine di supportare una serie di funzionalità aggiuntive (testo non ASCII, allegati non testuali, e messaggi con corpo composto). Questo è rilevante perchè i messaggi SIP sono in grado di trasportare dei body MIME, e lo standard MIME include anche una serie di meccaniche per garantire integrità o confidenzialità dei contenuti. Vengono usati dei certificati per identificare gli utenti finali sulla base del loro indirizzo email, che è parte del SIP-URI (es. *alice@domainA.com*): ciascun utente dispone di una propria chiave privata, ed il proprio certificato viene inoltrato al destinatario incapsulato come allegato MIME.

S/MIME mette a disposizione un insieme di funzionalità, due delle quali utilizzate da SIP: **Integrity and Authentication Tunneling** e **Tunneling Encryption**, che richiedono l'implementazione di una infrastruttura globale con

chiave pubblica S/MIME. Con questi meccanismi, si può garantire l'integrità dell'header incapsulando e cifrando l'intero messaggio SIP, inclusi gli header, nel body del messaggio esterno.

Come nel caso di IPsec, S/MIME genera un overhead considerevole nei messaggi SIP, che richiedono inoltre uno sforzo computazionale maggiore per essere processati, e non è in grado di garantire l'integrità e la confidenzialità dell'intero messaggio SIP a causa delle restrizioni sulla modifica dell'header, dato che i nodi intermedi devono avere accesso all'header SIP per poter processare ed inoltrare i messaggi verso la destinazione appropriata, e rimane suscettibile ad attacchi Man-in-the-Middle.

6.1.6. Secure RTP

SRTP estende il protocollo RTP con meccanismi di crittografia per garantire confidenzialità, integrità e autenticazione dei messaggi RTP tramite Digest con chiave pubblica, e protezione da attacchi replay utilizzando un valore di nonce (già visto implementato in HTTP Digest). Per quanto riguarda SIP, SRTP si occupa di proteggere esclusivamente il body del messaggio, e non l'intestazione dove avviene la segnalazione.

6.2. Difese della Rete VoIP

La protezione della segnalazione SIP è fondamentale quando si inoltrano dei pacchetti in una rete come Internet, una rete non sicura dove non si ha controllo delle entità che parteciperanno nell'inoltro dei pacchetti inviati. E' necessario quindi cercare di garantire l'autenticità dei messaggi e delle entità coinvolte, nonché la confidenzialità dello scambio di informazioni e la loro integrità, ed i protocolli visti in precedenza lavorano esattamente in quella direzione, con i relativi pro e contro.

Tuttavia, è strettamente necessario ad esempio *cifrare tutto il traffico che viene scambiato all'interno della propria rete VoIP?* La risposta è no, ma a

condizione che gli accessi ai servizi e alle risorse della rete, al traffico multimediale riservato e a dati sensibili, siano adeguatamente gestiti.

6.2.1. Sicurezza Logica

Per poter fare ciò, è fondamentale anzitutto schermare adeguatamente la propria rete privata **monitorando rigorosamente gli accessi**: si deve mantenere una lista di utenti, con relativi indirizzi, numeri di telefono, dati anagrafici, ecc. per poterli identificare, e registrare tutte le attività di ciascun utente tramite un'opportuna reportistica come un log (diario) di ciascun evento avvenuto nella rete.

Si associ a ciascun utente della rete un diverso livello di **privilegio di accesso** a diversi contenuti della rete basandosi su una gerarchia di responsabilità. Ad esempio, un utente finale dovrà avere un numero di privilegi di utilizzo ed accesso inferiore rispetto ad un amministratore di rete. Questo perchè una rete VoIP ha dei dati e dei flussi multimediali che richiedono diversi gradi di confidenzialità e di accesso a seconda della relativa importanza.

Per esempio, un identificativo di un utente o un suo indirizzo può essere salvato in chiaro in un database interno alla rete, ma informazioni sensibili quali la relativa password o un numero di carta di credito devono essere cifrati anche all'interno di una rete privata e protetta, in quanto dovrebbero essere strettamente confidenziali e non accessibili nemmeno da un amministratore di rete, nè tantomeno da altri utenti della rete. Una suddivisione logica della confidenzialità dei dati riduce drasticamente l'impatto di un attacco di **phishing**, in quanto un utente della rete ha accesso ad una quantità limitata di informazioni.

Infine, un'ulteriore **separazione del traffico dati dal traffico voce** in una rete VoIP evita che un attacco ad uno dei due tipi di traffico impatti anche sull'altro. E' opportuno, per esempio, mantenere le informazioni relative ad un utente della rete in un server diverso da quello utilizzato, ad esempio, per il forwarding. Nel caso di SIP, si pensi di separare il location database dal servizio di registrazione o di inoltra di messaggi, collocando quindi le informazioni di

locazione su un server diverso da un registrar SIP. Quest'ultimo può accedere ai dati nel location server tramite un'opportuna interrogazione.

Separare la logica delle funzionalità delle entità della rete e lo storage dei dati è un'utile **protezione da attacchi DoS**: un'offensiva atta ad inibire o neutralizzare un server SIP non andrebbe a colpire il server su cui sono solo collocati i dati. Si pensi di avere, ad esempio, due registrar server per una rete VoIP, ed uno (o più) location server. Un attacco DoS di tipo flooding o fuzzing nei confronti di uno dei registrar potrebbe metterlo fuori uso, ma il secondo server registrar o un redirect server potrebbero sempre utilizzare il location service.

6.2.2. Firewall

Proteggere una rete VoIP significa anche evitare che entità non desiderate possano accederne ai contenuti. Un firewall è un sistema software che garantisce protezione contro connessioni di rete non autorizzate, schermando una rete privata locale sia da attacchi esterni, sia da tentativi di connessione iniziati da applicazioni installate su macchine locali verso reti esterne.

La potenza del firewall risiede nel poter filtrare le connessioni di rete basandosi su una serie di diversi aspetti, che vanno dai tipi di protocolli utilizzati, a livello di trasporto o applicativo, agli indirizzi sorgenti e di destinazione dei partecipanti alla connessione, ecc. Un'opportuna configurazione, dettata dalle policy di sicurezza del gestore della rete VoIP, degli strumenti forniti da questo software consente quindi di controllare quali connessioni possono essere stabilite fra l'interno e l'esterno.

Il traffico di rete può essere inoltrato per passare tramite il firewall, solitamente situato su un'apparecchiatura ai bordi della rete (ad esempio un router) dove viene ispezionato e opportunamente filtrato prima di proseguire verso l'interno o l'esterno, a seconda della provenienza. La comodità di questo sistema è una centralizzazione del controllo delle connessioni in un unico punto della rete, mantenendo più facilmente hacker o altri intrusi dall'entrare o dall'uscire dalla rete protetta.

Nel caso più specifico di SIP, il firewall serve a bloccare chiamate in entrata o in uscita dalla rete VoIP. Lo scopo è quello di centralizzare la sicurezza della rete tramite il punto di controllo del firewall invece di dover configurare ciascun endpoint SIP con le policy di sicurezza.

Un firewall può quindi offrire una protezione agli utenti interni alla rete VoIP contro tutti quegli **attacchi sociali** provenienti dall'esterno, bloccando indirizzi e messaggi non autorizzati da contatti non voluti. DoS

6.2.3. Intrusion Detection System

IDS è un insieme di strumenti atto al rilevamento di intrusioni in un sistema o di una violazione delle sue regole, e si basa sul principio di far scattare un allarme quando ci sia un attacco in corso. Un network IDS si occupa nello specifico di esaminare il traffico di rete e monitorarne gli host, grazie ad un interfacciamento con un'apparecchio che inoltra il traffico, come un hub di rete o uno switch configurato con l'opzione di port mirroring. In questo tipo di sistema, i sensori sono collocati in quei punti che fanno da collo di bottiglia per il traffico della rete da monitorare, come i router di bordo: catturano tutto il traffico e analizzano pacchetto per pacchetto alla ricerca di contenuti sospetti. IDS più completi includono anche meccaniche per prevenire le minacce una volta rilevate. Non è detto che un'anomalia qualsiasi rappresenti una minaccia (es. un utente che ha semplicemente digitato un indirizzo errato).

Anche se lo scopo rimane quello di limitare il numero di intrusioni nella rete, un IDS è diverso da un firewall: il primo rileva le intrusioni, il secondo impedisce che accadano, e sono quindi rispettivamente programmi reattivi e proattivi.

6.3. Considerazioni

L'intestazione SIP, essendo esso un protocollo basato su testo in chiaro le cui informazioni possono essere facilmente *sniffate*, necessita di meccanismi di protezione per garantire l'autenticità dei messaggi scambiati e la loro integrità, e la confidenzialità e l'autenticità per le parti coinvolte. Tuttavia non si può cifrare end-to-end un intero messaggio, perchè alcune informazioni riguardanti la segnalazione, come i campi **Contact**, **To**, e **Via** sono fondamentali per l'inoltro dei messaggi da parte dei server SIP. L'intestazione dei pacchetti SIP può quindi essere cifrata su una base hop-by-hop da ciascun server SIP intermediario, così da garantire comunque la protezione da sniffing ed eavesdropping.

Non è necessario cifrare completamente ciascun pacchetto se questo viaggia all'interno di una rete privata protetta dove esiste un rigoroso controllo degli accessi ai relativi dati e servizi, tramite meccanismi quali **IDS** e **Firewall** che implementino le policy di sicurezza e corretto utilizzo della rete.

I pacchetti devono invece essere protetti quando lasciano l'infrastruttura protetta. SIP supporta meccanismi di **autenticazione basic** ed uno più robusto **digest**, utili ai proxy server per identificare gli utenti, ma non viceversa.

Un'estensione di SIP, **SIPS**, fa uso di **TLS** per garantire **autenticità, integrità, e confidenzialità**, con tutti i relativi vantaggi e limitazioni visti. Per protezioni più complesse e computazionalmente più pesanti, **IPSec** e **S/MIME** implementano un incapsulamento totale dei pacchetti trasmessi ed un'autenticazione basata su un'infrastruttura di scambio di chiavi.

Tutti i sistemi di sicurezza analizzati in questo capitolo servono a migliorare drasticamente la sicurezza del protocollo SIP, ma rimangono vulnerabili agli attacchi **Man-in-the-Middle**, in particolar modo se la terza parte in questione risiede al di fuori della rete protetta (ad es. nella rete Internet).

La tabella finale riporta una breve sintesi dei tipi di attacchi e delle contromisure attuabili.

Attacchi	Impatto	Soluzione
Eavesdropping	Perdita privacy e confidenzialità	Cifratura: TLS, S/MIME, IPSec
Virus e Software bug (Malformed Messages)	DoS / accesso non autorizzato	Aggiornamento software (patch), firewall, IDS.
Replay	DoS	Cifra hop-by-hop i campi Call-ID e CSeq, e utilizza valori nonce.
Spoofing (furto identità)	accesso non autorizzato	Autenticazione HTTP Digest
Message modification	DoS / perdita integrità	Cifratura: TLS, S/MIME, IPSec
SPAM, SPIT, SPIM	DoS	Firewall, IDS

Tabella 1: Sintesi di alcune categorie di attacco, impatto, e soluzioni [10].

Capitolo 7

7. Conclusioni

L'adozione delle tecnologie VoIP da parte di enti pubblici e privati è in rapido aumento. Le reti di servizi voce, multimediali, e di messaggistica offrono grande flessibilità e una varietà di servizi superiore rispetto alla rete telefonica generale, e ad un costo decisamente inferiore dovuto alle apparecchiature richieste e ai nuovi modelli di business adottati. Tuttavia, i sistemi VoIP hanno un'architettura e dei protocolli più complessi da implementare, che può portare facilmente a falle di sicurezza.

L'elaborato ha offerto una breve panoramica sulle **specifiche SIP** di base al fine di dare al lettore gli strumenti necessari per poter comprenderne la semantica generale ed il funzionamento. Ha mostrato poi un quadro generale su quelle che sono le **classi di vulnerabilità** delle reti VoIP, riportando una sintesi dei concetti definiti in maniera formale nella tassonomia VoIPSA, concentrandosi maggiormente sulla parte informatica e che più si avvicina ai problemi di sicurezza nell'ambito della segnalazione e del traffico multimediale. Successivamente, si è andati ad analizzare le tipologie di **attacchi a SIP** collocandole in una delle tre classi di vulnerabilità appropriate, e si sono mostrati i **tre esempi di attacchi a SIP** ritenuti più significativi, ovvero più completi come meccaniche da un punto di vista analitico sulle semantiche della segnalazione del protocollo. A seguire, si sono presentate alcune **contromisure** per prevenire e respingere gli attacchi al protocollo, ed alcune precauzioni da utilizzare all'interno della rete VoIP che implementa SIP come protocollo di segnalazione.

Per concludere la tesi, si propone al lettore una mappatura generica delle minacce e rischi dei sistemi VoIP, derivante da un'indagine sulle vulnerabilità e

sugli attacchi conosciuti che sono stati riportati su un vasto insieme di articoli scientifici.

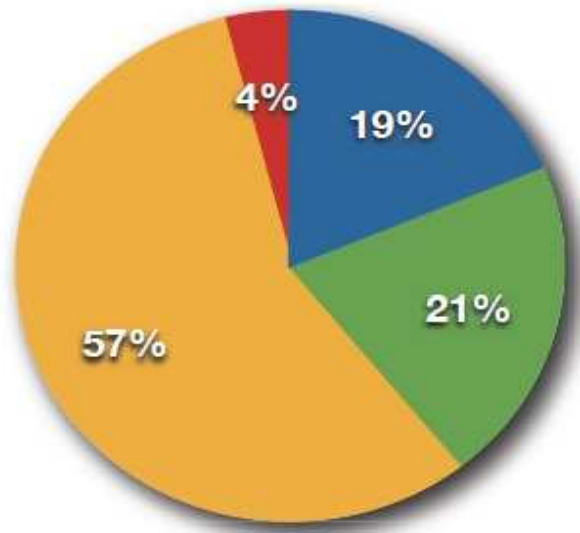
7.1. La Ricerca sulla Sicurezza nelle Reti VoIP

Angelos Keromytis, professore alla Columbia University di New York, direttore del Network Security Lab, membro attivo di IETF e IPSec Working Groups, e senior member di IEEE (Institute of Electrical and Electronics Engineer), ha condotto nel corso degli ultimi tre anni diverse indagini riguardo alle minacce e ai rischi nelle reti VoIP [2], [12], [13].

In uno di questi articoli [12] viene proposto un esame dello stato corrente della sicurezza VoIP in generale, con lo scopo di fornire un punto di partenza utile per comprendere l'argomento e una base per un'analisi più completa per il futuro nell'ambito della ricerca scientifica. L'articolo offre una breve panoramica su SIP come tecnologia utilizzata, una estrema sintesi della tassonomia VoIPSA, ed una discussione delle minacce a VoIP conosciute. L'esame è stato fatto su 215 vulnerabilità riscontrate in implementazioni SIP che sono state raccolte nel database CVE (Common Vulnerabilities and Exposures), un dizionario di informazioni sulla sicurezza pubblicamente conosciute, dal 1999 al 2009.

Keromytis descrive ciascuna vulnerabilità usando una tupla di tre valori (V,T,K) dove V indica la classe di vulnerabilità VoIPSA a cui concettualmente appartiene, T indica se la vulnerabilità riguarda l'aspetto di confidenzialità, integrità, oppure disponibilità del sistema, e K indica se la vulnerabilità deriva da un errore di protocollo, implementazione, o configurazione.

I risultati numerici ottenuti sono riportati nelle figure seguenti. La maggior parte delle vulnerabilità esaminate riguarda minacce DoS, attacchi che hanno un forte impatto sul funzionamento parziale o globale dei servizi della rete. Per il resto delle vulnerabilità si tratta in circa la metà dei casi di minacce di eavesdropping e hijacking (inteso come Conversation Impersonation and Hijacking, cioè Man-in-the-Middle), e per l'altra metà di minacce sociali.



- Social threats (1)
- Denial of Service (3)
- Physical Access (5)
- Eavesdropping, hijacking (2)
- Service Abuse (4)
- Interruption of Services (6)

Fig. 15: Suddivisione delle vulnerabilità secondo la tassonomia VoIPSA

La conseguenza logica di un numero elevato di attacchi DoS è un numero proporzionalmente alto di minacce alla disponibilità dei servizi, nonché una quantità sostenuta di casi in cui viene compromessa l'integrità del sistema, come in casi di DoS buffer overflow o di casi Man-in-the-Middle. In proporzione, il ridotto rilevamento di attacchi alla confidenzialità del sistema rappresenta un'inversione delle minacce percepite dagli utenti e dagli amministratori che, come suggeriscono gli aneddoti, si preoccupano tipicamente di problemi come intercettazione di chiamate e eavesdropping (Keromytis, [12]).

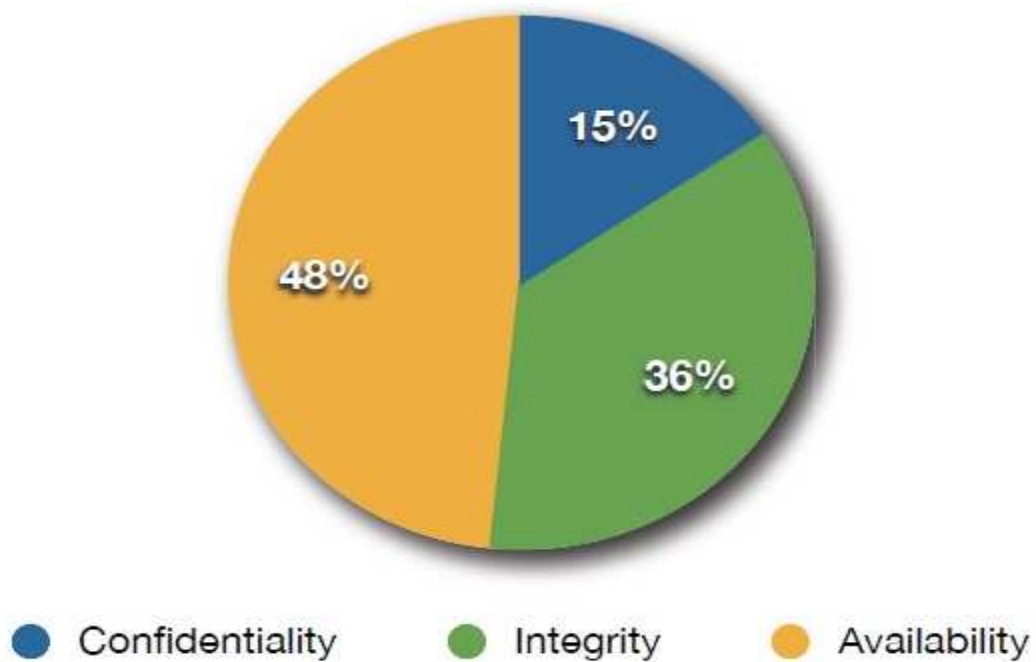


Fig. 16: Suddivisione delle minacce per classificazione “tradizionale”

Infine, viene mostrata la suddivisione delle vulnerabilità sulla base delle relative cause. La grande maggioranza dei problemi emerge da errori di implementazione, che non dovrebbe essere una sorpresa visto che le minacce conosciute sono estratte da una reportistica di bug-tracking. I problemi riguardanti la configurazione derivano da errori quali cattiva gestione dei privilegi e delle credenziali d’accesso. Nonostante vi siano poche minacce derivanti da errori di protocollo, semplicemente non c’è scusa per questo tipo di errori. Una percentuale, anche se relativamente bassa, indica una complessità del protocollo SIP ancora troppo elevata.

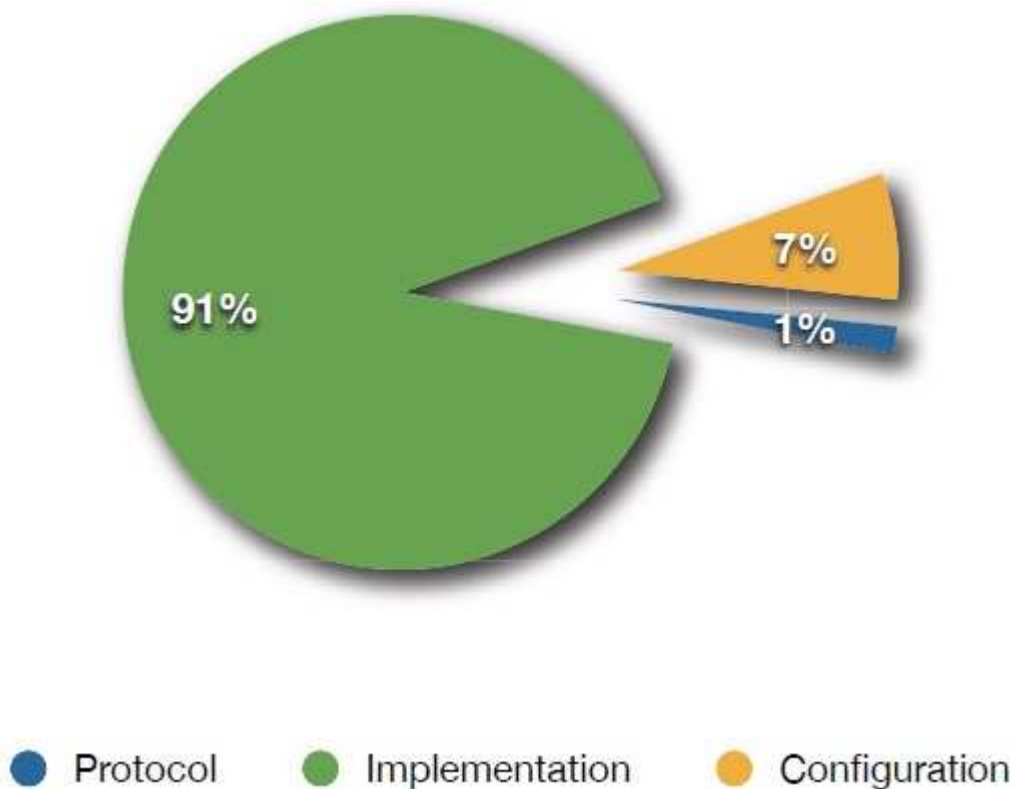


Fig. 17: Suddivisione delle minacce classificate per causa

Per concludere, si evince dall'esame effettuato da Keromytis che la maggior parte delle minacce per una rete VoIP che implementi il protocollo SIP come tecnologia di segnalazione derivino da attacchi DoS basati su problemi di implementazione della rete, vista la facilità con cui possono venire eseguiti contro gli utenti della rete. La causa di queste vulnerabilità sembra essere una implementazione inaccurata dovuta ad una complessità troppo elevata del protocollo di segnalazione e da tutti i componenti VoIP da dover integrare. L'uso di strumenti atti alla controffensiva di attacchi DoS, quali firewall e IDS, è un prezioso aiuto, ma devono essere adeguatamente accompagnati da misure di sicurezza di livello più alto, che agiscano sullo strato applicativo, come aggiornamenti software, patch, e antivirus per neutralizzare programmi malevoli (Keromytis, [12]).

7.2. Considerazioni Finali

SIP è un protocollo di segnalazione inventato per creare, modificare, e terminare sessioni di comunicazione voce e multimediali che, grazie al supporto per l'interazione con diversi componenti di rete, un'indipendenza dallo strato di trasporto (e dai media trasportati), e una base della segnalazione su testo in chiaro lo rendono una soluzione valida per la realizzazione di servizi VoIP.

Sebbene le specifiche di base SIP siano relativamente semplici e facili da implementare, il numero elevato di interazioni con altri componenti in uno scenario reale ha portato ad un'estensione drastica del protocollo e ad un'ulteriore crescita della complessità di implementazione, da cui derivano circa il 90% delle minacce alla rete di servizi VoIP. Maggiore la complessità del protocollo dovuta a varie estensioni, maggiore anche la difficoltà nel eseguire un debug completo, lasciando quindi ancora una piccola percentuale di errori intrinseci nel protocollo.

La tassonomia VoIPSA fornisce a ricercatori e ad amministratori di rete un utile strumento per conoscere quelle che sono le vulnerabilità alle implementazioni VoIP. Nella tesi corrente, si è andati a descrivere quelli che sono gli attacchi alla segnalazione SIP e a collocarli all'interno delle categorie descritte dalla tassonomia. Dall'analisi effettuata nella sezione precedente, si evince che attacchi DoS al protocollo SIP siano una minaccia reale, dovuti nella quasi totalità dei casi a cattive implementazioni. Come visto nel capitolo 5, attacchi DoS sono facilmente comprensibili e realizzabili tramite strumenti reperibili legalmente e gratuitamente sulla rete internet.

Comprendere le meccaniche di SIP è fondamentale per una corretta implementazione della segnalazione, ed è necessario che vengano prese di conseguenza tutte le contromisure necessarie per proteggere il traffico SIP e le relative apparecchiature. Non essendo possibile garantire una cifratura end-to-end dei parametri della segnalazione, si devono prendere delle precauzioni per cifrare questi valori, ove possibile, su una base hop-by-hop, utilizzando quei protocolli di sicurezza supportati da SIP per cercare di garantire l'autenticità delle entità

coinvolte, la confidenzialità e l'integrità delle informazioni fra loro scambiate. In una rete privata e protetta, si può favorire un'implementazione più agile ed efficiente solo se il traffico da proteggere rimane all'interno della rete. Una volta che i pacchetti SIP lasciano la rete protetta, magari per essere inoltrati nella rete Internet, si perde la garanzia di mantenere le tre proprietà della sicurezza appena elencate. E' essenziale in questo caso l'utilizzo di quei protocolli di sicurezza supportati da SIP che cifrano i messaggi scambiati e garantiscono l'autenticità e l'integrità della comunicazione. Tuttavia, attacchi di tipo Man-in-the-Middle dall'esterno della rete protetta rimangono un problema, che possono essere ridotti solo se il traffico viene inoltrato attraverso altre reti protette. Per reti protette si intendono delle infrastrutture protette adeguatamente da buone norme di sicurezza come la suddivisione logica tra traffico dati e voce, privilegi di accesso, firewall e sistemi di identificazione di intrusioni, che limitano prima di tutto le minacce DoS, e poi anche attacchi sociali e Man-in-the-Middle all'interno della rete.

In che direzione si muove dunque la ricerca sulla sicurezza dei sistemi VoIP? Keromytis propone in un altro survey [2] un esame di 245 pubblicazioni su questo argomento, concludendo che due aree specifiche della tassonomia VoIP, Denial of Service e Service Abuse, non hanno raccolto ancora il giusto impegno nella ricerca a soluzioni, in relazione all'importanza da lui attribuitagli dai suoi esami passati sulla sicurezza VoIP. Infine, cattive configurazioni di rete e bug di implementazione sono aree che meritano un interesse maggiore di quello raccolto finora (Keromytis, [2]).

Guardando al futuro allora, la prevenzione, l'intercettazione, e il neutralizzamento di attacchi DoS dovrebbe essere la priorità da affrontare. Un'offensiva di questo genere reca danni agli utenti che non possono usufruire dei servizi di rete VoIP, e dei servizi di rete in generale se questi non sono accuratamente separati nella logica del sistema, e crea danni economici ai gestori. Varrebbe la pena quindi concentrarsi su difese efficaci della rete VoIP privata e della realizzazione di collegamenti sicuri tramite reti protette per inoltrare il traffico verso una destinazione esterna. All'interno della rete protetta, mantenere un'implementazione semplice del protocollo di segnalazione non solo migliora le prestazioni dell'inoltro del traffico locale, ma rende più semplice evitare proprio

quegli errori di configurazione e implementazione che sono la causa degli attacchi presentati.

Bibliografia

1. SIP: Session Initiation Protocol, IETF Standard 3261, Giugno 2002.
2. Angelos D. Keromytis. *A Comprehensive Survey of Voice over IP Security Research*. Communication Surveys & Tutorials, IEEE, secondo trimestre 2012.
3. Aldo Campi. *SIP: Session Initiation Protocol*. AMS Campus, ALMA MATER STUDIORUM – Università di Bologna, 2008.
4. VoIP Security Alliance. *VoIP Security and Privacy Threat Taxonomy*. Public release 1.0, 24 October 2005.
5. Patrick Park. *Voice over IP Security*. Cisco Press, 9 Settembre 2008.
6. Wireshark, versione 1.8.6. <http://www.wireshark.org>
7. SIPp, versione 3.2. <http://sipp.sourceforge.net/>
8. Ulf Lamping , Richard Sharpe , Ed Warnicke. *Wireshark User's Guide*. http://www.wireshark.org/docs/wsug_html_chunked , 2012.
9. W. Werapun, A. Abou EI Kalam. *Solution Analysis for SIP Security Threats*. Multimedia Computing and Systems, 2009. ICMCS '09. International Conference.
10. El Sawda, Samer. *SIP Security Attacks and Solution: A state-of-the-art review* . Information and Communication Technologies, 2006. ICTTA '06.
11. SecureLogic. *Voice and Unified Communications State of Security Report 2012*. SecureLogic, May 6, 2012.
12. Angelos D. Keromytis. *Voice over IP: Risks, Threats, and Vulnerabilities*. Proceedings of the Cyber Infrastructure Protection (CIP) Conference, 2009.
13. Angelos D. Keromytis. *A look at VoIP Vulnerabilities*. ;login: Journal, Vol. 35, Febbraio 2010.

Ringraziamenti

Al Prof. Cerroni e al Prof. Campi, per il genuino interesse nell'insegnamento della loro materia, e per la disponibilità con la quale mi hanno seguito durante il corso e durante la tesi.

Ai miei, che hanno sponsorizzato la mia carriera universitaria e, alla fine, anche il mio futuro.

A quei compagni di corso che mi hanno aiutato a superare alcuni degli ostacoli del cammino universitario.

A chi ha avuto la pazienza di aspettarmi, e a chi continuerà ad averne in futuro se dovessi rimanere indietro. La vita è un cammino unico per ognuno di noi, ma l'arrivo è lo stesso per tutti. Tanto vale aspettarsi a vicenda e percorrerlo insieme.