UNIVERSIDADE FEDERAL FLUMINENSE

Rogério Ferreira da Cunha

Abordagens para escuta legal nas redes de Voz sobre IP

NITERÓI
2008

UNIVERSIDADE FEDERAL FLUMINENSE

Rogério Ferreira da Cunha

Abordagens para escuta legal nas redes de Voz sobre IP

Dissertação submetida ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre. Área de concentração: Processamento Distribuído e Paralelo.

Orientador:

Prof. Michael Stanton, Ph.D.

NITERÓI
2008

Ficha Catalográfica elaborada pela Biblioteca da Escola de Engenharia e Instituto de Computação da UFF

C972 Cunha, Rogério Ferreira da.
Abordagens para escuta legal nas redes de voz sobre IP / Rogério Ferreira da Cunha. – Niterói, RJ : [s.n.], 2008.
151 f.

Orientador: Michael Stanton.
Dissertação (Mestrado em Computação) - Universidade Federal Fluminense, 2008.

1. Protocolo de comunicação. 2. Voz sobre IP. 3. Protocolo de comunicação de rede. 4. Telefonia. 5. Escuta telefônica. 6. Interceptação telefônica. I. Título.

CDD 004.6

Abordagens para escuta legal nas redes de Voz sobre IP

Rogério Ferreira da Cunha

Dissertação submetida ao Programa de Pós-Graduação em Computação da Universidade Federal Fluminense, como requisito parcial para obtenção do título de Mestre. Área de concentração: Processamento Distribuído e Paralelo.

Aprovado por:

Prof. Michael Anthony Stanton, Ph.D. / IC-UFF
(orientador)

Prof. Célio Vinicius Neves de Albuquerque, Ph.D. / IC-UFF

Prof. Paulo Henrique Aguiar Rodrigues, Ph.D. / UFRJ

Prof. Ricardo Felipe Custódio, Ph.D. / PPGCC-UFSC

Prof. Ronaldo Lemos, Ph.D. / DIREITO-RJ-FGV

Niterói, 17 de março de 2008.

Para meus filhos, Bernardo e Arthur.

Agradecimentos

Agradeço à minha esposa, Ana Beatriz, pela compreensão e incentivos ao longo do caminho, ao meu orientador, Michael Stanton, pelas sugestões que auxiliaram na construção do trabalho e, finalmente, às secretárias do departamento, Ângela e Maria, pela cortesia de sempre.

Resumo

A escuta legal é uma ferramenta investigativa empregada quando os métodos de investigação tradicionais são insuficientes para produzir prova, desde que devidamente autorizada pelas autoridades competentes. Enquanto que na telefonia tradicional a escuta pode ser realizada, muitas vezes, apenas através de uma derivação da linha do assinante, em Voz sobre IP a questão pode tornar-se complexa, por conta das características intrínsecas de mobilidade, segurança e diferenciação de caminhos percorridos pelos tráfegos de sinalização e mídia. Em face do uso crescente dos sistemas de Voz sobre IP, legislações e normas têm surgido para regulamentar os aprovisionamentos dos conteúdos de voz interceptados pelos provedores de serviço, em decorrência das solicitações legais. Ainda assim, mesmo que uma determinada arquitetura forneça as condições necessárias para identificar e interceptar o tráfego em tempo hábil, atendendo a regulamentações, há ainda possíveis dificuldades na interpretação deste conteúdo se as mídias estiverem protegidas por mecanismos que garantam a privacidade. Ao contrário das redes telefônicas tradicionais, as características de criptografia estão padronizadas nos protocolos de Voz sobre IP, e o seu uso rotineiro poderá tornar o processo de escuta legal impraticável. Assim, esta dissertação apresenta as iniciativas sobre escuta legal e os detalhes específicos do protocolo SIP, incluindo seus mecanismos de segurança associados, evidenciando as dificuldades técnicas que seu uso pode impor à efetivação de escuta legal, para em seguida ser proposto um método alternativo de negociação das chaves de sessão, utilizadas na proteção das mídias, de forma a permitir seu armazenamento, possibilitando a continuidade deste instrumento investigativo. No final são apresentadas comparações da proposição com as demais possibilidades de recuperação de chaves pelos métodos padronizados, indicando as vantagens e desvantagens na aplicação dos casos em um sistema de custódia de chaves.

Palavras-chave: VoIP. SIP. Escuta legal. Interceptação. Kerberos.

Abstract

Lawful interception of telephone calls is a monitoring tool used where the approaches of traditional inquiry are insufficient to produce legal evidence, once properly authorized by the courts. Whereas in traditional telephone technology interpretation can be carried out by just tapping the subscriber's line, in Voice over IP the question can become complex, by its inherent characteristics of mobility, security and the varying paths traversed by signaling and media traffic. Due to the growing use of the Voice over IP systems, laws and technical standards have been created to regulate the intercepted voice traffic by service providers, in consequence of court orders. Nevertheless, even if a specific infrastructure might provide the necessary conditions to identify and intercept the traffic in a reasonable time, in accordance with the regulations, there are still possible difficulties in the interpretation of this content itself, if the media is using mechanisms to guarantee privacy. In contrast to the plain old telephonic system, Voice over IP protocols provide cryptography as a standard feature, and its routine use will make lawful interception process impractical. This dissertation describes the initiatives to provide lawful interception and details of the SIP protocol, including its associated security protocols, pointing up the technical difficulties to proceed with wiretapping in this environment. It presents a new proposal to manage session keys used to protect media streams, as a way to permit the key recovery. Finally, comparisons of the proposal are made with other standards-based alternatives for key recovery, indicating the advantages and disadvantages in the application of the cases where a key escrow system is used.

Key-words: VoIP. Lawful Interception. SIP. Kerberos.

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

ACK	:	Acknowledgement
ADSL	:	Asymmetric Digital Subscriber Line
AES	:	Advanced Encryption Standard
ALG	:	Application Layer *Gateway*
AOR	:	Address-of-record
API	:	Application Programming Interface
AS	:	Authentication Server
ASN.1	:	Abstract Syntax Notation number One
ATM	:	Asynchronous Transfer Mode
AVP	:	Audio Video Profile
B2BUA	:	Back *to* Back User Agent
BGP	:	Border Gateway Protocol.
BLD	:	Botton-Level Devices
BRI	:	Basic Rate Interface
CALEA	:	Communications Assistance for Law Enforcement Act
CallerID	:	Caller Identification Number
CAS	:	Common Associate Signaling
CCS	:	Common Channel Signaling
CDR	:	Call Detail Record
CM	:	Counter Mode
CODEC	:	Coder-Decoder
CPA	:	Central de Programa Armazenado
CRLF	:	Carriage Return and Line Feed
cRTP	:	Compressed Real Time Protocol
CS	:	Cryptography Session
CSB	:	Crypto Session Bundle
CSBID	:	Crypto Session Bundle Identification
DES	:	Data Encryption Standard
DH	:	Diffie-Hellman
DHCP	:	Dynamic Host Configuration Protocol
DNS	:	Domain Name System
DTLS	:	Datagram TLS
E911	:	Enhanced 911 emergency-calling system
ENUM	:	T*E*lephone *NU*mber *M*apping

ETSI	:	European Telecommunications Standards Institute
FBI	:	Federal Bureau of Investigation
FCC	:	Federal Communications Commission
FQDN	:	Full Qualified Domain Name
FTTH	:	Fiber To The Home
GSM	:	Global System for Mobile Communications
H.323	:	Audio-visual communication Standard Series H.
HMAC	:	Hash Message Authentication Code
HTTP	:	Hypertext transfer protocol
IAB	:	Internet Architecture Board
IANA	:	Internet Assigned Numbers Authority
IBCS	:	Identity-based Cryptography Standard
IBE	:	Identity-based Encryption
ICE	:	Interactive Connectivity Establishment
IESG	:	Internet Engineering Steering Group
IETF	:	Internet Engineering Task Force
IKE	:	Internet Key Exchange Protocol
ILD	:	Intermediate-Level Devices
IP	:	Internet Protocol
IPSEC	:	IP Security Protocol
ISDN	:	Integrated Services Digital Network
ITU	:	International Telecommunication Union
ITU-T	:	ITU Telecommunication Standardization Sector
IVR	:	Interactive Voice Response
KDC	:	Key Distribution Center
KEMAC	:	Key Data Transport Payload
KRB	:	Kerberos
KRB5	:	Kerberos Version No. 5
LEA	:	Law Enforcement Agency
LI	:	Lawful Interception
LS	:	Location Server
MD5	:	Message-Digest Algorithm No. 5
MEGACO	:	Media *Gateway* Control Protocol
MG	:	Media *Gateway*
MGC	:	Media *Gateway* Controller
MGCP	:	Media *Gateway* Control Protocol
MIKEY	:	Multimedia Internet Keying
MIME	:	Multipurpose Internet Mail Extensions
MIT	:	Massachusetts Institute of Technology

MitM	:	Man-in-the-middle
MMUSIC	:	Multiparty Multimedia Session Control
MTU	:	Maximum Transmission Unit
NAT	:	Network Address Translation
NETCONF	:	Network Configuration Protocol
NIST	:	National Institute of Standards and Technology
P2P	:	Peer-*to*-Peer
PCM	:	Pulse-Code Modulation
PFS	:	Perfect Forward Secrecy
PKCS	:	Public Key Cryptography Standards
PKG	:	Private Key Generator
PKI	:	Public Key Infrastructure
PPS	:	Public Parameter Server
PRI	:	Primary Rate Interface
PSD	:	Protocolo de Segurança de Dados
PSK	:	Pre-Shared Key
PSTN	:	Public Switched Telephone Network
QoS	:	Quality of Service
QSIG	:	Q signaling
RBAC	:	Role Based Access Control
RFC	:	Request for Comments
RP	:	Rendezvous Point
RSVP	:	Resource Reservation Protocol
RTCP	:	Real Time Control Protocol
RTP	:	Real Time Protocol
S/MIME	:	Secure Multipurpose Internet Mail Extensions
SA	:	Security Association
SAS	:	Short Authentication String
SAVP	:	Secure AVP
SCP	:	Service Control Point
SCTP	:	Stream Control Transmission Protocol
SD	:	Storage Device
SDES	:	Session Description Protocol Security
SDP	:	Session Description Protocol
SHA-1	:	Secure Hash Algorithm, Number 1.
SIP	:	Session Initiation Protocol
SPAM	:	Spamming
SPIT	:	Spam over Internet Telephony
SRTCP	:	Secure Real Time Control Protocol

SRTP	:	Secure Real Time Protocol
SS#7	:	Signaling System #7
SSH	:	Secure Shell
SSL	:	Secure Socket Layer
SSP	:	Service Switching Point
SSRC	:	Synchronization source
STP	:	Signal Transfer Point
STUN	:	Simple Traversal of User Datagram Protocol
TCP	:	Transmission Control Protocol
TCP/IP	:	Transmission Control Protocol / Internet Protocol
TDM	:	Time-Division Multiplexing
TEK	:	Traffic-Encrypting Key
TGK	:	TEK Generation Key
TGS	:	*Ticket* Granting Server
TGT	:	*Ticket* Granting *Ticket*
TLS	:	Transport Layer Security
TURN	:	Traversal Using Relay NAT
UA	:	User Agent
UAC	:	User Agent Client
UAS	:	User Agent Server
UDP	:	User Datagram Protocol
URA	:	Unidade de Resposta Automática
URI	:	Universal Resource Identifier
URL	:	Universal Resource Locator
VOIP	:	Voice over Internet Protocol
WiMAX	:	Worldwide Interoperability for Microwave Access
WPA	:	Wireless Fidelity Protected Access
XML	:	Extensible Markup Language
ZRTP	:	Media Path Key Agreement for Secure RTP

Sumário

Capítulo 1

Introdução

A Constituição Federal de 1989, no seu artigo 5°, inciso XII, assegura ao cidadão a inviolabilidade do sigilo das comunicações, sejam elas por correspondências, por sistemas telefônicos, telegráficos e de dados, com exceção aplicada aos casos de investigação criminal ou processual penal, quando, através de ordem judicial, poderá ser quebrado o sigilo da comunicação telefônica [1].

A Lei Nº. 9.296, de 24 de Julho de 1996 regulamenta o inciso XII da Constituição, detalhando os casos onde pode ser aplicada a interceptação legal da comunicação e os agentes autorizados a solicitar e autorizar o processo. Também determina que a "interceptação deva ocorrer em autos apartados, apensados aos autos do inquérito policial ou do processo criminal, preservando-se o sigilo das diligências, gravações e transcrições respectivas" [2].

Independente da histórica polêmica entre o direito do Estado sobre o direito individual, esta é uma condição estabelecida e aceita como meio auxiliar quando os métodos de investigação tradicionais são insuficientes para produzir prova.

No ambiente telefônico a quebra do sigilo pode ter dois objetivos: o primeiro apenas para identificar a relação de contatos estabelecidos, pelo acesso aos dados históricos dos sistemas de tarifação; o segundo, para interceptar e interpretar a comunicação telefônica, através da aplicação de sistemas de escuta, ou simplesmente "grampo".

A aplicação da escuta sobre as redes de telefonia tradicionais é algo tecnicamente simples de ser realizado. As escutas, em geral, quando autorizadas, contam com o apoio das empresas de telefonia, que permitem acesso do agente autorizado às centrais de comutação telefônica. Em alguns países, as agências de segurança contam com algumas linhas privativas entre os

centros de investigação e a principais centrais de comutação, permitindo a derivação da linha de um assinante ou acesso a um ponto de monitoração do equipamento comutador. Mesmo nos sistemas de telefonia celular, inclusive nos sistemas de segunda e terceira geração, que possuem criptografia na transmissão em espaço livre, não há dificuldades para realização de escuta, haja vista que as chamadas correm para um ponto central de comutação da operadora já sem o envelope criptográfico. Em função das regulamentações vários fabricantes programaram facilidades em suas centrais inteligentes de forma a derivar chamadas de um número telefônico para um ponto de monitoração.

Entretanto, os sistemas de Voz sobre IP oferecem novos desafios para viabilizar a execução de interceptação. Três pontos importantes podem ser destacados para ilustrar estes desafios: o primeiro devido ao fato dos sistemas de Voz sobre IP incorporarem nas suas especificações uma série de controles de forma a contornar possíveis ameaças aos requisitos de privacidade, integridade e autenticidade das comunicações de voz e vídeo. Se por um lado a telefonia tradicional não oferece mecanismos padronizados de sigilo que sejam amplamente utilizados, por outro, estas facilidades estão padronizadas, disponíveis e tornando-se nativas nas infra-estruturas de Voz sobre IP, gerando um desafio para manutenção da aplicabilidade do método investigativo legal. O segundo ponto ocorre pelas características intrínsecas dos protocolos Voz sobre IP, que fazem com que o caminho percorrido pela sinalização seja diferente daquele percorrido pela mídia digitalizada e transmitida na rede de pacotes. Além disto, os equipamentos comutadores de pacotes, por observarem apenas o endereçamento de destino da informação, não guardam o relacionamento entre o tráfego de sinalização e o respectivo tráfego da mídia. Por último, entra em cena o fator da localização física. Nas redes tradicionais o terminal do assinante é estático e reside nas proximidades da central telefônica ao qual o assinante pertence. Nas redes de Voz sobre IP o assinante pode estar em qualquer ponto da Internet e utilizar serviços de operadoras em outra parte do planeta. Passou a ser possível residir em um país e possuir um número telefônico público de outro. Da mesma forma, é possível ao assinante “carregar sua linha” para onde ele vá.

Então, considerar que a mesma estratégia traçada para viabilizar escutas em redes tradicionais possa ser aplicada aos sistemas de Voz sobre IP é desconhecer as complexidades do emergente sistema e simplificar excessivamente o problema. Desta forma, é necessária a correta compreensão do problema para uma eventual regulamentação dessa matéria. A análise das possibilidades técnicas, que permitam manter esta ferramenta investigativa num nível considerado suficiente, quando comparada à complexidade na implantação dos controles

necessários, é um fator importante na tomada de decisão pelo poder Público sobre a matéria, na medida em que o problema seja elucidado.

Neste sentido, esta dissertação apresenta as principais dificuldades envolvidas na adoção de mecanismos de escuta legal quando confrontadas com o emprego de Voz sobre IP, especificamente para o protocolo SIP, colocando o tema em discussão.

1.1 Objetivo

Grande parte dos métodos criptográficos empregados pelos protocolos de sinalização, como o H.323 e o SIP, utiliza técnicas de estabelecimento de chaves criptográficas de sessão que são descartadas ao término das conversações, inviabilizando qualquer recuperação sobre o material criptográfico coletado.

Sobre este contexto, a presente dissertação busca apresentar os desafios que a tecnologia de voz sobre IP apresenta à aplicação de técnicas de interceptação e interpretação, em especial quando do uso de criptografia. A dissertação inicia, no Capítulo 2, com a apreciação das principais iniciativas propostas e formalizadas para fins de interceptação e interpretação. Nos capítulos seguintes é apresentada uma das principais tecnologias utilizadas para distribuir sistemas de Voz sobre IP: o protocolo SIP, explorando suas características técnicas e os controles de segurança por ela adotada. Por fim, a dissertação irá discutir a viabilidade em realizar a interceptação e interpretação com base nesta tecnologia, incluindo o uso dos controles atuais. É então proposto um método que possibilita a interpretação da mídia criptografada, por intermédio de um sistema de distribuição de chaves centralizado, de tal forma a preservar e resguardar as chaves de sessão utilizadas na proteção das conversações telefônicas. Este método permite separar o ato da interceptação do conteúdo com o ato da interpretação, em momentos diferentes. No final são analisadas as vantagens e desvantagens na aplicação do sistema proposto, comparando-o com as possibilidades existentes, descrevendo vantagens e desvantagens.

As condições de contorno aplicadas às análises e as propostas efetuadas estão descritas detalhadamente no Capítulo 7.

1.2 Introdução aos sistemas telefônicos

A telefonia possui três fases na sua evolução tecnológica. A analógica, a digital e uma terceira, que compreende o emprego das redes de pacotes para transporte do serviço de voz.

1.2.1 Telefonia Analógica

A prestação de serviços públicos telefônicos iniciou [3] em 1876[1]. Na ocasião, o envio em meio elétrico da voz ocorria com o emprego de linhas de transmissão utilizando fios metálicos para interligar dois pontos terminados com transdutores eletroacústicos – os precursores dos atuais aparelhos telefônicos. Eram empregadas linhas em número igual ao interesse de comunicação de um assinante, implicando numa rede de distribuição física de cabos tendendo ao valor de `n(n-1)/2` cabos, sendo `n` é o número de assinantes, caso todos os assinantes desejassem acesso aos demais, criando uma rede totalmente conectada. Em 1878[2] [3], um aprimoramento desta abordagem, pela introdução das centrais de comutação manuais, permitiu a distribuição de redes com topologia em estrela, onde cada assinante era interligado à central mais próxima, apenas por um cabo metálico. O conjunto formado pelo aparelho telefônico e a linha física, entre a central e o assinante, ficou conhecido por enlace local ou linha de assinante.

Num primeiro momento, as centrais eram operadas manualmente em mesas comutadoras. Tais mesas eram integradas às demais por intermédio de meios de transmissão capazes de enviar múltiplos canais de voz[3] analógicos, simultaneamente, possibilitando a comunicação à distância, integrando regiões, cidades e países. O funcionamento era simples: um assinante alertava ao operador seu desejo em realizar uma chamada pelo envio de uma campainha, então o operador conectava seu fone de ouvido à linha do assinante para ser informado verbalmente sobre o destino de interesse[4]. Em seqüência, o operador alertava o destino, que ao atender, era interconectado ao chamador, dando início à conversação. Ligações entre regiões diferentes percorriam este algoritmo recursivamente, até que um caminho fosse criado entre a origem e o destino. O conjunto formado pela sinalização sucedida da criação de um caminho de comunicação, exclusivo entre a origem e o destino, é conceituado como “comutação de circuitos”, no qual todas as redes atuais de telefonia são baseadas.

---

[1] Segundo [3], a primeira linha instalada comercialmente ocorreu em Charlestown, Massachusetts, dando início a *Bell Telephone Company*.
[2] Segundo [3], a primeira central a se ter notícia foi instalada em New Haven, Connecticut, com vinte e nove assinantes.
[3] Foi estabelecido que um canal de voz analógico ocupasse uma faixa de freqüências nominal de 4.000 hertz, para ainda permitir a inteligibilidade e a capacidade de reconhecer o interlocutor. Isto representa um quinto do total das freqüências percebidas pelo ser humano.
[4] Os usuários eram identificados pelos próprios nomes e cidades onde moravam.

Uma chamada terminada pelo assinante era percebida e desfeita pelo operador, por um indicador visual de presença de corrente elétrica de enlace, uma vez que os telefones eram energizados pelos equipamentos centrais e consumiam energia apenas quando fora do gancho.

A evolução tecnológica conduziu a substituição das mesas de comutação manual por centrais eletromecânicas. Os aparelhos passaram a enviar pulsos em quantidades idênticas ao número do assinante de destino, permitindo às centrais interpretar os números e efetuar o direcionamento das chamadas de forma automática[5,6]. Os números discados pelo assinante eram armazenados em registradores nas centrais locais e enviados as demais centrais caso a chamada telefônica fosse destinada a outras regiões. De certa forma, o feito da automatização foi empregar o algoritmo executado manualmente, através de lógicas com dispositivos relês. Ao longo do tempo, a sinalização enviada ao assinante foi redesenhada para incluir envios de tons, como o tom de linha, tom de chamada, ocupado, congestionamento, desligamento e outros. Tal sinalização ficou conhecida como "sinalização de linha de assinante" e perdura até o presente, tendo sido parcialmente incorporada às tecnologias de Voz sobre IP.

1.2.2 Telefonia Digital

Com o desenvolvimento das técnicas de comunicação digital e da microeletrônica, por volta de 1970, a infra-estrutura de telefonia começou a ser paulatinamente substituída para comportar as novas centrais digitais. A voz passou a ser digitalizada[7] nas portas de acesso dos assinantes, na entrada das centrais telefônicas, para a grande maioria que permaneceu com terminais analógicos; ou digitalizadas no próprio terminal do assinante quando este possuísse um terminal digital. Com a voz digitalizada, técnicas temporais com base no TDM[8] serviram tanto para comutação entre linhas da mesma central quanto para o acesso aos canais de transmissão que integravam múltiplas centrais. Na pior condição, a parte analógica da

[5] Há uma história curiosa sobre a motivação de Almon B. Strowger por traz da sua invenção: a primeira central automática, patenteada em 1891. Ele suspeitava que os operadores da mesa telefônica da cidade onde morava, davam preferência para encaminhar as chamadas para seu concorrente, fazendo-o perder negócios no seu ramo de atuação: funerária.

[6] A primeira central automática foi instalada em 1892 em La Porte, Indiana, usando a patente da central de Strowger.

[7] Para digitalizar um canal de voz utiliza-se uma técnica conhecida por PCM, onde cada canal de voz é amostrado 8.000 vezes por segundo por um conversor analógico-digital, com cada amostra possuindo oito bits, totalizando uma taxa de 64Kbps. São necessárias oito mil amostras em um canal de voz para poder reconstituir a banda de um canal analógico, no destino; e oito bits por amostra para manter a qualidade num nível aceitável. Um processo similar é adotado no *Compact Disc* de áudio, que por precisar de mais qualidade, utiliza dezesseis bits ao invés de oito, amostrando a informação a uma taxa de 44.100/s, para cada canal estereofônico, de forma a comportar todas as freqüências audíveis.

[8] TDM é uma forma de compartilhar o acesso a um canal de transmissão digital, por múltiplos canais de entrada, onde é permitido a cada entrada ocupar uma fatia de tempo fixa e periódica do meio de transmissão. O Brasil adota um padrão que permite que um canal de transmissão, na sua taxa mais básica conhecida por E1, seja compartilhado por trinta canais de acesso. Se duas centrais telefônicas estão integradas por um canal E1, significa que em um dado instante de tempo poderá haver até trinta conversações simultâneas entre elas. Além deste valor, as centrais devem informar congestionamento.

conversação passou a existir apenas na seção referente às linhas dos assinantes, trafegando digitalmente no núcleo da rede, não importando quão distante fosse um assinante do outro.

A mudança para tecnologia digital não envolveu somente a forma como a voz passou a ser codificada e processada. Envolveu também o desenvolvimento de novos protocolos de sinalização entre assinantes e as centrais, e estas entre si. Também houve acréscimos nas funcionalidades disponíveis para os assinantes, agregando valor ao serviço prestado. Mesmo o assinante que permaneceu com um terminal analógico foi beneficiado com o surgimento da tecnologia digital.

Referente aos protocolos surgiu uma gama de padrões públicos. Dentre eles destacam-se o ISDN e a sinalização número 7, ou SS#7 [4]. O ISDN é empregado no acesso entre um assinante e a central, havendo dois padrões de acordo com o porte do assinante: um para acesso de dois canais digitais em uma única linha física e outro para uso de até trinta canais dentro de um mesmo meio físico[9]; ambos acompanhados de um canal adicional de sinalização. Um aspecto importante na mudança para a telefonia digital foi o uso da sinalização em canal associado e não mais dentro da banda utilizada para comunicação de voz, como feito no ambiente analógico.

A ITU-T padronizou o protocolo SS#7 [4] para uso entre centrais telefônicas públicas, permitindo que vários fabricantes construíssem equipamentos compatíveis entre si. A SS#7 acrescentou um novo patamar na telefonia digital, pela adoção de redes de sinalização totalmente separadas das redes de transmissão de áudio[10], aumentando a eficiência no processamento, encaminhamento das chamadas e ampliando o número de serviços suplementares possíveis nos sistemas telefônicos. As redes de sinalização eram compostas de equipamentos distintos, numa arquitetura próxima das redes de pacotes iniciais, como a rede *Frame-Relay*, onde os nós de comutação realizavam troca de sinalização entre os pontos de interesse de uma chamada, podendo também realizar translações de números com base nos prefixos numéricos dos serviços especiais.

A Figura 1-1 mostra um exemplo de topologia para uma rede SS#7. As linhas tracejadas representam os caminhos possíveis para sinalização, as linhas em negrito mostram o caminho exclusivo percorrido pelo áudio. No acesso aos assinantes a sinalização atravessa o mesmo meio físico do conteúdo.

---

[9] O primeiro caso é tecnicamente conhecido como acesso BRI, com capacidade de 128Kbps, contendo dois canais de voz e um de sinalização (2B+D). O segundo como PRI, com capacidade de 2Mbps, contendo trinta canais de voz e um de sinalização (30B+D). Os canais específicos para tráfego de voz/dados são conhecidos como B *"bearer channels"*.

[10] Esse tipo de separação entre sinalização e conteúdo ficou conhecido como sinalização de canal comum, ou CCS.

A SS#7 tornou-se um sucesso mundial, e várias entidades de padronização nacionais reescreveram o padrão estabelecido pela ITU-T, adaptando-o às necessidades locais. Atualmente a SS#7 forma a base das redes telefônicas públicas, tanto fixa quanto celular.

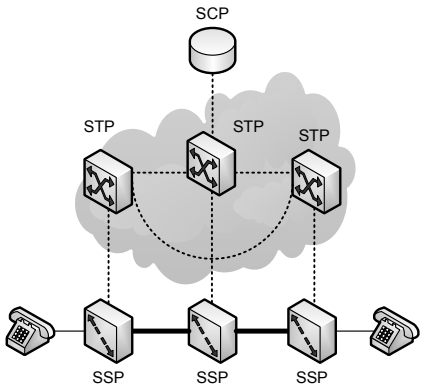

Figura 1-1 - Exemplo de topologia de uma rede SS7

Na arquitetura SS#7 há basicamente três equipamentos no processo de sinalização: o *Service Switching Point* (SSP), o *Signal Transfer Point* (STP) e o *Service Control Point* (SCP). Os SSP's são as centrais telefônicas compatíveis com SS#7 e capazes de separar a sinalização oriunda dos terminais dos assinantes, como exemplo, a "sinalização de linha" ou a ISDN, para ingressar na rede de sinalização, composta por uma série de nós STP's. Por sua vez, estes nós são conhecedores da topologia da rede e capazes de direcionar as chamadas para os SSP's de interesse[11], ou, de acordo com o prefixo numérico, para as bases de dados responsáveis pela translação numérica dos serviços inteligentes. Tais serviços são fornecidos por elementos SCP's, como: os prefixos 0800, 0300, cartões de chamada, consulta a crédito de assinante, registro e localização de assinantes, registro e localização de visitantes. Sendo que os dois últimos cumprem um papel fundamental em um problema básico da telefonia celular: localizar o assinante.

O advento da telefonia digital acrescentou funcionalidades programáveis pelos assinantes e aplicados nas suas respectivas centrais, como os serviços de "siga-me", bloqueio do identificador de chamada e chamada abreviada. Tais funcionalidades ficaram conhecidas como "serviços suplementares" e no Brasil com "facilidades da central" ou apenas "facilidades". Em vários países as facilidades citadas estão disponíveis no pacote básico do serviço telefônico prestado pela operadora de telecomunicações, sem custo adicional, como parte de sua estratégia de mercado.

[11] Podendo ser o SSP de destino da chamada e, eventualmente, outros que estejam no caminho dos canais de áudio, para operar suas matrizes de comutação.

O “siga-me”, quando programado, impede que as chamadas passem pela linha do assinante que está fazendo uso da facilidade. Correios de voz provocam um efeito parecido com o “siga-me”. Chamadas abreviadas impossibilitam a identificação do número de destino apenas observando a sinalização do assinante. O bloqueio do identificador de chamada idem. E assim por diante, de forma que o uso extensivo das facilidades acarretou em dificuldades no processo de escuta legal, ainda durante a transição da era analógica para a digital, motivando a criação da regulamentação CALEA, como será visto no próximo capítulo.

1.2.3 Telefonia empregando redes de Pacotes

A principal característica das redes de pacotes[12] é o seu comportamento estatístico no processamento do tráfego gerado pelas aplicações. Ao contrário das redes de telefonia, que precisam estabelecer um circuito entre a origem e o destino, criado através de sinalização específica, para dar início à conversação, as redes de pacotes trabalham num esquema de contenção dos meios de transmissão, sem sinalização prévia. Os pacotes enviados pelos usuários são colocados em filas dentro dos equipamentos comutadores, até o que o meio de transmissão esteja livre para atender ao próximo pacote enfileirado.

A desvantagem das redes por comutação de circuitos[13] é que a capacidade total dos canais de transmissão é dividida de forma igualitária pelo número de usuários previstos a ocupar o recurso, e assim permanece, mesmo que não sejam utilizados, gerando certo desperdício. As redes de pacotes, por outro lado, são capazes de conceder toda a capacidade de transmissão de um canal a um único usuário, se os demais não estiverem utilizando o recurso. Houve várias implementações de redes de pacotes. Dentre as tecnologias que obtiverem grande emprego comercial estão as redes *Frame-Relay*, ATM e as redes TCP/IP.

O movimento para emprego de voz em redes de pacotes ganhou força por duas necessidades não totalmente distintas: a primeira com foco na economia dos dispendiosos recursos de transmissão de longa distância, como visto anteriormente; a segunda quando operadoras de telefonia, operadoras de TV a cabo, corporações e os próprios usuários vislumbraram no acesso banda larga a possibilidade de envio não somente de dados, mas de serviços multimídia como Voz sobre IP. Em especial isso ocorreu após o surgimento dos

[12] São redes que executam a transmissão das mensagens subdividindo-as em blocos menores, conhecidos como pacotes, de modo que em um circuito de comunicação a transmissão dos pacotes, de diversas fontes distintas, possa ser intercalada.
[13] São redes que devem estabelecer um caminho, ou circuito, entre as partes de uma comunicação, previamente ao início da transmissão da informação. O circuito permanecerá dedicado à comunicação até que ocorra uma sinalização para desfazer o trajeto.

acessos de alta velocidade empregando tecnologias como o ADSL, *Cable Modem*, WiMAX e Fiber-to-the-Home (FTTH).

As centrais telefônicas digitais, que empregavam técnicas de acesso TDM nos recursos de transmissão, começaram a ser integradas através de *gateways* de acesso às redes TCP/IP, reduzindo drasticamente o custo na transmissão das informações de voz. Esse movimento tomou força com o surgimento de protocolos de controle de *gateways* distribuídos com o MGCP [5], permitindo que diversos fabricantes fornecessem soluções compatíveis entre si. O MGCP basicamente é constituído por dois elementos: o *media gateway controller* (MGC) e o *media gateway* (MG). O MGC é responsável por processar a sinalização convencional, em geral SS#7, oriunda dos pontos de troca de sinalização STP a ele integrados, para então decidir quais os melhores *MG's* para encaminhar os circuitos de voz por dentro da rede IP, reduzindo o custo da chamada de longa distância e número de centrais envolvidas no processo de comutação. O MG é o elemento responsável pela conversão da voz do formato tradicional TDM para uma das codificações possíveis e transportáveis na rede TCP/IP, conectado entre as centrais telefônicas e a rede. Equipamentos MGC modernos conseguem identificar[14] se o destinatário da chamada está localizado em um terminal puramente IP conectado à Internet, evitando o retorno da chamada para o mundo TDM. Notoriamente esta é uma tecnologia de transição, enquanto houver assinantes em redes de telefonia convencional.

De acordo com uma definição subjetiva empregada em diversas fontes consultadas, a técnica de integração do parágrafo anterior é mais conhecida pelo termo "Telefonia IP", ocupando uma posição mais abrangente que "Voz sobre IP", ou VoIP, empregado apenas aos casos onde o foco não é a integração de redes telefônicas e sim atender terminais puramente IP.

Para o aprovisionamento de assinantes puramente IP, ou assinantes de Voz sobre IP (VoIP), existem protocolos específicos para esta função, dos quais destacam-se o *Session Initiation Protocol* – SIP, especificado pela IETF, e o H.323 – especificado pela ITU-T. Ambos são similares na arquitetura e funcionamento básico, porém incompatíveis entre si na representação e processamento da sinalização. Os dois protocolos utilizam servidores para registro dos assinantes, permitindo mapear seu identificador lógico com o endereço IP momentaneamente utilizado. Os endereços IP, em muitos casos, são alocados aos terminais de forma dinâmica e temporária, e os protocolos de VoIP devem ser capazes de suplantar esta flutuação. O formato criado para os identificadores lógicos, pelo SIP e o H.323, pode

[14] Pela cooperação com protocolos como o SIP e o H.323.

acomodar nomes similares aos e-mails e números telefônicos para facilitar a integração com as redes PSTN.

No processamento da sinalização são empregados servidores com fins de iniciar, alterar ou terminar chamadas. No mundo H.323 estes servidores são conhecidos como *gatekeeper* e no mundo SIP a mesma função é realizada por servidores conhecidos por *Proxy, Location e Registar*. Numa mesma chamada podem ser empregados diversos servidores de sinalização em série, dependendo do número de provedores de serviço envolvidos ou das políticas empregadas para controle do fluxo de sinalização. Para uma chamada ser estabelecida há uma etapa inicial de sinalização, onde são realizadas atividades como: identificar o terminal remoto, negociar as características da mídia, as características do protocolo que irá transportar o áudio, mecanismos de segurança e outros.

Concluído o estabelecimento da chamada, a mídia é devidamente digitalizada, codificada e empacotada, na medida em que a conversação evolui, sendo trocada diretamente entre os terminais[15], utilizando um protocolo para transporte de tempo real conhecido com RTP, enviando-se pacotes em ambas as direções, numa taxa definida de acordo com codificador empregado. Um valor típico para a taxa de geração de pacotes para transportar o áudio de uma conversação telefônica sobre IP gira em torno de cinqüenta pacotes por segundo.

Por empregarem o mesmo conjunto de codificadores e utilizarem o padrão de transporte na rede, o SIP e o H.323 são compatíveis entre si no que concerne ao envio e recebimento de sinal de áudio e vídeo, mas não dispensam o emprego de conversores de sinalização entre os dois ambientes.

Um aspecto importante no emprego de VoIP é que o caminho percorrido pela sinalização, SIP ou H.323, não é necessariamente o mesmo caminho percorrido pela mídia. Num serviço público de Voz sobre IP, nem sempre o provedor do serviço VoIP será o mesmo que presta o serviço de acesso à Internet. Na verdade, a situação atual mostra o contrário: quase todos os provedores do serviço VoIP são diferentes do prestador do acesso, acentuando a questão do caminho. Um chamada em VoIP pode envolver múltiplos agentes, vários provedores de acesso e prestadores de serviço VoIP, como mostrado na Figura 1-2.

Um aspecto não menos importante é que os equipamentos roteadores tradicionais, tal qual a maioria dos empregados na Internet, mesmo que processando dados de sinalização e mídia para uma mesma chamada, por um mesmo caminho, não têm noção do vínculo entre esses

[15] A mídia pode também atravessar dispositivos intermediários como *Proxies* de mídia, transcodificação e elementos de controle de borda da rede.

dados. Tais equipamentos não interpretam informações além da camada de rede, inviabilizando a investigação e acompanhamento de tais correlações. A Figura 1-2 exemplifica a diferenciação de caminhos que pode ocorrer entre a mídia e a sinalização. Enquanto a sinalização atravessa os enlaces existentes entre os provedores de acesso de “A” e “B”, na direção do Servidor de Sinalização, a mídia percorre o caminho inferior.

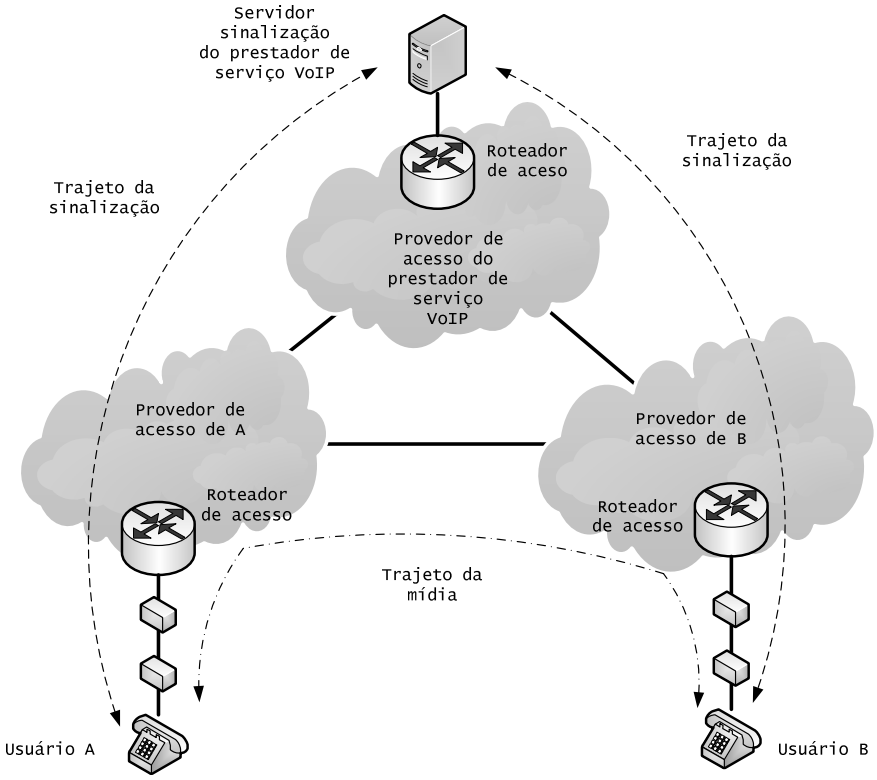

Figura 1-2 - Trajeto da sinalização e mídia

1.3 Organização do Trabalho

Os capítulos e seções restantes deste trabalho estão organizados da seguinte forma:

- O Capítulo 2 traz uma visão do cenário de como a escuta legal está sendo considerada mediante a transição tecnológica para as redes de Voz sobre IP. São apresentadas visões das iniciativas governamentais e de instituições de padronização, incluindo a principal legislação estadunidense, que vem provocando ações de adequação na indústria de tecnologia. Também são descritos alguns trabalhos publicados em periódicos e conferências, que foram pesquisadas durante o desenvolvimento desta dissertação. Alguns artigos contêm apenas ponderações, históricos e visão das técnicas

atuais para aplicação de sistemas de escuta e outros proposições para sobrepujar os desafios que as tecnologias de Voz sobre IP apresentam aos sistemas de escuta legal;

- O Capítulo 3 apresenta os detalhes do funcionamento do *Session Initiation Protocol* - SIP, com ênfase nos mecanismos adotados para iniciar, gerenciar e encerrar o recebimento da mídia de áudio. O capítulo também apresenta o funcionamento do protocolo de transporte de mídia em tempo real conhecido como *Real Time Protocol* (RTP), como fundamentação das discussões posteriores sobre o tema;
- O Capítulo 4 trata dos aspectos de segurança em Voz sobre IP, especificamente para o protocolo SIP, apontando os principais controles de segurança para manter a autenticidade, confidencialidade e integridade das comunicações. A maior parte dos controles é efetivada através da utilização de protocolos anteriormente padronizados para outras finalidades, que o SIP aproveita em seu benefício. Há um destaque para o funcionamento do protocolo de tempo real seguro, ou SRTP.
- O Capítulo 5 apresenta os protocolos de gerenciamento de chaves criptográficas aplicadas ao protocolo de transporte seguro da mídia, que complementam o funcionamento dos controles de segurança visto no capítulo anterior. O entendimento do funcionamento destes protocolos é importante para justificar o motivo do descarte das chaves criptográficas pelos padrões atuais.
- O Capítulo 6 é um estudo detalhado do funcionamento do protocolo de autenticação Kerberos, que vai ser usado para formar a base da proposição do capítulo seguinte.
- O Capítulo 7 estabelece proposições e estratégias de interceptação e interpretação das mídias de áudio para o protocolo SIP e uma forma alternativa para compor este protocolo de forma a tornar possível a preservação das chaves de sessão para posterior uso legal, sem comprometer o nível de segurança das especificações atuais.
- O Capítulo 8 contém as conclusões do trabalho e proposições para estudo futuro.

Capítulo 2

Cenário atual sobre escuta legal em VoIP

Este capítulo apresenta um retrato do cenário atual sobre como a escuta legal está sendo tratada mediante a transição tecnológica para o uso das redes de Voz sobre IP. Apresentando uma visão das iniciativas governamentais, dos órgãos de padronização e da indústria, em especial a legislação estadunidense CALEA [5], que tem provocado ações de adequação na indústria. Também apresenta algumas regulamentações, históricos e técnicas atuais para aplicação de sistemas de escuta.

É importante destacar um termo utilizado na literatura e três definições presentes neste texto. Para esta dissertação, escuta legal é um processo envolvendo a interceptação e a interpretação de uma comunicação telefônica de voz. A interceptação é o ato de aprovisionar um caminho físico ou lógico que viabilize e efetive o ato de gravação. A interpretação é o ato de decodificar a mídia de voz de forma a torná-la inteligível, principalmente se esta estiver protegida por algum mecanismo criptográfico. Após torná-la inteligível entra em cena o processo de interpretação semântica, executado pela perícia do conteúdo, que não tem relevância para este trabalho. Parte-se da premissa que todas as solicitações de escuta legal são autênticas e respaldadas em mecanismos legais, sem que seja detalhado como.

Na literatura consultada o termo interceptação legal[1] é empregado para qualquer tipo de conteúdo e não somente voz. A interpretação é dependente do estudo ou proposta analisada e eventualmente está inclusa no processo de *lawful interception* (LI) executado por um provedor ou pelo agente solicitante[2], após o fornecimento das chaves pela prestadora se esta tiver acesso ao material criptográfico.

[1] O termo em Inglês é *lawful interception*, ou LI. O termo *wiretapping* se refere à técnica utilizada, podendo ser de emprego legal ou não.
[2] O agente solicitante é conhecido na literatura como *Law Enforcement Agency*, ou LEA.

2.1 Privacidade, criptografia e o projeto *Clipper*

Privacidade é um tema que permeia as relações sociais há milênios. Escritos antigos revelam regras comunitárias criadas para que indivíduos não construíssem muros de altura inferior às janelas dos vizinhos nem às próprias. Mas foi a necessidade de comunicação à distância que fez com que estados, organizações, forças militares e correspondentes criassem mecanismos objetivando atestar a inviolabilidade da comunicação privada. Selos, portadores de confiança e métodos de criptografia surgiram como resposta às necessidades de privacidade.

Porém, o grande impulso ocorreu com o advento das comunicações por meio eletrônico, cujos precursores foram o telégrafo e o telefone, e mais recentemente a Internet. O contato entre pessoas e entidades começou a migrar do físico para o virtual, distante. Trocas de mensagens que demoravam semanas passaram para frações de segundo. De repente, conversas "ombro-a-ombro", carimbos, selos e *couriers* não mais serviam. Por outro lado, havia a percepção que muitas das tecnologias empregadas na transmissão da informação, em meio eletrônico, permitiam a fácil interceptação da comunicação privada, principalmente quando estas utilizavam meios de propagação como sistemas de rádio, satélites, linhas de transmissão em postes e todos os aparatos possíveis além do controle físico das partes.

Foi a criptografia a forma viável, tanto econômica quanto tecnicamente, de se manter a privacidade quando a comunicação passou a realizar-se em meio eletrônico, percorrendo meios de comunicação além do perímetro sobre os qual os interlocutores têm controle. Naturalmente outros dois fatores são de igual importância: a integridade e a autenticidade. Entretanto, a criptografia ganhou inicialmente as atenções, uma vez que alterar uma mensagem numa comunicação em tempo real é mais complexo que a simples escuta imperceptível da informação. A autenticidade só foi ganhar destaque com a disseminação dos negócios eletrônicos, muito depois do surgimento do e-mail.

A evolução dos sistemas computacionais fez com que diversos padrões criptográficos surgissem e fossem descontinuados. Mecanismos que eram amplamente usados desde 1970, como algoritmo DES[3], previsto para operar até 1998, começaram a mostrar fraquezas perante o crescimento exponencial da capacidade computacional. No ano de 2001 o NIST alterou o algoritmo padrão empregado pelos órgãos governamentais, no caso o DES, para o vencedor

[3] O DES é uma evolução do algoritmo Lúcifer, criado pela IBM em 1974.

de uma chamada pública[4] iniciada em 1997 e finalizada naquele ano. O novo algoritmo AES mostra-se virtualmente invulnerável à capacidade computacional existente[5] ou vindoura, num prazo indefinido, a menos que surja uma tecnologia que altere a histórica taxa de crescimento da capacidade computacional[6].

Prevendo a ampliação na complexidade em realizar escutas através de sistemas de comunicação protegidos por criptografia, o governo estadunidense promoveu o desenvolvimento de um mecanismo de *key escrow*[7], com o intuito de preservar a habilidade das agências de segurança em interceptar as comunicações privadas, alegando motivos de segurança nacional ou aplicação de escuta judicial. Foi criado um padrão que ficou conhecido como *Escrowed Encryption Standard*[8], desenvolvido pelo *U.S. Department of Commerce*, divulgado em abril de 1994. Tal padrão incluiu o desenvolvimento, pela NSA, de um algoritmo secreto conhecido como SKIPJACK, integrado em um circuito integrado (*chip*) inviolável com nome *Clipper*. O chip foi desenvolvido para ser empregado em telefones e equipamentos de dados, com capacidades de criptografia, tendo sido cogitado seu uso inclusive em computadores pessoais. Seu objetivo era preservar as chaves criptográficas utilizadas nas sessões seguras, de forma que apenas o governo estadunidense pudesse recuperá-las, impedindo, inclusive, este direito às empresas de telecomunicações.

O funcionamento consistia basicamente em transmitir a chave de sessão, devidamente criptografada com o SKIPJACK no início de cada transmissão, acompanhada do respectivo identificador do chip *Clipper* que realizou a operação. Pela consulta de uma base central de identificadores dos *chips* era possível decifrar a chave de sessão. Historicamente toda esta iniciativa ficou mais conhecida pelo nome *Clipper*.

Para incentivar a adoção do *Clipper*, o governo ofereceu vantagens às empresas que o empregassem em seus produtos, inclusive retirando seletivamente as restrições na exportação de equipamentos com criptografia.

Imediatamente surgiram oposições. A primeira pelos defensores dos direitos individuais, temendo que o *chip* fosse empregado em toda espécie de dispositivos, dando direito ao Estado efetivar a interceptação de toda e qualquer informação, concentrando muito poder numa

[4] Detalhes sobre a chamada pública podem ser encontrados em http://csrc.nist.gov/publications/nistbul/itl99-08.txt.
[5] Segundo a referencia [11], página 28, são estimados 30.000 anos para que um trilhão de processadores em paralelo varressem todas as combinações de uma chave de 128 bits, com cada processador executando o AES um trilhão de vezes por segundo.
[6] A capacidade computacional tem dobrado a cada dois anos, aproximadamente, seguindo a Lei de Moore.
[7] Key escrow é um mecanismo de custódia de chaves.
[8] O documento sobre o programa, identificado por: Federal Information Processing Standards Publication 185, pode ser visto em http://www.itl.nist.gov/fipspubs/fip185.htm.

forma que não poderia ser acompanhada ou fiscalizada, abrindo espaço para o uso indevido do mecanismo.

As empresas sediadas nos EUA consideraram que o uso do chip poderia torná-las menos competitivas no mercado internacional. O custo estimado para acréscimo do *Clipper* fora considerado alto, por conta de suas características secretas, e seus concorrentes externos não seriam obrigados a empregá-lo. Ainda mais grave, seria difícil convencer empresas e multinacionais de outros países a comprar um equipamento que contivesse um chip que só o governo estadunidense seria capaz de acessar para recuperar chaves criptográficas.

Especialistas consideravam que seria complexo e arriscado manter uma base de dados tão ampla, conforme referência [9]. Um sistema capaz de ter acesso às chaves que dariam acesso a informações de toda natureza seria um alvo natural. Havia também descrédito pelo elo de confiança humano no processo e as implicações que o mau verso poderia causar. A seção 2.2, descreve uma série de motivos pelos quais os sistemas de *key escrow*, ou custódia, devem ser preteridos.

As apelações e objeções foram tantas que o governo não teve outra opção se não abandonar paulatinamente o programa. Na prática, a maior parte dos equipamentos fabricados com o chip foi comprada pelo próprio governo. Após a descontinuidade do *Clipper*, dada a expectativa da chamada pública pelo NIST para adoção do novo algoritmo de criptografia, o SKIPJACK foi tornado público em 1998. Alguns dias depois foram apontadas vulnerabilidades[9] no processo de autenticação do chip para recuperação das chaves armazenadas.

Por volta de setembro do ano de 2000 o governo estadunidense iniciou uma mudança nas suas estratégias sobre criptografia, reduzindo a ênfase no controle sobre a exportação de criptografia para uso comercial, incentivando o emprego de soluções padronizadas e de prateleira. Por outro lado, ampliou-se o controle na exportação para uso governamental e militar.

2.2 Por que sistemas de custódia de chave são problemáticos

Toda iniciativa em regulamentar um mecanismo de custódia de chaves deve levar em conta alguns riscos inerentes, apenas pelo fato de existirem. Tais riscos não têm vínculo com a

[9] Biham, E., Biryukov, A., Shamir, A. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. EUROCRYPT 1999, pp12–23.

técnica empregada para produzir o resultado nem com a finalidade a que se destina. Indiscutivelmente uma infra-estrutura acompanhada de um sistema que venha a permitir a recuperação de chaves é mais insegura quando comparada à própria, desprovida desta função.

Para começar, a simples existência de um ponto central para repositório de chaves o promove automaticamente a um alvo natural. A manutenção de uma base de dados centralizada para arquivamento de chaves pode trazer riscos quanto à violação de parte ou do todo das informações que transitam na rede, caso as chaves utilizadas na liberação sejam comprometidas. Adicionalmente, este tipo de enfoque deposita uma forte confiança nos administradores dos sistemas.

Sistemas deste tipo são complexos para serem implementados e operados. Mesmo pequenas mudanças em sistemas podem provocar grandes vulnerabilidades. A história recente está repleta de casos de sistemas que demoraram anos de planejamento para serem subjugados em poucas semanas ou meses. O caso parece especial quando sistemas são desenvolvidos de forma fechada, sem a crítica da comunidade especializada. Há vários exemplos que mostram isso, como o próprio SKIPJACK, e mais recentemente os adotados para controle de direitos digitais[10].

A operação de um sistema de escuta legal pode levar a necessidade de cooperação entre múltiplas entidades organizacionais, inclusive entre países distintos, na articulação do resgate do material cifrado e das chaves custodiadas. Este é um dos motivos que levou a IETF a não considerar requerimentos para *wiretapping*. A comunicação pela Internet pode ocorrer por diversos domínios administrativos, com regimes jurídicos distintos, sendo impossível de ser resolvido por um mecanismo puramente técnico.

Sistemas de custódia também podem gerar problemas referentes à disponibilidade, desempenho e custo. Falhas nos mecanismos de custódia não podem comprometer o comportamento do serviço, nem gerar degradação na qualidade do serviço prestado. Mas o custo pode ser o fator decisivo. É complexo emplacar uma medida que acarrete em custos adicionais para os usuários e que possam tornar os fabricantes, provedores de acesso ou de serviço menos competitivos no mercado.

Há ainda uma armadilha sutil contida na própria existência de um sistema de custódia: se as chaves armazenadas e resgatadas para acesso aos conteúdos forem as mesmas utilizadas

[10] Dois mecanismos recentes implementados para compor o *Digital Rights Management* foram quebrados. O mecanismo adotado nos DVD´s e o criado para o *Blue Ray* e HD-DVD.

nos processos de assinatura digital pelos usuários, então os usuários podem repudiar[11] a autenticidade do material coletado, uma vez que as chaves não seriam de conhecimento exclusivo dos próprios. Seria um retrocesso nos mecanismos estabelecidos de assinatura digital atualmente existente.

Por fim, há um problema político e social, não discutido neste trabalho, sobre o papel de um sistema de recuperação de chaves numa sociedade livre. Certamente a decisão pela adoção de mecanismos para permitir a escuta legal em sistema de Voz sobre IP trará no bojo uma forte crítica da sociedade.

2.3 CALEA - Communications Assistance for Law Enforcement Act

Afetado com o crescimento da telefonia digital[12] e das redes de telefonia celular, que muitas vezes o impediam de executar as ordens de escuta, o *Federal Bureau of Investigation* (FBI), desde 1980, vinha realizando alertas para as instituições governamentais a fim de obter uma reavaliação da então existente regulamentação de escuta legal. O padrão vigente naquela ocasião era baseado literalmente na derivação da linha do assinante através de um extensor de "enlace local", algo que só funcionava sistematicamente se o assinante possuísse um terminal analógico fixo e sem nenhum serviço suplementar advindo da telefonia digital[13].

Então, em janeiro de 1995 foi promulgada a lei estadunidense CALEA [5] cujo objetivo é regulamentar a execução do ato da escuta, condicionando os fabricantes dos equipamentos de telefonia a modificar seus produtos, possibilitando às operadoras de telecomunicações suportar mecanismos de monitoração, eliminando as dificuldades encontradas pela difusão da telefonia digital, iniciada nos anos de 1960 e pelo posterior uso massivo das facilidades suplementares e serviços inteligentes, das redes ISDN e SS#7.

Havia também um fator "corporativista" que contribuiu para a promulgação da CALEA, como dito na referência [11], baseado na condição da ferramenta de escuta existir há mais de duas gerações de profissionais ligados à segurança pública, igualando o fato de perdê-la à

[11] Não-repúdio evita que alguém negue a imputabilidade por uma mensagem por ele assinada digitalmente.
[12] Pela ocorrência progressiva no uso dos serviços suplementares proporcionados pelas centrais digitais, como o redirecionamento de chamadas, que nem sequer faz com que as comunicações atinjam a linha do assinante. Outras facilidades, como bloqueio do identificador de chamada, chamadas abreviadas e chamada em espera também acrescentam dificuldades. Segundo [18], página 26, o FBI também experimentou um impacto operacional com o fim do monopólio existente no mercado estadunidense pela operadora AT&T.
[13] A seção 1.2 apresenta uma definição dos conceitos de telefonia.

retirada de uma ferramenta básica de trabalho, como retirar o estetoscópio de um clínico geral.

Segundo um estudo apresentado na referência [7], dez anos após a promulgação do CALEA ainda havia dificuldades na execução das ordens de escuta e o serviço de Voz sobre IP, prestado de forma gerenciada[14], aparece como a terceira tecnologia mais "impeditiva" quando comparada dentre oito causas mais comuns.

Em 2004, por ação de uma petição conjunta do FBI, do Departamento de Justiça e da Agência Anti-Drogas, o *Federal Communications Commission* (FCC) colocou em discussão pública um documento intitulado "*In the Matter of Communications Assistance for Law Enforcement Act and Broadband Access and Services*" [8], detalhando os tipos de provedores de serviço de telefonia afetados. Este documento culminou no ano de 2005 em uma portaria[15] que acabou por considerar os serviços gerenciados de Voz sobre IP, incluindo os provedores VoIP interconectados e os provedores de acesso banda larga à Internet, sujeitos à mesma regulamentação dos provedores de telecomunicações tradicionais e estabeleceu um prazo para adequação das infra-estruturas ao CALEA, independente da tecnologia empregada e sem o aporte financeiro governamental que houve em 1995.

Várias críticas surgiram com relação ao tratamento do CALEA às redes de Voz sobre IP, em especial a de Bellovin *et al*. [9], com destaque para a simplificação da aplicabilidade de sistemas de escuta legal, tratando as redes VoIP tal qual as redes PSTN, desconsiderando questões quanto aos aspectos de mobilidade, criptografia e características da mídia e sinalização percorrerem caminhos distintos. O cenário pode ser ainda mais complexo quando considera que empresas prestadoras de serviço VoIP podem não possuir representatividade legal no país onde foi emitida a legislação, tornando o ato inócuo e fomentando a fuga de usuários, prestadores de serviço e tecnologias.

2.4 A posição da IETF

Para responder aos pleitos realizados pelos próprios participantes dos grupos de trabalho[16] ligados ao desenvolvimento da telefonia sobre IP, dirigidos ao *Internet Engineering Steering*

[14] O FBI considera um serviço gerenciado de VoIP todo aquele que possui um provedor que atua como mediador para permitir o estabelecimento das chamadas, diferenciando do serviço P2P, conforme referência.
[15] USA. *Federal Register*, vol. 70, no. 197, 13 out. 2005, p. 59664.
[16] Os grupos de trabalho do IETF possuem como participantes profissionais autônomos, de instituições de ensino, pesquisa e empresas.

Group e ao *Internet Architecture Board*, no ano de 2000 a *Internet Engineering Task Force* (IETF) tornou pública sua opinião sobre o tema *wiretapping* através da RFC 2804 [40].

Para a IETF *wiretapping* é um evento que ocorre em três fases: coleta, filtragem e entrega de informação de uma comunicação onde as seguintes condições são satisfeitas: o usuário gerador e os receptores da informação desconhecem a ocorrência do evento; a expectativa do gerador é que a informação esteja sendo recebida e interpretada somente pelos receptores e, por último, que estes a mantém no nível de confidência de interesse entre as partes.

A RFC foi produzida com base numa lista de discussão e posteriores plenárias da IETF, exprimindo a opinião da maioria dos participantes. A opinião final foi que a IETF **não** irá posicionar-se sobre o tema, justificado pelos seguintes argumentos:

a. A sua falta de representatividade, como entidade geradora de padrões técnicos para a Internet, para desenvolver uma recomendação que surge pelo interesse legal dos países. Legislações sobre interceptação variam sobremaneira entre países, e independentes disto, todos adotam as recomendações da IETF, tornando complexa uma recomendação geral;

b. É de conhecimento que os padrões da IETF funcionam através de redes pertencentes a países distintos, operadas e mantidas por pessoas submetidas a diferentes regimes jurídicos e requisitos de privacidade. Assim, para continuidade da aceitação mundial da Internet e para melhor servir aos seus usuários a alternativa mais plausível é atribuir à Internet características de segurança bem entendidas por todos; traduzindo, na opinião da IETF, em manter a Rede livre de “pontos de observação” de segurança;

c. Na ausência do uso de criptografia nos terminais ou outro mecanismo de garantia de privacidade, existem várias formas em se executar a escuta de tráfego ao longo de seu trajeto, como espelhamento de tráfego, e outros, que podem ser aplicados a uma vasta gama de necessidades. A IETF não visualiza uma “solução de engenharia” para se realizar *wiretapping* quando os sistemas finais aplicarem medidas adequadas para proteger as comunicações;

d. A IETF acredita que adicionar requisitos de *wiretapping* significa um aumento na complexidade das aplicações, e que a prática demonstra contribuir para a redução da segurança dos sistemas, mesmo quando o recurso não é utilizado. De fato, Prevelakis e Diomidis [12] descreveram o que é hoje tido como o maior ataque contra um sistema de telefonia, favorecido pelas vulnerabilidades adicionadas pelo sistema de escuta legal implementado pelo fabricante;

e. A IETF já havia se pronunciado[17] a favor do uso de criptografia forte com a alternativa para manutenção da privacidade;

f. Por último, mesmo não tomando posição sobre o assunto, a IETF acredita que, se houver um padrão, este deve ser aberto e sujeito ao escrutínio da comunidade.

2.5 Publicações sobre o tema

A presente seção tem por objetivo relacionar alguns trabalhos disponíveis, pesquisados entre os principais periódicos, documentos da IETF e entidades normativas, sobre interceptação legal.

2.5.1 Publicações em entidades de padronização

2.5.1.1 Recomendações do ETSI

O *European Telecommunications Standards Institute*, através dos documentos TS 101 331, TR 101 943 e TS 101 671 e outros derivados, elaborou uma arquitetura genérica e interfaces padronizadas para suporte e interceptação legal de qualquer espécie de tráfego IP, por empresas prestadoras de serviço de telecomunicações, ou simplesmente, operadoras. Um dos objetivos da padronização é garantir que o processo ocorra de forma controlada e que as solicitações possam ser validadas e efetivadas agilmente. Os aspectos técnicos e detalhes da implementação da arquitetura proposta são deixados à parte, por serem dependentes das legislações nacionais. Um dos pontos chaves na arquitetura proposta é possibilitar a automatização do processo de LI de forma que a LEA possa realizar as configurações necessárias sem mesmo a interferência do operador. Um ponto importante é que o assinante alvo pode utilizar um serviço de acesso físico de um operador, onde deve eventualmente ocorrer a captura do tráfego, e ser usuário de um serviço VoIP de um segundo, por onde é processada a sinalização. Assim, se ambos aplicarem as recomendações é possível uma coordenação da LEA de forma ágil na interpelação entre operadores distintos.

2.5.1.2 RFC 3924

A RFC3924 [56] descreve uma arquitetura proposta pela empresa Cisco Systems que permita a interceptação de tráfego nas redes IP. A RFC é precedida por um aviso do IESG

[17] Apresentado na RFC 1984, não usado como fonte de consulta.

informando que esta não o considera um padrão para Internet e não foi revisado para servir como um padrão de segurança, integração com outros protocolos ou demais condições técnicas. O documento é uma resposta da empresa Cisco Systems às necessidades das empresas operadoras de telecomunicações terem que instrumentar suas redes de forma a habilitá-los a executar ordens legais de interceptação. É voltado para interceptação de qualquer espécie de tráfego IP, definindo uma arquitetura genérica o suficiente para ser aplicada independentemente da legislação local. A arquitetura inclui um conjunto de interfaces padronizadas.

Tal qual a recomendação do ETSI (seção 2.5.1.1), a RFC não detalha como deve ocorrer o processamento da informação capturada. Em especial, sugere que em havendo material cifrado o provedor deve recuperar as chaves, se possível, e entregar o material decifrado ao solicitante.

Um ponto de destaque do documento é a menção sobre a dificuldade em processar a interceptação nos casos onde o serviço suplementar[18] de redirecionamento de chamada[19] esteja programado. Nas redes de Voz sobre IP este serviço pode ser programado, por exemplo, a critério da identidade do chamador, fazendo com que um assinante tenha redirecionamentos condicionais. Este serviço suplementar é programado, em geral, no terminal e o identificador do terminal alvo do redirecionamento é conhecido apenas no momento do estabelecimento da chamada.

2.5.2 Artigos e conferências sobre Escuta legal

B. Karpagavinayangam, R. State e O. Foster [15], identificaram dificuldades na continuidade da escuta legal motivada pelas redes de Voz sobre IP. Os autores detalharam as diferenças básicas com as redes PSTN, sobressaltando os seguintes pontos: os trajetos diversos da sinalização e mídia percorridos na rede durante uma conversação; a possibilidade de um assinante estar em qualquer local da Internet; a multiplicidade de protocolos desenhados por órgãos de padronização distintos e o uso de mecanismos de segurança que podem ser empregados por diversos níveis do protocolo IP.

Os autores propuseram um modelo que possibilita a realização de LI centrada no protocolo SIP, mas ressaltando que a mesma proposta poderia ser aplicada ao protocolo

[18] "Serviço suplementar" o seu sinônimo "facilidade" são apresentados na seção 1.2.
[19] Conhecido como *call forwarding*

H.323. A proposta é uma combinação de arquitetura, de aprovisionamento de configurações de um pacote de gerenciamento em software composto por um analisador de protocolos, um módulo para troca de informações do SDP (seção 3.8, pág. 43) e um módulo mediador para desvio de tráfego nos *gateways* (seção 3.1, pág. 30).

Na parte de arquitetura os autores propuseram uma combinação de elementos a serem implementados nos provedores de serviço VoIP para coleta, armazenamento e entrega de chaves. Por fim, definiram um ponto de acesso administrativo para aplicação de configurações nos elementos de rede. Na parte de aprovisionamento é proposto o uso dos protocolos NETCONF[20] e RBAC[21]. Na parte de gerenciamento foi apresentado o mecanismo para trabalhar em cooperação com os servidores *Proxy* SIP de forma a alterar o SDP, desviando o tráfego para um ponto de encontro mais próximo do trajeto do tráfego RTP. Os autores consideraram que o provedor de acesso à Internet pode ser diferente do provedor de serviço VoIP, não implicando no funcionamento da proposição, contanto que todos apliquem a mesma arquitetura. Nada foi apresentado para possibilitar o resgate das chaves de criptografia, deixando a atividade a cargo do provedor responsável pelo serviço VoIP.

N. Thanthry, R. Pendse e K. Namuduri [13], deram destaque à eventual dificuldade de se realizar LI por conta do uso de criptografia na rede, e a dualidade entre o benefício da criptografia, prevenindo ações de desconhecidos, e a dificuldade que ela gera para acesso legítimo a este mesmo conteúdo. O artigo basicamente conteve-se em listar as dificuldades criadas pelos mecanismos de segurança para a viabilidade em realizar LI. Em seguida os autores citam alguns casos nos quais agências de segurança tiveram insucesso no processo de criptanálise e fizeram comentários sobre a evolução dos algoritmos de criptografia e o tempo computacional necessário para resgatar as chaves, justificando a dificuldade que este fator acrescenta. Na conclusão os autores comentaram que uma alternativa seria incluir um mecanismo de chave dinâmica entre assinante e *gateway*, de forma semelhante ao mecanismo utilizado pelos sistemas de telefonia celular GSM. As redes GSM criptografam o sinal para envio no enlace de rádio entre o aparelho portátil e a central e o decifram na estação rádio-base, sendo encaminhado no meio de transmissão fixa e de comutação livre do envelope criptográfico.

N. Thanthry, C. Goodrich e R. Pendse [14], apresentaram um sistema compatível com as determinações CALEA para redes de Voz sobre IP. Da mesma forma que N. Thanthry, *et al.*

[20] Para configuração de redes com XML, em substituição ao SNMP, definido na RFC 4741.
[21] *Role Based Access Control* é um mecanismo para implantação de regras de filtragem de forma segura, proposto pelo NIST, conforme http://csrc.nist.gov/groups/SNS/rbac/.

[13], os autores exploram o custo computacional na criptanálise pela exploração[22] das combinações possíveis para um determinado tamanho de chave, algoritmo de criptografia e capacidade computacional, com base num estudo feito por um laboratório de Departamento de Justiça estadunidense de 1999. A conclusão dos autores é que é inviável depender da capacidade computacional disponível, devendo existir algum mecanismo para preservar as chaves criptográficas utilizadas[23]. Em seguida os autores realizam a proposição de um mecanismo que implementa a preservação de chaves criptográficas entre os assinantes da rede, forçando-os a repassar as chaves acordadas para um ponto central do provedor, sob pena da não continuidade da chamada. O mecanismo é baseado no protocolo IKE [62] utilizado pelo IPSEC [63] para estabelecer uma associação segura. Os autores julgam como fator positivo da proposição a possibilidade de tornar disponível imediatamente o material criptográfico às agências; e como fatores negativos tanto a sobrecarga adicional de pacotes quanto o fato de a proposição ter que ser aplicada como um todo em vários provedores, sob pena de não funcionar.

A. Milanovic, S. Srbljic e I. Raznjevic, *et al*. [16] propuseram uma arquitetura distribuída para executar LI em redes de Voz sobre IP. A motivação para o desenvolvimento é feita sobre os problemas de múltiplos caminhos para sinalização e mídia e o efeito da mobilidade. O trabalho é dividido entre a apresentação de uma arquitetura e a definição de um protocolo para comunicação dos elementos distribuídos[24]. O modelo de arquitetura proposto pode ser aplicado a qualquer protocolo de sinalização, composto por três níveis hierárquicos: sendo um componente *Top Level Device* (TLD), alguns *Intermediate-Level Devices* (ILD) e diversos *Bottom-Level Devices* (BLD), acompanhados de *Storage Devices* (SD). TLD é o elemento por onde são feitas as solicitações de LI, coordenando os diversos ILD, que por sua vez interceptam as sinalizações para escolha do melhor ponto de coleta do tráfego, feito pelos BLD's e armazenados nos SD's. A arquitetura proposta é flexível de forma a acomodar o fato de uma chamada poder atravessar diversos operadores de telecomunicações. O protocolo proposto viabiliza a troca de conteúdos XML, com finalidade de enviar solicitações e respostas para coordenação da configuração entre as entidades da arquitetura, parecido com o NETCONF, transportado de forma segura e autenticada via SSL ou IPSEC. Uma vez

[22] Existe um fator de mérito para quantificar o número de operações necessárias para descobrir uma determinada chave tentando-se explorar todas as combinações possíveis, cujo nome é *workfactor*. Algoritmos considerados robustos possuem *workfactor* na ordem de grandeza de 2 elevado ao tamanho da chave.
[23] Na ocasião estava aberta uma chamada pública pelo NIST para substituição do algoritmo oficial DES, cujo vencedor foi definido em 2001, tendo sido adotado o AES.
[24] Definido pelo autor como Wiretap Information Exchange Protocol (WIEP).

capturado o tráfego, os autores não mencionam os problemas possíveis acarretados pela criptografia.

A. Milanovic, S. Srbljic e I. Raznjevic, *et al*. [17], apresentaram quatro propostas de interceptação baseado no protocolo H.323. Um ponto destacado pelos autores foi que, diferentemente das redes PSTN, um usuário de uma rede IP com conhecimento sobre o funcionamento das redes poderia detectar a ocorrência de interceptação, pelo trajeto de tráfego de mídia estabelecido no momento. A primeira proposta foi adotar pontos de coleta de tráfego interiorizados nos *gateways*[25] H.323 da rede. A segunda foi pela alteração da sinalização para o desvio da mídia de forma a passar por um *ponto de encontro* colocado dentro de uma zona administrativa. Esta proposta destaca-se por não necessitar alterações na rede exceto nos *gatekeepers*. Também se destaca por evitar que usuários que conheçam apenas o contato um do outro sejam capazes de deduzir se o endereço IP do ponto de conexão foi alterado, impossibilitando identificar o desvio das mídias para o *ponto de encontro*[26]. Um ponto negativo neste método é o efeito do retardo adicional ocasionado pela mudança do roteamento da chamada. A terceira proposta compreendeu a segunda, mas força que todas as mídias passem sempre roteadas por um ou mais pontos comuns, fazendo com que esta condição fosse pública e constante, portanto nada de diferente poderia ser detectado pelos assinantes mais "avançados". A desvantagem natural desta abordagem seria a carga nos pontos de roteamento. A quarta proposta foi a aplicação de dispositivos de captura trabalhando em modo promíscuo nos equipamentos de convergência dos usuários, como *switches*, de forma que o espelhamento fosse feito transparentemente. Uma vez capturado o tráfego, os autores não mencionaram os problemas acarretados pela criptografia.

S. Landau [18] descreveu o impacto das determinações do CALEA e da FCC sobre as infra-estruturas de Internet, uma vez que para segui-las seriam necessárias adaptações profundas nos protocolos, e que tais modificações poderiam reduzir características de segurança em geral. A autora apresentou um histórico sobre a LI na legislação estadunidense e citou alguns casos de utilização devida e indevida do recurso. Na página 30 foram apresentadas as dificuldades na interceptação ocasionadas por uma rede de pacotes como a Internet. O ponto central da exposição da autora passou a ser a incapacidade da arquitetura da Internet atual de suportar mecanismos de interceptação em seu núcleo, algo que na sua visão

[25] De uma forma similar os *gateways* H.323 operam como os *gateways* SIP, vistos na seção 3.1. Também são similares os funcionamentos de um *gatekeeper* H.323 e um *Proxy* SIP.

[26] Os autores não mencionaram que um agente do meio da mídia não necessariamente significa interceptação. Muitas vezes são aplicados dispositivos para realizar translação de endereços (NAT) e controle de segurança de perímetro, que também podem requerer interceptar a mídia pela alteração na sinalização. A RFC 3303 prevê mecanismos no meio do caminho para estas funções.

somente poderia ocorrer através de um "redesenho da arquitetura atual de Internet" [27], e não o questionamento em si da necessidade. Ela validou a exposição da IETF (seção 2.4, pág. 19) quanto ao risco de aumento da complexidade. Por fim, ela concluiu que a implementação de mecanismos de LI na Internet, para atender as exigências do CALEA, traria malefícios em grau superior àqueles que supostamente ocorreriam na segurança pública.

S. Bellovin, M. Blaze e E. Brickell, *et al*. [9], apresentaram uma critica ao CALEA do ponto de vista da viabilidade técnica em implantá-lo na Internet para os serviços de Voz sobre IP, destacando as diferenças entre o funcionamento das redes PSTN e VoIP, no que diz respeito a mobilidade e dificuldade em interceptar o tráfego de mídia, em especial, para interceptação de chamadas entre dois "*road warriors*" que constantemente alteram suas localizações. Para os autores implementar LI na Internet certamente implicaria numa cooperação entre uma porção muito larga da infra-estrutura de roteamento[28], gerando tanto dificuldade operacional quanto jurisdicional. O uso de mecanismos como *ponto de encontro* pode auxiliar em boa parte dos casos, mas a garantia só viria pela aplicação de um *wiretapping* nos roteadores de acesso próximos aos assinantes envolvidos, de forma que isto requer o uso de um mecanismo de aprovisionamento automático aplicado direto aos equipamentos roteadores. Como os equipamentos roteadores de fato não sabem distinguir o tráfego de interesse dentre quaisquer outros que estejam sendo roteados, qualquer vulnerabilidade no sistema de *wiretapping* poderia ser utilizada para obtenção de informações de qualquer natureza, tornando-se um atrativo forte para interesses escusos.

2.5.3 Artigos, conferências e teses sobre segurança e SIP

K. Ono e S. Tachimoto [19], apresentaram como soluções para o estabelecimento de um caminho seguro usando o protocolo SIP, como uma alternativa ao IKE/IPSEC, usado em aplicações como transferência de arquivos, *chatting* e *instant messaging*. A proposta dos autores consistiu no uso duas alternativas para proteção da sinalização e mídia muito próximas dos padrões atuais do protocolo SIP. A primeira seria através de um mecanismo com o TLS, onde o caminho da sinalização seria confiável. O segundo usaria o S/MIME. Em ambos a criptografia da mídia seria estabelecida por intermédio do SDP para transporte do material criptográfico, sem detalhar como. Em especial, os autores propõem três fatores de mérito para validar um mecanismo de segurança fim a fim: a praticidade na aplicação do

[27] Tradução nossa.
[28] Os autores faziam referência ao fato de uma chamada em VoIP poder ter múltiplos domínios administrativos envolvidos para ser efetivada.

mecanismo de gerenciamento de chaves; o tempo de estabelecimento do canal seguro; e a possibilidade da continuidade operacional de dispositivos que operem no meio do caminho, como agentes de NAT e *firewalls*.

R. Srinivasan, V. Vaidehi e K. Harish, *et al*. [20], apresentaram uma alternativa para autenticação do protocolo SIP, no lugar do atual *digest authentication*, por intermédio de um esquema de geração de chaves temporárias ofertadas pelos servidores de registro e autenticadas através de certificados digitais e carimbos de tempo.

2.6 Mecanismos adicionais de distribuição e custódia de chaves

Além do protocolo Kerberos, que está apresentado em detalhes no Capítulo 6, esta seção apresenta dois outros mecanismos que poderiam ser empregadas num sistema de distribuição e preservação de chaves, de forma a estabelecer comparações, na conclusão deste trabalho, com o método proposto no Capítulo 6.

2.6.1 Mecanismo de chave mestra

O mecanismo de chave mestra é baseado na geração de múltiplas chaves derivadas de uma mesma mestra, tal qual o conceito apresentado por T. Kiesler e L. Harn [21]. As múltiplas chaves são geradas através de uma função, de tal forma que não seja possível deduzir a chave mestra uma vez revelada uma ou mais chaves derivadas. Também, uma mensagem cifrada com uma chave derivada não pode ser decifrada por uma dupla derivação das chaves baseadas na mestra.

Uma chave mestra deve ser previamente conhecida pelas partes de uma comunicação a ser protegida, empregando-se a função de derivação para gerar chaves na taxa que a aplicação exigir, conhecida como taxa de *re-keying*. A métrica utilizada na taxa de *re-keying* varia com a aplicação, podendo ocorrer pelo número de mensagens ou bytes trocados ou tempo de vida da chave derivada.

A forma como as chaves mestras são negociadas previamente ao início da comunicação varia conforme a aplicação, mas em geral empregam-se configurações manuais.

2.6.2 Mecanismo de *Identity-based Encryption* (IBE)

O IBE foi idealizado por A. Shamir [74], em 1984. Na ocasião o autor propôs que a utilização de um identificador único para um usuário poderia servir como uma espécie de chave pública que protegesse as informações trocadas. Tal identificador poderia ser algo simples como um e-mail, criando uma alternativa atraente à complexidade na distribuição de chaves públicas existentes em outros mecanismos.

Foi através dos trabalhos de C. Cocks [76] e também de D. Boneh e M. Franklin [75] que surgiram os primeiros algoritmos que permitiram implementar as idéias de A. Shamir [74]. A proposição D. Boneh e M. Franklin [75], posteriormente, subsidiou a criação da RFC 5091 [77], padronizando o algoritmo para emprego em sistemas abertos.

Numa arquitetura IBE o remetente da mensagem gera uma chave pública, por demanda, com base em seu identificador público, a exemplo do e-mail. O receptor da informação deve obter, através de um serviço de geração de chaves privadas, a chave unicamente associada que permita decifrar a mensagem cifrada com seu identificador público, não sendo necessário que os processos de geração e obtenção de chaves ocorram de forma causal. Também, através de modificadores concatenados ao identificador público, o remetente pode determinar a validade inicial ou final de uma informação cifrada. É interessante ressaltar que o IBE dispensa a comunicação direta entre o remetente e o destinatário com o objetivo de estabelecer as chaves utilizadas na comunicação.

O foco do IBE é a criptografia das informações. Não sendo previstas formas de autenticação do remetente nem do destinatário perante o PKG, tampouco formas de se realizar a assinatura digital das mensagens trocadas, deixando estas funções para mecanismos complementares.

No esquema proposto na RFC 5091, uma infra-estrutura IBE é composta por dois servidores, um deles sendo o PKG acima e outro conhecido como PPS (*public parameter server*). O PPS serve para disponibilizar *parâmetros complementares*, unicamente associados à chave mestra de um PKG, para emprego no mecanismo de geração de chaves públicas pelo remetente. A chave pública é gerada pela composição dos *parâmetros complementares*, disponibilizados livremente pelo PPS, com o identificador público do destinatário e as políticas de restrição de datas. Em geral, os servidores PKG e PPS estão associados a um domínio DNS, integrando a identificação do serviço IBE com o identificador do usuário, caso este contenha um nome de domínio, a exemplo do e-mail.

A RFC 5091 recomenda que as conexões entre o remetente e o PPS e o destinatário e o PKG ocorram através de túneis TLS, de modo a tornar confiável a relação entre o domínio relacionado ao identificador e fornecedor dos *parâmetros complementares*. Da mesma forma, o TLS permite que o receptor tenha a garantia de que vai obter a chave privada do servidor correto.

Capítulo 3

Funcionamento de uma rede com SIP

Este capítulo apresenta em detalhes o protocolo de gerenciamento de sessões SIP, com ênfase nos mecanismos adotados para envio e recebimento da mídia de áudio, como meio de fundamentação das discussões posteriores sobre a temática que está sendo analisada neste trabalho.

Através do protocolo SIP é possível encaminhar as chamadas para um ponto de interceptação onde seja possível capturar e armazenar as mídias criptografadas para posterior interpretação do seu conteúdo, de forma que é importante o entendimento completo do seu funcionamento.

Alguns detalhes do protocolo pertinentes ao funcionamento das facilidades de mensagem instantânea e de gerenciamento de presença não serão tratados no trabalho.

3.1 Visão Geral do Protocolo

O *Session Initiation Protocol* - SIP presta-se ao estabelecimento, manutenção e término de sessões de áudio e vídeo, e suporta esquemas para uso de mensagens instantâneas e notificação de presença. O protocolo SIP está especificado na RFC3261 [43] e em diversos documentos complementares. O presente capítulo enfatiza o seu uso no gerenciamento da mídia de áudio nas aplicações de Voz sobre IP.

O SIP tem sua origem nos protocolos SMTP e HTTP e possui um esquema de mensagens codificadas textualmente, com campos delimitados pelos caracteres de retorno e alimentação de linha, simbolizados por CRLF. Pacotes contendo informações sobre o SIP, capturados na rede, poderiam ser lidos sem auxílio de um mecanismo de decodificação. Da mesma forma

que o HTTP, o SIP é baseado no esquema pergunta-resposta, através de mensagens padronizadas. O SIP é independente do protocolo de transporte, podendo operar sobre UDP, TCP, TLS ou SCTP. O UDP é usado por simplicidade e velocidade enquanto o TCP para proveito dos mecanismos de retransmissão. Mensagens SIP que ultrapassem o tamanho de 1300 Bytes, se o MTU do caminho for desconhecido, devem utilizar o TCP. O TLS destina-se ao uso de criptografia para proteção das mensagens entre os terminais e os servidores.

O SIP não atua sozinho para estabelecer as sessões de mídia: ele depende de uma série de protocolos especificados pela IETF ou a ITU. A Figura 3-1 apresenta um quadro geral com os principais protocolos que prestam auxílio no processo de estabelecimento de mídia e gestão da sessão. Muitos deles serão discutidos nas próximas sessões e capítulos.

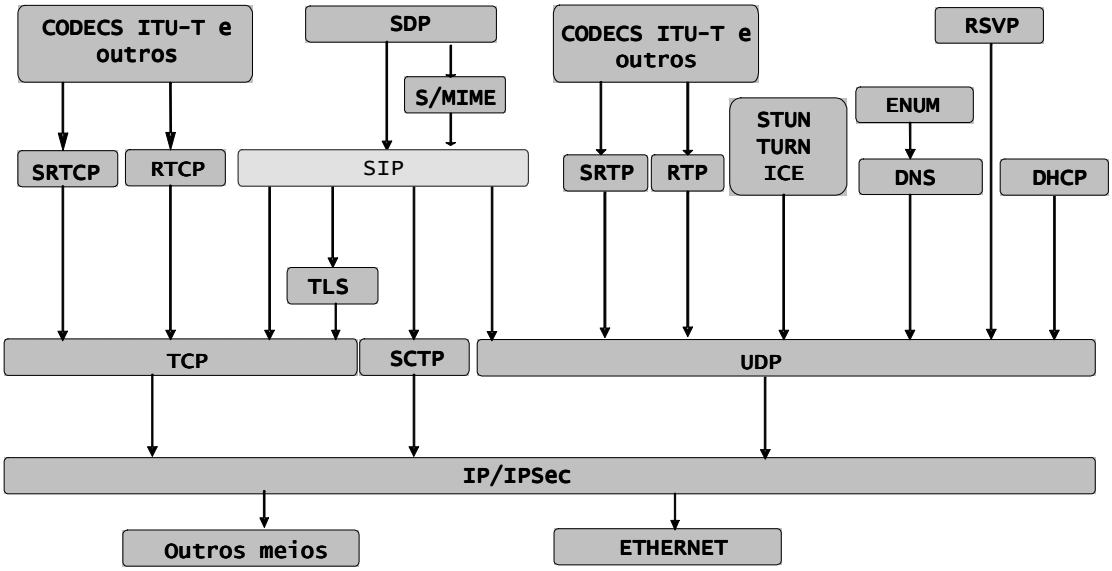

Figura 3-1 - Contexto do Protocolo SIP

A arquitetura do SIP prevê as seguintes entidades, estando ilustrada na Figura 3-2:

- **Usuário**. É a parte interessada em trocar tráfego, identificada através de um identificador universal chamado de URI de usuário ou AOR. Um AOR é o identificador típico que será utilizado pelo usuário para divulgá-lo em um diretório, cartão de visita, lista telefônica ou outro meio de publicação de contatos;
- **Terminal.** Também conhecido como agente usuário, identificado por um URI de dispositivo, ou UA, cumprindo simultaneamente as funções de cliente (UAC) e servidor (UAS), iniciando, mantendo e encerrando chamadas. Diferentemente do HTTP e demais protocolos que tem as funções cliente e servidoras distintas, o um

agente SIP pode enviar ou receber solicitações, cumprindo os dois papéis em uma mesma aplicação, agindo como um elemento P2P;

- **Servidor *Proxy***. É o elemento que processa as mensagens SIP enviadas pelos terminais, inspecionando o cabeçalho, com destaque para os campos *Request-URI* e o campo *To*, de forma a determinar o endereço efetivo do destinatário da chamada. O *Proxy* contata o servidor de localização, resolvendo o nome do usuário em um endereço IP ou em outro URI, reenviando a chamada para o terminal de destino ou para um próximo servidor *Proxy*, conforme a política de encaminhamento empregada;

- **Servidor de redirecionamento.** Cumpre a função de enviar as mensagens de reposta, da classe de redirecionamento de chamadas[1], para outro URI. Uma mensagem de redirecionamento recebida pelo UAC enseja o envio de um novo convite para o endereço especificado pelo redirecionador;

- ***Gateway***. Um *gateway* pode prestar serviços de tradução de sinalização ou transcodificação de mídia. Pela tradução da sinalização entende-se converter as requisições padrão do SIP para outro formato, como o H.323 ou para uma sinalização da rede telefônica PSTN. Como transcodificador, ele cumpre o papel de converter a codificação utilizada pela mídia para outro formato. Algumas vezes esta conversão inclui a troca de protocolo de transporte, por exemplo, quando convertendo do RTP para o mundo TDM das redes PSTN. Quando é necessária apenas a transcodificação, mantendo o RTP e a sinalização, tal elemento chama-se, mais comumente, de ***Proxy* de Transcodificação**;

- **Serviço de Localização**. É o elemento responsável por traduzir endereços AOR, em um URI para o ponto de contato onde o usuário encontra-se, incluindo endereços IP's, conforme solicitado pelos demais elementos. O protocolo entre o *Proxy* e o servidor de localização não é especificado pelo SIP;

- **Serviço de Registro**. É o serviço onde um terminal realiza o cadastro do AOR e do URI de contato onde o UA está localizado, normalmente ao ser inicializado na rede, anunciando sua existência e registrando o seu ponto do contato. O processo de registro no SIP é cumulativo.

[1] No SIP as mensagens são numeradas com três dígitos e dividas em cinco classes. Estas são identificadas pelo número da centena, conforme Tabela 3-3. Esta identificação foi baseada nos padrões SMTP e HTTP.

A Figura 3-2 mostra os componentes SIP. Sendo uma rede SIP o conjunto de elementos que funcionam sob uma mesma entidade administrativa. A distinção é apenas exemplificar do *gateway* na integração com redes distintas. Como redes SIP e H.323 operam no nível da aplicação, não há limite na distribuição da rede TCP/IP que as suportam.

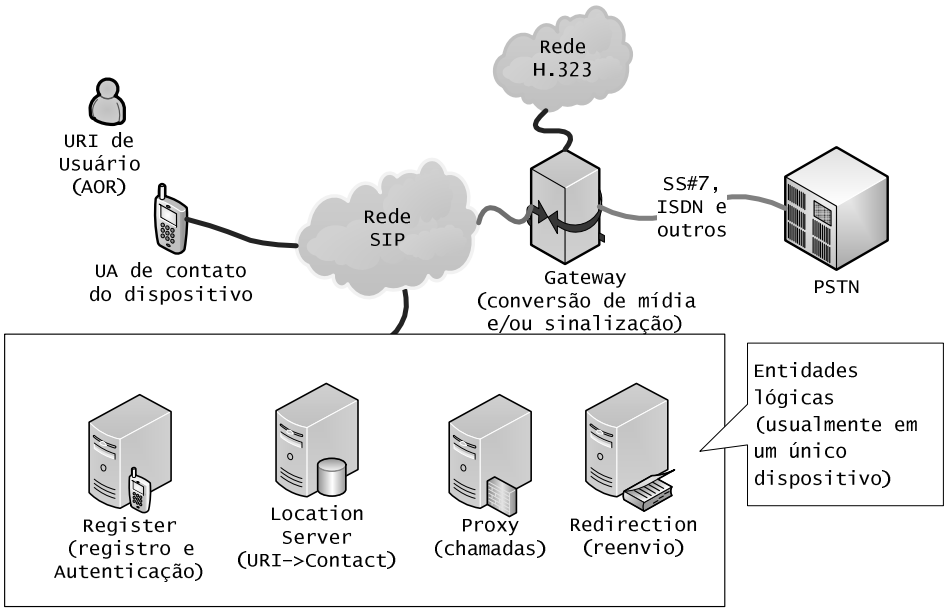

Figura 3-2 - Arquitetura SIP

3.2 Transações no SIP

O SIP opera baseado em transações, sendo que tais transações são divididas em duas grandes fases. A primeira compreende o convite para uma nova sessão, pela mensagem de INVITE, até sua efetivação, pela resposta “200 OK”, ou erro. A segunda fase é composta por todas as demais transações que podem objetivar encerrar a sessão atual, acessar alguma facilidade, como transferência, chamada em espera ou para alterar alguma característica em curso da mídia. Todas as transações deste segundo momento estão inclusas no que o SIP chama de **diálogo**. Durante a transação de estabelecimento do diálogo, o protocolo SDP [67] é utilizado para negociar os detalhes das mídias que serão utilizadas na sessão, como o CODEC utilizado, a taxa de amostragem em que irá operar o codificador e os detalhes do protocolo de transporte. As portas UDP que serão utilizadas para tráfego da mídia são negociadas dinamicamente para uso por ambos agentes usuários, fazendo com que o caminho da mídia seja diferente do caminho da sinalização, efetuando um desenho que a RFC 3261 chama de “trapezóide SIP”, conforme Figura 3-3.

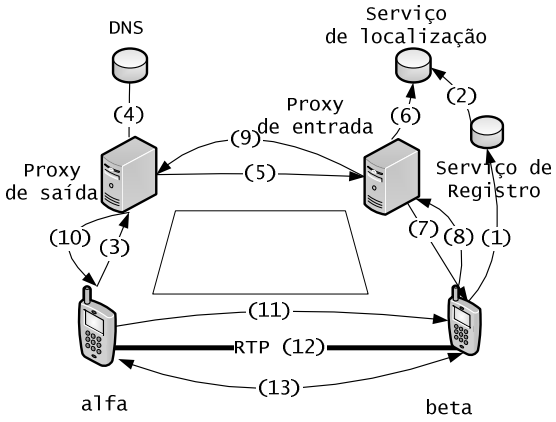

Figura 3-3 - Ordem de Mensagens do SIP

3.3 Funcionamento do protocolo SIP

A Figura 3-3 apresenta de forma numerada as mensagens trocadas pelo protocolo SIP para o estabelecimento de uma chamada. Em primeiro lugar o usuário *beta* deve cadastra-se no serviço de registro (1), que por sua vez atualiza esta informação num serviço de localização[2] (2). Quando uma chamada solicitada por *alfa*(3), conforme Figura 3-5, com a descrição da mídia, é encaminhada para seu servidor *Proxy* de saída, este deve localizar o servidor *Proxy* de *beta* através de uma consulta DNS (4). O convite é encaminhado para o servidor *Proxy de entrada* (5), que por sua vez resolve os dados sobre o *Request*-URI (6), localizando o ponto de contato momentâneo, encaminhando a chamada para o destino (7). *Beta* retorna uma mensagem de sinalização pelo mesmo caminho de chegada (8, 9,10) contendo um aviso que o usuário está sendo avisado da chamada, composto usualmente por mensagens numeradas "100 Trying"[3], "180 Ringing". Caso *beta* atenda, a chamada será estabelecida pelo envio por *beta* da mensagem "200 OK" anexado da descrição SDP da mídia. Neste ponto a transação de abertura de diálogo é encerrada. A mensagem de ACK[4] (11) é uma concordância de *alfa* sobre as condições negociadas, dando início à abertura da mídia (12) entre *alfa* e *beta*, sem que os demais elementos participem deste processo. A tripla INVITE, 200, ACK é chamada na RFC 3261 de "*three way handshake*". O encerramento da chamada ocorre pelas mensagens BYE, confirmada com um "200 OK", trocadas pelas partes no caminho (13). Dois pontos de destaque devem ser ressaltados: a mídia via RTP é enviada diretamente entre os UA; e nem toda sinalização percorre os caminho dos *Proxies*. De fato, somente a transação de abertura é vista pelos *Proxies*. Políticas instaladas em um *Proxy*

[2] Na prática, os serviços de registro e localização são integrados ao *Proxy*, em um único servidor.
[3] Mensages "100 Trying" são processadas entre vizinhos no caminho da sinalização, e não são roteadas.
[4] Mensagens de ACK que sucedem o INVITE não são confirmadas.

podem determinar se as demais transações pertencentes ao diálogo deverão ou não atravessar o *Proxy* que tiver interesse em permanecer no caminho da sinalização, conforme opção definida pelo protocolo (seção 3.5, pág. 38).

3.4 SIP na modalidade P2P

A Figura 3-4, apresenta uma troca de mensagens entre dois dispositivos SIP sem a presença de um *Proxy*, conectados por uma rede IP, operando na modalidade P2P, de forma a detalhar o formato das principais mensagens. O lado *Alfa* envia uma mensagem INVITE que contém todos os detalhes para início de uma chamada multimídia.

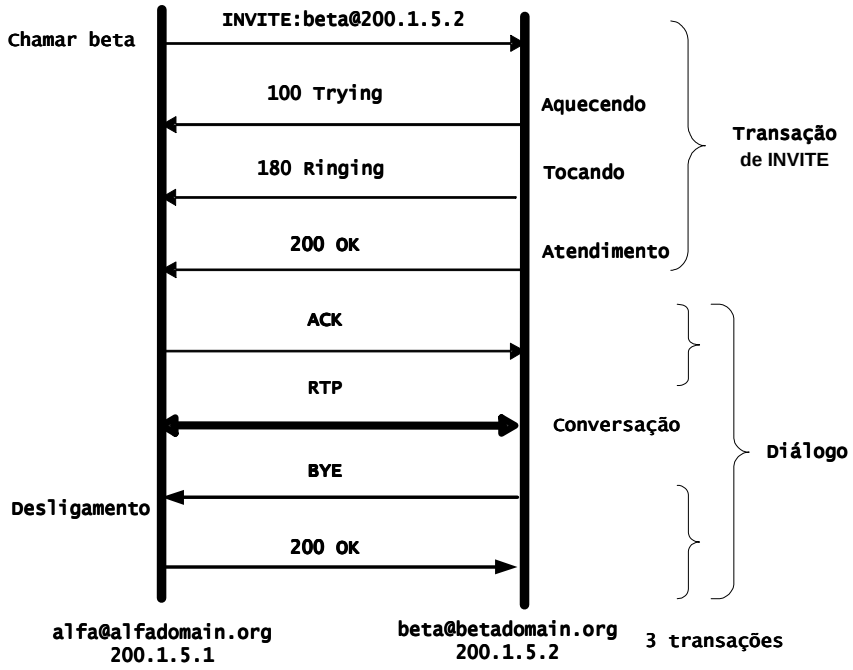

Figura 3-4 - Fluxo de Mensagens SIP operando em P2P

```
INVITE sip:beta@200.1.5.2 SIP/2.0
Via: SIP/2.0/UDP 200.1.5.1:5060;branch=z9hG4bKfw19b
Max-Forwards: 70
To: Beta B <sip:beta@200.1.5.2>
From: Alfa A <sip:alfa@alfadomain.org>;tag=76341
Call-ID: 123456789@200.1.5.1
CSeq: 1 INVITE
Subject: Atenda por favor...
Contact: <sip:alfa@200.1.5.1>
Content-Type: application/sdp
Content-Length: 158

v=0
o=Alfa 2890844526 2890844526 IN IP4 200.1.5.1
s=Phone Call
```

```
c=IN IP4 200.1.5.1
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Figura 3-5- INVITE

Como o SIP possui seus campos em formato texto, o exemplo acima representa o que seria visto através da captura do respectivo pacote por um analisador de protocolos tipo o *wireshark*[5] ou o *ngrep*[6]. Assim como o HTTP, todos os campos são terminados em CRLF. A distinção entre o SIP e o SDP, que segue em anexo no corpo da mensagem, ocorre por um duplo CRLF. O SIP suporta esquemas de MIME para composição de vários anexos a mensagem, seguindo a RFC 2045 [36].

Na Figura 3-5 a primeira linha representa o método de requisição, o ***Request*-URI**, e contém o tipo da requisição (INVITE) do pacote, a URI de destino e a versão do protocolo (SIP2.0). O primeiro campo *Via* contém o *hostname* do originador do pacote, que pode ser o lado *Alfa* da comunicação ou algum agente intermediário. Além disto, é informada a versão do protocolo[7], o tipo de transporte usado (UDP) e a porta[8]. O parâmetro *branch* é um identificador da passagem desta mensagem pelo *Proxy*, permitindo a correlação das mensagens de retorno. *Max-forward* indica quantos *Proxies* são admitidos no trajeto, e deve ser decrementado por cada possível *Proxy*, provendo um algoritmo simples de detecção de loops na camada de aplicação. Os campos *To* e *From* indicam o originador e receptor da mensagem, respectivamente. Quando um rótulo é usado para fins de visualização do nome do originador na tela do receptor, a URI seguirá delimitada pelos caracteres “<” e “>”. *Cseq* indica o número da transação ao qual esta mensagem pertence, seguida da referida mensagem de requisição que fez a abertura da transação. Os campos acima são considerados essenciais para todas as mensagens SIP e os demais são opcionais ou dependentes do tipo de requisição. O campo *Contact* é requerido nas mensagens INVITE e contém a URI SIP do terminal do originador (UA), permitindo que ele seja localizado diretamente. O *Subject* não cumpre outra função senão permitir ao receptor decidir se deve ou não atender a chamada, baseado em algum informativo. Os campos *Content-Type* e *Content-Length* indicam o protocolo que segue e o tamanho respectivo em bytes. Neste caso é indicado como protocolo o SDP descrevendo os CODECS possíveis para uso pelo originador, que será detalhado na seção 3.8.

[5] Encontrado em http://www.wireshark.org/.
[6] Encontrando em http://ngrep.sourceforge.net/
[7] A versão 2 significa a atual RFC 3261.
[8] A porta 5060 é padrão para o SIP em UDP e TCP. A porta 5061 é padrão para o SIP com TLS.

O SIP possui uma família de mensagens de resposta padronizadas pelo número da centena, tal qual o HTTP. Neste caso a resposta contém a mensagem "180 *Ringing*", ilustrada na Figura 3-6. Esta mensagem possui praticamente uma cópia do INVITE, sem alterar a ordem de *To* e *From*, haja vista que ela ainda pertence à transação INVITE, conforme mostra o campo *CSeq*. Adicionalmente aparece a definição do atributo *tag* no campo *To.*

Os campos *Caller*-ID e os *tags* posicionados nos campos *To* e *From* servem para identificar unicamente um diálogo, e devem ser mantidos por todos os elementos da rede SIP que desejarem manter o estado de uma chamada. O *tag* para o campo *From* e o *Caller-ID* são gerados pelo chamador, enquanto que o *tag* do campo *To* é gerado pelo recebedor da chamada.

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 200.1.5.1:5060;branch=z9hG4bKfw19b
To: Beta B <sip:beta@200.1.5.2>;tag=a53e42
From: Alfa A <sip:alfa@alfadomain.org>;tag=76341
Call-ID: 123456789@200.1.5.1
CSeq: 1 INVITE
Contact: <sip:alfa@a.alfadomain.org>
Content-Length: 0
```

Figura 3-6 - Exemplo de uma mensagem de retorno de ring

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 200.1.5.1:5060;branch=z9hG4bKfw19b
To: Beta B <sip:beta@200.1.5.2>;tag=a53e42
From: Alfa A <sip:alfa@alfadomain.org>;tag=76341
Call-ID: 123456789@200.1.5.1
CSeq: 1 INVITE
Contact: <sip:alfa@200.1.5.1>
Content-Type: application/sdp
Content-Length: 155

v=0
o=Alfa 2890844528 2890844528 IN IP4 200.1.5.2
s=Phone Call
c=IN IP4 200.1.5.2
t=0 0
m=audio 60000 RTP/AVP 0 a=rtpmap:0 PCMU/8000
```

Figura 3-7 - Exemplo de Mensagem de Atendimento

Como a chamada foi aceita pelo destino a Figura 3-7, "200 OK," foi enviada de volta indicando o atendimento. No protocolo SDP segue o tipo de CODEC aceito pelo destinatário, conforme será visto na seção 3.8.

Uma mensagem ACK é enviada pelo originador para confirmar o recebimento da mensagem "200 OK", dando início do diálogo, conforme Figura 3-8. Neste instante, ambas as

partes conhecem os CODECS e as portas de uso pelo RTP onde serão aceitas as mídias, iniciando a conversação.

```
ACK sip:beta@200.1.5.2 SIP/2.0
Via: SIP/2.0/UDP 200.1.5.1:5060;branch=z9hG4bK321g
Max-Forwards: 70
To: Beta B <sip:beta@200.1.5.2>;tag=a53e42
From: Alfa A <sip:alfa@alfadomain.org> ;tag=76341
Call-ID: 123456789@200.1.5.1
CSeq: 1 ACK
Content-Length: 0
```

Figura 3-8 - Exemplo de uma mensagem ACK

A chamada é encerrada por *beta*, através do envio da Figura 3-9, que é também confirmada como uma resposta "200 OK", não representada em exemplo.

```
BYE sip:alfa@200.1.5.1 SIP/2.0
Via: SIP/2.0/UDP 200.1.5.2:5060; branch=z9hG4bK392kf
Max-Forwards: 70
To: Beta B <sip:alfa@200.1.5.1>;tag=76341
From: Alfa A <sip:alfa@200.1.5.2>;tag=a53e42
Call-ID: 123456789@200.1.5.1
CSeq: 1 BYE Content-Length: 0
```

Figura 3-9 - Exemplo de uma mensagem de BYE

3.5 Chamadas através de *Proxy*

O grande motivador de um *Proxy* reside no originador não conhecer o endereço IP do destinatário, necessitando que alguém o auxilie neste processo. AOR são semelhantes a endereços de e-mail: é impossível determinar, a priori, em que terminal ele será lido. O *Proxy* é o agente intermediário que auxilia esta conversão. O SIP utiliza um esquema de identificação do usuário do formato `<usuário>@<domínio>` conhecido como SIP URI, inutilizando, na maioria dos casos, o esquema de resolução de nomes diretamente através de um DNS. Esta denominação facilita o desacoplamento entre o local físico e o usuário, provendo o conceito de mobilidade e por conseqüência a capacidade em realizar e receber chamadas em qualquer ponto de contato registrado.

Um *Proxy* não inicia nem é ponto de encerramento de solicitações e também não trata das informações referentes às mídias. O SIP URI é capaz de assimilar esquemas de numeração como o descrito no padrão ITU E.164[9] para o nome de usuário, facilitando o mapeamento direto entre os dois identificadores e reduzindo o processamento no roteamento da chamada. Uma forma não detalhada neste texto que permite mapear identificadores E.164 com URI SIP

[9] A Norma ITU E.164 define o plano de numeração das redes PSTN, como lidamos atualmente.

cujo nome de usuário literal é utilizar os serviços de DNS que implementem a recomendação ENUM, presente na RFC 3761 [54].

A Figura 3-10 ilustra a troca de mensagens por intermédio de um *Proxy*. Num primeiro passo o UA resolve o endereço do servidor SIP que responde pelo seu domínio, por uma resolução simples de DNS[10] e encaminha o INVITE.

O *Proxy* pode resolver diretamente o usuário de destino, se este pertencer ao mesmo domínio, acessando a base do serviço de localização. Se não, ele deve tentar uma resolução para o domínio contido no *Request*-URI, determinando o *Proxy* que responde por um determinado domínio, para o serviço SIP, através de uma resolução DNS. Este processo é descrito em [45].

Ao encaminhar o INVITE para o destino ou outro *Proxy* intermediário, o *Proxy* processando a mensagem deve acrescentar seu próprio identificador no campo *Via*, empilhando mais uma linha. Com isto, o receptor poderá saber para quem vai encaminhar a mensagem de resposta, e os *Proxies* intermediários os seus predecessores respectivos, assim por diante, até atingir a origem. Este aspecto assegura que toda a sinalização na transação de que dá início ao diálogo percorrerá sempre o mesmo caminho na volta.

As respostas obtidas às mensagens de requisição são preenchidas com a relação de *Proxies* informadas durante o processo do parágrafo anterior, copiando-se todos os campos *Via*, presentes na requisição. Cada *Proxy* deverá desempilhar sua entrada da relação e enviar para o próximo. Como regra, quaisquer mensagens de retorno devem ser repassadas para o "último *Proxy*" listado no topo da pilha de campos "Via" no cabeçalho, até que se atinja a origem. Durante a vida de uma chamada as partes envolvidas devem guardar uma série de variáveis que representam o estado atual da chamada.

A RFC condiciona anexar ao campo Via o parâmetro "received=<IP>", permitindo dispensar resoluções de nomes e URI's nas mensagens de resposta. Este campo também tem relevância na descoberta de um agente NAT no trajeto da sinalização.

[10] Através de um registro DNS tipo SRV ou, se não existir, do tipo A.

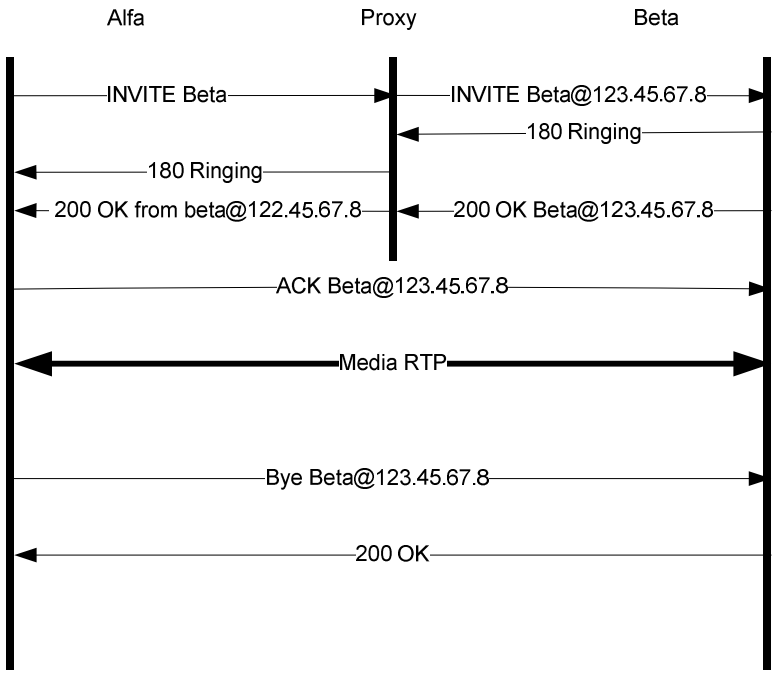

Figura 3-10 - Troca de sinalização com um *Proxy*

A Figura 3-11 é uma alteração da Figura 3-5, mostrando como seria o INVITE no momento (7) da Figura 3-3Figura 3-3 - Ordem de Mensagens do SIP. A diferença consiste na presença dos campos Via, inseridos pelos *Proxies*, de forma a forçar que as mensagens de retorno referentes a esta transação os atravessem. Nas mensagens de resposta (ex.: 200 OK) o terminal de destino copia todos os campos "Via". Cada *Proxy* deve retirar sua entrada durante o roteamento na volta. O campo *branch* serve para um *Proxy* determinar de que passagem a mensagem processada pertence, prevendo a possibilidade de uma transação circular mais de uma vez por um mesmo servidor[11].

```
INVITE sip:beta@betadomain.org SIP/2.0
Via: SIP/2.0/UDP proxy.betadomain.org:560;branch=z9hG4bKfadcc;
received=200.1.5.10
Via: SIP/2.0/UDP proxy.afadomain.org:560;branch=z9hG4bKfaabb;
received=200.1.5.12
Via: SIP/2.0/UDP alfadomain.org:5060;branch=z9hG4bKfw19b;
received=200.1.5.1
Max-Forwards: 70
To: Beta <sip:beta@betadomain.org>
From: Alfa <sip:alfa@alfadomain.org>;tag=76341
Call-ID: 123456789@alfa.alfadomain.org
CSeq: 1 INVITE
Contact: <sip:alfa@alfa.alfadomain.org>
Content-Type: application/sdp
Content-Length: 158
```

[11] Isto é plenamente possível, até que *Max-Forwards* chegue a zero.

```
v=0
o=Alfa 2890844526 2890844526 IN IP4 alfa.alfadomain.org
s=Phone Call
c=IN IP4 200.1.5.1
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Figura 3-11 - INVITE com *Proxy*

Uma diferença adicional no INVITE anterior àquele definido na Figura 3-5 é que o usuário *alfa* utiliza como *Request*-URI o valor sip:*beta@beta*domain.org, algo que não poderia ser traduzido diretamente em um endereço IP via um serviço de DNS.

3.5.1 Características complementares de um *Proxy*

Proxies podem operar no modo ***stateful*** ou no modo ***stateless***. No modo *stateful* ele mantém o estado da transação, temporizando, reenviando ou gerando mensagens necessárias para a conclusão da transação. No modo *stateless* ele cumpre apenas o papel de roteador de mensagem e tradutor de URI's.

Outra capacidade de um *Proxy stateful* é realizar um *fork* de uma requisição objetivando atingir vários pontos de contato simultaneamente para um único INVITE do assinante. Por exemplo, em momentos em que um AOR estiver registrado em mais de um URI de contato, forçando que mensagens de INVITE sejam enviadas simultaneamente para todos os pontos de contato até o usuário atender em algum ponto. O *Proxy* ao identificar o atendimento envia uma mensagem de CANCEL, terminando as demais transações pendentes.

Proxies podem ainda cumprir a função de tarifação, resolver solicitação com base em endereços E.164 e aplicar políticas, autorizações e demais funções de filtragem.

3.6 Forçando o roteamento de mensagens pelo *Proxy*

A presença do campo *Contact*: na resposta "200 OK" dá o ensejo ao usuário chamador enviar a mensagem de ACK diretamente para o destino, como exemplificado na Figura 3-11[12], acarretando na exclusão do *Proxy* do caminho de troca de sinalização nas transações pertencentes ao diálogo (seção 3.2, pág. 33). Para evitar esta característica e permitir o controle completo durante toda a chamada, possibilitando a bilhetagem correta, o *Proxy* deve

[12] O hostname contido no campo *Contact*: <sip:*alfa@alfa.alfa*domain.org>, que é *alfa.alfa*domain.org pode ser resolvido por uma pergunta ao DNS por um registro do tipo "A", conforme [30].

inserir o campo *Record-route* na mensagem de INVITE informando que ele deve ser mantido durante as demais trocas de sinalização. Os *Proxies* subseqüentes que tiverem o mesmo interesse acrescentam seus respectivos URI's no campo *Record-route*, ao final, separado por ponto-e-vírgula. A recomendação atual é que o *Request*-URI não seja alterado e que o campo *Record-route* seja preenchido durante a requisição de INVITE. O campo *Route* surge nas requisições que abrem as transações durante o diálogo, contendo os valores preenchidos no campo *Record-route*, marcando o roteamento desejado. Tal efeito é conhecido como *loose-route*, por não coibir que outros *proxies*, além daqueles que subscreveram no campo *Record-route*, possam estar no trajeto da mensagem[13].

3.7 Registro

É a primeira etapa em qualquer componente terminal SIP que intencione ser integrado a uma rede, ao invés de apenas ter serviços P2P. O processo de registro ocorre através do simples envio de uma mensagem do tipo REGISTER conforme exemplificada abaixo, sucedida pela recepção de "200 OK" se tudo ocorrer de acordo.

```
REGISTER sip:alfa.alfadomain.org SIP/2.0
Via: SIP/2.0/UDP 200.201.202.203:5060;branch=z9hG4bKus19
Max-Forwards: 70
To: Alfa B <sip:alfab@alfadomain.org>
From: Alfa B <sip:alfab@alfadomain.org> ;tag=3431
Call-ID: 23@200.201.202.203
CSeq: 1 REGISTER
Contact: sip:alfab@200.201.202.203
Content-Length: 0
```

Figura 3-12 – Registro

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 200.201.202.203:5060;branch=z9hG4bKus19
To: Alfa B <sip:alfab@alfadomain.org>;tag=8771
From: Alfa B <sip:alfab@alfadomain.org>;tag=3431
Call-ID: 23@200.201.202.203
CSeq: 1 REGISTER
Contact: <sip:alfab@alfadomain.org >;expires=3600
Content-Length: 0
```

Figura 3-13 - Confirmação de Registro

O *Request*-URI é a URI do servidor de registro. *To*: e *From*: normalmente contém o mesmo valor e representam o AOR. O campo *Contact* representa o local onde o usuário se

[13] Por políticas empregadas em um domínio administrativo, pode-se requerer que as mensagens de sinalização atravessem um *Proxy* que não esteja listado no campo *Record-Route*, desde que os listados façam parte do roteaomento.

encontra no momento, criando um aspecto circunstancial quando comparado aos campos *To* e *From*.

3.8 O protocolo SDP

O SDP, descrito na RFC 2327 [67] serve para comunicar as possíveis mídias que os terminais suportam e a ordem de preferência entre elas. O receptor da descrição SDP deve concordar com alguma das mídias apresentadas, evoluindo num processo chamado de oferta-resposta. Este método determina como os participantes vão acordar a descrição das mídias. Um participante constitui uma oferta dos tipos de mídia suportados, características de segurança e outros, acompanhados do endereço e porta por onde ele deseja receber o conteúdo, através do SDP. A outra parte responde sobre esta oferta, indicando apenas as opções que ele suporta ou colando em zero o número da porta das características não desejadas.

Incompatibilidade entre a relação de CODECS informados pelo INVITE e o receptor, podem fazer com que este envie uma resposta "606 Not Acceptable", forçando o originador a uma das condições:

- Enviar novo INVITE, com novos CODECS, o que seria pouco provável, pois ele já teria apresentado antes;
- Reenviar a chamada através de um *proxy* com habilidade de transcodificação;
- Desistir (via mensagem CANCEL).

Incompatibilidades no processo de requisição-resposta, que não possam ser deduzidos imediatamente pelo INVITE e conduzam a não continuidade do diálogo devem ser processadas como encerramento do INVITE através de uma resposta da classe 2XX, seguida de um BYE.

O SDP pode ser carregado em mensagens SIP do tipo INVITE, ACK[14], PRACK ou UPDATE e nas respostas 18X e 200.

Mensagens 18X acompanhadas de SDP servem para estabelecer a mídia antes do fechamento da transação de INVITE de forma a permitir a escuta de sinalização dentro da banda quando a chamada for atravessar algum *gateway* com a rede PSTN. Os padrões das redes PSTN utilizam parte da sinalização "dentro da banda" e isso pode influenciar no curso

[14] Apenas se o SDP não tiver sido enviado no INVITE.

da chamada e no comportamento do usuário chamador, como, por exemplo, as mensagens de *ring*, caixa postal sem tarifação e outras que ocorrem antes dos sinais de atendimento, congestionamento ou ocupado. O PRACK serve para confirmar o SDP durante o fechamento pré-maturo do áudio, principalmente quando operando sobre UDP.

O SDP serve para estabelecer as seguintes informações quanto a uma sessão unicast:

- Endereço IP, a família do protocolo e a versão, onde a conexão ocorrerá;
- As portas RTP utilizadas em ambos os lados;
- Uma ou mais mídias, como áudio, vídeo, *whiteboard* e outros;
- O esquema de codificação da mídia (PCM a-Law, MPEG-II e outros)

Abaixo segue um exemplo de uma mensagem SDP

```
v=0
o=Alfa 2890844526 2890844526 IN IP4 alfa.alfadomain.org
s=Phone Call
c=IN IP4 100.101.102.103
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Figura 3-14 - Exemplo de SDP

3.8.1 Campos utilizados pelo protocolo SDP

Muitos dos campos definidos na RFC 2327 não têm sentido na RFC 3264 [46], sendo que alguns são mantidos apenas para efeito de compatibilidade. Por fim, o SDP assume que algum outro protocolo o irá transportar e manterá o contexto dos acontecimentos, como feito nas aplicações SIP.

- **Origem (o=)**: contém informações que identificam unicamente o originador e a sessão, no formato: *o=username session-id version network-type address-type*, sendo que username pode ser o nome do host, o usuário ou apenas "-". Em geral recomenda-se que *session*-id e version sejam *timestamps* extraídos via NTP.
- **Session (s=)**: contém o nome da sessão, com qualquer seqüência não nula de caracteres. Sem interesse para unicast;
- **Conexão(c=)**: contém informações sobre onde deve ser terminada a sessão, na forma *c=network-type address-type connection-address*,

- **Tempo (t=)**: contém o tempo (NTP) previsto de início e término da sessão. Valores em zero indicam “permanente”.
- **Chave (k=)**: estabelece via SDP uma chave de sessão para o RTP, representada na forma k=*method:encryption-key*, onde método de criptografia pode ser *clear, prompt, base64 e uri*. Com *prompt*, o usuário será interrogado para digitar uma chave comum.
- **Informação de mídia (m=)**: na forma *m=media port transport format-list*, podendo correr mais de uma linha num mesmo SDP.

A composição do campo *m=* pode conter os seguintes valores para mídia, transporte e format-list:

- ***Audio*****:** para uso em fonia ou áudio com alta definição.
- ***Video*****:** envio de mídias codificadas com H.26X ou outros protocolos.
- ***Application***: permite a invocação de alguma aplicação para processamento deste canal RTP (exemplo: *whiteboard – wb*)
- ***data***: para informações do canal de dados da mídia, que em geral não serão mostrados pelo usuário. *Closed Caption* e outros.
- ***telephone-event*****:** para envios de eventos de teclado (tons de DTMF) por intermédio da RFC2833 [41].
- ***control***: quando desejar o uso do protocolo de controle dos recursos de mídia, para síntese e reconhecimento de voz.
- ***Port***: contém a porta onde a conexão é aguardada
- ***Transport***: pode conter o valor RTP/AVP, *real time protocol/audio video profile*, indicando que a relação de CODECS segue o descrito na recomendação [51].
- ***Format-list***: informa a relação de CODECS suportados. Todos os CODECs contidos na lista devem ter seu campo atributo (a=rtpmap:), com as condições de configuração do próprio, como: taxa, número de bits, codificação e etc. Os números estabelecidos para os CODECS são padronizados.

A tabela abaixo sintetiza os campos do SDP.

Campo	Nome	Obrigatório
v=	Versão	√
o=	Dono e criador da sessão	√
s=	Propósito da sessão	√
i=	Informação sobre a sessão	×
u=	URL da sessão	×
e=	E-mail	×
p=	Número do telefone	×
c=	Informação sobre a conexão	√
b=	Banda	×
t=	Tempo de início e fim da sessão	√
r=	Número de repetições	×
z=	Timezone	×
k=	Chave de criptografia	×
a=	Linhas de atributo	×
m=	Informações da mídia	×

Tabela 3-1 - Campos do protocolo SDP

3.9 Mensagens de Requisição e Respostas do SIP

Por regra toda transação no SIP deve ser encerrada com uma resposta, a exceção das transações ACK, que seguem um INVITE[15]. As respostas servem tanto para notificar sobre o andamento de uma transação, pela classe 1XX, quanto para encerrar uma transação, pelas classes 2XX em diante.

Transações de CANCEL podem ser emitidas por UAC's a qualquer tempo, desde que a transação ainda esteja em aberto. Ao receber um CANCEL um UAS não pode mais emitir uma resposta sobre a transação original, e sim um "200 OK" sobre o CANCEL ou um "*481 Transaction Does Not Exist*".

Mensagens ACK que sucedem um INVITE podem conter uma descrição SDP, desde que o INVITE não contenha uma, servindo para o UAC postergar suas definições, em função daquelas indicadas na mensagem de retorno 200 OK oriunda do UAS. Em geral esta estratégia é adotada quando o UAC depender do UAS para definir alguma característica da mídia ou informação referente às condições de QoS da parte chamada. Métodos adicionais podem ser usados para esta finalidade se os UA suportarem, como visto adiante.

As seguintes requisições estão especificadas na RFC 3261 [43].

[15] O INVITE é a única transação cuja mensagem de resposta da classe 2XX em diante recebe um ACK. Tais ACKs, por sua vez, não recebem respostas.

REQUSIÇÃO	Descrição
INVITE	Requisita uma sessão (criação de uma chamada)
ACK[16]	Confirma qualquer resposta final (das classes 2xx,3xx,4xx,5xx,6xx) de um INVITE
OPTIONS	Questiona as facilidades de um terminal
CANCEL	Cancela uma transação pendente
BYE	Termina uma sessão
REGISTER	Envia as informações de terminal para o servidor

Tabela 3-2- Mensagens de Requisição do SIP

As seguintes classes de resposta estão padronizadas:

RESPOSTA	Descrição
1XX	Informativos de aprovisionamento (ex.: 180 Ringing)
2XX	Sucesso na solicitação (ex.: 200 OK)
3XX	Redirecionamento (ex.: 302 Moved Temporarily)
4XX	Erro no lado cliente (ex.: 404 Not Found)
5XX	Erro no lado servidor (ex.: 504 Server Time-out)
6XX	Falha global (ex.: 603 Decline)

Tabela 3-3 - Classes de Mensagem de Resposta

O SIP estabelece as seguintes em termos de comportamento no de envio de requisições e respostas padronizadas, por tipo de agente:

Propriedade	Servidor de Redirecionamento	Servidor *Proxy*	UAS	Servidor de Registro
Atua como UAC	Não	Sim	Não	Não
Retorna 1XX	Sim	Sim	Sim	Sim
Retorna 2XX	Não	Sim	Sim	Sim
Retorna 3XX	Sim	Sim	Sim	Sim
Retorna 4XX	Sim	Sim	Sim	Sim
Retorna 5XX	Sim	Sim	Sim	Sim
Retorna 6XX	Não	Sim	Não	Não
Permitido inserir o Campo Via	Sim	Sim	Sim	Não
Aceita ACK	Sim	Sim	Sim	Não

Tabela 3-4 - Comportamento no envio de mensagens

[16] O ACK só é fim-a-fim para confirmar uma reposta 2XX. Os demais são processados pelos *Proxies*.

3.9.1 Transações adicionais do SIP

Documentos complementares da IETF acrescentaram transações para conter condições especiais, das quais para este trabalho as seguintes são relevantes:

3.9.1.1 PRACK

Provisional ACK [44] são respostas do UAC às mensagens de aprovisionamento da classe 1XX geradas pelo UAS, a exceção da mensagem "100 Trying", quando for requerida confirmação. As respostas de 101 a 109 são representadas doravante de 1X[1-9].

Mensagens de 1X[1-9] e PRACKs podem carregar descritores de mídia SDP, num processo de oferta-resposta, permitindo a configuração dos detalhes da sessão e envio de mídia prematuramente. Em geral, este mecanismo é empregado para que o usuário chamador possa receber a mídia antes do encerramento do INVITE, quando a chamada terminar em um *gateway* PSTN, possibilitando a escuta das sinalizações *"dentro da banda"* oriundas das redes de telefonia convencionais. Tais sinais podem conter tons de chamada, mensagens de caixa postal, avisos sobre tarifação extra e outros, antes do envio de sinais como o atendimento ou congestionamento. Em geral estas sinalizações prévias ao atendimento permitem que o usuário altere o comportamento da chamada, como por exemplo, não permitir o redirecionamento para caixa postal do assinante, evitando uma cobrança de tarifa extra, como nos sistemas celulares.

Um UAS pode gerar uma resposta de aprovisionamento 1X[1-9], se assim o UAC suportar, pela indicação na mensagem de INVITE do *tag* 100rel, ou se o UAC insistir, pela presença do campo "*Required: 100rel*" no INVITE. *Proxies* podem interceder respondendo com 1X[1-9], a menos que o *tag* esteja presente no campo *To.*

Um UAS pode tomar a iniciativa de enviar um 1X[1-9], se estiver presente o *tag* 100rel no INVITE. Ao enviar uma mensagem 1X[1-9], o UAS pode requer uma confirmação pelo UAC, pelo retorno de um PRACK, sempre que houver indicação de um número de seqüência *RSeq* e o campo "*Supported: 100rel*" no corpo da mensagem 1X[1-9]. Um PRACK deve ser respondido pelo UAS com um 200 OK, encerrando o processo.

Uma mensagem 1X[1-9] pode terminar o processo oferta-resposta do SDP para um INVITE anterior, permitindo a ação prematura da mídia mesmo antes de encerrar o INVITE, se o UAC indicar com um tag 100rel. Um INVITE não acompanhado de um SDP pode fazer

com que a oferta venha do UAS e que a resposta parta do UAC por intermédio de um PRACK, invertendo o processo.

3.9.1.2 UPDATE

Mensagens de UPDATE [48] podem ser geradas tanto pelo UA chamador quanto pelo destinatário enquanto a transação de INVITE não tiver sido encerrada. Um UPDATE é realizado quando for necessária uma atualização ou criação de características da mídia que não podem ser cumpridos apenas pela simples troca das mensagens INVITE, 200 OK e ACK, em casos onde: deseja-se abrir prematuramente a mídia, estabelecer características sobre QoS, segurança ou alterações no aspecto da mídia como colocá-la em mudo e outros. Alterações de qualquer característica da sessão, após o encerramento da transação de INVITE, devem ocorrer através de um **re-INVITE**, com um novo SDP.

As partes tomam ciência sobre o suporte ao UPDATE pela presença do referido método no campo “Allow” no INVITE, que força o UAS a enviar uma mensagem de aprovisionamento, caso ele suporte o método. Em seguida, tanto o chamador quanto o chamada podem enviar mensagens de UPDATE em qualquer direção para ajuste das condições necessárias. Tais mensagens devem ser reconhecidas com um 200 OK, referente ao número CSeq presente no respectivo UPDATE. Por fim, a transação de INVITE é encerrada como de costume. Há algumas regras para evitar *condições de corrida* em relação ao fechamento da transação de INVITE ou outras solicitações pendentes, através de uma resposta de aprovisionamento, de forma que novos UPDATEs só possam ser enviados ou respondidos se as demais mensagens intermediárias estiverem fechadas, inclusive alguma mensagem de aprovisionamento.

3.10 O protocolo RTP

O RTP define um formato padrão para envio e recebimento de mídia sobre o protocolo UDP. Sua definição está contida na RFC3550 [50]. Ao contrário dos protocolos de aplicação, o RTP não definiu uma porta específica do UDP para seu uso exclusivo, tornando complexa sua identificação na rede para tratamento por dispositivos que realizam funções de *firewall* e NAT. Cabe ao protocolo de sinalização (SIP/SDP ou H.323) estabelecer tanto as portas em uso quanto o início e término de uma sessão. O RTP opera em conjunto com o RTCP, que utiliza a porta seguinte àquela definida para o RTP, de maneira que uma parte possa prover à

outra relatórios sobre a qualidade da sessão com informações sobre tráfego, perda de pacotes e variação do atraso.

O fato de o RTP utilizar alocação dinâmica de portas além de dificultar a operação em serviços de NAT pode também atrapalhar a classificação do tráfego para emprego de qualidade de serviço[17]. Mas a IETF, prevendo a possibilidade dos terminais estabelecerem múltiplos canais de mídia entre as partes, inclusive usando o mesmo CODEC, deve ter avaliado que seria complexo identificar a que fluxo um determinado pacote recebido por um lado da conversação poderia pertencer, não fosse pela diferenciação da porta UDP utilizada.

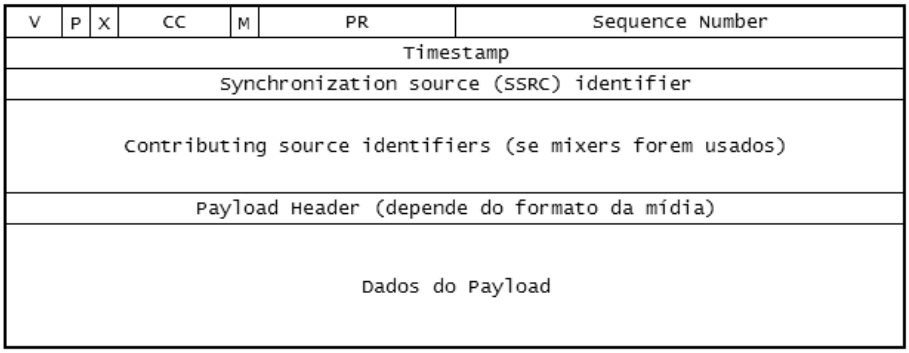

Figura 3-15 – Quadro RTP

Os campos têm a seguinte interpretação:

- **PT (payload type)**: que serve para definir o tipo de CODEC sendo transportado. A RFC 4288 [61] define como deve ser especificada uma mídia e como ela deve ser registrada no IANA. Os valores usados no PT são aqueles definidos pela negociação ocorrida através do protocolo SDP por intermédio do campo *format-list* (seção3.8.1, pág. 44). A Tabela 3-1 apresenta alguns exemplos de tipos pré-definidos, de acordo com a RFC 3551 [51].

Perfil	CODEC	Taxa	Descrição
0	PCMU	8000	ITU G.711 PCM com µLaw
2	G721	8000	ITU G.721 ADPCM 32Kbps
3	GSM	8000	Padrão Europeu para redes GSM 13Kbps
31	H.261	90000	ITU H.261 Vídeo

Tabela 3-5 - Exemplos de perfis pré-definidos para AVP

- **Sequence Number:** representa um inteiro de 16bits que é incrementado a cada pacote enviado, permitindo ao receptor identificar pacotes fora de ordem e perdas.

[17] As aplicações podem marcar o campo ToS com oDSCP apropriado, porém se este papel for deixado para a rede, poderá haver dificuldades na identificação do tráfego.

- **SSRC:** É um número aleatório que identifica a fonte geradora da informação, para efeitos de sincronização.
- **V:** a versão do protocolo RTP
- **P:** padding
- **X:** flag que informa se o pacote foi estendido para suportar informações de múltiplas fontes
- **CC:** número de identificadores de fonte que estão anexados.
- **M:** marcador para uso, por exemplo, para indicar início de detecção de silêncio.
- **CSRC:** é a relação das fontes usadas para computar a presente mídia por um processo de mixagem.

Para um protocolo como o G.711, o RTP deve preencher o campo PT com o valor zero. Na condição típica o G.711 envia oitenta bytes por amostragem, com cada intervalo de 10 milissegundos por amostragem. Cada pacote RTP transporta duas amostragens, totalizando 20 milissegundos de áudio, gerando um tráfego de cinqüenta pacotes por segundo. A tabela abaixo ilustra o comportamento típico dos principais CODECS disponíveis.

Parâmetro	G.711	G.723.1	G.726	G.728	G.729
Taxa da saída do CODEC (*codec bit rate*), após codificação do sinal.	64000	6300	32000	16000	8000
Tamanho da amostra de dados na saída por rodada de codificação realizada.	80 Bytes	24 Bytes	20 Bytes	10 Bytes	10 Bytes
Tempo de duração para coletar e processar cada amostra, ou Intervalo de amostragens.	10ms	24ms	5ms	5ms	10ms
Número típico de amostras acumuladas para envio ao RTP (que pode ser ajustável), por pacote.	2	1	4	6	2
Tamanho do *Payload* enviado por pacote RTP.	160 bytes	24 bytes	80 bytes	60 bytes	20 bytes
Tempo de conversação contido em cada *Payload* do RTP.	20ms	30ms	20ms	30ms	20ms
Taxa de pacotes por segundo enviado à rede pelo RTP.	50	34	50	34	50

Tabela 3-6 - Taxa de alguns CODECs[18]

[18] Fonte: http://www.cisco.com/en/US/tech/tk652/tk698/technologies_tech_note09186a0080094ae2.shtml

3.11 Integração com a Rede PSTN

Interessado na integração com as redes PSTN, a IETF patrocinou vários grupos de trabalhos para estudar formas de conversão de sinalização, cooperação do plano de numeração, transporte de sinalização PSTN por dentro da rede IP e relacionamento entre identificadores SIP URI e o endereçamento E.164. Os resultados destes grupos foram o desenvolvimento de alguns protocolos de integração, conforme apresentado abaixo.

3.11.1 Resolução com ENUM

Para permitir a tradução entre endereços E.164 e um URI *alfa*numérico, como um SIP URI, um alias H.323 ou um endereço de e-mail foi desenvolvido o mecanismo ENUM [54] para uso através dos serviços de DNS, que consiste de: um método para representação de números E.164 em nomes de DNS; numa arquitetura de nomes reversos; e um mecanismo de resolução. Um número E.164 pode ser representado em um nome pela inversão do número separado dígito a dígito e complementado com o domínio ao qual pertence, então, por exemplo, um número E.164 +552143215678 teria a representação 8.7.6.5.1.2.3.4.1.2.5.5.*e164.arpa*, para o domínio público *e164.arpa*. Em termos de arquitetura, os domínios são delegados para entidades no nível onde estas possuírem o controle administrativo[19] de acordo com o plano de numeração Internacional e *Country Code*. A resolução de nomes ocorre por perguntas DNS para registros do tipo NAPTR [30], que retornam os URIs relacionados a este número. Do ponto de vista do SIP, os *Proxies* são elementos chaves no processo de resolução ENUM, auxiliando os *gateways* no direcionamento das chamadas oriundas da rede PSTN para o mundo IP quando os usuários não forem identificados diretamente pelos respectivos números E.164. No sentido contrário das chamadas, eventualmente os *Proxies* podem evitar que chamadas direcionadas a endereços E.164 atinjam a rede PSTN se um determinado usuário possuir um URI ativo na rede IP, associado ao endereço E.164 de destino.

3.11.2 Protocolo TRIP

O TRIP [42] surgiu pela complexidade na escolha de um determinado *gateway* quando do encaminhamento de chamadas destinadas ao mundo PSTN, num cenário onde há uma grande

[19] O domínio **5.5.e164.arpa**, por exemplo, é de interesse para o Brasil

diversidade de opções para rotas de saída. Vários aspectos podem afetar o julgamento do melhor ponto de saída, dentre eles: o custo da chamada, a preferência do entroncamento, políticas do provedor, políticas de usuário e acordos bilaterais entre provedores. Num domínio capaz de suportar TRIP deve haver pelo menos um *Location Server* (LS) responsável por propagar e importar as rotas telefônicas para auxílio aos servidores Proxy. De forma a permitir que um LS tenha conhecimento das possíveis rotas PSTN e dos respectivos *gateways* associados, o TRIP estabeleceu um mecanismo de troca de rotas entre LS's de domínios distintos, utilizando o protocolo BGP como meio de transporte, livre da ocorrência de *loops*. O TRIP também inclui mecanismos de troca de informações entre LS internos em um mesmo domínio. A especificação determina o formato para as mensagens OPEN, UPDATE, NOTIFICATION e KEEPALIVE do BGP, a forma de representação das rotas E.164[20] e técnicas de agregação de prefixos para reduzir o número de rotas enviadas a frente. As rotas em geral terão o formato "$N_1...N_n||X_1..X_m$", onde $N_1...N_n$ é o prefixo telefônico que pode ser atingido através de um LS, concatenado com $X_1..X_m$, sendo este qualquer número com exatamente m dígitos, simbolizado por m caracteres "ponto" consecutivos, p.ex: "2222....". Todos os prefixos devem estar associados a um *Next Hop*, podendo ser o endereço do *Proxy* ou *gateway* responsável para encaminhar as chamadas. Agregações podem fazer com que o *Next Hop* seja alterado para ocultar a diversidade à frente. Um aspecto importante sobre o TRIP é que este não faz referência de como um determinado LS toma ciência das rotas associadas aos *gateways* pertencentes ao mesmo domínio, ou seja, ele não especifica as forma de registro por parte do *gateway* dos troncos de voz associados ao próprio.

3.11.3 Protocolo TGREP

O protocolo TGREP [71] é uma iniciativa para complementar o TRIP, possibilitando automatizar o processo de registro dos troncos telefônicos coordenados por um *gateway* em um LS responsável pelo domínio. A especificação divide o papel do LS em dois: o *Ingress* LS e o *Egress* LS. O *Ingress* é responsável por coletar as informações dos *gateways* submetidos à mesma autoridade administrativa, ou seja, os prefixos telefônicos que podem ser atingidos por um determinado *gateway* e a respectiva proximidade[21]. O *Ingress* injeta no *Egress* as rotas, propagando-as para os demais domínios ou LS's do mesmo domínio, via o protocolo TRIP. O TGREP especifica o mesmo protocolo de roteamento, formato de

[20] Outras opções além do E.164 também são suportadas, devido a variações em determinados países.
[21] Em tese, qualquer número telefônico público pode ser alcançado por qualquer *gateway*; o fator proximidade entre em cena para computar o custo na escolha de um *gateway* sobre outro, pela região onde estão instalados.

mensagem, endereçamento e critérios de agregação daqueles existentes no TRIP, facilitando a interoperação. O TGREP acrescenta a definição de consolidação, que é reunir as rotas internas de um domínio para repasse aos demais, sem descontinuidade, diferentemente do que ocorre num processo de agregação. Para as chamadas IP oriundas de outros provedores, o TGREP é o elemento que vai permitir ao *Egress* decidir para qual *gateway* encaminhar a chamada.

3.12 SIP e NAT

SIP e NAT, de certa forma, são incompatíveis entre si. Basta observar as diversas mensagens exemplificadas neste texto para perceber que várias informações sobre endereçamento são passadas na camada de aplicação: seja nos campos do SIP, como o *Via*, *Contact*, *Refer-To* e outros, seja nos campos do SDP, como ponto de contato do RTP e porta. O efeito do NAT em SIP e possíveis soluções podem ser vistos no artigo [35].

3.13 Agente B2BUA

Um *back-to-back user agent* é um elemento previsto na RFC 3261 para operar entre dois agentes usuários (UA), repetindo todas as mensagens SIP e as mídias. Um B2BUA deve ser capaz de alterar os campos necessários do SIP de forma a posicionar-se no meio do caminho, processando tanto todas as mensagens SIP quanto a mídia trocada entre as partes. Em outras palavras, para um UAC de um terminal ele simula um UAS, e vice-versa. A Figura 3-16 apresenta um B2BUA entre dois terminais, mas nada o impede de operar entre um terminal e um servidor *Proxy*. Um B2BUA foi constituído com intuito de fornecer os seguintes serviços:

- Gerenciamento de chamadas, como bilhetagem, transferência e outros;
- Adaptação de protocolos na camada de transporte;
- Ocultar detalhes de uma rede privativa ou realizar NAT inclusive na aplicação;
- Transcodificação de formatos de áudio, quando há incompatibilidade entre as partes.

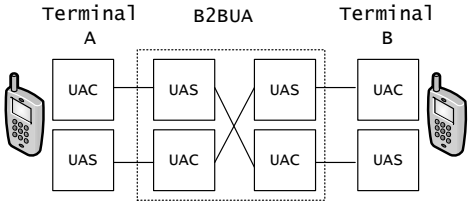

Figura 3-16- B2BUA

Capítulo 4

Aspectos de Segurança em VoIP

O SIP contém diversas funcionalidades de segurança que podem ser dispostas pelas aplicações de forma a conter algumas vulnerabilidades conhecidas para os sistemas de Voz sobre IP. Este capítulo apresenta uma síntese das vulnerabilidades possíveis e, em seguida, os controles de segurança que podem ser utilizados pelo protocolo. São dois os interesses na exposição deste tópico na dissertação: alguns destes controles, quando aplicados, poderão dificultar a execução dos atos de escuta legal; e o entendimento do funcionamento permitirá lançar proposições para conter ou superar esta contraposição.

4.1 Riscos e ameaças em Voz sobre IP

Esta seção apresenta algumas ameaças encontradas na literatura, em especial aquelas descritas por D. Richard Kuhn, et al. [22], e que podem sujeitar uma infra-estrutura de Voz sobre IP a riscos na segurança do serviço prestado. A existência de tais ameaças continua servindo como insumo na especificação dos controles de segurança aplicados em VoIP, que são analisados nas seções subseqüentes deste capítulo especificamente para o SIP.

Em geral, as ameaças são categorizadas através do balanço entre o custo, a complexidade, e a probabilidade da sua ocorrência contra o valor da informação a ser obtida ou o ativo a ser preservado. A ameaça deve levar em conta os potenciais agentes, como empregados de uma empresa, concorrentes, *script-kiddies* e outros. Esta relação pode levar a identificação, atenuação ou anulação de uma ameaça. As práticas de segurança recomendam avaliar o risco medindo a probabilidade de uma ameaça ser efetivada pelo valor da informação envolvida. Esta identificação é o primeiro passo na busca das respostas às perguntas básicas sobre o que

e como proteger, servindo como direcionador para o que deve ser realizado para prevenir e proteger os sistemas ora sendo implantados ou analisados.

O risco é a medida da exposição de alguma informação ou ativo à efetivação de uma determinada ameaça, balanceada com o valor da informação, desde que o sistema em questão possua a vulnerabilidade sobre a qual a ameaça ser perpetrada.

As redes de VoIP herdaram as vulnerabilidades[1] existentes do protocolo TCP/IP e acrescentaram outras que foram descobertas na medida do uso e da crítica da comunidade. Do ponto de vista das redes Internet, pela indeterminação do agente, qualquer vulnerabilidade pode representar um risco. Portanto, quando apontadas devem ser eliminadas ou mitigadas no menor tempo possível.

Das ameaças existentes para uma rede VoIP, podemos dividi-las em duas classes: aquelas que são herdadas do próprio TCP/IP e as específicas de VoIP. O NIST classificou as ameaças nas seguintes subclasses:

- **Sociais**. São aquelas que não estão ligadas a fatores técnicos, mas sim às formas de engenharia social, de responsabilidades das partes, e do uso inadequado dos recursos, como o recém surgido SPAM em correios de voz, ou SPIT [32].
- **Monitoração**. É a classe de ataques que permitam a interceptação parcial ou total da troca de sinalização e/ou mídia sem alterar o seu conteúdo, sem o conhecimento das partes. Nesta classe estão os ataques por captura de tráfego, registro ilegal de mídia e outros.
- **Interceptação e modificação**. Nesta classe estão os ataques que englobam as ações acima adicionando alterações no tráfego de sinalização ou mídia. Nesta classe estão exemplos como:
    - *Bloqueio de chamadas*. São técnicas que possibilitem o descarte ou recusa de qualquer parte do protocolo de forma a impedir a comunicação.
    - *Roteamento indevido de chamadas*. São as técnicas de redirecionamento de uma parte do protocolo de forma a desviar a comunicação para outro ponto.
    - *Alteração da conversação*. São as formas de alterar a mídia de forma a mudar ou alterar determinado conteúdo.
    - Degradação. Ocorre pela redução na qualidade de serviço.

[1] As vulnerabilidades em questão são aquelas intrínsecas da especificação do protocolo e não das ocasionadas pelas aplicações.

- *Despersonalização* e *Seqüestro*. São as formas onde a identidade de uma das partes é alterada, pela injeção, remoção, adição ou remoção de alguma parte do processo de comunicação. Pode ser também apenas alteração do status de uma das partes.
- *Uso indevido ou fraude de tarifação*. São as formas que tentam alterar a bilhetagem em alguma forma, ou usar serviços privilegiados de rede sem a devida autorização.

- Interrupção de Serviço. Nesta classe encontram-se as tentativas em gerar indisponibilidade atuando sobre alguma característica das redes e não de usuários específicos. Exemplos são o DoS, DDoS, Intrusão, Consumo de Recurso, ataques a infra-estrutura, como os sistemas de energia.

Para mitigar os riscos acima o NIST apresenta uma série de recomendações, que estão sumarizadas abaixo:

- Adequação da Rede, através das seguintes ações:
  - É recomendado o uso de uma rede segregada para disponibilização dos serviços de voz.
  - Uso de *firewall*s com suporte de um ALG.
  - Uso de SSH para acesso ao gerenciamento.
  - Retirar os acessos para gerenciamento remoto dos agentes usuários (UA), baseados no protocolo HTTP.
  - Estudo dos riscos envolvidos e da classificação do sistema de informação. Antes de realizar investimento na ampliação do grau de segurança ofertado dentro de um domínio administrativo, a primeira ação é avaliar quais são os riscos envolvidos e quais os ativos devem ser protegidos ou preservados, de forma a direcionar esforços e investimentos.
  - Implementação de sistemas *backups*.
- Uso de técnica de seguranças nos sistemas adicionais como, por exemplo, a aplicação de WPA em redes *Wireless*.
- Uso intensivo das recomendações de segurança dos protocolos de Voz sobre IP.

O uso intensivo das recomendações de segurança dos protocolos de Voz sobre IP, que naturalmente dependem do protocolo em questão, serão exploradas nas próximas seções especificamente para o protocolo SIP. Ao mesmo tempo em que elas cumprem um papel

fundamental em evitar a exploração indevida e maliciosa das informações e recursos, elas também podem servir como impeditivos na aplicação do recurso legal de interceptação e interpretação.

4.2 Serviços de autenticação do SIP

O SIP prevê mecanismos de autenticação da sinalização do registro e dos métodos usados para estabelecer, alterar e encerrar uma chamada, que são descritos na RFC2617 [39]. O esquema mais básico de autenticação usado é o *digest authentication*, através do paradigma desafio-resposta. Neste esquema, ao realizar uma requisição, o servidor de registro ou o *Proxy*[2] retornam uma mensagem da classe 4XX, solicitando uma autenticação, colocando em um campo[3] um desafio contendo um valor *nonce* e o *realm*. O *nonce* serve como um fator aleatório para evitar que a mesma resposta em múltiplas autenticações, o *realm* possui um valor alfanumérico em geral idêntico ao nome qualificado do servidor. O UA computa um valor de retorno sobre o desafio e retorna a requisição para o referido servidor, incrementando o contador de transação, CSeq, da mensagem. A determinação do valor de retorno é computada segundo, na forma simplificada, pela RFC2617 [39], conforme o seguinte:

```
request-digest = <">< MD5( MD5(A1), nonce":" MD5(A2) ) ><">
A1       = username ":" realm-value ":" password
A2       = Method ":" URI
```

Onde *nonce* é o valor enviado pelo servidor ao cliente, único por requisição, *realm* é o domínio ao qual o *Proxy/Register* pertence e a senha é uma informação secreta compartilhada somente entre as partes.

A Figura 4-1 apresenta as trocas de mensagem entre as partes para um registro.

[2] O *Proxy* retorna uma mensagem do tipo "`407 Proxy Authentication Required`". O *Register* retorna uma mensagem do tipo "`401 Unauthorized`".
[3] O *Proxy* utiliza o campo "`Proxy Authenticate`" e o *Register* o campo "`WWW-Authenticate`", para enviar o desafio.

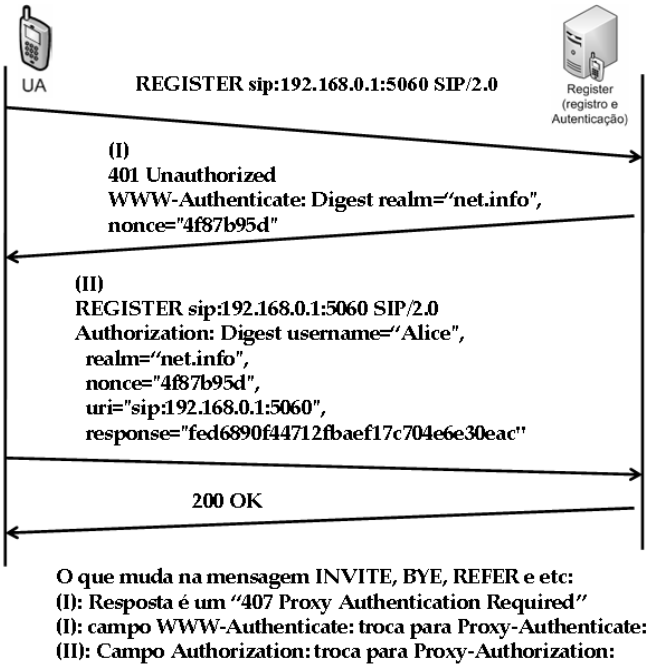

Figura 4-1 - Autenticação de usuário

Conforme dito acima, o SIP define que todas as transações de registro e as que podem alterar uma chamada podem ser autenticadas, isso inclui os métodos REGISTER, INVITE REFER, BYE e outros, mas exclui as requisições ACK e CANCEL, gerando uma exposição a alguns tipos de ataque. É recomendável que a sinalização seja protegida de forma complementar, com mecanismos como o TLS ou o DTLS, que serão vistos adiante.

O esquema de autenticação *Digest* está sendo utilizado pela totalidade dos fabricantes de terminais SIP, sendo atualmente o método efetivamente empregado para autenticar o UA.

4.3 Protocolo TLS

O TLS, definido na RFC4346 [64], é baseado no antigo SSL, que foi desenvolvido pela Netscape. Ele utiliza o protocolo TCP ou o SCTP para transporte e hoje está associado com o uso de serviços de Web seguros como o HTTPS. O TLS acrescenta duas camadas no topo do TCP, a primeira é a camada de transporte do TLS, para envio de mensagens cifradas, e a segunda refere-se ao protocolo que estabelece o handshake prévio. O TLS assegura a privacidade, integridade e protege as partes contra ataques do tipo *replay*[4]. A parte referente ao handshake é usada para estabelecer a conexão, negociar o esquema de criptografia e

[4] Ataques do tipo *replay* são efetuados pela repetição de informações capturadas na rede, originais ou levemente modificadas, rumo ao destinatário. Números de seqüência podem ser forjados para simular a continuidade na troca da mensagem em tempo real ou as mensagens podem ser armazenadas para envio posterior. Ataques deste tipo podem objetivar uma autenticação forjada ou forçar repostas dedutíveis do destinatário, que possam favorecer ataques contra sistemas com criptografia.

autenticar os dois lados. Durante a negociação as partes podem trocar certificados digitais que serão usados para autenticação – obrigatório ao menos para o servidor. A computação da chave de criptográfica de sessão envolve várias etapas e diversos pacotes percorrendo as duas direções, podendo causar alguma latência no processo. A figura abaixo mostra as mensagens sendo trocadas. O KDF é uma função de derivação de chaves e MAC o código de autenticação da mensagem – ou hash.

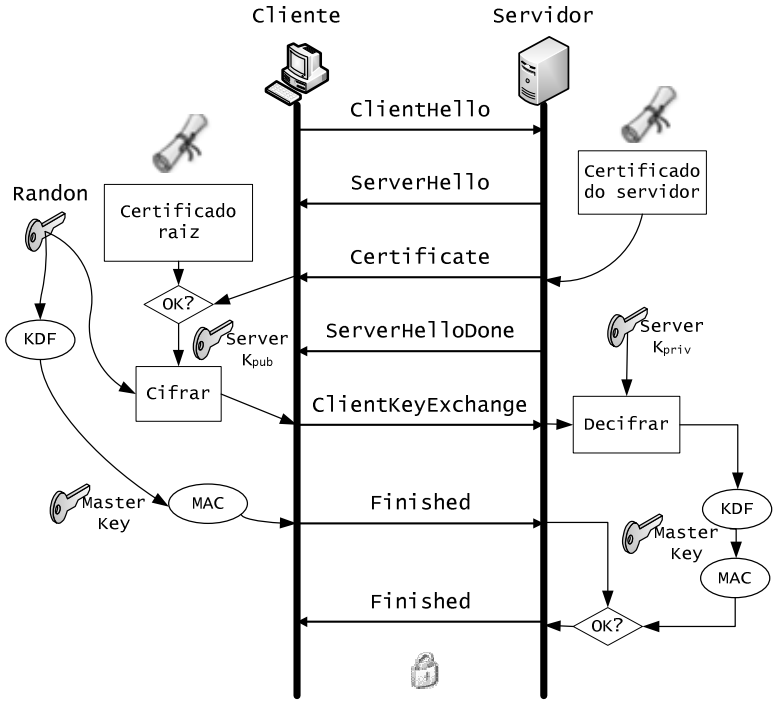

Figura 4-2 - ***Handshake*** **do TLS**

A RFC3261 recomenda o uso do TLS para *Proxies*, redirecionadores e registradores, protegendo a sinalização entre os terminais contra quebra de confidencialidade, integridade e ataques do tipo *replay* e do envio de mensagens de requisição não autenticadas, que possam provocar algum tipo de negação de serviço, *call blocking* e demais ameaças pertinentes.

Um ponto contra o uso do TLS é a necessidade do provimento de uma infra-estrutura de chaves públicas e o aprovisionamento destas chaves nos terminais, podendo levar a um problema de escala e custo. Alguns fabricantes utilizam certificados apenas nos servidores e autenticam o UA por *digest*, operando conforme o fluxo de mensagens da Figura 4-2. Para uso do TLS o assinante deve sinalizar sua chamada como o esquema "`sips`:" na *Request-URI*.

Um ponto óbvio, mas importante, é que o TLS realiza proteção salto a salto da sinalização. Assim, havendo mais de um *Proxy* no caminho da sinalização, será necessário que os *Proxies*

intermediários estabeleçam entre si canais seguros, sob pena de a chamada falhar. Outro aspecto é que todas as mensagens de sinalização estarão disponíveis internamente aos *Proxies* no trajeto, tanto para apreciação quanto para alteração do que for necessário[5]. A Figura 4-3 mostra uma representação de uma chamada com múltiplos *Proxies* via TLS.

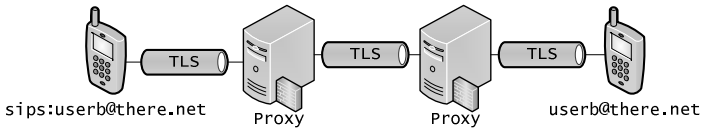

Figura 4-3 - Chamada com esquema "sips:"

4.4 Protocolo DTLS

O DTLS, definido na RFC 4347 [65], foi desenvolvido para prover o mesmo nível de proteção para o protocolo de transporte UDP. O DTLS é similar ao TLS em quase tudo, incluindo a necessidade de estabelecimento de sessões salto a salto. A diferença fundamental ocorre pela maior tolerância do DTLS a perdas de pacotes.

4.5 O uso do S/MIME

Herdado do SMTP, o SIP adotou o esquema MIME e S/MIME, permitindo a proteção e integridade da mensagem, podendo incluir parte do cabeçalho SIP e o corpo SDP. Quando presente, o S/MIME é indicado pela presença dos tipos MIME *multipart/signed* e *application/pkcs7-mim.* Diferente do TLS o S/MIME é capaz de prover confidencialidade fim a fim para as informações críticas.

A aplicação de S/MIME pode ocasionar dificuldades para elementos intermediários que desejam inspecionar o SDP, em especial equipamentos do tipo *firewall* que possam estar realizando abertura seletiva de portas RTP e/ou tradução de endereços NAT também na camada de aplicação, conforme seção 3.12.

4.5.1 Certificados com S/MIME

O S/MIME requer o uso de certificados digitais emitidos para os usuários, isto é, que eles sejam associados ao URI de usuário, aumentando a complexidade e custo da infra-estrutura.

[5] Exemplo: para cumprir a inclusão do campo Via, obrigatoriamente. É obrigatório que os campos Request-URI, CSeq, Via, Contact, *To*, *From* e Call-ID estejam visíveis ao *Proxy*.

Alternativamente, os usuários devem dispor de um chaveiro para armazenar as chaves dos interlocutores de interesse ou ter acesso a algum serviço de diretório para obtenção dos certificados.

Não havendo um serviço de diretório, o SIP pode ser utilizado para propagar os certificados entre as partes, anexando o certificado no padrão X.509[6] com a respectiva chave pública, utilizando um anexo assinado com S/MIME. Para envio dos certificados são permitidos o uso das transações de INVITE, no início da chamada, ou OPTIONS, paralelamente ao INVITE.

O UAC originador deve enviar, na requisição de início da transação de troca de certificados, o corpo da mensagem com um S/MIME *multipart/signed*. Por sua vez, o UAS do receptor deve verificar a validade do certificado utilizando o certificado raiz instalado previamente. Sendo válido o receptor ainda compara se o certificado foi emitido em nome do URI que consta no campo *From*[7]. Se for diferente, a recomendação é que o usuário presente no receptor seja perguntado se deseja continuar a chamada – assumindo o risco. Sendo igual, o terminal receptor deve enviar o seu certificado para a primeira parte, procedendo da mesma forma[8]. Se tudo correr bem, ambos podem armazenar os certificados nos seus respectivos chaveiros, para futuras transações.

O S/MIME também pode ser usado para garantir a integridade e/ou privacidade do cabeçalho SIP, inclusive omitindo alguns dos campos do cabeçalho SIP que não seriam de interesse dos elementos intermediários. A RFC 3261 recomenda que os únicos campos disponíveis para processamento pelos *Proxies* sejam: *To*, *From*, Call-ID, Cseq, Contact e o Via. A intenção de cifrar parte do cabeçalho é preservar informação de interesse apenas por parte dos usuários finais, como *Accept-Language*, *Subject* e outros.

A figura abaixo ilustra um INVITE protegida com S/MIME.

```
INVITE sip:bob@b.example.com SIP/2.0
To: <sip:bob@b.example.com>
From: <sip:alice@a.example.com>;tag=4bba1f0d
Via: SIP/2.0/UDP
     127.0.0.1:5070;branch=z9hG4bK-c87542-558422834-1--c87542-;rport
Call-ID: 132bb895019d4536
CSeq: 1 INVITE
Contact: <sip:alice@a.example.com:5070>
Max-Forwards: 70
Content-Disposition: attachment;handling=required;filename=smime.p7
```

[6] X.509 é um padrão para certificados digitais, encontrado em http://www.itu.int/rec/T-REC-X.509/en

[7] Por isto o certificado deve ser emitido em nome do AOR, ou seja, para o exemplo da Figura 4-4, O *Common Name* do Certificado deve ser em nome de CN=bob@b.example.com

[8] Numa mesma transação o campo *From* e *To* não podem ser invertidos, então quando o terminal chamado envia seu certificado ao chamador este deve comparar se o certificado foi emitido para a mesma URI que consta no campo "*To*".

```
Content-Type:
    application/pkcs7-mime;smime-type=enveloped-data;name=smime.p7m
User-Agent: SIPimp.org/0.2.2 (curses)
Content-Length: 385

*****************
*    BINARY     *
*****************
```

Figura 4-4- Exemplo de SDP protegido com S/MIME[9]

4.6 Protocolo IPSEC

O IPSEC é um mecanismo de segurança ponto a ponto que atua na camada de rede, prestando serviços de integridade e privacidade quando necessário às aplicações, tendo sido especificado na RFC4301 [63]. O IPSEC adota o protocolo IKE, descrito na RFC 4306 [62], para autenticação e troca das chaves criptográficas entre as partes interessadas em estabelecer uma comunicação segura. A autenticação pode ser realizada tanto com uma chave compartilhada quanto com certificados digitais. O próprio IKE encarrega-se de transportar os certificados dispensando o uso de serviços de diretório, mas quando necessário, é prevista a consulta as listas de certificados revogados[10]. Como atua na camada de rede, o IPSEC pode ser empregado em qualquer tipo de aplicação e tem sido muito utilizado na criação de redes virtuais privativas (VPN) sobre a Internet. VPN's com IPSEC são caminhos criados, conectando dois elementos em geral em redes diferentes, como se estivessem conectados diretamente um ao outro, por onde o tráfego de interesse será direcionado de forma protegida.

O uso do IPSEC cria um obstáculo virtualmente intransponível à possibilidade de interpretar as mensagens por ele protegidas, inviabilizando estratégias de LI em sistema de VoIP. Por outro lado, há uma série de restrições no uso do IPSEC em soluções de Voz sobre IP, principalmente relativas ao IKE, considerado complexo e lento para aplicações multimídia[11].

4.7 Mecanismo de criptografia nativo do RTP

O protocolo RTP [69] prevê um mecanismo de criptografia nativo na sua especificação, que permite a confidencialidade de ambos RTP e RTCP. Todos os octetos do RTP, incluindo

[9] Fonte: http://tools.ietf.org/html/draft-jennings-sip-sec-flows-03

[10] Listas de certificados revogados são relações publicadas pelas entidades certificadoras responsáveis pela emissão de certificados, de forma semelhante ao processo adotado nos antigos catálogos de cartões de crédito revogados.

[11] Há uma proposta de padrão em andamento na IETF que comenta o problema da complexidade e latência do IKE e propõe uma forma alternativa de negociação. http://ietfreport.isoc.org/idref/draft-saito-mmusic-ipsec-negotiation-req/.

o cabeçalho, podem ser criptografados. A desvantagem é a eliminação da capacidade em compactar o cabeçalho RTP[12].

O algoritmo padrão para uso na criptografia é o DES-CBC [24]. Quando o RTP foi projetado, imaginava-se ser suficiente o uso do DES. Mas, o avanço na capacidade computacional o tornou insuficiente para prover confidencialidade. É recomendado que as implementações, além de suportarem o DES - obrigatório pela RFC como nível básico de compatibilidade, suportem outro algoritmo mais robusto, como o AES [26].

O RTP não define um mecanismo para troca de chaves de criptografia. Esta atividade é de responsabilidade do protocolo de estabelecimento de sessão.

O esquema da codificação do RTP pode ser visto na Figura 4-5. O quadro mais a esquerda mostra o pacote como seria antes da criptografia nativa do RTP; ao centro o pacote com o enchimento necessário para permitir a operação pelo DES-CBR; à direita a parte hachurada representa a parte do pacote criptografada.

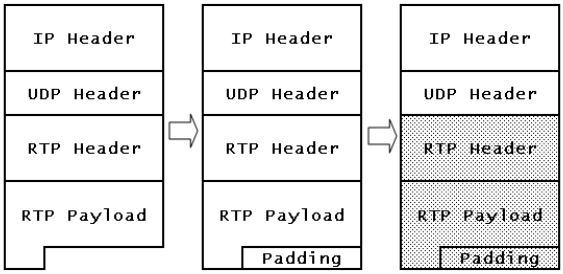

Figura 4-5 - Criptografia para o RTP

No caso do RTCP, é prevista a inserção de um vetor de inicialização (IV), ou "random prefix", de forma a evitar a exposição a ataques sobre a parte inicial do material, que possa ser previamente conhecido. A Figura 4-6 apresenta, na parte hachurada, o RTCP ao final do processo critpográfico.

[12] Casner, S.;Jacobson, V. Compressing IP/UDP/RTP Headers for Low-Speed Serial Links, RFC 2508, February 1999.

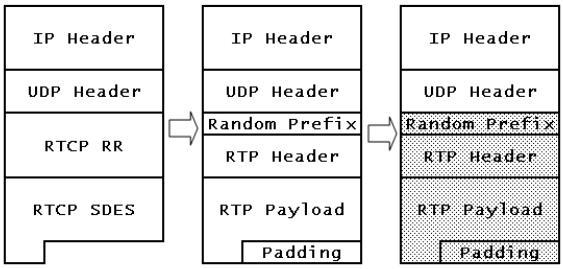

Figura 4-6 - Criptografia para o RTCP

Na prática, o esquema nativo de criptografia previsto pelo RTP deu vazão a uma nova especificação, conhecida por SRTP, descrito na seção seguinte.

4.8 O protocolo SRTP e o SRTCP

O Secure RTP (SRTP), descrito na RFC3711 [53], estabelece um novo mecanismo proteção da mídia, provendo tanto autenticidade, integridade e confidencialidade ao RTP. O SRTP foi desenhado de forma a operar mesmo em situações onde os circuitos no trajeto da mídia possuam relativamente alta taxa de perdas de pacotes, onde seja necessário realizar compressão do cabeçalho RTP[13] e com baixo custo computacional. Para o SRTP operar é necessário definir previamente aquilo que a RFC chama de contexto criptográfico, sendo formado pelo conjunto das chaves mestras, *salting* e alguns outros dados que irão compor o estado do processo de criptografia. O SRTP é acompanhado de dois contextos padronizados e descreve como outros contextos podem ser acrescentados posteriormente. Nos contextos especificados, foi adotado o algoritmo AES para realizar a criptografia do *payload* RTP e RTCP, gerando uma proteção maior na confidencialidade da informação, relativo ao mecanismo anteriormente existente. O segundo contexto opera no modo NULL, que significa não alterar o RTP.

De forma a permitir que o SIP pudesse negociar o contexto criptográfico, através do protocolo SDP, os autores do SRTP registraram no *Internet Assigned Numbers Authority* (IANA) o perfil RTP/SAVP para uso pelo protocolo SDP, através do parâmetro *transport* do campo “`m=`”.

Um dos pontos positivos do SRTP reside em sua complexidade computacional pequena, permitindo que seja implementado tanto em software quanto em *hardware*, de forma que os

[13] O método de criptografia original do RTP incluía o cabeçalho, eliminando a possibilidade de comprimi-lo.

dispositivos portáteis possam aplicá-lo em software sem um significativo aumento na carga de processamento e, conseqüentemente, no consumo de energia.

O SRTP provê criptografia apenas protegendo o *payload* do RTP, conforme Figura 4-7. Do ponto de vista funcional, o SRTP intercepta os pacotes RTP criados antes do envio para o UDP de forma a acrescentar a confidencialidade, integridade e autenticidade nos pacotes RTP e o mesmo é feito na recepção. Do ponto de vista da aplicação o SRTP opera de forma transparente.

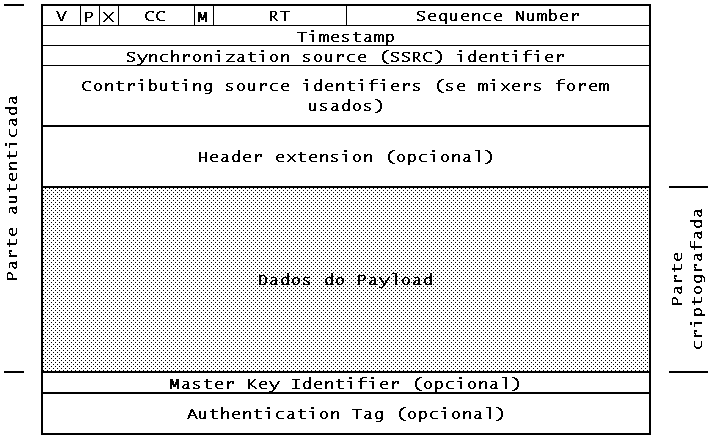

Figura 4-7 - Quadro SRTP

Nos contextos atualmente definidos, o processo de criptografia de um quadro RTP ocorre apenas pela operação XOR da informação presente no *payload* do pacote com uma chave conhecida como *key stream* (K_s), gerada com o mesmo tamanho, não ocasionando acréscimo no tamanho final do pacote SRTP, em relação ao RTP. O único acréscimo ocorre pela inclusão dos campos opcionais *Master key identifier* e *Authentication tag*. Esta estratégia também dispensa o uso de *bits* de enchimento comumente aplicados nos algoritmos que operam em blocos.

Há dois contextos criptográficos pré-definidos e que devem ser suportados por todos os elementos que utilizam o SRTP: o primeiro é pelo uso de AES com autenticação e integridade usando o SHA-1 com 160 bits de comprimento. O segundo é pelo uso do *NULL stream*, ou seja, a informação não é criptografada[14]. Com uso do AES, é permitido que ele opere em dois modos: o modo contador e o modo `f8`.

O SRTP inclui na sua especificação um algoritmo de derivação de chaves sobre uma chave mestra, não precisando de protocolo auxiliar para esta função. Este algoritmo é aplicado

[14] A operação lógica ou-exclusivo (XOR) de um bit com zero resulta no próprio bit.

para evitar que excessivo material seja cifrado com a mesma chave, gerando ocasionalmente uma chave de sessão que será utilizada no processo de proteção da carga. O contexto define qual será a taxa de troca de chaves derivadas, mensurada pelo número limite de pacotes cifrados com a mesma chave de sessão. O mecanismo de derivação é tal que de posse da chave sessão não é possível determinar a chave mestra.

O cabeçalho SRTP e quaisquer extensões são enviados sem criptografia, e respeitam as definições do RTP, visto na seção anterior, permitindo sua compressão, ao contrário do padrão de criptografia nativo, descrito na seção 4.7, pág. 63. Da mesma forma que o modo nativo do RTP, a chave de criptografia deve ser obtida por intermédio do protocolo de estabelecimento de sessão. O campo *Master key identifier* pode ser usado para enviar informações sobre a chave mestra atualmente utilizada, quando for fornecida mais de uma pelo processo de gerenciamento de chaves.

Ambos o transmissor e o receptor devem manter um contexto de criptografia, que consiste em:

a. Conhecer os algoritmos criptográficos e de assinatura a serem utilizados, que nos contextos pré-definidos consistem do AES e o SHA-1;

b. A chave mestra, que deve ser estabelecida pelo protocolo de estabelecimento de sessão, no caso o SIP;.

c. A chave usada para *salting*, conforme será visto adiante;

d. Um vetor de inicialização, IV, apenas para uso com o modo f8 do algoritmo AES, como será visto a frente, definido pelas informações do cabeçalho RTP

e. Um contador de pacotes enviados `I,` de 32 bits, com o número de vezes que o contador de seqüência do pacote RTP (o campo *sequence number de 16bits)* transbordou. Desta forma `I = 2^16 * ROC + SEQ, onde ROC` conta os transbordos (*rollover counter*);

f. A taxa de derivação da chave de sessão. Para evitar o acúmulo de material criptografado com a mesma chave, o SRTP prevê um mecanismo de gerenciamento de chaves intermediárias, que são geradas baseadas na chave mestra. Desta forma, a revelação de uma chave derivada não ocasionará na decifração das demais mensagens, nem na descoberta da chave mestra. Porém, o acesso à chave mestra é suficiente para derivar as demais chaves.

O processo de criptografia ocorre nas etapas:

i. O sistema é iniciado com uma ou mais chaves mestras (M), supridas por outro protocolo que não o SRTP. Originadas da chave mestra, chaves de sessão (Z_n) são derivadas pelo resultado da aplicação de uma função pseudo-aleatória, numa taxa definida pelo contexto. Desta forma, o SRTP determina sobre quantos pacotes uma mesma chave de sessão poderá ser usada. Um ponto importante é que caso existam mais que uma chave mestra suprida ao SRTP, as partes deverão sinalizar qual chave mestra está sendo usada para derivar a chave de sessão correta, via o campo *Master key identifier*, sincronizando as partes;

ii. É determinada uma chave de seqüência, *key stream*[15] (K_s), que é determinada pela computação do número de seqüência do pacote RTP, do valor *salting* pré-estabelecido, e a chave Z_n;

iii. O *payload* do RTP é cifrado numa operação *bit* a *bit* XOR com K_s e transmitido na rede.

A computação de K_s é realizada pelo AES e depende da forma em que ele opera. Há duas formas previstas: o modo contador e o modo f8.

No modo contador, é gerado um valor intermediário IV de 128bits, pela operação `IV = (salting * 2^16) XOR (SSRC * 2^64) XOR (I * 2^16)`. Este valor, por sua vez, é utilizado para determinar a chave Z_n, com base na função `AES(A,B)`. Onde `AES()` é a operação do algoritmo AES utilizando a chave A sobre a mensagem B. I é o contador de pacotes definido no item "e".

Desta forma K_s, é computado com comprimento igual ou superior ao *payload* RTP, tendo comprimento múltiplo de 128 bits, pela fórmula: K_s = [AES(Z_n, IV) || AES(Z_n, IV + 1 mod 2^128) || AES(Z_n, IV + 2 mod 2^128) ...].

Quando operando no modo contador, é importante assegurar que cada pacote seja criptografado com uma única K_s, durante toda a vida da sessão. A presença do campo SSRC e o número de seqüência com transbordo são fundamentais para este fim.

No modo f8, o valor K_s é gerado da seguinte forma:

- Determina-se o resultado de $Z_n \oplus$ Salt;

[15] O objetivo é que cada pacote seja cifrado com uma chave que não será mais repetida, aproximando-se da abordagem "on-time-pad" [24].

- É gerado um Vetor de Inicialização, pela computação de: `IV = 0x00 || M || PT || SEQ || TS || SSRC || ROC,` onde `M, PT, SEQ, TS e SSRC` são tirados do cabeçalho RTP e `ROC` sendo o *rollover counter*;
- Utiliza-se o valor $Z_n \oplus$ `Salt`, como chave do AES para cifrar o vetor de inicialização (IV);
- O resultado é chamado de vetor de inicialização interno e é utilizado para gerar os blocos de 128 bits, em numero suficiente para ultrapassar o tamanho do *payload*;
- Cada bloco ainda passa pela uma operação XOR com um contador de 128 bits, J, inicializado em zero e incrementado a cada iteração;
- O primeiro bloco é cifrado pelo AES usando a chave Z, e os demais são alimentados pelo bloco anterior, conforme exemplifica a figura abaixo. Até que K_S seja superior ao tamanho do *payload*.

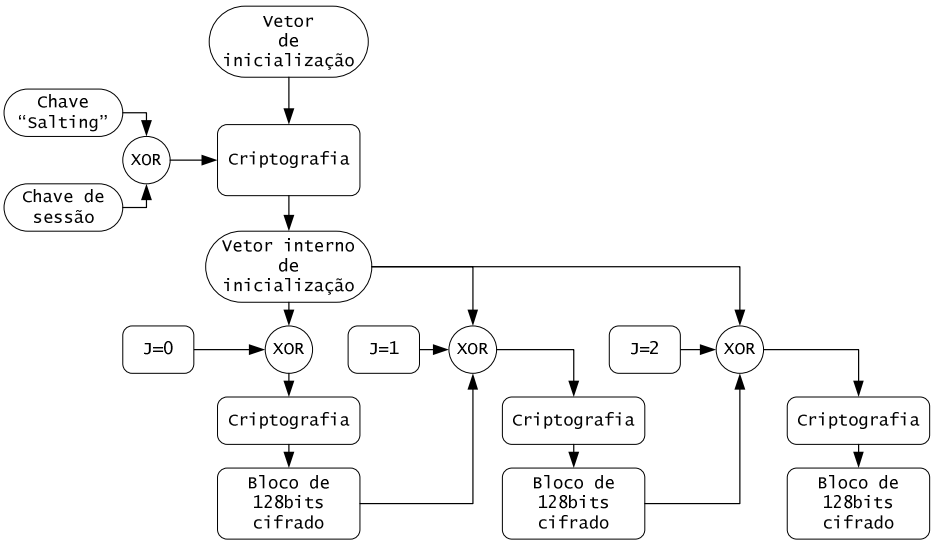

Figura 4-8 - AES no Modo f8 [16]

A escolha do uso do modo contador ou do modo `f8` deve ser feita pelo protocolo de inicialização da sessão, via o mecanismo de gerenciamento de chaves. O modo contador possui um desempenho melhor em circunstancias de taxa de erro elevadas, e é o método padrão.

[16] Fonte: http://www.packetizer.com/rfc/rfc3711/

O SRTP ainda provê mecanismos de proteção do RTCP, conhecido como SRTCP. Os primeiros 64 bytes permanecem intactos, conforme Figura 4-9. Um bit “E” visto na Figura 4-9, é utilizado para sinalizar a presença de criptografia do *payload*.

O processo de criptografia do SRTCP é similar ao SRTP, com a única exceção na troca do campo “SRTCP index” no lugar do número de seqüência. Da mesma forma que o SRTP, a chave mestra M deve ser apontada pelo protocolo de sessão.

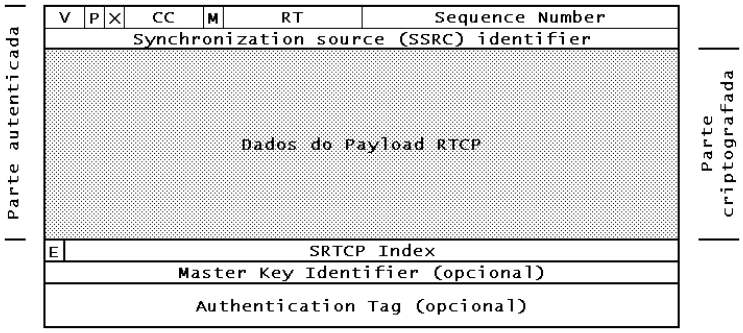

Figura 4-9 - O quadro SRTCP

Finalizando, a autenticação do SRTP e o SRTCP permitem aos protocolos atestar a integridade dos pacotes e evitar ataques do tipo “*replay*”. A autenticação ocorre pela aplicação do algoritmo HMAC-SHA1, definido na RFC 2104, que produz um resultado de 160 bits truncados em 80 bits, sendo seguidamente estampado no campo “authentication data”. O HMAC é calculado sobre o *payload* e o cabeçalho, incluindo o número de seqüência de pacote, ou o SRTCP *index*, gerando o mecanismo de integridade e proteção *anti-replay*.

A Figura 4-10 apresenta o processo SRTP como um todo, atuando sobre o RTP.

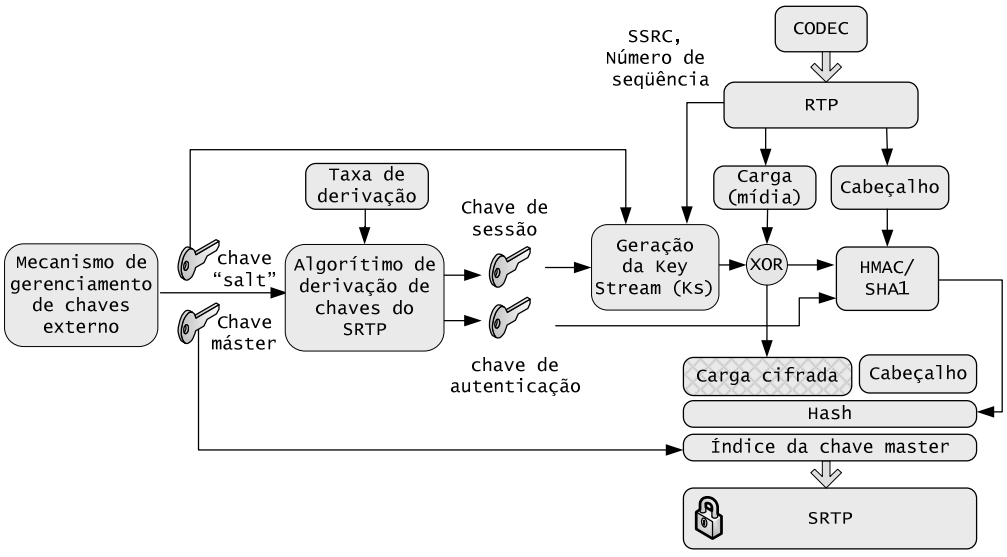

Figura 4-10 - Ilustração do Processo SRTP

Capítulo 5

Gerenciamento de chaves de sessão

O objetivo nesta seção é apresentar os mecanismos existentes para estabelecimento e manutenção das chaves mestras e o contexto que serão usadas pelo protocolo SRTP, ou seja, os mecanismos de gerenciamento de chaves. Não há uma definição universal para um método único. Havendo algumas formas de se realizar esta função e que devem ser aplicadas de acordo com o cenário em análise, oferecendo, cada uma, vantagens e desvantagens.

Até o presente momento vimos que há três formas de prover confidencialidade a mídia: o uso de IPSEC, o uso do recurso nativo de criptografia do RTP e o novo SRTP. Os métodos em VPN possuem em sua especificação mecanismos próprios estabelecer chaves de sessão[1], porém eles não serão discutidos neste trabalho, a menos na conclusão. O protocolo nativo RTP não é empregado na prática em função da enorme vantagem do SRTP.

Para o caso do SRTP, até o presente momento, as seguintes opções estão disponíveis:

- Uso de chaves estabelecidas dinamicamente por intermédio do protocolo de gerenciamento de chaves MIKEY;
- Uso do protocolo de sessão para passagem do contexto de criptografia, que no caso SIP, ocorre pela extensão SDES do SDP;
- Uso do protocolo ZRTP.

Nas próximas sessões as opções acima serão detalhadas.

[1] Para o IPSEC o protocolo IKE e para o SSL, o SSL *handshake protocol*.

5.1 Protocolo MIKEY

O MIKEY [55] foi uma iniciativa da IETF para produzir um protocolo de gerenciamento de chaves de sessão para ser utilizado nas aplicações multimídia, emergentes, de forma a permitir o estabelecimento de chaves mestras. Até aquele momento o IKE, utilizado pelo IPSEC, o TLS-Handshaking e demais trabalhos, na consideração do MMUSIC[2], não comportavam as necessidades das aplicações multimídia[3]. O SDP, naquela ocasião ainda não havia sido estendido para suportar a negociação de chaves via SDES e o campo "k=" existente mostrava-se inadequado e inextensível.

O MIKEY é independente da funcionalidade do protocolo de criptografia ao qual ele presta serviço, podendo estabelecer qualquer conjunto criptográfico. O MIKEY pode ser transportado por outros protocolos, como por exemplo o SDP. Para transportar e trocar informações sobre o material criptográfico o MIKEY pode utilizar três metodologias: *pre-shared key* (PSK), PKI e o Diffie-Hellman [38].

A RFC3830 categoriza o PSK como o processo mais eficiente para o estabelecimento de chaves, e dispensa uma infra-estrutura de chaves públicas para distribuição de certificados entre os diversos agentes. O Diffie-Hellman [38], dos três, é considerado o processo mais complexo computacionalmente e devido a sua característica de gerar chaves que não possam ser dedutíveis[4] a partir chave mestra, acaba não sendo adequado ao uso em sessões de grupo. A abordagem PKI requer o uso de certificados digitais, podendo afetar o custo e a complexidade na gestão da infra-estrutura.

5.1.1 Definições para o protocolo MIKEY

Algumas definições foram estabelecidas e são necessárias para entendimento do protocolo, são elas

- Protocolo de Segurança de Dados (PSD): é o protocolo que irá proteger o dado, como o IPSEC ou o SRTP;
- Data security association (*Data* SA): informações mantidas pelo PSD, contendo chaves, como o TEK e um conjunto de parâmetros e políticas, dependente do PSD;

[2] MMUSICMulti-Party Multimidia Session Control Working Group
[3] As opções naquele momento eram consideradas como tendo excessiva latência no processo de estabelecimento de chaves e não suportavam criação de chaves para grupos, por multicast.
[4] Esta é a característica PFS – Perfect Forward Secrecy.

- Sessão criptografada (CS): é um fluxo de dados unidirecional, protegido por uma única instancia de um SP, contendo um "Data SA";
- Chave criptográfica de dados (TEK): é chave usada pelo PSD para proteger os dados, ou usada para derivar chaves internamente ao protocolo do PSD;
- Coleção de sessões (CSB): é um conjunto de CS que possuem o mesmo conjunto de chaves TGK e parâmetros;
- Chave de Geração de TEK (TGK): é uma chave acordada entre as partes, associadas a um CSB. Da TGK a TEK pode ser originada, sem nenhuma necessidade de comunicação entre as partes;
- TGK *re-keying*: é o processo de renegociação da TGK, e conseqüentemente da TEK;
- *Salting key*: uma string randômica usada para proteger contra ataques tipo dicionário, que possa eventualmente ocorrer em um PSD;
- *Initiator* (I) e *Responder* (R) são as partes envolvidas no processo MIKEY;
- Há ainda identificadores para sessão (CSID) e para o conjunto (CSBID).

O MIKEY opera em etapas, conforme Figura 5-1, sendo que a fase mais ao topo é dependente do método adotado e será explorada nas próximas sessões. As seguintes da figura, realizam a derivação da TEK e de formação do *Data* SA, e não serão tratadas neste texto por não afetarem as proposições do Capítulo 7.

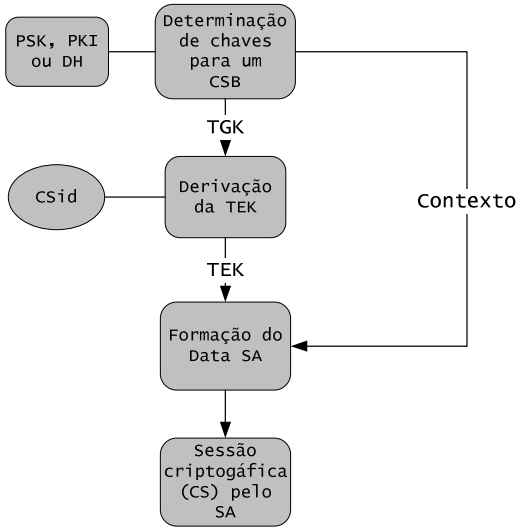

Figura 5-1 - Etapas do MIKEY [5]

[5] Fonte: http://tools.ietf.org/html/rfc3830

5.1.2 Determinação de um CSB por PSK

Este método é o mais simples de todos e compreende apenas a configuração estática das chaves de sessão nos terminais. Tal chave é utilizada para autenticar as mensagens trocadas entre as partes e para estabelecer uma chave de sessão, num processo contendo apenas duas trocas de mensagem, como visto abaixo.

Para descrição do processo, a seguinte definições são empregadas:

- HDR: O cabeçalho da mensagem MKEY, incluindo o CSBid e informações específicas para o SP;
- T: o *timestamp* do lado do *Initiator*;
- ID_x: a identidade de x | x=i⇒*initiator*, x=r⇒*responder;*
- RAND: um número gerado de forma pseudo-aleatória para compor uma incerteza na geração do MAC;
- SP: políticas e parâmetros necessários pelo SPD;
- MAC(K,M): um campo de integridade sobre a mensagem M, gerado pela chave de autenticação K;
- K_{psk}: a chave conhecida previamente por ambas as partes;
- encr_key: a chave de criptografia da mensagem, derivada de K_{psk};
- auth_key a chave usada para autenticar a mensagem, derivada de K_{psk};
- PRF(K, n) a função pseudo-aleatória que gera uma enésima chave sobre K;
- KMAC: é a parte protegida do pacote, ou *payload*, contendo um ou mais TGK's, sendo composto por:

$$K_{enc\_key} = PRF(K_{psk}, n)$$

$$K_{auth\_key} = PRF(K_{psk}, n+1)$$

$$KMAC = K_{enc\_key}[TGK] \,||\, MAC(K_{auth\_key}, HDR)]$$

O funcionamento está ilustrado figura abaixo. O *handshake* ocorre apenas pela troca de duas mensagens: I_MESSAGE e R_MESSAGE.

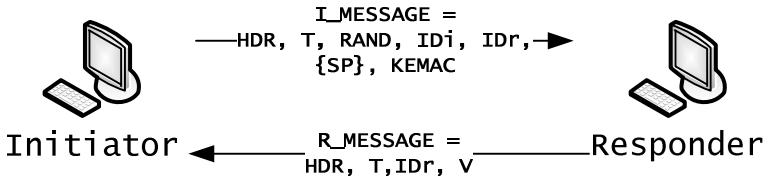

Figura 5-2 – CSB por PSK

No retorno, o responder adiciona o campo V, que é um MAC gerado sobre toda a mensagem com a chave auth_key, derivada pelo mesmo processo pelo responder.

$$V = MAC(K_{auth\_key}, I\_MESSAGE)$$

Com isto, ambas as partes conhecem um mesmo TGK e parâmetros necessários para prosseguir com o algoritmo MIKEY.

5.1.3 Determinação de um CSB por PKI

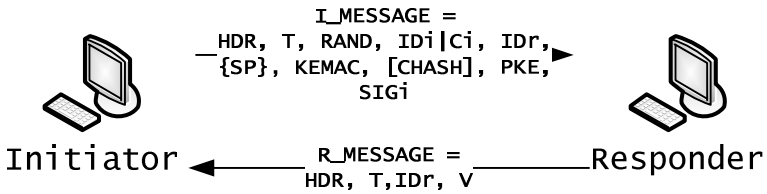

Figura 5-3 - CSB por PKI

Neste caso, temos:

- K_{pubr} a chave pública do responder;
- CHASH o identificador da chave pública de R (representada por), se ele possuir mais de uma.
- $PKE = K_{pubr}(env\_key);$
- env_key uma chave de envelope, que o *initiator* gera aleatoriamente. env_key é usada no *initiator* para gerar uma enc_key e uma auth_key, tal qual sessão anterior;
- Então, KMAC é gerado por:

$$KMAC = K_{enc\_key}[TGK, IDi\ ]||\ MAC(K_{auth\_key}, HDR);$$

- ID_i é o identificador do *initiator* conforme aquele especificado no certificado digital emitido em favor dele (C_i);

- SIG_i é uma assinatura feita por i com sua chave privativa K_{privi}

$$SIG_i = K_{privi}\ [I\_MESSAGE];$$

- V é calculado da mesma forma que a seção anterior.

A chave de envelope estabelecida acima pode ser usa posteriormente como uma *pre-shared* para futuro contato entre as partes, mas a RFC não recomenda fazer uma cópia desta informação por longo prazo, e deixa por conta da implementação determinar o tempo máximo de armazenamento.

5.1.4 Determinação do CSB por DH

O método DH é recomendado apenas para chaves ponto a ponto. Além disto, é necessário que as partes possuam certificados digitais[6]. O iniciador remete os valores trocados publicamente em um negociação por DH para o *responder* e uma assinatura da sua mensagem com seu certificado, tal qual o SIG_i anterior.

O *responder* devolve uma assinatura SIG_r e demais parâmetros de retorno do DH. Sendo as assinaturas válidas, neste ponto, ambas as partes são capazes de determinar TGK pelo processo DH.

5.1.5 O transporte do protocolo MIKEY

A RFC 3830 [55] contém todas as regras necessárias para criação do *payload* para formação das mensagens do MIKEY, mas não especifica a forma como ele dever ser transportado. O SDP foi estendido pela RFC 4567 [68] de forma a incorporar o MKEY como protocolo para negociação de chaves de sessão, estabelecendo o campo “a=” para enviar o *payload* do MIKEY, conforme ilustrado na mensagem SDP abaixo.

```
v=0
o=alice 2891092738 2891092738 IN IP4 w-land.example.com
s=Cool stuff
e=alice@w-land.example.com
t=0 0
c=IN IP4 w-land.example.com
a=key-mgmt:mikey AQAFgM0XflABAAAAAAAAAAAAAAsAyONQ6gAAAAAGEEoo2pee4hp2
UaDX8ZE22YwKAAAPZG9uYWxkQGR1Y2suY29tAQAAAAAAAQAk0JKpgaVkDaawi9whVBtBt
0KZ14ymNuu62+Nv3ozPLygwK/GbAV9iemnGUIZ19fWQUOSrzKTAv9zV
m=audio 49000 RTP/SAVP 98
a=rtpmap:98 AMR/8000
m=video 52230 RTP/SAVP 31
```

[6] DH é susceptível ao “man-in-the-middle”.

```
a=rtpmap:31 H261/90000
```

Figura 5-4 - SDP transportando o MIKEY [7]

O protocolo de gerenciamento de sessão deve realizar uma análise sintática sobre o campo `a=` para recuperar os valores nas mensagens I e R e enviá-los ao processo executando as atividades de gerenciamento de chaves. Em termos de transações do SIP, a figura abaixo ilustra as trocas de mensagem entre duas partes.

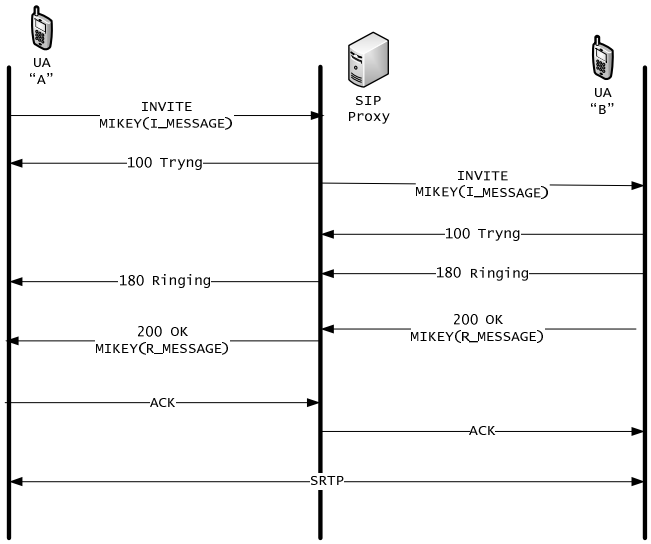

Figura 5-5 - Fluxo de mensagens SIP com MIKEY

A RFC 4567 estabelece que quando operando com SIP, se uma oferta SDP (seção 3.8, pág. 43) for rejeitada a resposta deve incluir o motivo e a negociação deve reiniciar através de outra oferta-resposta por um re-INVITE ou um UPDATE. Se a negociação for inviável, o receptor deve enviar um "*408 Not Acceptable Here*". Caso mais de um mecanismo de gerenciamento de chaves seja suportado, múltiplas entradas para o campo `a=key-mgmt` podem ser enviadas, cada qual com seu mecanismo, listadas na ordem de preferência. A resposta deve conter o mecanismo que vai ser utilizado, obedecendo a preferência indicada e as possibilidades do receptor, do contrário, seguirá um "408 *Not Acceptable Here*". *Re-keying* das chaves mestras devem ocorrer sempre através da negociação de uma nova oferta-resposta (seção 3.8, pág. 43), possivelmente através de um UPDATE. A RFC 4567 recomenda para evitar corte na mídia, uma vez que a outra parte pode iniciar o envio do material criptográfico antes de chegada da resposta, à adoção de definições de pré-condições conforme descrito na RFC 5027 [70].

[7] Fonte: http://www1.tools.ietf.org/html/draft-ietf-mmusic-kmgmt-ext-15

5.2 O Protocolo SDES – Secure Description for Media Stream

O SDES foi um meio articulado pela IETF para estabelecer as informações que irão compor o contexto de como ocorrerá a criptografia da mídia pelo protocolo SRTP (seção 4.8, pág. 65). O SDES, descrito na RFC 4568 [69], e é meramente uma extensão ao protocolo SDP (seção 3.8, pág. 43), de forma a definir o esquema para estabelecimento de material criptográfico em sessões *unicast*. Sessões *multicast* devem ser tratadas em estudo complementar, representando um ponto em aberto.

O SDES não propõe ser um protocolo de estabelecimento de chaves autenticadas, ou AKE, podendo sofrer com alterações ao longo do caminho, exatamente porque o material a ser usado pela criptografia segue em *texto aberto*. A RFC 4568 recomenda que o SDP seja protegido preferencialmente em seu trajeto pelo TLS (seção 4.3, pág. 59) e S/MIME (seção 4.5, pág. 61) e eventualmente pelo protocolo IPSEC (seção 4.6, pág. 63).

O principal fator para uso do SDES é a simplicidade na implementação. Consistindo na inclusão do atributo "*crypto*" para uso adicional no campo "a=" (seção 3.8.1, pág. 44) do SDP, de forma a especificar os parâmetros necessários para descrever o material que formará o contexto criptográfico. O campo "k=" (seção 3.8.1, pág. 44), estabelecido anteriormente pelo SDP com a finalidade de transportar a chave criptográfica, é insuficiente para esta finalidade e não é utilizado no SIP. O SDES foi criado sendo expansível, acomodando futuras necessidades de negociação para outros algoritmos ou contextos.

A definição do atributo *crypto* é a que segue:

```
a=crypto:<tag> <crypto-suite> <key-params> [<session-params>]
```

Onde os campos possuem a seguinte interpretação:

- *Tag*: usado para identificar um atributo *crypto*, dentre diversas opções ofertadas por uma parte, isto servirá para o receptor apontar qual opção aceita, no esquema oferta e resposta (seção 3.8, pág. 43);
- *Crypto-suite*: representa os algoritmos de criptografia e autenticação a serem utilizados, conforme Tabela 5-1.
- *Key-params*: define um ou mais conjuntos de chaves para o *crypto-suite* em questão. O `key-params` pode ser subdividido em:

- `<key-method>":"<key-info>`. O único `key-method` definido na RFC receber o valor foi o `inline`, informando que todo o material criptográfico necessário segue no campo `key-info`.
- Quando mais de uma chave é especificada, cada chave conter um identificador, tal que seja permito usá-la como referência no campo *Master Key Identifier* do SRTP (seção 4.8, pág. 65).

- *Session-Params*, é usado para acertar complementos aplicáveis ao contexto.

Se `key-method` possuir o valor **inline** então key-info conterá os seguintes componentes:

- "inline:" <key||salt> ["|" lifetime] ["|" MKI ":" length];
- key||salt:concatenação da master key e SALT, base64;
- lifetime: número máximo de pacotes usando a mesma chave
- MKI:length: O valor para apontar esta chave no campo MKI e o tamanho do campo MKI em bytes
- Exemplo para o caso AES_CM_128_HMAC_SHA1_80:
  `inline:d0RmdmcmVCspeEc3QGZiNWpVLFJhQX1cfHAwJSoj|2^20|1:4`

Abaixo seguem dois exemplos de uma mensagem SDP contendo uma descrição SDES especificando o perfil RTP como sendo o valor RTP/SAVP, indicando o uso do protocolo SRTP.

```
v=0
o=jdoe 2890844526 2890842807 IN IP4 172.20.25.100
s=-
c=IN IP4 172.20.25.100
t=0 0
m=video 51372 RTP/SAVP 31
a=crypto:1 AES_CM_128_HMAC_SHA1_80
  inline:d0RmdmcmVCspeEc3QGZiNWpVLFJhQX1cfHAwJSoj|2^20|1:32
m=audio 49170 RTP/SAVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_32
  inline:NzB4d1BINUAvLEw6UzF3WSJ+PSdFcGdUJShpX1Zj|2^20|1:32
m=application 32416 udp wb
a=orient:portrait
```

Mensagem 5-1 – SDES [8]

É importante notar a presença no Request-URI do esquema SIP seguro (sips:), solicitação para proceder via TLS. O campo "a=" em negrito mostra a especificação do processo criptográfico a ser usado, indicando: o uso do AES, no modo contador – CM (seção 4.8, pág. 65), o uso de autenticação para o protocolo através do algoritmo HMAC_SHA1.

[8] Fonte: http://www.faqs.org/rfcs/rfc4566.html

É necessário especificar pelo menos um atributo crypto para cada mídia onde for necessária criptografia, como no exemplo acima. Sendo que a ordem em que são apresentados os parâmetros *crypto=*, para uma mesma mídia, também define a ordem de preferência entre os esquemas ofertados. Cada esquema representa como será cifrada a mídia na direção de quem envia. Na resposta, a outra parte pode indicar os atributos aceitos ou rejeitar sua oferta. O receptor deve depois estar apto a receber o tráfego cifrado conforme acerto.

Para que a criptografia seja nas duas direções, a outra parte deve iniciar outro processo de oferta-resposta do SDP em seu benefício.

É possível que após a oferta possa ocorrer o envio prematuro da mídia, mas como ainda não há uma definição exata de qual atributo *crypto* será consenso, poderá haver cortes da mídia. Para evitar esta situação a RFC5027 [70] acrescenta pontos na negociação para evitar o início da transmissão sem que ambas as partes tenham entendimento mútuo do contexto criptográfico.

5.2.1.1 Alterando as chaves durante a sessão

É previsto que durante o diálogo as partes efetuem a troca da chave de sessão, temporariamente. Isto ocorre pela re-envio do SDP com o novo contexto criptográfico entre ambas as partes, realizando o processo de re-INVITE.

5.2.2 Parâmetros de chave padronizados para o SRTP com SDES

A Tabela 5-1 apresenta todos os possíveis *crypto-suites* atualmente especificados para o SRTP. O *crypto-suite* indiretamente define o contexto de transporte a ser utilizado, como o tempo de vida da chave, o tamanho do MAC e outros.

	AES_CM_128_ HMAC_SHA1_80	AES_CM_128_ HMAC_SHA1_32	F8_128_HMAC_ SHA1_80
Master key length	128 bits	128 bits	128 bits
Master salt length	112 bits	112 bits	112 bits
SRTP lifetime	2^48	2^48	2^48
SRTCP lifetime	2^31 packets	2^31 packets	2^31 packets
Cipher	AES Counter Mode	AES Counter Mode	AES F8 Mode
Encryption key	128 bits	128 bits	128 bits
MAC	HMAC-SHA1	HMAC-SHA1	HMAC-SHA1
SRTP auth. tag	80 bits	32 bits	80 bits
SRTCP auth. tag	80 bits	80 bits	80 bits
SRTP auth. key len.	160 bits	160 bits	160 bits

SRTCP auth. key len.	160 bits	160 bits	160 bits

Tabela 5-1 – Crypto-suites [9]

5.3 O Protocolo ZRTP

ZRTP é um mecanismo desenvolvido com a capacidade em negociar as chaves de sessão diretamente entre os terminais, empregando as portas negociadas para o RTP e utilizando as possibilidades de extensão do cabeçalho RTP. Após a negociação das chaves, o ZRTP utiliza o SRTP para proteção da mídia. O objetivo do ZRTP foi introduzir uma técnica de proteção que não fosse dependente de uma relação de confiança gerenciada centralizadamente pelo provedor do serviço. O ZRTP foi elaborado e desenvolvido por Phil Zimmermann, o criador do mecanismo de segurança de e-mails conhecido como *Pretty Good Privacy*[10].

O ZRTP especifica dois esquemas possíveis para estabelecer uma chave mestra: o Diffie-Hellman com PSK e o Diffie-Hellman com SAS, sem emprego de mecanismos de PKI.

O modo PSK tem as mesmas características das demais abordagens vistas neste trabalho e seu funcionamento baseia-se no envio à outra parte os números acordados publicamente pelo Diffie-Hellman [38]., através de um MAC operado com a chave compartilhada, similar ao MIKEY com PSK (seção 5.1.3, pág. 75). Desta forma a outra parte terá condições de atestar a ausência do *man-in-the-middle*.

No modo DH com SAS o *man-in-the-middle* é evitado pela geração dois *hashs,* com comprimentos curtos, tipicamente de dois caracteres, dos valores trocados publicamente para uso pelo Diffie-Hellman [38], um para cada direção. Parecido com a forma anterior, mas sem o emprego de chaves compartilhadas. Tais *hashs* são apresentados visualmente para os interlocutores, via o display do terminal. Os interlocutores lêem um para o outro o hash, via o canal de voz[11] RTP. Se ambos concordarem que a leitura é idêntica ao visualizado no display é porque provavelmente o DH obteve sucesso em estabelecer o canal seguro.

No ZRTP este *hash* é conhecido como *Short Authentication String* - SAS, e possui tipicamente 16 bits, que dá uma chance a cada 65536 do atacante não ser reconhecido.

[9] Fonte: http://www.networksorcery.com/enp/rfc/rfc3711.txt
[10] Há uma polêmica interessante quando da criação deste mecanismo, que envolve o direito do Estado, do indivíduo e privacidade, podendo ser consultada em http://en.wikipedia.org/wiki/Pretty_Good_Privacy.
[11] Neste momento o canal ainda não está criptografado, mas o RTP estabelecido. Se ambos concordarem com o hash, o ZRTP converte o RTP para SRTP.

Capítulo 6

A arquitetura Kerberos

6.1 Motivação

O objetivo deste capitulo é analisar as características funcionais da arquitetura Kerberos [58], que será utilizada na proposição feita no Capítulo 7.

6.2 Introdução ao Kerberos

O objetivo primário do Kerberos é autenticar os participantes de uma rede adicionando a facilidade de "*single sign on*", ou seja, feita a primeira autenticação todos os serviços registrados no mesmo domínio em que é obtida a concessão poderão ser acessíveis sem repetidas apresentações das credenciais de longo prazo do usuário, durante um período máximo de tempo estabelecido na primeira autenticação. O protocolo Kerberos foi concebido para operar sem creditar a autenticação apenas em características como as asserções dos hosts, endereçamento, controle de acesso físico, e também no fato das informações poderem ser interceptadas, modificadas ou inseridas durante o trânsito nas redes. O Kerberos baseia seu processo de autenticação no conhecimento prévio de chaves compartilhadas, únicas entre os hosts participantes e os serviços de autenticação por eles providos. Na especificação atual todos os mecanismos de assinatura e criptografia são realizados através de algoritmos simétricos. No padrão inicial são suportados os algoritmos DES, 3DES e RC4. O AES foi apropriado recentemente. O Kerberos negocia dinamamicamente o algoritmo para a transação entre cada parte.

Inspirado no algoritmo de Needham-Schroeder [24], sua descrição está contida na RFC 4120 [58], que apresenta a versão 5 do protocolo. Quando um cliente `C` deseja obter acesso a

um determinado serviço S, esse deve solicitar ao servidor Kerberos TGS que atende ao seu domínio uma *chave de sessão* K a ser compartilhada entre C e S. K é entregue pelo TGS a C de duas maneiras, uma protegida pela chave compartilhada entre C e o TGS e a outra protegida com a chave compartilhada entre o TGS e S. Esta última é conhecida como *ticket* e é decifrável apenas por S. De posse do *ticket*, C solicita acesso a S encaminhando em anexo o *ticket*, usando o meio de transporte que aplicação final especificar, e um componente *autenticator* atestando que o cliente conhece a chave que está dentro do referido *ticket*. Ao decompor o *ticket* pelo uso da chave compartilhada com o TGS, S recupera a chave K para em seguida atestar que o *autenticator* foi gerado corretamente, deduzindo simultaneamente que C tem o conhecimento de K e que C foi autenticado pelo TGS do domínio. Este mecanismo serve para assegurar a autencidade de C perante S e vice-versa, mas nada diz respeito a autorização de C nos serviços disponibilizados por S, sendo este um problema da aplicação.

O Kerberos adiciona à proposta de Needham-Schroeder a presença de um outro servidor de autenticação de mais alta hierarquia, conhecido como AS, que distribui um *ticket* "maior" chamado de TGT, permitindo aos *hosts* solicitarem *tickets* de serviço ao TGS, da mesma forma que no parágrafo anterior, mas evitando o uso constante das senhas compartilhadas de longo prazo, em favor do uso do TGT no seu lugar, conforme será visto abaixo. As mensagens também incluem carimbos de tempo para evitar a repetição indevida dos *tickets* por outros agentes.

6.3 Componentes do Kerberos

O Kerberos estabelece alguns elementos em sua arquitetura, que são:

- **Domínio, ou *realm* (R):** é a composição de servidores, clientes e serviços de autenticação administrados por uma mesma autoridade identificada por um nome, em geral, similar a um domínio qualificado do DNS.
- ***Principal* (P):** é a identificação de um usuário ou de um serviço. O esquema de nomes para compor um *Principal* é apresentado na seção 6.4.
- **O cliente (C)**, é o processo executado em benefício do usuário e que solicita permissão para acesso aos serviços.

- **Serviço:** é o recurso disponibilizado em rede por uma aplicação qualquer, cliente-servidora ou P2P. O uso de Kerberos em P2P é previsto na especificação, com algumas condições que serão vistas à frente.
- **Servidor do Serviço (S):** é a parte que disponibiliza um serviço e precisar saber quem o está acessando.
- ***Ticket* (T):** é uma "mensagem" protegida que pode ser lida apenas pelo destinatário, em geral um servidor, com informações adicionais suficientes de forma a permitir garantir a autenticidade da mensagem. Por fim, um *ticket* pode encapsular outros *tickets* destinados a terceiros.
- **Kerberos Authentication Server (AS):** é o serviço que emite e envia os *tickets* iniciais para os solicitantes.
- ***Ticket* Granting *Ticket* (TGT):** É o *ticket* inicial emitido pelo AS ao cliente, sem o qual o cliente não poderá ser autenticado para os serviços de interesse.
- **Chaves (K):** O Kerberos possuis três tipos de chave: as de longo prazo, definida entreo o AS e *Principal*, utilizadas no processo de obtenção do TGT, as de sessão, usadas no processo de obtenção de TGS e as sub chaves trocadas entre o cliente e servidor. As chaves de longo prazo são geradas pela transformação da senha do usuário, por uma função padronizada na API Kerberos, conhecida como *String2Key*[1].
- **Chave compartilhada (K_{AB}):** uma chave conhecida única e exclusivamente pelos *Principals* A e B.
- ***Ticket* Granting Server (TGS):** Após a obtenção de um TGT, *tickets* adicionais para acesso aos serviços de rede devem ser obtidos através do TGS.
- **Key Distribution Center (KDC):** é a coleção dos serviços TGS e AS numa infra-estrutura real. Em geral o TGS e o AS estão reunidos num mesmo elemento físico.
- **Flags (F):** um conjunto de bits utilizado para sinalizar ações descritas na seção 6.9.
- **etype:** especifica os algoritmos de criptografia suportados por uma das partes.

[1] Não existe a Key2String. O processo é irreversível.

6.4 Formato dos nomes no Kerberos

Há três designações importantes para o Kerberos: o *realm*[2], o *Principal* e a instância. Todas as entidades em uma instalação Kerberos devem ser unicamente identificadas através de um *Principal*, incluindo usuários, computadores, serviços, servidores. Cada Principal está associado com uma chave de longo prazo, como uma senha.

Principals são estruturados de forma hierárquica iniciando com um o nome do usuário ou serviço, seguido de um nome de instância opcional para a entidade, de forma que os dois juntos, anexados ao domínio, formem uma identidade universal. Por convenção, um *realm* é o domínio DNS de uma autoridade em letras maiúsculas. O formato final então fica sendo:

```
<nome|serviço>/[<instância>]/[complemento]/.../[complemento]@<DOMAIN_NAME>
```

Onde complemento seriam identificadores adicionais específicos para a necessidade do serviço. *Principals* associados a *hosts* devem conter o texto “host” no campo <serviço|nome> e o nome FQDN do *host*, na instância. O Kerberos reserva uma série de nomes especiais de serviço, como o “*krbtgt*” para identificar o TGS em um domínio.

6.5 Solicitanto o TGT ao Kerberos

O processo de obtenção de um TGT é o primeiro passo para o ingresso de um cliente numa infra-estrutura autenticada por Kerberos. Em geral, este processo é realizado apenas uma vez, na inicialização do dispositivo e só volta a ser repetido quando o tempo de validade do TGT expirar. Até lá o cliente não utilizará mais sua chave de longo prazo, a menos para serviços de troca de senha, que seria a segunda motivação para acesso ao AS. Ao término deste interação, todos os acessos aos serviços serão feitos ao TGS pelo uso do *ticket* resultante desta troca de mensagens: o TGT.

Para obter um TGT, o cliente realiza uma solicitação ao AS através de uma mensagem em texto aberto, conhecida com KRB_AS_REQ, contendo o seguinte: o *Principal* do cliente, P_c; a hora local do cliente, h_c; o principal do TGS ou P_{TGS}; um *nonce* gerado pelo cliente; um flag INITIAL para demostrar que as chaves de longo prazo serão utilizadas no processo; demais *flags*, se necessário; um campo opcional de autenticidade, chamado PA_DATA, para evitar solicitações indevidas ao AS intencionando uma superexposição de material criptografado

[2] Chamado de *realm* para distinguir do domínio DNS.

com a senha compartilhada; a relação *etype* dos algoritmos de criptografia suportados pelo cliente, em ordem de preferência, entre outros campos menos relevantes.

Ao receber o KRB_AS_REQ o AS verifica se o *Principal* do cliente está registrado, se sua hora h_C é compatível com seu relógio[3] local, se o PA_DATA é consistente, se estiver presente, dentre outras detalhes. Não sendo, ele envia um KRB_ERROR contendo o motivo da rejeição. Caso positivo, o AS gera uma mensagem KRB_AS_REP com as seguintes informações: A chave de sessão entre o cliente e o TGS, $K_{C:TGS}$; o tempo de vida L do *ticket*; o *Principal* de TGS, P_{TGS}; o *nonce* gerado pelo cliente; o *ticket* para uso com o TGS, TGS_C ; o *Principal* do cliente e seu *Realm.*

Considerando $A \rightarrow B: M$, uma mensagem M enviada de A para B; que IP_A é o endereço IP de A; $K[M]$ uma mensagem cifrada com a chave K; e que um valor entre os símbolos "<","> " é opcional; e `msg_type` o identificador da mensagem sendo enviada, então as seguintes mensagens são trocadas entre as partes:

$KRB\_AS\_REQ\ (C \rightarrow AS)$: { msg_type, $P_C, R_C, <P_s>$, h_C, P_{TGS}, nonce, $<IP_C>$, <PA_DATA>, etype}

AS gera o TGT_C:

$TGT_C = \{K_{AS:TGS}[F_C, K_{C:TGS}, R_C, P_C, h_{AS}, L, IP_C], P_{TGS}, R_{TGS}\}$,

e envia para C:

KRB_AS_REP(AS→C): { msg_type, $K_{C:AS}[F_C, K_{C:TGS}, P_{TGS}, R_{TGS}, h_{AS}, L, IP_{TGS}]$, TGT_C}

C decifrará os valores: $K_{C:TGS}$, P_{TGS}, L, mas não terá condições de decifrar TGT_C. Neste ponto, C possui um TGT_C do domínio ao qual pertence ou se registrou. As mensagens trocadas estão desenhadas abaixo. A RFC 4120 recomenda que neste momento o cliente estabeleça um "*time skew*" do seu relógio local com o do servidor, e que esta informação seja tratada no nível da aplicação cliente para correção nas próximas solicitações, sem requerer alterar o relógio do dispositivo local.

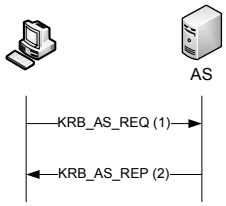

Figura 6-1 - Obtendo um TGT

[3] O Kerberos aceita uma margem de cinco minutos na hora relativa, por padrão. Mas é configurável.

6.6 Solicitando um *Ticket* de serviço ao TGS

Uma vez de posse do TGT_C, conforme sessão anterior, C irá solicita ao TGS autenticação em algum serviço disponível no domínio ao qual pertence ou registrou-se temporariamente via relação de confiança, quando for de seu interesse, através da mensagem KRB_TGS_REQ. Havendo concordância por parte do TGS, este envia para o cliente uma mensagem KRB_TGS_REP, caso contrário um erro KRB_ERRO, com o respectivo motivo. Então, para solicitar autenticação em um serviço em S, C procede da seguinte maneira:

$$KRB\_TGS\_REQ\ (C \rightarrow TGS):\ msg\_type\ ,\ P_c,\ K_{C:TGS}\ [Flags,\ h_c],\ P_s$$

$$KRB\_TGS\_REP\ (TGS \rightarrow C):\ msg\_type,\ K_{C:TGS}[K_{C:S},\ L_s,\ P_s],\ T_s$$

Onde Ls é o tempo máximo de vida para chave $K_{C:S}$ e $T_{s,}$ o *ticket* para o serviço em S, definido como sendo $Ts = K_{S:AS}\ [\ IPc,\ Ls,\ PC,\ KC:S\ ],$ então C, decifra sua parte e armazena $T_{s,}$ indecifrável para C.

É importante ressaltar que, deste ponto em diante, a definição de como a aplicação enviará para o servidor o *ticket* T_S é diferente para cada aplicação.

Outro ponto é que por padrão o *ticket* de serviço é protegido com a chave de longo prazo do servidor, representada por $K_{S:AS.}$

Tickets podem ser usados sucessivamente, sem novas solicitações, desde que dentro do prazo de validade. *Tickets* podem ser solicitados com direito a renovação, pelo ajuste do *flag* apropriado (seção 6.9, pág. 90).

6.7 Autenticação do Serviço

Para autenticar-se no serviço, C envia a S o *ticket* T_S, mas usando um formato padrão, através das mensagens KRB_AP_REQ e opcionalmente o KRB_AP_REP. Há um *flag* a ser especificado pelo cliente, chamado de *mutual-required* que força o envio pelo servidor da mensagem KRB_AP_REP, que de outra forma seria opcional. As mensagens abaixo mostram o início ao processo de autenticação ao serviço requisitado:

$$KRB\_AP\_REQ(C \rightarrow S):\ \{msg\_type,\ AP\_OPTIONS,\ T_s,\ K_{C:S}\ [autenticator]\}$$

Onde AP_OPTIONS contém dois *flags*: USE-SESSION-KEY, indicando se o *ticket* estiver protegido pela *session key* do TGT do destino, ou pela chave de longo prazo; e o *flag* MUTUAL-REQUIRED, indicando se a autenticação mútua será requerida pelo retorno da mensagem KRB_AP_REP.

O campo *autenticator* contém principalmente: $\{P_c, R_c, h_{KDC}$, subkey, <seq_number>$\}$ cifrados com a chave de sessão criada para o cliente e o servidor. O destaque ocorre por conta de uma *subkey*, gerada pelo cliente com tamanho arbitrário, que pode ser usada pela aplicação.

O servidor pode retornar um erro KRB_ERROR, por alguma inconsistência, ou, caso seja requisitada mútua autenticação, retornar a mensagem KRB_AP_REP contendo o seguinte:

$$KRB\_AP\_REP(C \rightarrow S): \{msg\_type, K_{C:S}[\ hC, <subkey>, <seq\_number>]\}$$

A Figura 6-2 apresenta a troca de mensagens necessárias para autenticação de um serviço, em especial, as mensagens KRB_AP_REQ e KRB_AP_REP.

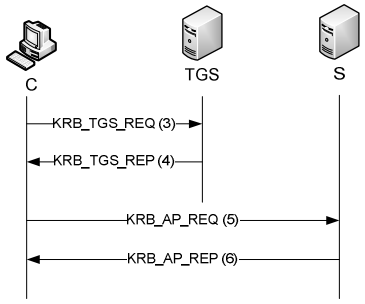

Figura 6-2 - Acessando um serviço

Tickets enviados aos servidores, que de certa forma são passivos no processo acima, são emitidos cifrados com a chave de longo prazo, uma vez que ele não está solicitando nada, permanecendo em espera. Caso o *flag* USE-SESSION-KEY seja utilizado, é porque o *ticket* está cifrado com a chave de sessão associada ao TGT do servidor. Esta opção é empregada em autenticações P2P. Caro haja a presença de uma *subkey* na mensagem de retorno é porque o servidor deseja utilizar uma outra chave na direção do tráfego para o cliente.

. É relevante observar que as mensagens KRB_AP_REQ, REP e todas as outras mensagens do Kerberos que não as da família KRB_AS e KRB_TGS, devem ser transportadas por algum mecanismo próprio da aplicação. O Kerberos não define um padrão para transportar tais mensagens, como abrir *sockets* específicos. O que o Kerberos faz é

prover em sua API, chamadas para aplicação processar o *ticket* e recuperar os seus componentes da mesma forma que para compor a mensagem, chamados de *parse* no linguajar da programação.

A mensagem KRB_ERROR pode ser transportada nas duas condições, dependendo da fase da negociação.

6.8 Kerberos entre múltiplos domínios

Na medida do crescimento do número de usuários e da ocorrência da necessidade de autenticar acessos a serviços em domínios distintos, que muitas vezes representam outras fronteiras organizacionais, o modelo acima se torna insuficiente. Para contornar esta limitação, o Kerberos prevê um mecanismo de relação de confiança possibilitando que um *ticket* possa ser expedido em favor de *Principals* de outros domínios. Essa abordagem assegura a autenticidade de um *Principal* perante outro, mas, como de praxe, não diz respeito ao nível de autorização ou acesso, sendo este um problema das políticas do domínio visitado.

A relação de confiança é estabelecida pelo compartilhamento direto de chaves entre *Ticket Granting Servers* de domínios distintos, devendo ser criada uma chave para cada direção na relação a ser estabelecida. Um *Principal* de um TGS tem a seguinte aparência `krbtgt/hostname.domain.tld@domain.tld`

Para estabelecer uma relação de confiança do domínio *alpha* para o domínio *beta*, bi-direcional, deve-se criar dois *Principals*, cada qual com uma chave compartilhada, em ambos TGS's:

```
krbtgt/alpha.tld@beta.tld (1)
krbtgt/beta.tld@alpha.tld (2)
```

Para o TGS do domínio *alpha*, a entrada (2) será utilizada para as requisições de *ticket* dos usuários em `alpha.tld`, que almejam serviços em `beta.tld`, e a entrada (1) para cumprir a direção contrária. Quando um cliente em *alpha.tld* solicitar um serviço em *beta.tld*, ele irá receber primeiro um *Ticket* para acesso ao TGS do *realm* remoto, assinado com a chave interdomínio associada ao *Principal* (2), enviando a solicitação de serviço diretamente para o TGS da rede *beta*. Este processo é conhecido como *direct cross-realm*, o que pode levar a uma condição "*full mesh*"[4] na existência de várias relações de confiança. Outra forma prevista

[4] Onde todos os servidores Kerberos teriam relação de confiança com todos os demais.

ocorre pela criação de um caminho transitivo na relação de confiança, compondo uma estrutura hierárquica, de forma a acomodar redes de confiança mais complexas. Clientes percorrem recursivamente a estrutura até obterem um *ticket* para um principal idêntico ao domínio do TGS que o emitiu. Este processo pode estar combinado com a árvore de nomes dos domínios, tal qual o DNS, ou configurado estaticamente nos servidores. A abordagem *direct cross-realm* pode ser trabalhada em conjunto como o esquema transitivo, para minimizar a latência no processo de autenticação, quando de interesse.

KDC, se configurados desta forma, podem ajustar um *flag* TRANSITED-POLICY-CHECKED no TGT emitido para informar que a procedência da verificação foi dentro da linhagem de relação de confiança prevista. Mas cabe ao servidor de aplicação decidir pelo aceite do TGT.

É prevista uma situação onde um cliente registre um Principal em outro domínio, dispensando a necessidade de uma relação de confiança, quando a necessidade não ultrapassar um reduzido número de clientes.

6.9 Opções para emissão de *Tickets*

O Kerberos estabelece alguns sinalizadores para o processamento do *ticket*, ofertando as seguintes possibilidades de uso:

- *Proxiable e Proxy Tickets:* Algumas vezes poderá ser necessário para um principal permitir a um serviço realizar operações em seu benefício. O serviço terá que assumir temporariamente a identidade do Principal para um propósito específico. Assim, o Principal deverá garantir o direito do serviço em transformar-se num *Proxy*. Um *Proxiable ticket* é emitido por um AS quando assim indicado pelo *Principal*, na suposição que ele queira conceder a terceiros o direito de emitir *tickets* de serviço em seu nome. Um *Proxy ticket* é emitido a um TGS, por intermédio de um TGT *Proxiable*, solicitado pelo *Principal*. O *Principal* de origem deve especificar o *Principal* do serviço que terá a procuração para solicitar TGT's em seu benefício e os endereços de origens pelos quais os *tickets* poderão ser expedidos. Opcionalmente os endereços podem não ser especificados. A única atividade de um serviço que tenha recebido um TGS com *Proxy* habilitado **não** poderá fazer é emitir um novo TGT em nome do *Principal* que o transformou em *Proxy*.

- *Forwardable tickets*. Opera da mesma forma que o anterior. A diferença é que o serviço poderá emitir inclusive novos TGTs em nome do *Principal*, em benefício de outros clientes Kerberos. Assim, por exemplo, um usuário nômade poderá acessar um dispositivo de dados diferente e receber o mesmo nível de serviços de autenticação, se tal terminal receber um TGT enviado pelo serviço ocupando o papel de *Proxy*. Ou seja, o segundo cliente não precisa solicitar um novo TGT por que o serviço com *Proxy* o fará, quando necessário. A vantagem deste processo é que o usuário não precisará lançar mão da sua chave de longo prazo no destino.
- *Renewable Tickets:* Um *ticket* pode receber uma data de renovação superior à data de validade. A renovação pode ser solicitada ao KDC antes que expire a validade e pode ou não ser aceita, mediante restrições administrativas.
- *Postdated Tickets*: é um *ticket* que não é válido até uma data posterior, permitindo que determinados *tickets* sejam emitidos para trabalhos agendados, como backup e outros eventos.
- ENC-TKT-IN-SKEY: Visto na seção 6.10.

6.10 Kerberos autenticando redes P2P

Há certo desentendimento quanto ao fato do Kerberos prestar-se apenas a autenticação do tipo cliente-servidor. Um servidor está apto a receber *tickets,* dos clientes, a qualquer instante, pois estes estão cifrados com sua chave de longo prazo que deve ser mantida enquanto o serviço estiver disponível. Clientes, por outro lado, nem sempre vão desejar ou conseguir reter as chaves de longo prazo ($K_{C:AS}$) após o primeiro *login* e a obtenção do TGT (usado para solicitar serviços e dispensar o reuso da chave de longo prazo). Desta forma, numa operação P2P, tais clientes ficariam impossibilitados de receber *tickets* assinados pela chave de longo prazo de outros clientes P2P. Há vários motivos para os clientes não desejarem reter a chave de longo prazo: pode ser por questões de segurança, direito de acesso das aplicações dentro do sistema operacional do cliente, porque algum hardware de autenticação foi utilizado apenas num primeiro instante, ou qualquer outro fator. Então, para resolver esta questão, a RFC 44120 [58] determina que, quando necessário, *tickets* possam ser emitidos cifrados pela chave de sessão utilizada para obtenção do TGT do referido cliente ($K_{C:TGS}$).

Para realizar tal feito, um cliente “A” que quiser solicitar acesso a outro “B”, pode remeter em anexo o TGT de B, solicitando ao KDC emitir um novo *ticket* com a chave de sessão

contida neste TGT. A mensagem KRB_TGS_REQ tem a possibilidade de transportar outros *tickets* em anexo, que no caso, seria o próprio TGT de B. A solicitação deve ter habilitado o *flag* ENC-TKT-IN-SKEY, informando o desejo em emitir o *ticket* de serviço baseado na chave de sessão do *ticket* em anexo.

Ao enviar o novo *ticket* gerado pelo TGS para "B", "A" deve enviar um KRB_AP_REQ contendo o *flag* `USE-SESSION-KEY` de forma que "B" decifre o *ticket* com sua chave de sessão referente ao TGT. Assim, "B", este será capaz de decifrar o *ticket* porquanto mantiver a senha compartilhada com o TGS, criada pelo AS, quando da obtenção do TGT.

O Kerberos não descreve a maneira como "A" terá acesso prévio ao TGT de "B", ficando esta atividade por conta da aplicação. Mas o processo é considerado seguro, na medida em que o TGS irá emitir o novo *ticket* baseado no *Principal* e *Realm* contidos no TGT e não na sugestão do cliente, na possibilidade de ser falsa. O fator decisivo reside em ninguém mais tendo acesso à chave de sessão entre "B" e o TGS.

6.11 Mensagens adicionais

O Kerberos possui adicionalmente as seguintes mensagens:

- KRB_SAFE: pode ser usada pelos clientes quando estes requerem a habilidade de detectar modificações nas mensagens trocadas entre as partes. Como qualquer outra mensagem, existe uma REQuisição e uma REsPosta, ou KRB_ERRO. Nestes envios as partes trocam entre si um *checksum* de uma mensagem trocada anteriormente, para assegurar a integridade das partes não protegidas;
- KRB_PRIV: também é usada com o mesmo contexto utilizando as *subkeys*;
- KRB_CRED: utilizado para envio das credencias do usuário, contendo o *ticket* e uma parte autenticada com a chave de sessão compartilhada entre o servidor e o cliente e a chave associada com o *ticket*.

6.12 Kerberos, Rede e NAT

6.12.1 DNS

Um servidor KDC, se não for configurado previamente no cliente, deve ser localizado na rede através de uma resolução DNS para um registro do tipo SRV. Uma entrada SRV em um registro DNS deve conter o seguinte formato

```
service.protocol.realm TTL class SRV priority weight port target
```

Um exemplo de busca de um servidor TGS através do comando *nslookup*, por exemplo, pode ser feito, de acordo com o ilustrado na figura abaixo:

```
nslookup –type SRV _kerberos._tcp.domainxyz
```

Para resolução de nomes de um host em seu equivalente Principal, usando a busca por um nome canônico, a RFC4120 [58] não recomenda o uso de servidores DNS se estes não empregarem mecanismos seguros, como o DNSSEC[5].

6.12.2 Rede

Como visto anteriormente, o KDC necessita colocar o serviço de autenticação em processo de escuta para acesso pelos clientes aos serviços de emissão de *tickets* TGT e TGS. Forma reservadas as portas 88 TCP ou UDP pela IANA para troca das mensagens KRB_AS_REQ/REP, KRB_TGS_REQ/REP e KRB_AS_ERROR. Quando operando sobre UDP as partes devem observar certas cautelas na temporização e reenvio de mensagens, se for necessário. A vantagem do UDP é dispensar o processo de *three-way handshaking*.

O envio propriamente do *ticket* para o servidor, como dito anteriormente, é de responsabilidade da aplicação.

6.12.3 NAT

NAT impõe uma restrição grave ao uso de Kerberos por conta da presença do endereço IP no *Ticket* emitido. Se o tipo de NAT feito puder operar apenas um endereço externo, então é possível configurar o servidor para anunciar este endereço. De outro modo, os clientes devem solicitar a emissão de *tickets* do tipo "*addressless*", previsto na RFC4120 [58].

[5] DNSSEC está especificado nas RFC 4033, RFC 4034, e RFC 4035.

6.13 Pontos adicionais

Há iniciativas para utilizar pares de chaves RSA no lugar de senhas compartilhadas, similar ao protocolo SSH, para a primeira autenticação [66]. O AS passaria a manter uma tabela associativa entre os *Principals* e as respectivas chaves RSA públicas dos participantes de um domínio. Os TGTs são emitidos pelo AS e apenas a chave de sessão para o uso com o TGS segue protegida pela chave pública do Cliente, economizando recurso computacional no uso da criptografia assimétrica. Os demais processos de emissão de *tickets* pelo TGS seguem conforme especificado anteriormente. Eventualmente os clientes poderiam autenticar o AS responsável pelo domínio com o uso de certificado digital aplicado apenas ao servidor.

6.13.1 Diferenças entre o Kerberos 4 e 5

As diferenças fundamentais são:

- O Kerberos 4 utiliza o DES para o processo criptográfico, para executar a função simbolizada acima por `K[M]`. A versão 5 mantém o DES por compatibilidade, mas especifica também o AES como algoritmo de trabalho, conforme RFC 3962 [57].
- O Kerberos 5 adota a notação ASN.1, para condicionar o protocolo a operar independente da plataforma.
- O Kerberos 4 cifra os *tickets* com a chave do cliente, condição esta que foi considerada desnecessária na versão 5.
- O Kerberos 4 utiliza o caractere ponto (“.”) no lugar do caractere barra (“/”) para separar o nome da instância.

Por compatibilidade um KDC na versão cinco deve prover serviços de translação de *tickets* através do serviço “krb524”, desde que a versão 4 esteja trabalhando com DES.

Capítulo 7

Proposições

O presente capítulo apresenta algumas estratégias para interceptação e interpretação legais da conversação telefônica quando ao menos um dos interlocutores for usuário de um serviço de Voz sobre IP. Faz parte do objetivo das proposições não reduzir o nível de segurança no serviço prestado, nem das especificações existentes, evitando que o mesmo ato seja realizado de forma não autorizada[1]. Pela extensão do assunto e diversidade de protocolos existentes as proposições ficarão contidas ao protocolo SIP, que foi apresentado com detalhes nos capítulos anteriores. As condições de contorno onde as proposições podem ser aplicadas estão definidas na seção 7.2.

Este capítulo está estruturado na seguinte forma: primeiro são apresentadas as condições de contorno, em seguida são analisadas as possibilidades de interceptação e interpretação das mídias RTP. No aspecto da interceptação são vistas as possibilidades de desvio de tráfego para um ponto de interceptação, conhecido como *ponto de encontro*[2], e outros mecanismos extras, acompanhados de uma compilação e aprofundamento da literatura consultada. Num terceiro ponto, são apresentadas as possibilidades de interpretação do material coletado quando este empregar os mecanismos atualmente disponíveis para proteção da mídia pelo protocolo SIP. No final do capítulo é apresentado um método adicional para preservação das chaves de sessão através do mecanismo de distribuição centralizada de chaves, com uso do protocolo Kerberos combinado com o SIP e SDP.

[1] A escuta sem a devida autorização judicial viola o direito constitucional presente no Art. 5º, inciso XII, da Constituição Federal; mas há controvérsias quanto a ser considerado crime pelo código penal, se a informação não for divulgada a outrem, segundo a referência [33]. Porém, este é um assunto que foge do tema.

[2] Chamado de *rendezvous point* na literatura consultada.

7.1 Relação entre interceptação e interpretação

Para tornar o processo de escuta efetivo, são necessárias duas medidas: a primeira para permitir que o tráfego gerado pelo usuário possa ser **interceptado** em algum ponto, durante o seu trajeto. Uma vez ocorrendo a interceptação, a segunda visa tornar possível que a mensagem seja **interpretada** ao ponto de tornar-se inteligível para uso em um processo investigativo. Em Voz sobre IP, ao contrário das redes PSTN, as duas medidas oferecem desafios distintos para atender ao objetivo da escuta legal, e serão analisadas separadamente.

7.2 Condições de Contorno

7.2.1 Quanto ao protocolo

As considerações são válidas apenas para o protocolo SIP, e todas as afirmações têm este pressuposto. O outro protocolo aberto aplicado em grande escala em Voz sobre IP é o H.323, que guarda similaridades com o SIP, inclusive por utilizar os mecanismos de transporte RTP e SRTP, mas são necessárias análises detalhadas na sinalização para aplicar as mesmas considerações no processo de gerenciamento de chaves de sessão. A. Milanovic, S. Srbljic, et al. [17] apresentam algumas estratégias ligadas ao H.323, principalmente nas questões de interceptação, pelo redirecionamento do tráfego RTP a um ponto de encontro. Os demais protocolos públicos em uso no momento são o MGCP[3] e o H.248/MEGACO[4] que compartilham o emprego do RTP/SRTP e o uso do mecanismo SDP. Diferentemente do SIP, estes cumprem um papel com foco no controle de *gateways* num ambiente de integração e transporte da rede PSTN sobre a rede de telefonia IP[5], através do processamento das informações de sinalização provenientes das redes de PSTN. O MGCP tem sido empregado também no controle de *gateways* residenciais em ambientes de integração. Uma visão do funcionamento do MGCP e H.248/MEGACO foi apresentada na seção 1.2.3.

Novas infra-estruturas estão sendo lançadas no mercado, especialmente as redes IMS[6] e as redes 3GPP[7], que utilizam o SIP como protocolo de sinalização, tendo potencial para aplicar as sugestões deste trabalho. Mas pelo fato delas incluírem um serviço de autenticação

[3] Andreasen, F.;B. Foster. Media *Gateway* Control Protocol (MGCP) Version 1.0, RFC 3435, January 2003.
[4] Groves, C.; Pantaleo, M.; Anderson, T.; Taylor, T. *Gateway* Control Protocol Version 1, RFC 3525, June 2003.
[5] O ITU trata o termo "telefonia IP" como mais abrangente que VoIP. VoIP é considerado o serviço prestado para o usuário final, e telefonia acrescenta as características de controle de *gateway*, convergência de serviços inteligentes e outros detalhes.
[6] IP Multimedia Subsystem. Disponível em http://www.3gpp.org/.
[7] The 3rd Generation Partnership Project. Disponível em http://www.imsforum.org/.

específico[8], pode ser que o processo de distribuição de chaves proposto tenha que sofrer adaptações.

7.2.2 Quanto aos serviços prestados

Serviços de VoIP, atualmente, podem ser classificados em duas categorias: gerenciados e não gerenciados. Os serviços gerenciados são aqueles prestados para o público em geral, e que seguem as políticas do prestador. Neste caso enquadram-se as corporações e os provedores de serviço VoIP público, que empregam um plano de numeração telefônico através do qual um assinante pode ser localizado em toda rede VoIP ou PSTN. Os serviços não gerenciados são aqueles aprovisionados pelo próprio usuário e não contam com uma infra-estrutura de servidores. Neste caso enquadram-se as iniciativas P2P. Os serviços P2P são complexos para serem identificados e controlados, e muitas vezes são procurados por possibilitarem o anonimato[9] das partes, tanto no registro quanto no uso dos serviços. Uma abordagem proposta para lidar com o problema pode ser vista em **[27]**.

Do ponto de vista do provedor de serviço, são considerados apenas os cenários de Voz sobre IP gerenciados por alguma autoridade administrativa. Tais serviços, por oferecem capacidade de comunicação com terceiros, necessitam alocar para os seus usuários, ou assinantes, identificadores de forma que estes possam ser acessíveis, no mínimo, pelos demais assinantes da mesma rede. Do ponto de vista do serviço prestado, e não do custo, há no mercado quatro modalidades básicas de assinatura para serviços VoIP, que são estratificadas conforme os parágrafos seguintes.

7.2.2.1 Cenário I – VoIP restrito

Modalidade restrita é aquela em que os provedores fornecem facilidades VoIP apenas para comunicação com demais assinantes IP da mesma unidade administrativa. Os identificadores são em geral privativos e associados ao ponto de contato do assinante pelo registro dinâmico na inicialização da aplicação cliente. É comum o uso de protocolos proprietários e fechados. Vários provedores neste cenário empregam protocolos proprietários, como o famoso Skype[10].

[8] As redes 3GPP e IMS estabeleceram o serviço DIAMETER como padrão para autenticação e autorização de acesso.
[9] O anonimato pode ocorrer de duas formas: o locutor não deseja ser identificado por ninguém, inclusive pelo seu correspondente; ou o locutor requer apenas que terceiros não consigam identificar ou interpretar sua comunicação. Esta dissertação considera o termo “anônimo” ou “anonimato” aquele aplicável apenas ao segundo caso.
[10] O Skype também presta serviços em outras modalidades, com adição de tarifa.

7.2.2.2 Cenário II – VoIP integrado

Uma segunda modalidade permite que os assinantes sejam contatados por usuários de outros prestadores da Internet, puramente através das redes IP. Nesta modalidade é necessário que os assinantes recebam um identificador universal, como um URI SIP. É necessário o emprego de um protocolo padrão para voz sobre IP, como o SIP ou o H.323. A localização do assinante de destino ocorre pelo auxílio de *Proxies*, como visto na seção 3.5. Este é o cenário que irá prevalecer na possibilidade da migração total dos usuários PSTN.

7.2.2.3 Cenário III – Telefonia IP unidirecional

A terceira modalidade pode compreender a primeira ou a segunda, com a facilidade adicional em permitir que o assinante realize chamadas para a rede PSTN, por intermédio de *Gateways*. Os assinantes não possuem identificação própria na rede PSTN, e o identificador de chamadas, ou CallerID na direção da rede pública é o número chave alocado para o acesso PSTN ao referido *Gateway*, se não for bloqueado. Os gateways são escolhidos segundo critérios de menor custo[11] ou por acordos bilaterais entre provedores. Há certa complexidade na escolha e configuração das rotas de saída, que podem ser resolvidas pela aplicação das técnicas constantes na seção 3.11. Chamadas realizadas de volta para identificador de chamada apresentado na PSTN são rejeitadas ou atendidas por um serviço de URA ou IVR, pelos quais pode-se discar o número interno do assinante na rede VoIP, permitindo, de certa forma, chamadas na direção PSTN-VoIP, sem o emprego de um plano de numeração E.164 que permita ligações diretas.

7.2.2.4 Cenário IV – Telefonia IP bidirecional

A quarta e última modalidade, sendo a mais completa em termos de serviços prestados, inclui as facilidades da primeira ou segunda e integralmente a terceira. Esta permite ao assinante receber e realizar chamadas para a rede PSTN. O assinante deve ser universalmente identificável na rede PSTN, atribuindo-lhe um número telefônico, conforme prescrito pela norma ITU-T E.164, com um código país e um código de área. Vários provedores, por simplicidade e escala, alocam na configuração do terminal instalado no cliente, seja ele um Softphone ou um Hardphone, apenas um URI, que pode ser o próprio e-mail do assinante. A associação ente o número telefônico e o identificador URI são realizados centralizadamente

[11] Chamado de “least cost routing”

através de um serviço de translação de endereços E.164 para SIP URI, como o ENUM (seção 3.11.1, pág. 52).

7.2.3 Quanto aos aspectos da arquitetura

Do ponto de vista da arquitetura, o protocolo SIP pode operar perfeitamente sobre um esquema P2P (sem uso de *Proxy*), possibilitando a um usuário realizar sessões de mídia apenas pelo conhecimento do endereço IP do terminal remoto (seção 3.4, pág.35). Neste caso, as dificuldades para monitorar a comunicação são maiores. Inclusive, a Internet provê mecanismos para despersonalizar o endereço IP empregado nas comunicações, como as redes TOR[12]. Os critérios e estratégias para rastrear tais chamadas estão fora do escopo desta dissertação. Algumas idéias existem neste sentido, e podem ser consultadas na referência [28].

A arquitetura analisada neste trabalho é estrita para o protocolo SIP e é similar àquela desenhada na Figura 3-2. Mais restritivamente, o funcionamento do *Proxy* (seção 3.5, pág.38) deve ocorrer de modo *stateful* de forma a manter o estado pelo menos da transação de início da chamada. O roteamento de chamadas pelo critério *loose-route* do SIP 3.6, pág. 41), é um critério a ser preservado se for do interesse manter a bilhetagem completa da chamada, determinando o momento da desconexão e duração das conversações; de outro modo ele é indiferente. Os terminais devem ser configurados para aceitar chamadas estritamente pelos *Proxies* da operadora, através da configuração do *Proxy de entrada* e do *Proxy de saída*, aplicados no terminal do assinante. De outro modo, não haveria controle completo sobre o caminho que a sinalização SIP percorreria durante a transação que estabelece o diálogo, ao menos. Em termos práticos, as restrições acima são vantajosas para um provedor, uma vez que há um interesse prático na realização da correta bilhetagem das sessões: o faturamento.

7.3 Quebra do Sigilo da Bilhetagem

Adotando as restrições acima, se for do interesse Público apenas preservar o registro das ligações telefônicas, para efeitos da "quebra do sigilo sobre a conta telefônica", ou bilhetagem, basta regulamentar o volume de dados que deve ser mantido, em termos de histórico. No termo técnico estes dados são conhecidos como *Call Detail Record* (CDR). Para os provedores de serviço isto significa estabelecer a capacidade de armazenamento necessária e a implantação dos mecanismos *loose-route* como padrão no roteamento do *Proxy* de

[12] Detalhes em http://en.wikipedia.org/wiki/Tor_%28anonymity_network%29

tarifação. Tecnicamente esta é uma facilidade disponível por padrão nos servidores *Proxy* de mercado. O acúmulo e processamento do CDR pode requerer eventualmente a implantação de software adicional.

7.4 Interceptação de Tráfego de Voz sobre IP

Uma das características inatas da rede PSTN é o fato da localização física dos terminais ser praticamente fixa. Nesta condição baseia-se o serviço E911[13] nas redes estadunidenses. Pelo identificador da chamada – CallerID – os sistemas fazem uma busca reversa no catálogo telefônico e retornam o endereço do assinante em questão de segundos. Isto é possível porque a relação entre o número E.164 e o local onde se encontra o terminal tem associação espacial de longo prazo; muitas vezes décadas. Mesmo as redes celulares tornaram-se capazes de fornecer a localização[14] do assinante dentro da precisão necessária para o objetivo do serviço, conforme definido na recomendação FCC E-911, Phase II[15].

Ao contrário das redes PSTN e celulares, os usuários de uma rede VoIP podem estar localizados em qualquer local geográfico sem contar com sistemas de posicionamento. Do ponto de vista funcional, à exceção das características de QoS, não importa a localização do assinante e sim que ele faça o registro devidamente autenticado para efetuar e receber chamadas. Muitas vezes o assinante pode ser nômade ou pode estar em trânsito, levando a uma relação entre o local físico e o terminal com prazo curto de duração, anulando completamente as investidas convencionais. Isto é um conceito fundamental e irrevogável de mobilidade provocado pelo advento da Internet; inefável nas redes de Voz sobre IP. Então, a interceptação não pode ocorrer nos moldes convencionais; e a depender do grau de mobilidade talvez nem possa ocorrer.

A estratégia mais indicada para permitir sucesso na interceptação é posicionar um ou mais pontos de *encontro na rede*, de tal forma que o *Proxy* possa redirecionar o ponto de contato da mídia RTP através deste dispositivo. Esta sugestão foi apresentada nos trabalhos [15], [16], [17]. Esta abordagem, do ponto de vista da sinalização levaria a solução plena do problema, não fossem eventuais retardos introduzidos pelo desvio de tráfego RTP, o que poderia

[13] O órgão FCC estadunidense estabeleceu recentemente que os provedores VoIP devem anunciar claramente para os assinantes que o serviço ora sendo assinado não conta com as facilidades E.911. Há estudos em andamento para permitir um grau mínimo de localização de assinantes na rede IP, como mostra a Schulzrinne, H. e Arabshian, K. em "**Providing Emergency Services in Internet Telephony**", disponível em http://www1.cs.columbia.edu/~knarig/911.pdf.

[14] Um sistema em operação para localização de assinantes celulares em uso pelas operadoras T-Mobile e Cingular Wireless pode ser visto em http://www.trueposition.com/e-911.php.

[15] Documento pode ser consultado em http://www.fcc.gov/Bureaus/Wireless/News_Releases/2001/nwl0127a.pdf.

prejudicar a característica fundamental da interceptação: realizá-la sem que os interlocutores a percebam. Caso o retardo introduzido seja insuficiente para ser percebido pelos interlocutores, então a sinalização SIP pode prestar-se facilmente a este papel. Através de manipulações pelos *Proxies* nos dados contidos no SDP, o caminho do RTP pode ser alterado de forma a inserir um agente B2BUA ou dispositivo de captura de pacotes, com capacidade de armazenamento para gravar as mídias RTP ou SRTP que o atravessem. Agentes B2BUA foram apresentados na seção 3.13.

Proxies previamente configurados para interceptar chamadas de um assinante devem alterar os anexos SDP dentro das mensagens INVITE, 18X, PRACK e ACK (seção 3.9, pág. 46), modificando o ponto de contato da mídia RTP, para ambas as partes, incluindo um agente B2BUA como uma espécie de *man-in-the-middle*. Alterando-se os campos "c=" do SDP e as portas RTP sinalizadas pelo campo "m=", em ambas as direções, é a única medida necessária para direcionar o RTP/SRTP de ambos os lados. Basta ao agente B2BUA tomar ciência desta sinalização para que ele possa realizar as traduções de endereço e colocar as devidas portas RTP em modo de escuta, de forma transparente, executando a gravação dos referidos pacotes para posterior uso. Há exemplos de softwares que permitem gravar o RTP para posterior interpretação, como o "*rtpmon*", "*rtpdump*" e "*rtpplay*", que podem ser encontrados através do endereço ftp://mm-ftp.cs.berkeley.edu/pub/rtpmon/. A referência [17] alerta que um usuário mais experiente poderia deduzir a presença do elemento de escuta, chegando a propor que todo o tráfego de um provedor passe por um dispositivo no meio do caminho.

O fluxo de mensagens apresentado na Figura 7-1 mostra como deve ser o roteamento das mensagens SIP de forma a viabilizar o ponto de encontro. "Y" e "X" são as duas[16] interfaces do elemento B2BUA. C= IPx, Px representam o ponto de contado do RTP, ou seja o IP e a Porta do dispositivo x, que são traduzidos ao passar pelo agente. Pelo desenho o elemento B2BUA simula, para o User Agent (UA) do lado "A", ser o UA do lado "B" e vice-versa, inclusive interceptando a mídia.

[16] Uma interface seria suficiente. Duas são apenas para ilustrar.

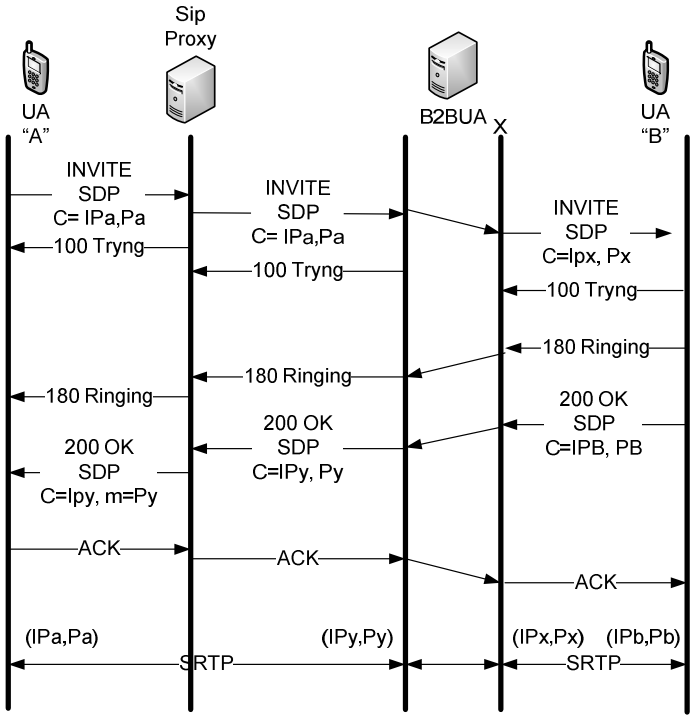

Figura 7-1 - Desviando o tráfego da mídia

Como dito anteriormente, um ponto relevante no processo de escuta é que os usuários não percebam que o processo está sendo conduzido. O fator que poderia ser determinante para este ponto seria inserir um roteamento na mídia cujo caminho incluísse excessivo retardo na conversação. De forma a conter esta condição, seria necessária a instalação de agentes B2BUA distribuídos junto aos diversos provedores de acesso, de forma a reduzir o caminho inserido, podendo implicar em custo adicional para o valor do serviço, a menos que a função fosse incorporada nos elementos de redes, isto é, os roteadores. Há uma crítica forte contra as iniciativas em tentar alterar algum elemento roteador para suportar esta facilidade, pelas possíveis vulnerabilidades que poderiam ser acrescidas ao sistema, como a autora Landau, S. [18] descreve.

Uma alternativa para conter a necessidade de reservar um ou mais recursos de hardware pelos provedores, minimizando o custo adicional para os assinantes, seria alocar um B2BUA por demanda, conforme determinação "ad-hoc" do provedor onde o assinante encontra-se no momento. O provedor hospedando momentaneamente o assinante, então, faria a instalação do B2BUA num ponto estratégico e informaria os endereços alocados ao B2BUA, para programação do processo de roteamento das chamadas no *Proxy* do prestador do serviço. A solução naturalmente exige a cooperação entre entidades administrativas distintas e

eventualmente não seria de fácil execução. Ter que lidar com diversos provedores na coordenação de um processo de escuta é uma das principais motivações da criação das diversas arquiteturas apresentadas na seção 2.5.

O último ponto a ressaltar no aspecto da interpretação é que a sinalização não pode estar protegida pelo protocolo S/MIME, visto na seção 4.5. Esta é única forma de proteção da sinalização de forma fim a fim padronizada e empregada no mercado. A sua presença impede que o SDP seja interpretado e que os parâmetros da conexão sejam alterados. Neste caso a opção seria tentar seguir os moldes da telefonia convencional: derivar o tráfego da mídia em algum ponto do trajeto, uma vez que não seria possível alterar as informações contidas no SDP.

7.5 Interpretação do Tráfego de Voz sobre IP

Caso a mídia seja transportada sem criptografia, as considerações acima são suficientes também para a interpretação. De outra forma é necessário resgatar as chaves criptográficas utilizadas durante a sessão SRTP. A forma como a chave pode ser resgatada depende do protocolo de gerenciamento de chaves adotado, podendo ou não ter solução.

A possibilidade de resgate pressupõe a existência de um sistema de custódia de chaves, que é um local onde as chaves são armazenadas para posterior acesso pelo pessoal legalmente qualificado, utilizando um processo seguro não discutido em detalhes neste trabalho.

Na seção 7.6 serão analisadas as possibilidades de resgate das chaves criptográficas pelo emprego dos protocolos que estão em uso no mercado e padronizados na IETF. Nas seções subseqüentes será dada ênfase ao método proposto.

7.6 Opções atuais para lidar com o resgate de chaves

Esta seção apresenta as possibilidades que os protocolos de segurança definidos pelo IETF podem oferecer para preservar as chaves de sessão ao final de uma comunicação de voz empregando o protocolo SRTP.

O uso do SDES, apresentado na seção 5.2, pág. 78, quando empregado, permite a qualquer *Proxy* registrar as chaves negociadas para a proteção da mídia. A única hipótese em proteger o SDES (e todo o SDP), até mesmo dos *Proxies*, é pelo emprego do S/MIME. As demais opções possíveis para proteção da sinalização, que são o TLS e o DTLS, cumprem o papel de

evitar que as chaves e demais informações sofram problemas com a integridade e confidencialidade no trajeto entre o UAC e o *Proxy*, e entre *proxies* sucessivos, até o destino, mas não interferem na interpretação das mensagens pelos *Proxies* no trajeto da sinalização.

O uso do MIKEY com PKI, apresentado na seção 5.1.3, pág. 75, pode inviabilizar qualquer tentativa de armazenamento das chaves criptográficas, por conta do emprego de certificados emitidos para os interlocutores, a menos que as chaves privadas sejam armazenadas em um sistema de custódia, para posterior resgate. Como o TGK é protegido por um envelope, e este envelope pela chave pública do *Responder*, no KMAC, torna-se inviável o acesso ao conteúdo.

O MIKEY com DH também depende dos certificados, mas mesmo que as chaves fossem custodiadas, não haveria como descobrir as chaves de sessão, por conta das características de *Perfect Forward Secrecy* (PFS) do Diffie-Hellman [38]. A existência dos certificados é apenas para que as partes autentiquem uma a outra, evitando a vulnerabilidade do "*man-in-the-middle*" do Diffie-Hellman.

A tabela abaixo sintetiza as possibilidades de interpretação do conteúdo pelo emprego das opções de segurança da sinalização e mídia, definidos pela IETF e apresentados no Capítulo 5.

Protocolo	**Chave recuperável.**	**Vulnerável ao MitM**	**Funcional[17] entre Domínios**	**Requer PKI**
SDES	Sim, Sem TLS/DTLS	Sim	Sim, mas Inseguro.	Não
MIKEY PSK	Sim, pelos administradores[18]	Não	Não	Não
MIKEY PKI	Não, há menos que as chaves privadas sejam depositadas em um sistema de custódia.	Não	Sim, Com certificados rastreáveis.	Sim
MIKEY DH	Não	Não	Sim	Sim
ZRTP	Não	Improvável	Improvável	Não

Tabela 7-1 - Comparação entre padrões

[17] Funcional no sentido de ser prático e ao mesmo tempo seguro, principalmente entre domínios distintos
[18] Pelo administrador da infra-estrutura VoIP.

Então, o único protocolo que permitiria o resgate da chave seria o SDES, caso este não seja protegido pelo S/MIME, mostrando uma severa restrição na perpetuação da escuta legal caso os demais padrões sejam adotados em larga escala pelo mercado.

A adoção do SDES, com ou sem S/MIME, ou das formas do MIKEY, são mutuamente exclusivas e dependem da especificação do projetista de uma determinada infra-estrutura de Voz sobre IP com SIP. Muitos fabricantes codificaram todas as opções citadas em seus produtos, deixando a cargo do mercado a decisão da escolha. Tal decisão depende do grau de segurança que se pretende aplicar a uma da infra-estrutura, envolvendo riscos e custos. O SDES é uma opção menos segura, por permitir que as chaves possam ser capturadas por quem processar a sinalização, porém fácil de implementar, não necessitando o uso de uma infra-estrutura de chaves públicas. O MIKEY com PSK sofre de problemas graves de escala, e as demais opções do MIKEY dependem do investimento de uma infra-estrutura de chaves públicas (PKI), para administrar os certificados aprovisionados para os terminais dos usuários. Para completar, certificados digitais geram custos adicionais por assinante aprovisionado e devem ser renovados periodicamente.

O ZRTP pode ser inviável em casos onde as partes de uma chamada telefônica não se conheçam previamente, mas desejam estabelecer um canal seguro, haja vista que uma não conhece o timbre da voz da outra.

7.7 Método proposto

Uma forma ainda não padronizada para suplantar o problema da preservação das chaves mestras, ainda tendo a capacidade de funcionar entre múltiplos domínios, sem requerer uso de PKI, mantendo a complexidade computacional baixa, pelo uso apenas de algoritmos simétricos, seria utilizar uma combinação do protocolo SRTP com o Kerberos.

O Kerberos pode substituir parcialmente a função de um gerenciador de chaves descrito no Capítulo 5 para emitir chaves compartilhadas, numa espécie de “chave compartilhada por demanda”. Uma vez que as partes tenham estabelecido as chaves compartilhadas elas poderão utilizá-las para viabilizar a ação do protocolo MIKEY com PSK ou então para proteger a passagem de informações pelo SDES. Eventualmente o Kerberos poderia dispensar por completo o uso do MIKEY ou SDES, mas faltaria desenvolver uma forma extra de passagem das demais informações necessárias na configuração dos *crypto-suites* necessários para o SRTP.

A proposta vai analisar apenas a integração do Kerberos com o SDES, valendo-se da capacidade do SDES em ser extensível. A mesma idéia, entretanto, poderia ser facilmente aplicável ao MIKEY-PSK. A preferência pelo SDES advém da sua simplicidade na implementação.

Basicamente a idéia consiste em acrescentar uma estrutura de distribuição de chaves – KDC, para estabelecer as chaves compartilhadas entre usuários. Estas chaves seriam usadas para proteger a chave contida na negociação do SDES, que por sua vez, ficaria responsável por configurar o SRTP. A forma como SDES opera com o SRTP já foi descrita na seção 5.2, pág. 78.

Se o KDC for desenhado para remeter as chaves compartilhadas, enviadas aos terminais via os *tickets*, também para um sistema de custódia, então será possível reverter o processo de criptografia. Para a comunicação entre o KDC e o sistema de custódia, pode-se usar um método padronizado qualquer para estabelecer um canal seguro, como o uso de IPSEC ou SSL devidamente autenticado com certificado da agência custodiante.

Visando melhor atender aos possíveis cenários de integração de VoIP descritos na seção 7.2.2, a solução proposta foi divida em duas. A primeira, quando ambos os usuários são assinantes de serviços de voz sobre IP, atendendo aos cenários VoIP restrito e integrado[19]. A segunda, sendo a mais abrangente das soluções, serve para os cenários que envolvem telefonia IP, ou quando o UA não suportar reter a chave de longo prazo usada para associação ao KDC. A segunda pode englobar a primeira, mas por agilidade no processo de obtenção da chave compartilhada, ambas são apresentadas em separado.

O Kerberos possui um mecanismo específico para emissão de *tickets* em redes P2P, visto na seção 6.10. Do ponto de vista do protocolo Kerberos, segundo a definição da RFC4810, a diferença básica na autenticação cliente-servidor para autenticação entre dois clientes (P2P), ocorre por conta do cliente não conseguir reter a chave de longo prazo. A RFC 4810 comenta que após o primeiro "*login*" alguns dispositivos podem não reter a chave de longo prazo[20] ficando impossibilitados de receber *tickets* de serviço. Esta diferenciação nada tem haver com a natureza da aplicação que será beneficiada pela chave compartilhada[21], no processo de autenticação. Por sorte, manter a chave de longo prazo por um cliente SIP pode não ser necessariamente um problema, haja vista que os clientes precisarão manter suas senhas de

[19] Desde que usando o protocolo padrão SIP.
[20] Seja por segurança, seja por capacidade computacional restrita.
[21] Ou seja, não guarda relação com o fato da aplicação que requer ser autenticada é do estilo cliente-servidor ou P2P.

autenticação para responder às solicitações de autenticação feitas pelos servidores de registro e *Proxies*, com visto na seção 4.2, para as mensagens de INVITE, REGISTER e outras.

Apesar desta dissertação não abordar a questão de como o KDC sincronizará sua base de *Principals* com os URI's cadastrados em um servidor de Registro SIP, é totalmente plausível que as duas informações estejam sincronizadas por um mecanismo extra. De modo que a senha de autenticação no SIP, para realizar o método da seção 4.2, seja a mesma utilizada para derivar a chave de longo prazo do Kerberos. Assim não ocorreria o problema citado na seção 6.10. Mas numa suposição do dispositivo terminal não conseguir reter as senhas, a segunda solução resolveria o problema.

Mas o principal fator motivador da segunda solução advém da dificuldade de um SIP UA, representado fisicamente por um *softphone* ou *hardphone* não conseguir determinar, a priori, onde sua chamada irá terminar no mundo IP quando o destino final for um assinante da rede PSTN, principalmente se for considerado um ambiente onde ocorra a integração de vários provedores compartilhando infra-estruturas de *gateways* PSTN. Os mecanismos automáticos para determinação do melhor *gateway* para dar continuidade a uma chamada no mundo PSTN foram apresentados na seção 3.11, página 52. Protocolos como TRIP e o TGREP podem ocultar completamente a identificação do *gateway* que cumprirá este papel, dada as possibilidades de agregação e consolidação das rotas telefônicas propagadas entre domínios distintos, não dando condições a um *Proxy* de saída de um provedor ter a visibilidade exata sobre a rede associada à sua. A Internet é um favorecedor na criação de acordos de cooperação mútua entre provedores VoIP, no sentido de compartilharem infra-estruturas de *gateways* em regiões diferentes, de forma que os recursos operados diretamente por um provedor sejam reduzidos, sem que ele perca alternativas de chamadas de menor custo em locais onde não há ponto de presença próprio. Afinal, este pode ser considerado o principal atrativo de marketing da atualidade: "Ligações para mais de duzentas cidades com custo de ligação local". Alguns provedores vão além: fornecem abordagens "*All you can eat*" para as cidades com ponto de "presença", tornando a capilaridade um fator de sobrevivência no mercado: quanto mais pontos de presença, melhor, o que justifica as parcerias.

Não sabendo onde uma chamada será terminada na rede IP, sendo o destino final dela na rede PSTN, especialmente para o Kerberos, faz sentido pensar em iniciar a solicitação para abertura de um canal seguro por um processo reverso: do *gateway* para o assinante VoIP. Muitos dos mecanismos apresentados na seção 7.6 não levaram em consideração este possibilidade.

Poder-se-ia questionar porque cifrar chamadas que atravessam *gateways* PSTN, uma vez que elas **não** vão seguir cifradas no mundo da telefonia tradicional. Há pelo menos uma boa razão para isto: a indeterminação. Supondo que alguém desejasse "espiar" as chamadas de um usuário VoIP. Se a abordagem fosse coletar as conversações na rede PSTN, então o *agente* enfrentaria alguns desafios: principalmente pela indeterminação do número discado, do *gateway* escolhido pela rede, do circuito alocado pelo *gateway* (entre os diversos que ele possua), pela diversidade de operadoras PSTN conectadas e de localidades. Com tudo isto, o agente poderia concluir que o melhor seria interceptar seu tráfego ainda enquanto IP, realizando um *wiretapping* em seu acesso local à Internet. Os mecanismos mais modernos de escuta na rede de telefonia digital baseiam-se em implantar dispositivos de *wiretapping* integrados nas centrais telefônicas de assinante, exatamente para combater o problema da dificuldade em espiar uma linha de assinante, como discutido em 1.2.2. Se espiar a linha é complexo, o que dirá o tronco. Este é uma das origens para criação das legislações como o CALEA (seção 2.3, pág. 18).

Adicionalmente, algumas facilidades disponíveis internamente nas redes de VoIP também podem conduzir a mesma situação de indeterminação, como o redirecionamento para um serviço de caixa postal, entre outros. Por estes motivos a autenticação reversa (seção 7.7.2, pág. 111) é considerada mais completa, porém mais complexa.

7.7.1 Solução para usuários de VoIP

Para tornar as explicações mais concisas, as seguintes definições são adotadas:

- "**A**" representa o URI do assinante chamador;
- "**B**" representa o URI do assinante chamado;
- $\mathbf{P_A}$ ou $\mathbf{P_B}$ são os respectivos *Principals;*

Neste contexto o assinante de destino seria alguém que pudesse ser identificado por um URI[22], da mesma forma que o assinante de origem. Supondo que ambos **não** tenham problemas em armazenar as chaves de longo prazo necessárias para o Kerberos, então quando **A** desejasse realizar uma chamada segura, poderia solicitar um *Ticket* para acesso ao *Principal* diretamente associado ao URI de interesse. A regra de construção do *Principal* é flexível o suficiente para acomodar a sintaxe do URI SIP, de modo que o mapeamento poderia ser

[22] Lembrando que uma URI pode ser um texto como o e-mail ou uma seqüência numérica como um E.164.

direto. Mais detalhes sobre este assunto podem ser vistos na seção 7.7.6. Não há necessidade em identificar o URI do ponto de contato, até porque o assinante de destino poderá estar registrado simultaneamente em vários terminais. O *Proxy* se encarregará em realizar um *fork* [23] da chamada quando necessário.

Para o assinante **B** o ponto importante e necessário é que o UA conheça a chave de longo prazo, quando o usuário fizer a autenticação no terminal. Não faz diferença em quantos terminais **B** está registrado.

Para **A**, o primeiro passo é obter o TGT com o seu respectivo AS. Para em seguida solicitar um TGS para acesso ao *Principal* do usuário **B**, ou TGS_B.

O fluxo de mensagens na Figura 7-2 ilustra como seria a realização de uma chamada entre componentes de uma rede SIP/KDC, independente de estarem ou não no mesmo domínio[24], para a realização de uma chamada segura. Na proposta, a mensagem de INVITE do SIP ficaria encarregada de transportar tanto TGS_B quanto o SDP contendo uma extensão do SDES, utilizado para transportar as informações sobre o contexto criptográfico, contendo todos os parâmetros de configuração adicionais para o SRTP. A extensão proposta para o SDES está descrita na seção 7.7.4. Como apresentado na figura, as informações contidas nas mensagens Kerberos KRB_AP_REQ e KRB_AP_REP seriam suficientes para proteger o SDES modificado. É o conhecimento da chave compartilhada de **A** e **B** com o KDC, que faz com que o mecanismo seja seguro.

Considerando que o TGT é obtido na inicialização do dispositivo, ou quando da autenticação de um usuário num determinado terminal SIP, então o processo proposto acrescenta apenas duas novas mensagens, pela inclusão do Kerberos, em relação àquelas necessárias para estabelecer uma chamada SIP convencional. Estas duas mensagens são aquelas necessárias para estabelecer as chaves que serão utilizadas entre as partes, na conversação segura, sendo as duas primeiras da Figura 7-2. Para futuras chamadas entre esses dois assinantes, dentro do período de validade do *ticket*, as duas mensagens adicionais seriam completamente dispensáveis, a menos que tais informações tenham que ser descartadas por falta de memória nos terminais.

Se for necessário o envio de novas mensagens SDES, dentro do período de duração da chamada, para, por exemplo, realizar o *re-keying* imposto pelo contexto do SRTP (ver seção

[23] Se for necessário realizar a busca de um assinante em diversos terminais simultaneamente, como descrito na seção 3.5, página 38.
[24] Desde que estabelecida uma relação de confiança vista no Capítulo 7.

4.8, página 65), é importante que novas *subkeys*, contidas nas mensagens Kerberos, sejam negociadas.

A Figura 7-2, mostra a troca de mensagens de uma forma macro. Para detalhes de como o KRB_AP_REQ e o KRB_AP_REP seguem na mensagem SIP deve-se consultar a seção 7.7.3. Para detalhes de como o SDES deve ser alterado, deve-se consultar a seção 7.7.4.

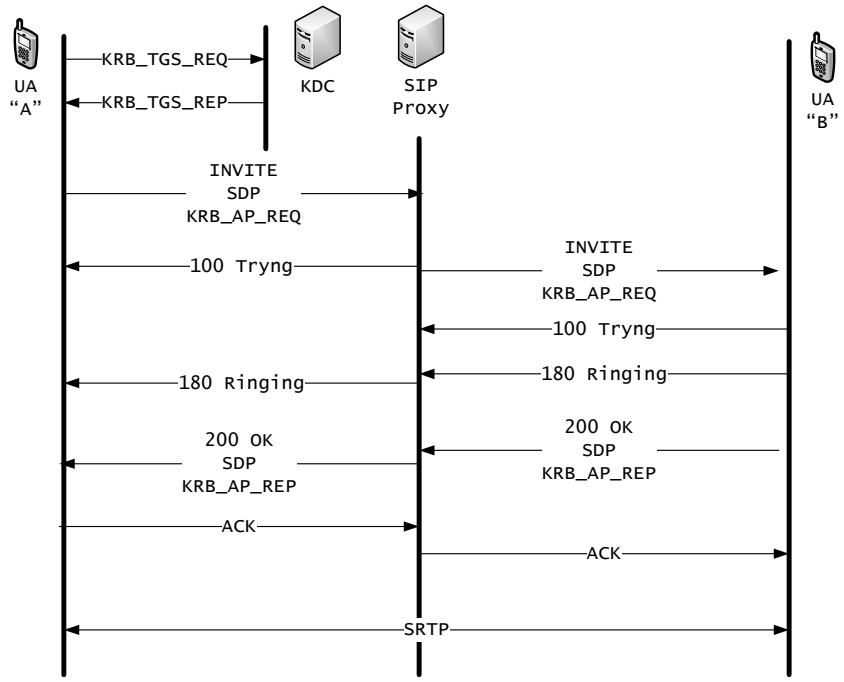

Figura 7-2 – Solução A, para SIP e Kerberos

Numa condição onde **B** esteja em outro domínio KDC (*realm)*, identificável por seu URI, **A** terá que solicitar um *Ticket* no TGS pertencente ao re*a*lm de B, através de um *cross-realm ticket*, requisitado no domínio de **A**. De forma a gerar o seguinte fluxo inicial de mensagens, a mais em relação à figura anterior.

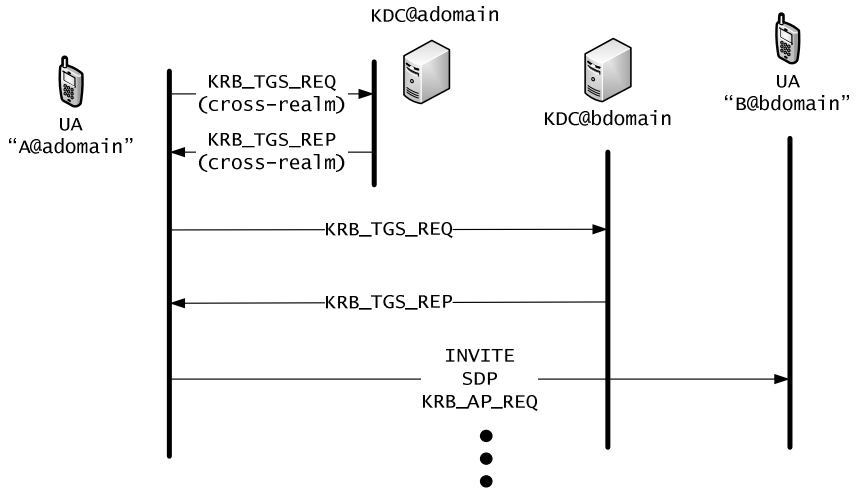

Figura 7-3 - Solução A com *cross-realm*

Da mesma forma que no caso anterior, futuras comunicações ente **A** e **B** poderão dispensar este processo contanto que o *Ticket* expedito pelo domínio de **B** não expire.

7.7.2 Solução com autenticação reversa

Este caso é aplicável quando as chamadas, em um dos lados, terminarem em gateways com a rede PSTN[25]. Este caso substitui por completo solução descrita em 7.7.1, mas acrescenta uma latência adicional na negociação do canal seguro.

Quando **A** iniciar a chamada de INVITE, ele deverá anexar o seu TGT, como forma de sinalização do seu interesse no estabelecimento reverso da negociação para o canal SRTP. O INVITE poderá ou não seguir com a parte do SDP referente à descrição da mídia e CODECS suportados, se for requerido abertura prematura do áudio para detecção de eventos de sinalização dentro banda, conforme apresentado na seção 3.8 página 43, podendo inclusive conter ofertas para o SDES no formato nativo. Levando a duas condições de negociação, descritas a seguir.

Primeiro, se o SDP for enviado logo no INVITE, as trocas de mensagem do Kerberos deverão aguardar o final do processo de oferta-resposta, encerrado com o envio de respostas de aprovisionamento da classe 18[1-9], seguindo a recomendação do protocolo, como apresentado na seção 3.9.1.1. Em seguida deve ser enviada uma mensagem de UPDATE, vista na seção 3.9.1.2, pelo usuário **B**, requerendo alteração no processo criptográfico com uso das proteções adicionadas pelo Kerberos.

[25] Ou gateways para outros tipos de sinalização.

Num segundo caso, se o SDP não for enviado no INVITE, a negociação do SDP associado ao Kerberos poderá ser realizada na troca das mensagens 200 OK (referente ao INVITE) e o ACK, mas sob pena de não ser possível abrir o canal de áudio prematuramente. Enviar mensagens SDP no ACK é um ponto facultativo se o INVITE não contiver um SDP, conforme seção 3.8. Eventualmente um UPDATE poderia ser enviado numa mensagem intermediária para permitir a abertura do canal de áudio prematuramente.

A Figura 7-4 ilustra como seria a troca de mensagens quando houver interesse por **B** em estabelecer previamente a mídia encerrando a oferta que seguiu no INVITE. O *ticket* obtido por **B** é enviado na mensagem KRB_AP_REQ, por onde é feita a passagem da *subkey*. Na mesma mensagem o SDP pode conter a extensão protegida do SDES, uma vez que **A** poderá decodificar KRB_AP_REQ com a chave associada ao TGT. É recomendado que **B** ajuste o *flag* MUTUAL_REQUIRED para forçar o retorno de KRB_AP_REP, que de outro modo não seria remetido. É importante também que **A** forneça sua própria *subkey* que protegerá o SDP na direção de **B**, evitando reutilizar a mesma chave.

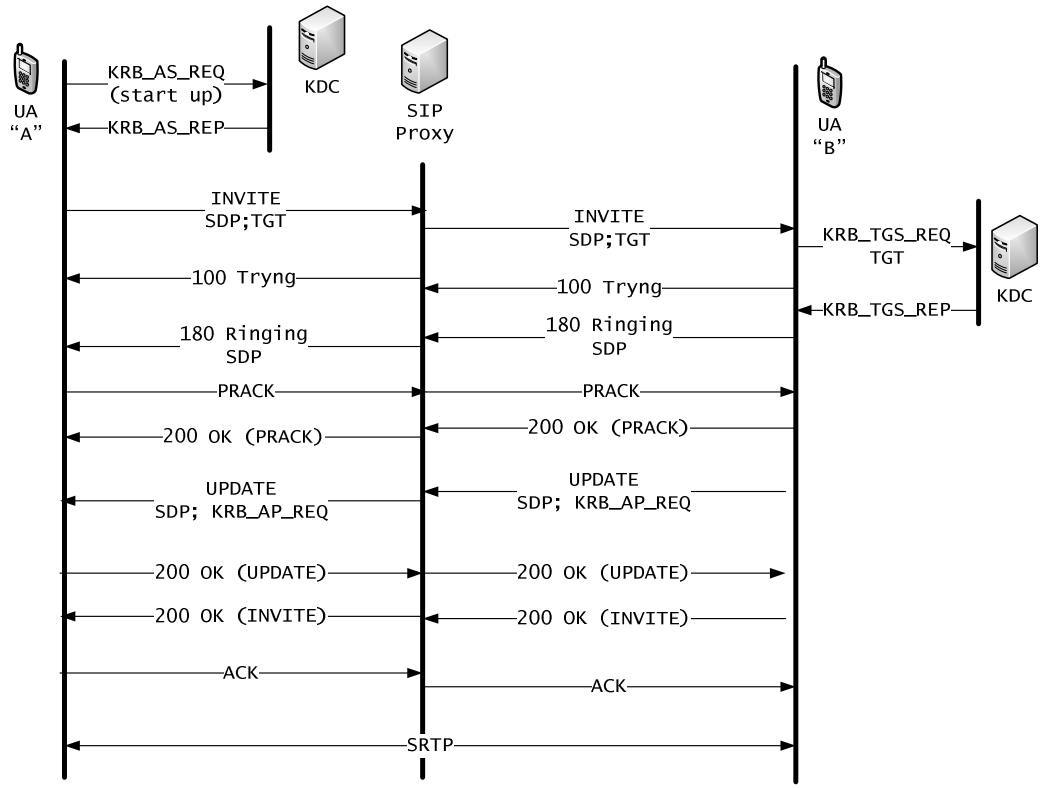

Figura 7-4 - Troca por UPDATE

Futuras atualizações da chave para atender ao *re-keying* podem ser efetivadas com novos UPDATES contendo novas *subkeys*.

Se **A** e **B** pertencerem a domínios distintos, **B** deverá identificar no TGT o *realm* que emitiu o *ticket* para então obter um TGT no KDC de **A**, através de uma solicitação pela relação de confiança com seu KDC. O principal de **A** pode ser extraído do campo *From* da transação de INVITE.

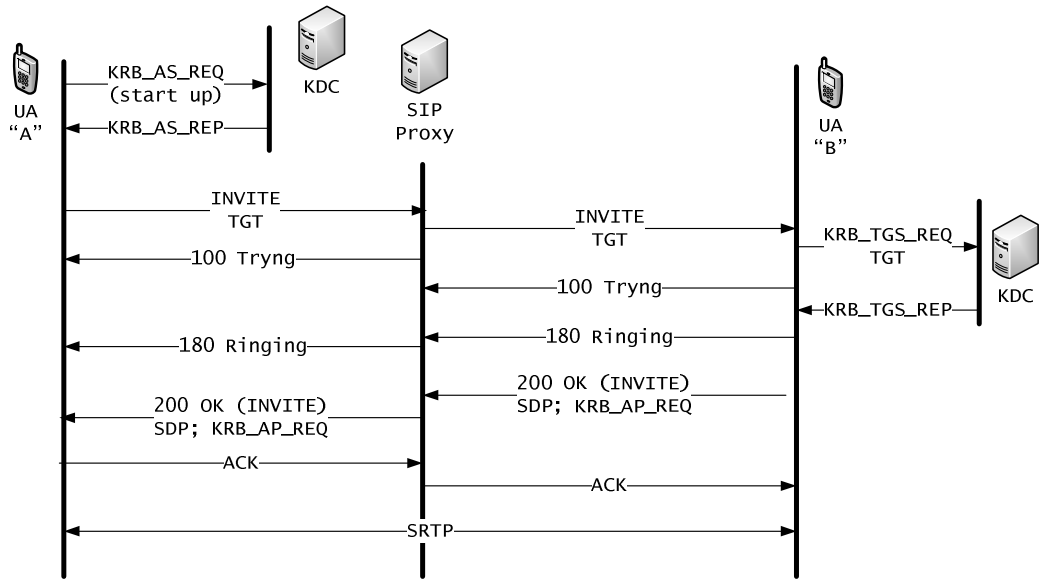

Figura 7-5 - Negociação com INVITE sem SDP

A Figura 7-5 apresenta como seria uma troca de mensagens onde o INVITE não continha descrições de SDP, fazendo com que a oferta-resposta ocorresse direto nas mensagens 200 OK e ACK, como preconiza a RFC 3261.

7.7.3 Transporte de mensagens Kerberos pelo SIP

Como visto na seção 6.6, o Kerberos delega à aplicação transportar as mensagens que não sejam para finalidade de receber um TGT ou solicitar um *ticket*. A proposição para o envio das mensagens KRB_AP_REQ, KRB_AP_REP, KRB_ERROR e KRB_CRED seria repassá-las sobre o próprio SIP, reduzindo o volume de mensagens trocadas. A RFC 3261 considera o padrão MIME *multipart* para envio de múltiplos corpos na mesma mensagem algo totalmente normal, o que é uma herança benéfica do SMTP. O Kerberos codifica suas mensagens usando o ASN.1, transformadas em um bloco binário após o processamento pelas rotinas da API Kerberos. Para este resultado ser transportado em MIME, deve ser codificado em base64.

A Mensagem 7-1 apresenta como seria tal representação de um pacote que seguisse as recomendações desta seção, transportando o Kerberos como MIME, montada com base nos exemplos de C. Jennings [59].

O receptor desta mensagem, ao decodificar o texto em base64, faria uma passagem a rotina Kerberos dentro do terminal SIP, que por sua vez teria como identificar a mensagem e proceder com a chamada da API correta, por conta do campo identificador que antecede qualquer mensagem Kerberos.

```
INVITE sip:bob@b.example.com SIP/2.0
To: <sip:bob@b.example.com>
From: <sip:alice@a.example.com>;tag=4bba1f0d
Via: SIP/2.0/UDP
     127.0.0.1:5070;branch=z9hG4bK-c87542-558422834-1--c87542-;rport
Call-ID: 132bb895019d4536
CSeq: 1 INVITE
Contact: <sip:alice@a.example.com:5070>
Max-Forwards: 70
MIME-version: 1.0
Content-type: multipart/mixed; boundary="frontier"

--frontier
Content-type: application/sdp
Content-Length: 158

v=0
o=Alfa 2890844526 2890844526 IN IP4 a.example.com
s=Phone Call
c=IN IP4 200.1.5.1
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000

--frontier
Content-type: application/krb5
Content-transfer-encoding: base64

PGh0bWw+CiAgPGhlYWQ+CiAgPC9oZWFkPgogIDxib2R5PgogICAgPHA+VGhpcyBpcyB0aGUg
Ym9keSBvZiB0aGUgbWVzc2FnZS48L3A+CiAgPC9ib2R5Pgo8L2h0bWw+Cg==
--frontier—
```

Mensagem 7-1 - Mensagens Kerberos via MIME.

Na ocasião em que este texto estava sendo redigido não havia na IANA um registro para o tipo MIME *application/krb5* ou qualquer texto relativo ao Kerberos, mas nada impede seu registro.

7.7.4 O relacionamento com o SDP e a oferta e resposta

O processo do Kerberos integrado ao SIP compreende parte do problema. Ele sozinho não conseguiria configurar o SRTP nem estabelecer um denominador comum nos *crypto-suites* suportados por ambos interlocutores. Assim como o SIP reaproveitou o HTTP e o SMTP, da mesma forma não é necessário redefinir um novo protocolo completo de gerenciamento de chaves. Para tal, este trabalho reaproveita a definição do SDES, apresentado na seção 5.2, que possui uma previsão para acomodar necessidades futuras.

Todas as proposições para os *crypto-suites* serão reaproveitadas, conforme 5.2.2. A mudança fundamental ocorre por um novo método `key-method` para indicar que a chave do SDES estará protegida por uma operação XOR com a *subkey* acordada pelo Kerberos, na respectiva direção que flui a oferta ou resposta. Por este motivo é fundamental não repetir *subkeys* quando da oferta de novas chaves. A referência [24], na página 13, alerta para os efeitos negativos na repetição da chave ou uso de chaves menores que o conteúdo a ser cifrado quando aplicado a operação de soma módulo dois, bit a bit. É importante que as *subkeys* sejam geradas em tamanho superior aos tamanhos das chaves mestras e *salting* concatenadas, que nos contextos presentes na seção 5.2.2 representam 128 + 112 bits, num total de 30 bytes. O Kerberos recomenda, para maior incerteza na geração das *subkeys*, utilizar as chaves de sessão no processo da função pseudo-aleatória.

Uma mensagem SDES estendida teria a seguinte aparência: `a=crypto:tag crypto-suite key-params <session-param>`. Para os valores `a=crypto:tag crypto-suite` e <`session-param`>, as definições seriam as mesmas da seção 5.2, pág. 78. No valor `key-params` seria incluído a indicação de que as chaves estão cifradas em XOR com a chave compartilhada na direção da mensagem, se esta existir. `Key-params` é definido como `<key-method> ":" <key-info>`. Atualmente o único `key-method` definido é o "`inline`" e `key-info` segue com todas as informações para composição do contexto. A proposta aqui é criar um novo `key-method`, que pode ser registrado na IANA, como por exemplo, o valor "`xor-inline`" e preservar a sintaxe de `key-info`. Desta forma uma mensagem SDES teria a aparência apresentada na Figura 7-1, também montada com base nos exemplos de C. Jannings [59].

```
INVITE sip:alice@example.com SIP/2.0
To: <sip:alice@example.com>
```

```
From: <sip:bob@example.com>;tag=4bba1f0d
Via: SIP/2.0/UDP
     100.1.1.2:5060;branch=z9hG4bK-c87542-558422834-1--c87542-;rport
Call-ID: 132bb895019d4536
CSeq: 1 INVITE
Contact: <sip:bob@example.com:5070>
Max-Forwards: 70
MIME-version: 1.0
Content-type: multipart/mixed; boundary="frontier"

--frontier
Content-type: application/sdp
Content-Length: 234

v=0
o=Alfa 2890844526 2890844526 IN IP4 100.1.1.2
s=Phone Call
c=IN IP4 100.1.1.2
t=0 0
m=audio 49170 RTP/AVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
  xor-inline:WVNfX19zZW1jdGwgKCkgewkyMjA7fQp9CnVubGVz|2^20|1:4
a=crypto:2 F8_128_HMAC_SHA1_80
    inline:MTIzNDU2Nzg5QUJDREUwMTIzNDU2Nzg5QUJjZGVm|2^20|1:4;
    inline:QUJjZGVmMTIzNDU2Nzg5QUJDREUwMTIzNDU2Nzg5|2^20|2:4
a=rtpmap:0 PCMU/8000

--frontier
Content-type: application/krb5
Content-transfer-encoding: base64

jk+wkgoo+RHk90O16gO7FPFq4QIT/PGh0bWw+CiAgPGhlYWQ+CiAgPC9oZWFkPgogIDxib2R5Pg
ogICAgPHA+VGhpcyBpcyB0aGUgYm9keSBvZiB0aGUgbWVzc2FnZS48L3A+CiAgPC9ib2R5Pgo8L
2h0bWw+Cg==
--frontier—
```

Mensagem 7-2 - Mensagem de envio das opções SDES

O exemplo mostra múltiplas ofertas pelo assinante **B**, para proteger a mídia **m=** logo acima, numeradas pelos *tags* 1 e 2. A primeira opção é a preferencial e indica que a chave está protegida. A segunda segue o modelo convencional do SDES e propõe duas chaves mestras com MKI=1 e 2 em texto limpo. Esta seria uma forma de proposição num modelo de transição onde alguns clientes não suportassem Kerberos.

O segundo anexo do SIP contém o KRB_AP_REQ, com a *subkey* utilizada para transformação do `xor-inline`.

Ao receber esta mensagem, **A** interpretaria a oferta e escolheria uma das opções possíveis na ordem sugerida. Caso **A** fosse um UA "Kerberized" [26], certamente escolheria a opção 1. No retorno, **A** enviaria para **B** a mensagem contendo apenas o valor escolhido para a mídia

[26] Jargão utilizado na Internet para aplicações que foram portadas para suportar Kerberos.

em questão, propondo uma nova chave de sessão na sua direção, também devidamente protegida, como no exemplo que segue:

```
200 OK SIP/2.0
To: <sip:alice@example.com>
From: <sip:bob@example.com>;tag=4bba1f0d
Via: SIP/2.0/UDP
     100.1.1.1:5060;branch=z9hG4bK-c87542-558422834-1--c87542-;rport
Call-ID: 132bb895019d4536
CSeq: 1 INVITE
Contact: <sip:bob@example.com:5070>
Max-Forwards: 70
MIME-version: 1.0
Content-type: multipart/mixed; boundary="frontier"

--frontier
Content-type: application/sdp
Content-Length: 234

v=0
o=- 289082323 234442334 IN IP4 100.1.1.1
s=-
c=IN IP4 100.1.1.1
t=0 0
m=audio 53210 RTP/AVP 0
a=crypto:1 AES_CM_128_HMAC_SHA1_80
  xor-inline: PS1uQCVeeCFCanVmcjkpPywjNWhcYD0mXXtxaVBR|2^20|1:4
a=rtpmap:0 PCMU/8000

--frontier
Content-type: application/krb5
Content-transfer-encoding: base64

jk+wkgoo+RHk90O16gO7FPFq4QIT/+U4XNIfgzW80RI0f6fr6ShiS/h2T
TNxfaYrkddA3DJEIlZvep175/PSfL91DqItU8T+wBwhHTV2Iw8O1s+NVCHXV
Ym9keSBvZiB0aGUgbWVzc2FnZS48L3A+CiAgPC9ib2R5Pgo8L2h0bWw+Cg==
--frontier—
```

Mensagem 7-3 - Mensagem SDP de retorno com campo xor-inline

7.7.5 Modelo Integrado do UA com o Kerberos

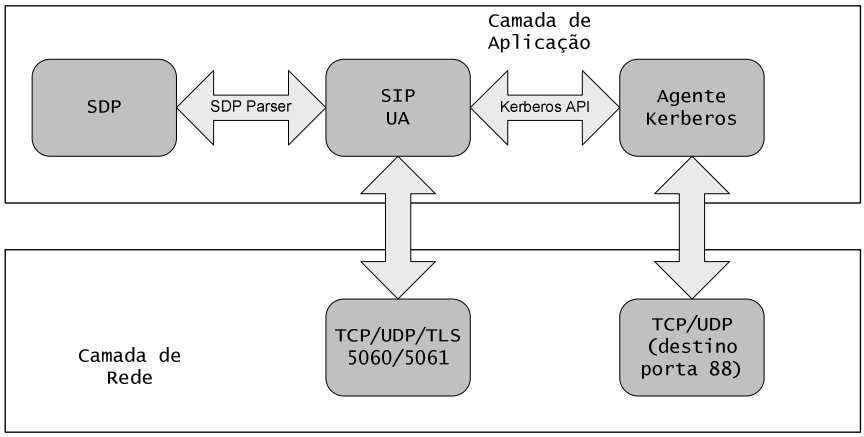

Figura 7-6 - SIP UA "Kerberized"

A figura acima mostra como seria um SIP UA "kerberized"[27]. A presente proposta não menciona alterações em servidores *Proxy* e de redirecionamento, além disto, há um bom grau de compatibilidade na comunicação com UA que não suportem o mesmo mecanismo, permitindo uma adoção progressiva da proposição. O papel do Kerberos poderia ir além, incluindo proposições para autenticação dos UA nos serviços SIP.

7.7.6 A compatibilização do Principal e o SIP URI

Um ponto importante é compatibilizar o URI SIP com o cadastro no KDC, de forma que os *tickets* sejam emitidos para o destino correto da chamada. Do ponto de vista do Kerberos, pelo fato da sintaxe do Principal ser muito flexível, é totalmente viável compatibilizar os dois esquemas. Adicionalmente, vale ressaltar que a decisão do *Principal* para o qual será solicitado o *Ticket* pode ser alterada pelo resultado de eventuais mensagens de redirecionamento enviadas pelo *Proxy* (seção 3.9, pág. 46).

Um aspecto não comentado é a sincronização entre a base de *Principals* registrados e a base do serviço de registro SIP. Naturalmente ambas poderiam ser sincronizadas, inclusive na geração de chaves de longo prazo geradas pelas mesmas senhas de autenticação nos serviços SIP. Mas isto é uma atividade focada na implementação e não no objetivo de preservar as chaves para escuta legal.

[27] Este é o jargão usado na Internet para referenciar uma aplicação qualquer que tenha sido alterada para suportar Kerberos.

7.7.7 A logística na interpretação do SRTP

O procedimento para interpretar a comunicação sobre o SRTP inicia pelo resgate da chave compartilhada $\mathbf{K_{AB}}$, arquivada em algum sistema de custódia, atribuída para aos dois agentes usuários **A** e **B**, gerada pelo KDC. Também é fundamental o registro das mensagens de INVITE, UPDATE, 200 OK e ACK que contiverem anexos MIME com as mensagens do tipo KRB_AR_REQ e KRB_AP_REP, ou descrições SDES com *inline*, se o cliente não acordar pelo *xor-inline* na transação de estabelecimento do diálogo. Sem este armazenamento seria impossível reverter o processo.

De posse do material acima e da mídia SRTP completa, incluindo o cabeçalho RTP, coletada pelos métodos da seção 7.4 ou outra forma qualquer de *wiretapping*, proceder-se-ia da seguinte forma:

a. Se **A** e **B** acordaram **não** usar o SDES "estendido", basta utilizar o material que passou em texto aberto e passar para o passo "e".

b. Senão, deve-se resgatar as chaves $\mathbf{K_{AB}}$ do sistema de custódia.

c. Decifrar as mensagens KRB_AR_REQ e KRB_AR_REP com a chave $\mathbf{K_{AB}}$ para recuperar as duas *subkeys*.

d. Reverter o XOR no campo *xor-inline* para cada direção.

e. Atentar para a taxa de *re-keying* e para o volume de dados coletados. No contexto definido no exemplo da Mensagem 7-3, o *re-keying* ocorrerá a cada 2^{20} bytes. Numa conversação telefônica utilizando o CODEC G.711, segundo a Tabela 3-6, são enviados 50x80 = 400 bytes/segundo, que levaria a um *re-keyng* em aproximadamente 43 minutos.

f. Deve-se então utilizar o material do contexto para gerar a chave de sessão e autenticação derivadas pelo SRTP, conforme seção 4.8, que junto com as informações do cabeçalho vão permitir reverter o processo.

Um ponto importante neste processo é que não é necessário ter conhecimento das chaves e senhas entre os assinantes e seus respectivos KDC's e servidores *Proxies*, utilizadas nos processos de autenticação. Indo além, é importante que tais senhas sejam de conhecimento apenas dos usuários, não permitindo a custodia das mesmas.

7.8 Modificações necessárias no Kerberos

As implementações Kerberos implementam cachês para armazenamento temporário dos *tickets* emitidos, realizando o descarte pela validade respectiva, de forma que, para recuperação das chaves, há a necessidade de armazenar tais informações de forma duradoura em um sistema de custódia. Segundo referencia [71], a forma e o local onde os tickets do KDC são armazenados dependem da implementação. A implementação original realizada pelo MIT utiliza o próprio sistema de arquivos do servidor KDC para a função de cachê.

Uma forma viável e segura para o processo de custódia é alterar o código do Kerberos de modo que este submeta a um serviço de custódia os *tickets* devidamente criptografados pelo uso de uma chave pública contida num certificado digital emitido para o serviço custodiante, facilitando a recuperação do conteúdo. Num sistema baseado em bibliotecas abertas isto seria tão simples como integrar as funções de escrita do cachê para em paralelo remeter tais informações com o uso, por exemplo, da API disponibilizada pelo projeto OpenSSL [73].

7.9 Complementaridades sobre a técnica

O objetivo do capítulo é analisar as possibilidades de interceptar e preservar as chaves criptográficas, pelo uso dos protocolos existentes, ou do protocolo proposto. Mas para efetivar um instrumento real de escuta legal ainda são necessários vários passos como a regulamentação, a implementação, a criação de processos operacionais de coordenação das atividades de LI, dentre outras etapas, que estão além deste trabalho. Apenas como sugestão para uma implementação de uma arquitetura, é recomendável manter as chaves de sessão expedidas pelo KDC, armazenadas de forma protegida do acesso alheio, realizando imediatamente o envio das chaves para um sistema de custódia, através de um canal seguro, autenticado por um certificado da agência custodiante, como visto em 7.8.

Adotando-se tal proposição, a possibilidade de interpretar o conteúdo ficaria condicionada a uma operação tripla, oferecendo mais consistência ao dispositivo legal. Em primeiro lugar, a mídia deveria ser capturada, ainda cifrada. Tal ação, em geral, é uma responsabilidade das diligências, acompanhadas pelas equipes técnicas dos provedores de acesso ou serviço, usando as sugestões contidas na seção 7.4. Em segundo lugar, seria necessário que a prestadora do serviço de VoIP fornecesse os registros da sinalização trocadas pelo assinante, contendo as chaves negociadas. Adotando o método proposto, tais chaves estariam protegidas

e invioláveis para quem as capturasse. Por fim, seria necessário que as chaves expedidas pelo KDC, nos *tickets* que ele emitiu para um determinado assinante, fossem recuperadas no sistema de custódia. Esta última etapa poderia ser executada somente pela agência custodiante, dona do certificado digital que protegeu as informações do KDC. Somente quando três fatores estivessem juntos a interpretação semântica poderia ocorrer.

Isto faria com que as diligências e as prestadoras de serviço não conseguissem interpretar semanticamente o conteúdo enquanto coletassem as informações, ainda que involuntariamente. Os autos seriam de fato apartados e o sigilo das diligencias e das gravações estaria preservado. Em outro local, reservado, quem realmente precisasse desta informação, procederia juntar as três partes e proceder com a interpretação.

Capítulo 8

Conclusões

8.1 Avaliação do mecanismo proposto

Seja qual for o partido que as literaturas consultadas tenham tomado sobre a possibilidade técnica de efetivar um sistema de escuta legal, tacitamente têm entre si uma concordância: não negam o valor positivo da intenção sobre a preservação da segurança pública. Das fontes consultadas que se propõem a viabilizar um mecanismo, há sensíveis diferenças entre as abordagens sobre como implementar estratégias de interceptação em ambientes de Voz sobre IP. Muitas das iniciativas parecem preocupar-se com a coordenação do processo de forma ágil, a ponto de apresentarem arquiteturas e interfaces de acesso para a resposta rápida às necessidades das agências de investigação. Apesar das fontes aceitarem como ponto crítico no processo de interceptação as dificuldades face ao uso da criptografia, poucas sugerem algum mecanismo que possa auxiliar na preservação das chaves. Das fontes consultadas apenas os artigos [13]e [14] apresentam uma solução para o problema.

Partindo deste ponto, o presente estudo procurou mecanismos que pudessem realizar a preservação de chaves e, ao mesmo tempo, assegurassem a privacidade nas comunicações. Com base na análise dos padrões de segurança da mídia, o estudo concluiu que das possibilidades existentes, nenhuma poderia ser aplicada num sistema de escuta legal de forma a responder à altura os problemas que um sistema de custódia pode gerar. Assim, a pesquisa buscou uma alternativa que pudesse responder a estas questões. Para tanto, algumas premissas foram adotadas: a primeira em não alterar equipamentos ou dispositivos presentes no núcleo da rede de forma a permitir a derivação de tráfego, reduzindo o risco adicionado. Segundo

evitar a criação de um novo protocolo, e sim aproveitar as técnicas experimentadas e existentes. Desta forma, houve a convergência para o aproveitamento do Kerberos.

De forma a apoiar a conclusão de que a alternativa proposta pode oferecer um diferencial, se escolhida como um mecanismo de preservação de chaves, em um sistema de escuta legal, as seções seguintes fazem uma comparação do método proposto com as técnicas padronizadas, avaliando as medidas necessárias para utilizá-las em sistemas de custódia de chaves.

8.1.1 Comparação da sugestão com SDES

O único mecanismo padronizado que permite o processo de interceptação sem necessitar criar um ambiente adicional para custódia das chaves seria o SDES. Porém, o SDES sofre de uma fraqueza quase incompatível com o motivo de sua existência: ele realiza a passagem das chaves criptográficas sem proteção. Mesmo com a sugestão feita pelos autores em proteger o caminho da sinalização com mecanismos como o TLS ou o DTLS, ainda assim, não haveria garantia sobre a continuidade deste processo quando chamadas são realizadas através de vários domínios. Políticas implantadas em mecanismos de *Proxy* podem fazer com que determinados trechos da sinalização sejam utilizados mesmo sem o suporte de tais mecanismos. Ainda assim, o fato das chaves passarem pelos equipamentos *proxies* de múltiplos provedores pode constituir por si só um risco.

Há sugestões para manter o SDES protegido por S/MIME. Mas esta estratégia pode inviabilizar a operação correta do protocolo SIP quando atravessando um mecanismo de NAT, visto na seção 3.12, pág. 54. O NAT é amplamente utilizado e vai permanecer até a substituição do IPv4 pelo IPv6. Ainda assim, o uso desta combinação inviabilizaria o resgate das chaves, a menos que fosse complementado por um sistema de custódia das chaves privadas utilizadas nos certificados digitais, podendo incorrer em alguns problemas descritos na seção seguinte.

A vulnerabilidade do SDES pode favorecer que o mercado escolha outras opções padronizadas nas implementações que requeiram maior segurança.

Se por um lado, a adoção do SDES é a opção mais econômica para acesso às chaves de sessão, por outro, a possibilidade de realizá-la inteiramente por quem interceptar o tráfego torna-a similar à forma existente na telefonia convencional, onde qualquer um pode fazer uma escuta, autorizada ou não. A sua combinação com o Kerberos, de acordo com proposta

apresentada em 7.7, pode conceder-lhe o crédito necessário tanto para atender as implementações mais exigentes, quanto para mecanismos de escuta legal, pelo acréscimo de controle no acesso das chaves. Isto pode representar um diferencial numa possível aceitação pública de uma regulamentação sobre a matéria.

8.1.2 Comparação da sugestão com MIKEY

Conforme 5.1, o MIKEY pode suportar três alternativas de negociação das chaves: o *pre-shared* (PSK), via infra-estrutura de chaves públicas (PKI) ou Diffie-Hellman. A única condição que possibilitaria a continuidade da LI, nestes casos[1], seria empregar um mecanismo de custódia, que pudesse reter as chaves compartilhadas ou as chaves privadas associadas às chaves públicas dos certificados.

O uso do MIKEY com chaves compartilhadas peca por fator de escala. É pouco recomendável dar crédito a uma chave secreta compartilhada entre múltiplos terminais, no longo prazo. Pessoas mudam, administradoras trocam de empresas, equipamentos são enviados para reparo, enfim, há uma diversidade de razões práticas para retirar o crédito do método. A situação pode ficar pior, quando pensamos em ambientes integrados com múltiplos provedores de serviços, com estruturas organizacionais distintas. Também não é prático pensar em utilizar chaves compartilhadas configuradas por demanda, pois há certa complexidade em comunicar este tipo de informação de forma segura. Integrar o MIKEY-PSK com um sistema de custódia requeria o emprego de um agente em cada terminal SIP, dado que não há um vínculo com um sistema central de gerenciamento de chave, trazendo problemas de autenticação e escala graves.

O MIKEY com PKI, por empregar infra-estrutura de chaves públicas, é extremamente robusto, permitindo a autenticação mútua e requerendo um volume pequeno de transações para estabelecimento do canal seguro. Porém, o emprego de PKI acarreta em custos de investimento e operacional. O uso de certificados é um sucesso em autenticação de servidores e transações eletrônicas pela Web, mas pode ser complexo empregá-los em aparelhos telefônicos, que ocorrem aos milhões. Quando os certificados ficarem amplamente difundidos como mecanismo de identificação pessoal, pode ser que o desdobramento para proteção de sistemas complementares seja um passo fácil de realizar, diluindo o custo do investimento. Custo é um problema no acirrado mercado de telecomunicações. Se os assinantes dos

[1] Exceção para o Diffie-Hellman.

sistemas de telefonia convencional reclamam da obrigatoriedade de pagar assinaturas dos serviços telefônicas tradicionais, provavelmente vão agir da mesma forma para custear os valores de investimento nos certificados. Há ainda o custo operacional em decorrência da limitação da validade dos certificados.

Contudo, o contraponto mais complexo no emprego do MIKEY-PKI combinado com um sistema de custódia é a condição de reter as chaves privadas relacionadas aos certificados emitidos. O procedimento atual de geração de certificados não envolve a retenção das chaves privadas, que na maior parte dos casos, nem chegam ao conhecimento das certificadoras[2]. Uma regulamentação neste sentido envolveria uma mudança num processo consolidado mundialmente. Possivelmente, muitas certificadoras não têm infra-estrutura adequada para esta função. Outro ponto negativo desta hipótese diz respeito à eventual anulação da característica de não-repúdio, se o certificado for utilizado também para assinaturas digitais[3], dado que a chave privada seria de conhecimento de outros que não seu proprietário, como discutido na seção 2.2.

O MIKEY com Diffie-Hellman impossibilita a custódia das chaves, como apresentado na seção 7.6.

8.1.3 Comparação com o emprego de chaves mestras

A utilização de chave mestra, cujo conceito foi apresentado na seção 2.6.2, poderia substituir completamente o Kerberos, como forma de proteção do meio de comunicação entre um sistema de custódia e o UA SIP, permitindo o depósito das chaves de sessão empregadas na criptografia da mídia[4].

Esta idéia, entretanto, oferece desvantagens se for observado que o comprometimento da chave mestra poderia conceder o direito a terceiros em derivar todas as chaves empregadas no processo de comunicação entre o sistema custódia e o cliente SIP, e, por conseqüência, todas as chaves criptográficas utilizadas na proteção das comunicações. Apesar do Kerberos também depender de uma informação compartilhada entre as partes: a chave de longo prazo

[2] Tecnicamente, certificados são assinados mediante a apresentação de um arquivo de requisição de assinatura, no formato PKCS #10 definido na RFC 2315, que não inclui a chave privada do solicitante.
[3] Eventualmente o mesmo certificado utilizado pelo assinante para registrar-se num serviço VoIP, via um *smartcard*, SIM Card ou outro dispositivo, poderia ser utilizado para outros fins. Esta característica poderia ser desejável para unificação e redução de custos, facilitando o emprego de PKI.
[4] Neste caso, as chaves de sessão para proteção da mídia seriam estabelecidas pelo próprio protocolo de gerenciamento de chaves do SIP, como o SDES, MIKEY e o ZRTP, que então deveriam ser depositadas no sistema de custódia, por intermédio de uma chave mestra.

para o processo de autenticação[5], o comprometimento das chaves de longo prazo não permitiria acesso às comunicações passadas, pois uma não é derivada da outra. Existe um padrão para emprego de chaves RSA [66] no processo de autenticação, que pode fazer com que nem a chave de longo prazo seja necessária ser armazenada no KDC, reduzindo drasticamente os efeitos do comprometimento de um determinado servidor, dado ser este um dos fatores de maior rejeição na adoção de um sistema de custódia, conforme seção 2.2.

A favor do Kerberos estão: a possibilidade de trabalhar entre domínios distintos, com integração pelos mecanismos de DNS; a possibilidade de ser transportado pelo próprio SIP, ao passo que um sistema de chaves mestras necessitaria de desenvolvimento ou aproveitamento de protocolos que auxiliem tanto na gestão de relações de confiança quanto no transporte das informações protegidas; no Kerberos não é necessário que dois terminais revalidem as chaves obtidas para comunicação entre partes, desde que o ticket obtido seja válido, reduzindo o tráfego para ponto central de custódia, ao passo que no mecanismo de chave mestra, para cada chave de sessão gerada pelos mecanismos de gerenciamento de chaves do SIP, deve ocorrer uma transmissão para o sistema de custódia; o Kerberos não necessita manter informações de estado dos clientes, ao passo que no sistema de chave mestra as chaves são geradas em seqüência e os lados devem manter informações sobre a próxima chave derivada a ser gerada, evitando renegociar o estado inicial; por fim, o Kerberos tem embutido um sistema de autenticação.

8.1.4 Comparação com o emprego de IBE

O *Identity-based Encryption*, apresentado na seção 2.6.2, oferece uma alternativa interessante para ocupar a função do Kerberos no papel de distribuição de chaves, dado que o IBE propõe-se a definir a chave pública de uma comunicação pelo conhecimento do identificador do usuário, que poderia ser a própria URI SIP. Uma vez gerada uma chave pública o *Private Key Generator* (PKG) poderia operar de forma muito parecida com a proposição do Kerberos neste trabalho, realizando a remessa das chaves privadas para um sistema de custódia[6]. Conforme a RFC 5091 os serviços PKG/PPS[7] são previstos para operar de forma integrada com o DNS, facilitando o processo de localização e obtenção dos *parâmetros complementares* associados a um PKG. Também, de forma muito parecida ao Kerberos, não há necessidade em manter informações de estado dos clientes.

[5] Não há uma chave global de autenticação e sim diversas chaves compartilhadas, uma para cada cliente de um KDC.
[6] Poderia também apenas registrar e enviar as chaves públicas, caso a chave mestra do PKG fosse depositada em custódia.
[7] *public parameter server*

A favor do Kerberos estão: este já possui um processo de autenticação embutido no protocolo, ao passo que o PKG precisa do auxílio de mecanismos como o TLS, sugerido na RFC 5091, tanto para autenticar a comunicação entre usuário originador e o PPS, quando autenticar a comunicação entre o recebedor e o PKG, o que pode acarretar no acréscimo de tempo durante o estabelecimento das chamadas; o Kerberos combinado com o SDP pode dispensar o emprego de PKI, ao passo que por necessitar do TLS, o IBE mantém o vínculo com o uso de PKI na solução de VoIP; através do Kerberos é possível negociar uma chave simétrica, com uso imediato pelo SRTP, reduzindo o número de mensagens para estabelecimento da chamada, ao passo que uma infra-estrutura IBE objetiva a geração de chaves públicas e privadas, cabendo aos clientes de uma comunicação em voz sobre IP negociar as chaves simétricas para uso pelo protocolo SRTP; o Kerberos estabelece as chaves caso a caso, para emissão dos *tickets*, e estas chaves são descartadas ao final da validade de um *Ticket Granting Ticket* (TGT), não havendo uma chave tal que se comprometida poderia permitir a interpretação indevida de todas as mensagens, ao passo que os *parâmetros complementares* são associados à chave mestra de um PKG, podendo o conhecimento indevido desta chave permitir a terceiros reproduzirem todas as chaves privadas, para todas as comunicações de voz que foram protegidas durante o período de vigência da respectiva chave mestra.

8.1.5 Como o método colabora para um sistema de custódia

A solução proposta neste trabalho ocupa uma lacuna aberta pelas técnicas atuais, sendo uma candidata potencial para responder algumas das questões citadas no item 2.2.

Com o emprego do Kerberos haveria a possibilidade de manter as chaves necessárias para reverter os processos, sem uma dependência do conhecimento das chaves de longo prazo utilizadas no processo de autenticação, gerando um efeito de *forward secrecy*, parecido como o Diffie-Hellman [38], onde a exposição indevida de uma chave não comprometeria as chaves anteriores nem as futuras, negociadas durante as conversações de um usuário.

A adoção do procedimento recomendado na seção 7.8, forçando a junção de três partes coletadas, possivelmente, em pontos distintos, pode auxiliar no combate aos efeitos negativos na concentração da administração do sistema de custódia.

Kerberos é uma arquitetura aberta e disponível há vários anos. Seus padrões estão disponibilizados publicamente, tendo sido experimentado em redes de grande porte. Por

exemplo, a Microsoft emprega Kerberos no processo de autenticação dos serviços de rede baseados no *Active Directory*, presente deste a entrada do sistema operacional Windows 2000. O *Active Directory* é amplamente difundido em ambientes corporativos de grande porte, mostrando, na prática, a capacidade do Kerberos em comportar grande escala de tráfego, disponibilidade e balanceamento. A integração com SDES não requer mudanças na interface deste com o SRTP, reduzindo o número de alterações necessárias nos clientes para implementar a sugestão.

O modelo, como herança do SDES, aceita que múltiplas condições de negociação de chaves sejam ofertadas durante a sinalização, como visto em 7.7.4, facilitando a integração com sistemas que não suportem a sugestão e permitindo a continuidade operacional pela falta de disponibilidade da infra-estrutura de KDC. Comparado com as demais soluções padronizadas, o custo é provavelmente inferior àquele necessário para implantar uma PKI, e possivelmente comparável ao SDES.

Também, como dito anteriormente, as chaves utilizadas nos processos de autenticação dos terminais SIP e no sistema KDC não precisam ser custodiadas. Há sugestões, conforme a RFC 4556[8] em utilizar Kerberos e PKI para o processo inicial de autenticação, fortalecendo a motivação para a interdependência das chaves de longo prazo com o material custodiado, evitando o problema relativo ao não-repúdio.

8.1.6 Desvantagens da sugestão

O primeiro a se destacar é que o Kerberos depende que os participantes da rede estejam minimamente sincronizados, a priori. Mesmo com a abordagem de corrigir a defasagem de tempo após a obtenção do primeiro *ticket*, ainda assim há um valor mínimo de discrepância que pode ser aceito pelo sistema.

O segundo ponto, e o mais relevante, é que as chaves compartilhadas no Kerberos são geradas sobre senhas. E senhas estão sujeitas a ataques de dicionário, furto e engenharia social. Políticas empregadas pelo mecanismo de troca de senhas do Kerberos podem gerar algumas regras de formação, para fugir do óbvio, ou para reduzir o tempo de vida de uma senha. Mecanismos alternativos têm sido estudados para melhorar este aspecto, dentre eles a recente RFC4556. O próprio processo de autenticação do SIP sofre deste problema, e carece de alternativas apropriadas.

[8] Zhu. L.; Tung. B. Public Key Cryptography for Initial Authentication in Kerberos (PKINIT) , RFC4556, June 2006.

Por fim, a aplicação do mecanismo é dependente da análise do assunto pelo Estado e uma posterior regulamentação. Certamente a regulamentação será um incentivo para que organizações e indivíduos sejam compelidos a buscar métodos alternativos para manutenção da comunicação segura. Este é um efeito descrito na RFC 2804[40] e que não pode ser combatido com uma solução de engenharia.

8.2 Trabalhos futuros

8.2.1 Possibilidades de uso dos atributos do Kerberos

Forwardable e *Proxyable Tickets*: O Kerberos pode emitir TGTs que sejam transferíveis de um host ao outro, desde que previamente informado no TGT. Desta forma, pode ser dispensável um novo processo de obtenção de um TGT em um segundo host, facilitando o deslocamento de um usuário de um ponto ao outro. O SIP admite que o usuário registre cumulativamente UA com o seu AOR, e esta facilidade pode ser associada a uma emissão de TGTs reenviáveis. Este aspecto pode ser pesquisado em maior detalhe para redução da necessidade do uso das chaves de longo prazo em ambientes onde talvez seja melhor não utilizá-las.

Também não foi considerada a emissão de *tickets* que possam ser delegados a terceiros, ou seja, *tickets* solicitados por alguém, os quais, quando enviados aos pontos de interesse, podem ser repassados a terceiros. Isso é uma funcionalidade interessante para ser combinada com as facilidades de serviço do SIP, como conferência a três, transferência e conferências em multicast, de forma a diminuir o processo de troca de mensagens através do KDC e para manutenção da criptografia quando utilizando estes serviços.

Para complementar este trabalho, de forma a torná-lo mais consistente como uma proposição de um sistema de interceptação legal, há uma série de aspectos que devem ser pesquisados e desenvolvidos. Por exemplo, há inferências no texto sobre custo que carecem de uma análise econômica apropriada. Também não se discute detalhes de como implementar um mecanismo de custódia seguro e de estratégias para restauração das chaves armazenadas. Não há um estudo que quantifique os fatores de desempenho do protocolo proposto, do tráfego gerado ou da capacidade de armazenamento.

8.2.2 Uso dos serviços telefônicos

Um ambiente telefônico é acompanhado de uma diversidade de serviços como transferência, chamada a três, teleconferência, chefe-secretária, acampamento e outros mais. Este trabalho não analisou como a integração com o Kerberos pode auxiliar na manutenção da criptografia mesmo quando da alteração das características da sessão.

Bibliografia

[1] BRASIL, Constituição (1988). **Constituição da República Federativa do Brasil.** Organização do Texto: Glaucia Carvalho, Regina Quaresma. 4° ed. Rio de Janeiro: Forense Editora, 2006. 925p.

[2] BRASIL, **LEI Nº 9.296, de 24 de julho de 1996**. Presidência da República. Casa Civil. Subchefia para Assuntos Jurídicos. Disponível em: http://www.planalto.gov.br/ccivil/LEIS/L9296.htm.

[3] AT&T Corporation. **The History of AT&T**. Disponível em http://www.corp.att.com/history/.

[4] ITU-T. **Introdution to CCITT Signalling System No. 7**. Q.700. March, 2003. Disponível em http://www.itu.int/rec/T-REC-Q.700-199303-I/en.

[5] Foster, B.; Andreasen, F. **Media Gateway Control Protocol (MGCP) Version 1.0**, RFC3435. January, 2003.

[6] USA, Congress of the United State of America. **Communications Assistance for Law Enforcement Act - CALEA**, Publication Law No 103-414, 18 Stat. 4209, 1994. Disponível em http://www.fcc.gov/calea/.

[7] USA, U.S. Department of Justice, Office of Inspector General, Audit Division. **Implementation of the Communications Assistance for Law Enforcement Act**, Audit Report 06-13, March 2006. Disponível em http://www.usdoj.gov/oig/reports/FBI/a0613/index.htm.

[8] USA, U.S. Department of Justice**. In the Matter of Communications Assistance for Law Enforcement Act and Broadband Access and Services**. ET Docket No. 04-295, RM 10865, November, 2005. Disponível em http://hraunfoss.fcc.gov/edocs_public/attachmatch/FCC-04-187A1.doc.

[9] Bellovin, S; Blaze, M.; Brickell, E.; Brooks, C.; Cerf, V.; Diffie, W.; Landau, S.; Peterson, J.; Treichler, J. **Security Implications of Applying the Communications Assistance to Law Enforcement Act to Voice over IP**. [S.I]: The Information Technology Association of America ITAA, June 2006. Disponível em http://www.itaa.org/news/docs/CALEAVOIPreport.pdf.

[10] Albeson, Hal; Anderson, Ross; Bellovin M, Steven; Benaloh, Josh; Blaze Matt; Diffie, Whitfield; Gilmore, John; Neumann G., Peter; Rivest L., Ronald; Schiller I. Jeffrey; Schneier, Bruce. **The Risks of Key Recovery, Key Escrow, and Trusted Third-Party Encryptation**. Final Report. May 27,1997. Disponível em http://www.schneier.com/paper-key-escrow.html.

[11] Diffie, Whitfield; S, Landau. **Privacy on the Line:** The Politics of Wiretapping and Encryptation**.** Updated and Expanded. USA: Ed. MIT Press, 2007.

[12] Prevelakis, V.; Diomidis,S. **The Athens Affair**. IEEE Spectrum, NY, volume 44, No. 7, Pages 18-25, July 2007.

[13] Thanthry, N.; Pendse, R.; Namuduri, K.; **Voice over IP security and law enforcement,**., pp. 246 – 250, 11-14 Oct. 2005.

[14] Thanthry, N.; Goodrich, C.; Pendse, R.; **CALEA Compliant Secure Voice Over IP System**. In: Carnahan Conferences Security Technology, 40, 2006. Annual IEEE International, Oct. 2006, p. 191 – 196.

[15] Karpagavinayagam, B; State, R; Festo, O.; **Monitoring Architecture for Lawful Interception in VoIP Networks**. In: Second International Conference on Internet Monitoring and Protection (ICIMP 2007), IEEE Computer Society, pag 5, 2007.

[16] Milanovic, A.; Srbljic, S.; Raznjevic, I.; Sladden, D.; Skrobo, D.; Matosevic, I. **Distributed System for Lawful Interception in VoIP Networks**, EUROCON 2003. Computer as a Tool. The IEEE Region 8, Volume 1, Issue , 22-24 Sept. 2003.

[17] Milanovic, A.; Srbljic, S.; Raznjevic, I.; Sladden, D.; Skrobo, D.; Matosevic, I. **Methods for Lawful Interception in IP telephony networks based on H.323**. EUROCON 2003. Computer as a Tool. The IEEE Region 8. Volume 1, Issue , 22-24, , p. 198 – 202, Sept. 2003.

[18] Landau, S. **Security, Wiretapping, and the Internet**. IEEE Security & Privacy. Volume 3, no. 6, Page(s): 26 – 33. Nov - Dez. 2005.

[19] Ono, K; Tachimoto, S. **SIP Signaling Security for end-to-end Communication**, .Asia-Pacific Conference on Communications. APCC'03. Volume 3, n. 21-24 Sept. 2003.

[20] Srinivasan R.; Vaidehi V.; Harish K.; *et al***. Authentication of Signaling in VoIP Applications**. Asia-Pacific Conference on Communications. APCC'05.Pages(s): 530-533, Oct. 2005.

[21] Kiesler, T.; Harn, L. **Cryptographic master-key-generation scheme and its application to public key distribution.** Computers and Digital Techniques, IEE Proceedings. Vol.139, No.3, page(s): 203- 206, May 1992.

[22] D. Richard Kuhn, Thomas J. Walsh, Steffen Fries, **Security Considerations for Voice Over IP Systems, Recommendations of the National Institute of Standards and Technology - NIST**, Special Publication 800-58, 2005. Disponível em http://csrc.nist.gov/publications/nistpubs/800-58/SP800-58-final.pdf.

[23] Perkins, Colin. **RTP: Audio and Video for Internet**. USA: Addison Wesley Professional, June, 2003.

[24] Schneier, Bruce. **Applied Cryptography:** Algorithms and source code in C. Second edition. USA: John Wiley & Sons, 1996.

[25] F. de Alencar, Edgard. **Teoria das Congruências**. São Paulo: Nobel, 1986.

[26] NIST. **Advanced Encryption Standard (AES)**. [S.I.]: Federal Information Processing Standard, FIPS-197, Nov. 2001. 51p. Disponível em: http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf.

[27] Bryam, D.A, Lowekamp, B.B. **Descentralizing SIP**, ACM Queue, March 2007.

[28] S. Chen; X. Wang; S. Jajodia. **On the Anonymity and Traceability of Peer-to-Peer VoIP Calls**. IEEE Network Magazine. Vol 6, No. 5, September, October 2006.

[29] M.Barnes: **A Mechanism to Secure SIP information inserted by Intermediaries**, October 2003, Internet Draft Work in Progress. Disponível em http://search.ietf.org/Internet-drafts/draft-barnes-sipping-sec-inserted-info-01.txt.

[30] Liu, C.; Albitz, P. **DNS e BIND.** 5th Ed. [S.I.]: O'Reilly, May 2006.

[31] George, J. **DNS Configuration**. http://mit.edu/sip/sip.edu/dns.shtml. May 12, 2003.

[32] Hansen, Markus; Hansen, Marit; Moeller, Jan; Rohwer, Thomas; Tolkmit, Carsten, Waack, Henning. **Developing a Legally Compliant Reachability Management System as a Countermeasure against SPIT**, Third Annual VoIP Security Workshop, Berlin, June 2006.

[33] Neto Linhares, Benon. **Da escuta telefônica clandestina.** Disponível em http://jus2.uol.com.br/doutrina/texto.asp?id=192, Sem data.

[34] Jae Cheon Han; Wook Hyun; Sun Ok Park; Il Jin Lee; Mi Young Huh; Shin Gak Kang. **An application level *gateway* for traversal of SIP transaction through NATs.** ICACT 2006. The 8th International Conference on Advanced Communication Technology, 2006. Volume 3, N 20-22 Feb. 2006 Page(s):4 pp.

[35] Aurel Constantinescu, M.; Croitoru, V.; Oana Cernaianu, D.. **NAT/*Firewall* traversal for SIP: issues and solutions.** ISSCS 2005. International Symposium on Signals, Circuits and Systems. 2005.Volume 2, 14-15 July 2005 Page(s):521 – 524.

[36] Freed, N; N. Borenstein. **Multipurpose Internet Mail Extensions (MIME) Part One:** Format of Internet Message Bodies: RFC 2045. November 1996.

[37] Krawczyk, H.; Bellare, M.; R. Canetti. **HMAC: Keyed-Hashing for Message Authentication**, RFC 2104, February 1997.

[38] Rescorla, E. **Diffie-Hellman Key Agreement Method**. RFC 2631, June 1999.

[39] Franks, J.; Hallam-Baker, P.; Hostetler, J.; Lawrence, S.; Leach, P.; Luotonen, A.; L. Stewart. **HTTP Authentication: Basic and Digest Access Authentication**, RFC 2617, June 1999.

[40] IAB; IESG. **IETF Policy on Wiretapping**, RFC 2804, May 2000.

[41] Schulzrinne, H.; S. Petrack. **RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals**, RFC 2833, May 2000.

[42] Rosenberg, J.; Salama, H.; M. Squire. **Telephony Routing over IP (TRIP)**, RFC 3219, January 2002.

[43] Rosenberg, J.; Schulzrinne, H.; Camarillo, G.; Johnston, A.; Peterson, J.; Sparks, R.; Handley, M.; E. Schooler. **SIP: Session Initiation Protocol**, RFC 3261, June 2002.

[44] Rosenberg, J.;H. Schulzrinne. **Reliability of Provisional Responses in Session Initiation Protocol (SIP)**, RFC 3262, June 2002.

[45] Rosenberg, J.; H. Schulzrinne. **Session Initiation Protocol (SIP):** Locating SIP Servers, RFC 3263, June 2002.

[46] Rosenberg, J.; H. Schulzrinne. **An Offer/Answer Model with Session Description Protocol (SDP)**, RFC 3264, June 2002.

[47] Srisuresh, P.; Kuthan, J.; Rosenberg, J.; Molitor, A.; A. Rayhan. **Middlebox communication architecture and framework**, RFC 3303, August 2002.

[48] Rosenberg, J. **The Session Initiation Protocol (SIP) UPDATE Method**, RFC 3311, October 2002.

[49] Rosenberg, J.; Weinberger, J.; Huitema, C.; R. Mahy. **STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)**, RFC 3489, March 2003.

[50] Schulzrinne, H.; Casner, S.; Frederick, R.; V. Jacobson. **RTP: A Transport Protocol for Real-Time Applications**, STD 64, RFC 3550, July 2003.

[51] Schulzrinne, H; S. Casner. **RTP Profile for Audio and Video Conferences with Minimal Control**, STD 65, RFC 3551, July 2003.

[52] Huitema, C.. **Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)**, RFC 3605, October 2003.

[53] Augher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman. **The Secure Real-time Transport Protocol (SRTP)**, RFC 3711, March 2004.

[54] Faltstrom, P; M. Mealling. **The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)**, RFC 3761, April 2004.

[55] Arkko, J.; Carrara, E.; Lindholm, F.; Naslund, M.; K. Norrman. **MIKEY: Multimedia Internet KEYing**, RFC 3830, August 2004.

[56] Baker, F.; Foster, B.; Sharp, C. **Cisco Architecture for Lawful Intercept in IP Networks**. RFC3924, October 2004.

[57] Raeburn, K. **Advanced Encryption Standard (AES) Encryption for Kerberos 5**. RFC3962, February 2005.

[58] Neuman, C.; Yu, T.; Hartman, S.; and K. Raeburn. **The Kerberos Network Authentication Service (V5)**, RFC 4120, July 2005.

[59] Jennings, C. **Example call flows using SIP security mechanisms Internet**. IETF Draft, July 16, 2005.

[60] Crocker, D.; P. Overell. **Augmented BNF for Syntax Specifications: ABNF**, RFC 4234, October 2005.

[61] Freed, N.; J. Klensin. **Media Type Specifications and Registration Procedures**, BCP 13, RFC 4288, December 2005.

[62] Kaufman, C., Ed**. Internet Key Exchange (IKEv2) Protocol**, RFC 4306, December 2005.

[63] Kent, S.; Seo, K. **Security Architecture for the Internet Protocol**. RFC 4301. December 2005.

[64] Dierks, T; E. Rescorla. **The Transport Layer Security (TLS) Protocol Version 1.1**, RFC 4346, April 2006.

[65] Rescorla, E; Modadugu, N. **Datagram Transport Layer Security**, RFC 4347, April 2006.

[66] Zhu. L.; Tung. B. **Public Key Cryptography for Initial Authentication in Kerberos** (PKINIT) , RFC4556, June 2006.

[67] Handley, M.; Jacobson, V.; C. Perkins. **SDP: Session Description Protocol**, RFC 4566, July 2006.

[68] Arkko, J.; Lindholm, F.; Naslund, M.; Norrman, K.; E. Carrara. **Key Management Extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP)**, RFC 4567, July 2006.

[69] Andreasen, F.; Baugher, M.; D. Wing. **Session Description Protocol (SDP) Security Descriptions for Media Streams**, RFC 4568, July 2006.

[70] Andreasen, F.; D. Wing. **Security Preconditions for Session Description Protocol (SDP) Media Streams**, RFC 5027, October 2007.

[71] Bangalore, M.; Kumar, R.; Rosenberg, J.; Salama, H.; D.Shah, **A Telephony *Gateway* REgistration Protocol (TGREP)**, Work in Progress, January 2007.

[72] Garman, Jason. **Kerberos: The Definitive Guide.** [S.I.]: O'Reilly, August 2003.

[73] Open Source toolkit implementing the Secure Sockets Layer (SSL v2/v3) and Transport Layer Security (TLS v1). OpenSSL. Disponível em http://www.openssl.org/.

[74] Shamir, A. **Identity-based cryptosystems and signature schemes**. In Proc. of CRYPTO'84, LNCS vol. 196, pp. 47-53, 1984.

[75] Boneh, D.; Franklin, M. **Identity-based encryption from the Weil pairing**. In Proc. of CRYPTO'01, LNCS vol. 2139, pp. 213-229, 2001.

[76] Cocks, C. An **Identity Based Encryption Scheme Based on Quadratic Residues**, Proceedings of the 8th IMA International Conference on Cryptography and Coding, 2001.

[77] Boyen, X.; Martin, L. **Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems**, RFC 5091, December 2007.