Universidade Federal do Rio de Janeiro

Núcleo de Computação Eletrônica

Fábio Ferrão Ribeiro

SEGURANÇA EM VOZ SOBRE IP: Apresentação e análise dos protocolos SRTP, ZRTP e IPSec

Rio de Janeiro

2010

Fábio Ferrão Ribeiro

SEGURANÇA EM VOZ SOBRE IP:
Apresentação e análise dos protocolos SRTP, ZRTP e IPSec

Monografia apresentada para obtenção do título de Especialista em Gerência de Redes de Computadores no Curso de Pós-Graduação Lato Sensu em Gerência de Redes de Computadores e Tecnologia Internet do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro – NCE/UFRJ.

Orientador:

Moacyr Henrique Cruz de Azevedo M.Sc., UFRJ, Brasil

Rio de Janeiro

2010

Fábio Ferrão Ribeiro

SEGURANÇA EM VOZ SOBRE IP:
Apresentação e análise dos protocolos SRTP, ZRTP e IPSec

Monografia apresentada para obtenção do título de Especialista em Gerência de Redes de Computadores no Curso de Pós-Graduação Lato Sensu em Gerência de Redes de Computadores e Tecnologia Internet do Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro – NCE/UFRJ.

Aprovada em Março de 2010.

Moacyr Henrique Cruz de Azevedo, M.Sc., UFRJ, Brasil

Pelas dificuldades encontradas para concluir esta monografia, dedico a mesma a todos os professores que de uma forma ou de outra estão sempre prontos para orientar seus alunos a atingirem seus objetivos.

AGRADECIMENTOS

Agradeço ao meu Deus, em primeiro lugar, pois Ele é a razão da minha vida e se não fosse a confiança que deposito n'Ele, certamente nem estaria escrevendo estes agradecimentos; à minha esposa pelo apoio e por estar sempre ao meu lado; ao meu filho, meu segundo grande presente de Deus; aos meus pais e meu irmão pela ajuda em momentos difíceis; aos professores pela dedicação em ensinar; e a todos que de alguma forma tornaram este caminho mais fácil de ser percorrido.

RESUMO

RIBEIRO, Fábio Ferrão. **SEGURANÇA EM VOZ SOBRE IP: Apresentação e análise dos protocolos SRTP, ZRTP e IPSec.** Monografia (Especialização em Gerência de Redes e Tecnologia Internet). Núcleo de Computação Eletrônica, Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2010.

Esta monografia visa elucidar a relação entre segurança e desempenho e tem por objetivo mostrar a importância de implementar protocolos que viabilizam a segurança do tráfego VoIP. Para esclarecer tal questão, este trabalho inicia com um histórico sobre a tecnologia VoIP, mostrando suas vantagens e desvantagens. A seguir são apresentados os protocolos de segurança SRTP, ZRTP e IPSec, explicando seus funcionamentos, mostrando suas características e suas funcionalidades. Por fim, este trabalho analisa a aplicação dos protocolos citados em conjunto com o tráfego VoIP.

ABSTRACT

RIBEIRO, Fábio Ferrão. **SEGURANÇA EM VOZ SOBRE IP: Apresentação e análise dos protocolos SRTP, ZRTP e IPSec.** Monografia (Especialização em Gerência de Redes e Tecnologia Internet). Núcleo de Computação Eletrônica, Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2009.

This monograph intend to elucidate the relation between security and performance and her aim is to show the importance of the implement protocols that make the security of VoIP traffic possible. To clear up this question, this work begins with a historic about VoIP technology, showing her advantages and disadvantages. Next this work is showing SRTP, ZRTP and IPSec security protocols, explaining their operation, displaying their features and functionalities. Finally, this work analyses the application of the security protocols with VoIP traffic.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS OU SIGLAS

3DES	Triplo Data Encrytion Standard
AES	Advanced Encryption Standard
AES-128	Advanced Encryption Standard-128 bits
AES-256	Advanced Encryption Standard-256 bits
AH	Authentication Header
AS	Associação de Segurança
CFB	Cipher Feedback
DES	Data Encryption Standard
DH	Diffie-Hellman
DH3k	Diffie-Hellman 3072 bits
DH4k	Diffie-Hellman 4096 bits
DoS	Denial of Service
ESP	Encapsulation Security Payload
HMAC	Hash Message Authentication Code
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IP	Internet Protocol
IPSec	Internet Protocol Security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
MD5	Message-Digest algorithm 5
MIC	Message Integrity Code
MIKEY	Multimedia Internet Keying
MKI	Master Key Index
OFB	Output Feedback
PBX	Private Branch eXchange
PGP	Pretty Good Privacy
QoS	Quality of Service
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol
SAS	Short Authentication String
SDP	Session Description Protocol
SHA1	Secure Hash Algorithm – 160 bits
SHA-256	Secure Hash Algorithm – 256 bits
SIP	Session Initiation Protocol
SRTCP	Secure Real-time Transport Control Protocol
SRTP	Secure Real-time Transport Protocol
TCP	Transport Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecomunications System
VoIP	Voice over IP
VPN	Virtual Private Network
ZRTP	Zimmermann Real-time Transport Protocol

SUMÁRIO

1 INTRODUÇÃO

1.1 HISTÓRICO

Antigamente os canais de comunicação eram praticamente dedicados à voz, com pouquíssimo tráfego de dados. Os anos foram passando e a mudança foi acontecendo, o que antes era pouco aumentou muito até o ponto em que o tráfego de dados passou a ser muito maior do que o de voz. Isso levou as operadoras de redes de comutação de pacotes a pensarem em uma alternativa para trafegar voz sobre suas redes de dados já estabelecidas, tendo em vista que a largura de banda exigida para a voz é pequena em comparação com a largura de banda já existente nas redes de dados. A partir desta “revolução tecnológica” surge a idéia de voz sobre IP – VoIP (Voice over IP).

Atualmente a tecnologia de voz sobre IP é uma realidade e isso se deve às vantagens oferecidas, que dentre elas estão:

- redução de custos com a implantação e manutenção, em comparação ao alto custo para manter um PBX (Private Branch Exchange), pois a arquitetura de voz sobre IP utiliza a rede de dados já existente;
- redução de custos, principalmente em ligações de longa distância, onde quanto maior a distância maior a economia;
- mobilidade que o usuário de voz sobre IP tem em utilizar o seu número de qualquer lugar com acesso à Internet.

Estas vantagens fazem muitos usuários pensarem em trocar a utilização da telefonia convencional pela telefonia IP.

Ainda existem algumas desvantagens na utilização de VoIP, como o fato de ser dependente de energia elétrica, porém com bons sistemas de redundância para geração de energia, esta desvantagem passa desapercebida, principalmente em um ambiente corporativo.

A grande preocupação atualmente é a segurança do tráfego VoIP, pois como a voz está sendo transportada sobre IP, este tráfego está sujeito aos mesmos ataques sofridos por pacotes IP, além de ataques mais específicos para voz sobre IP, como por exemplo o Eavesdropping (Escuta), Call Hijack (Sequestro de Chamada), Call Fraud (Fraudes nas Chamadas) e Spam over IP Telephony (Spam sobre Telefonia IP).

Atualmente, mesmo com a falta de mecanismos de segurança no IPv4 (Internet Protocol version 4), existem mecanismos que, agregados ao protocolo, podem ajudar com eficiência na segurança do tráfego IPv4, tornando o tráfego VoIP bastante seguro.

Fazer o transporte seguro do tráfego VoIP é o grande desafio desta tecnologia, pois quando os mecanismos de segurança são aplicados o pacote VoIP aumenta significativamente de tamanho por causa dos cabeçalhos que são adicionados, causando sobrecarga *(overhead)* na rede, o que pode prejudicar o desempenho das chamadas VoIP, tendo em vista que este tráfego é de tempo real e por isso deve ser transportado no menor atraso possível.

1.2 MOTIVAÇÃO

O tráfego de voz sobre IP a cada dia ganha força dentro das organizações e em uso doméstico devido à grande redução de custos que se obtém com a

utilização desta tecnologia, porém está sujeito à falhas de segurança do protocolo IP e, desta forma, o referido tráfego pode ser facilmente interceptado e pessoas indesejáveis podem ouvir as chamadas. Por isso, existe a necessidade de se conhecer os mecanismos de segurança para o tráfego de voz sobre IP.

1.3 OBJETIVO

Esta monografia tem por objetivo apresentar e analisar os protocolos de segurança SRTP (Secure Real-time Transport Protocol), ZRTP (Zimmermann Real-time Transport Protocol) IPSec (Internet Protocol Security), de forma a conhecer suas características, além das vantagens e desvantagens, levando em consideração que serão aplicados sobre o tráfego VoIP, que funciona em tempo real.

Os protocolos SRTP e ZRTP foram escolhidos para a realização deste trabalho porque ambos viabilizam a segurança no nível de conversação da chamada, que é realizada em RTP, são, portanto, próprios para serem utilizados em chamadas VoIP. O IPSec viabiliza a segurança em nível IP, protegendo todo tráfego IP entre o host de origem e o de destino, sendo o protocolo de segurança próprio para o protocolo IPv4 e nativo nas implementações de IPv6 (Internet Protocol version 6).

2 REFERENCIAL TEÓRICO

2.1 A TECNOLOGIA VOIP

A Comunicação de voz em redes IP, conhecida como VoIP, consiste no uso das redes de dados que utilizam o conjunto de protocolos das redes IP, para a transmissão de sinais de voz em tempo real na forma de pacotes IP.

Nas redes IP são implementados protocolos adicionais de sinalização de chamadas e transporte de voz que permitem a comunicação com qualidade tão boa quanto a fornecida pelas redes convencionais dos sistemas públicos de telefonia comutada ou de telefonia móvel.

A idéia da tecnologia é utilizar uma rede IP como rede telefônica, com algumas funcionalidades adicionais. Ao invés da comunicação acontecer em uma rede de comutação de circuitos (comunicação sem fila ou atrasos intermediários) essa aplicação permite a comunicação entre duas partes em uma rede de comunicação de pacotes (a Internet) [2]. Como a Internet é uma rede global, isso significa que se pode usar essa tecnologia em qualquer lugar do mundo [4].

De acordo com o órgão de padronização norte-americano FCC (Federal Comunication Comission):

> "Tecnologias VoIP, incluem aquelas utilizadas para facilitar a telefonia IP, que permitem a transmissão da voz em tempo-real e outras aplicações baseadas no uso da voz. Tecnologia VoIP é utilizada quando, numa comunicação de voz, pelo menos em uma parte do percurso desta comunicação é feita por pacotes IP, utilizando tecnologia IP e redes IP. Pode-se prover VoIP sobre Internet pública (aberta) ou sobre redes privadas IP. VoIP pode ser transmitido utilizando qualquer tipo de meio (ex. cobre, cabo, fibra, radiofreqüência, etc.). Diferentemente da tradicional telefonia por comutação de circuitos, onde é estabelecido um circuito dedicado entre os pontos para a transmissão de voz, VoIP conta com a comutação de pacotes que divide a transmissão de voz em pacotes (empacotamento) e os envia através da rota mais rápida disponível. Desta forma, VoIP utiliza mais eficientemente a largura de

banda disponível que os circuitos telefônicos comutados e possibilita aos provedores manter uma única rede IP para dados e voz.” [17].

Com isso, verifica-se que o tráfego de voz está deixando a exclusividade de ser transportado em uma infra-estrutura de telefonia convencional, onde os canais para voz são exclusivos, não se tem atrasos ou filas e a qualidade da voz e da chamada é excelente, para ser transportado em redes IP, redes onde existem atrasos fim-a-fim, onde os pacotes passam por filas nos equipamentos ocasionando atrasos, onde o caminho percorrido pelos pacotes de uma mesma chamada pode ser diferente, o que pode causar uma falta de sincronismo na chegada dos pacotes ao destino. Estes fatores são preocupantes principalmente quando esta rede IP é a Internet.

Mesmo com todos os desafios descritos acima, a tecnologia é muito vantajosa, permitindo a redução significativa dos custos com telefonia, mobilidade, flexibilidade e inúmeros outros benefícios. Estas vantagens normalmente só são atingidas quando o conceito de QoS (Quality of Service) é implementado na rede IP, permitindo assim que o tráfego VoIP possa ter prioridade nas filas dos equipamentos, que os pacotes de uma chamada possam ser encaminhados sempre para o mesmo canal para que haja sincronismo na chegada dos pacotes ao destino, que os pacotes de uma nova chamada possam ser encaminhados para o canal menos utilizado de forma a diminuir o atraso na entrega destes pacotes, para assim atingir uma ótima qualidade na chamada de voz sobre IP.

Para que a voz trafegue sobre redes IP, ela passa por alguns processos, dentre eles estão a conversão da voz de sinal analógico para dados através de *codecs* (codificadores e decodificadores). Uma vez que a voz está em forma digital, ela é introduzida em pacotes de dados e enviada através das redes IP utilizando

protocolos de transporte como o UDP (User Datagram Protocol*)* e o RTP (Real-time Transport Protocol). Quando este tráfego chega ao destino os pacotes são reordenados e convertidos de volta para a forma analógica [11].

A transmissão de voz na rede IP necessita de certas propriedades como: baixa latência (atraso) origem-destino, baixa variação da latência (jitter), taxas de perdas de pacotes e erros de bits baixas, por isso o canal de transmissão deve ser muito bom, não necessariamente a largura de banda do canal deve ser alta, pois o consumo de banda de uma chamada é baixo, porém deve ser livre de ruídos ao máximo. Essas necessidade são fundamentais porque o tráfego VoIP é realizado em tempo real e se algum dos parâmetros supracitados estiver alto, ocorrerá falhas na interatividade da chamada, fato que pode inviabilizar a mesma.

2.2 SEGURANÇA NA TECNOLOGIA VOIP

A tecnologia utilizada para transmissão de voz em redes IP representa um novo ponto para pesquisas sobre os potenciais problemas de segurança da informação. Se esta tecnologia for comparada com a telefonia convencional, onde a interceptação da comunicação deve ser feita por meio físico através de escutas, o fato de colocar a voz em pacotes IP e transportá-los através da rede os torna mais acessíveis e fáceis de interceptar.

Segurança e eficiência são muitas vezes requisitos conflitantes. Apesar de haver áreas e aplicações na Internet onde o impacto desses mecanismos de segurança é menor, as aplicações em tempo real, como VoIP, podem ser seriamente afetadas [6]. Com a introdução de vários cabeçalhos no pacote IP para

viabilizar a proteção dos dados, o desempenho deste tráfego pode ser prejudicado e em muitos casos pode ser inaceitável para transmissões em tempo real.

A segurança da informação é fundamentada em alguns conceitos que devem ser levados em consideração no momento de colocar um serviço na rede. A implementação de todos certamente oferecerá uma forte segurança para o pacote transportado. É claro que se o intuito é de máxima proteção ao tráfego VoIP, o mesmo deve ser tratado de acordo com estes conceitos, que são:

- *Integridade* – o receptor deve receber os pacotes originalmente enviados sem qualquer alteração em seu conteúdo. Uma terceira parte não deve ser capaz de modificar os pacotes em trânsito. Essa definição é estritamente aplicada no caso de sinalização VoIP (estabelecimento da chamada). Contudo, no caso da mídia em si (conversação na chamada), a perda de integridade dos pacotes pode ser tolerável.
- *Privacidade* – uma terceira parte não deve ser capaz de ler os dados destinados ao receptor [5].
- *Autenticidade* – o transmissor e o receptor das mensagens de sinalização ou mídia VoIP devem estar seguros que o interlocutor com o qual conversam é efetivamente quem ele diz ser [5].
- *Disponibilidade/Proteção contra ataques de negação de serviço (DoS – Denial of Service)* – o serviço VoIP deve estar disponível aos usuários a qualquer tempo. Usuários ou dispositivos maliciosos ou que se comportem de forma inadequada não devem ser capazes de interromper o serviço. A minimização de ataques de DoS demanda medidas para proteger os recursos VoIP e proteger a rede IP em operação [5].

Duas formas são comumente abordadas para que o tráfego VoIP seja transportado de forma segura. Uma é a inserção de mecanismos na funcionalidade dos protocolos de sinalização. O H.323 (protocolo de sinalização VoIP) possui a recomendação H.235 que diz quais os algoritmos de criptografia e autenticação devem ser utilizados na comunicação de gatekeepers, terminais e gateways. O SIP (Session Initiation Protocol) (outro protocolo de sinalização VoIP) possui soluções que devem ser inseridas nas mensagens de controle e no canal de mídia entre os proxies e os agentes usuários. Esses mecanismos são implementados na camada de aplicação [11].

O outro tipo de mecanismo é a utilização de VPN (Virtual Netwok Private) para a transmissão do tráfego de voz. As redes virtuais privadas implementam seus protocolos de segurança na camada de rede do modelo TCP/IP (Transport Control Protocol/Internet Protocol), viabilizando a integridade, autenticidade e confidencialidade do canal TCP e UDP/RTP. Este mecanismo não é exclusivo da tecnologia VoIP, podendo ser utilizado com outros tipos de mídias.

Dentro de um pacote VoIP SIP a opção pela criptografia da mídia pode ser sinalizada pelo parâmetro k do SDP (Session Description Protocol) (RFC 2327), nos seguintes formatos:

- k=clear: <chave de criptografia>
- k=base64: <chave de criptografia codificada>
- k=prompt: solicita a chave

O SDP é usado no corpo da mensagem SIP para descrever os parâmetros de uma sessão multimídia. Essa informação inclui o tipo de sessão (áudio, vídeo ou

ambos) e parâmetros como codecs ou portas necessárias para estabelecer um fluxo de mídia [5].

Em todos estes casos a chave acaba passando em claro na rede se a sinalização não estiver protegida. Para proteger a chave de criptografia da mídia a sinalização SIP deve ser criptografada e para isso existem duas estratégias: hop-a-hop com utilização de IPSec; fim-a-fim com o uso de chave compartilhada ou mecanismo de chave pública [12].

A fim de melhor conhecer como funcionam os protocolos que viabilizam a segurança para o tráfego VoIP, os próximos capítulos estão destinados aos protocolos SRTP, ZRTP e IPSec que, como já citado, oferecem segurança nas camadas de aplicação e rede.

3 SRTP (SECURE REAL-TIME TRANSPORT PROTOCOL)

3.1 INTRODUÇÃO

O SRTP é uma variante do RTP, que pode oferecer confidencialidade, autenticação de mensagem e proteção contra ataques do tipo repetição para o tráfego RTP, e para o controle do tráfego RTP, chamado RTCP (Real-time Transport Control Protocol) [14]. Como o RTP é estreitamente relacionado com o RTCP, que é usado para controlar as sessões RTP, o SRTP também possui um protocolo "irmão" denominado SRTCP (Secure Real-time Transport Control Protocol). O SRTCP provê para o RTCP as mesmas funcionalidades de segurança que o SRTP provê para o RTP.

O SRTP oferece algumas melhorias para a segurança da mídia, tais como: confidencialidade e integridade para RTP/RTCP, criptografando os respectivos campos de carga útil *(payload)*; possibilidade de atualização das chaves de sessão periodicamente, estrutura que permite atualizações com novos algoritmos de criptografia; e segurança para aplicações unicast e multicast.

O SRTP provê um framework para encriptação e autenticação de mensagens de fluxos RTP e RTCP.

O SRTP é independente de uma implementação de pilha RTP específica e de um padrão de gerenciamento de chaves específico [7].

O SRTP não especifica como as chaves são trocadas entre o transmissor e o receptor. Os sistemas de gerenciamento de chaves estão fora do escopo da especificação SRTP. No caso de VoIP, o protocolo de sinalização pode trocar as chaves antes que o SRTP inicie. Para usar o protocolo de sinalização para troca de

chaves deve-se tornar o mesmo seguro através de TLS (Transport Layer Security), IPSec ou métodos similares. De outra maneira as chaves que o SRTP usa podem ser expostas a hackers [15].

Esta troca de chaves inicia na mensagem de sinalização *Invite.* Um dos métodos que tem sido utilizado com o SRTP é o MIKEY, protocolo de gerenciamento de chaves, que suporta três tipos de acordo de chaves: chave pré-compartilhada *(pre-shared key),* encriptação por chave pública *(public-key encription)* ou infra-estrutura de chave pública e DH (Diffie-Hellman). O outro método que pode ser utilizado com o SRTP é o ZRTP que será abordado no próximo capítulo.

O SRTP provê um acréscimo na segurança, conforme abaixo:

- Confidencialidade para RTP e RTCP através da encriptação da carga útil de cada um [7];
- Integridade para pacotes inteiros RTP e RTCP, junto com proteção contra ataques de repetição [7];
- Possibilidade de atualizar a chave de sessão periodicamente, o que limita a quantidade de textos cifrados produzidos por uma chave fixa que podem ficar disponíveis para um adversário criptoanalisar [7];
- Framework extensível que permite a melhora com novos algoritmos de criptografia [7];
- Derivação de chave da sessão segura com função pseudo-randômica para ambos os fins [7];
- Uso de chaves de salto (*salting keys)* para proteger contra ataques com pré-computação [7];
- Segurança para aplicações RTP unicast e multicast [7].

3.2 FORMATO DO ENCAPSULAMENTO

O SRTP especifica o formato de encapsulamento para pacotes RTP protegidos, bem como quais partes do pacote RTP são criptografadas e autenticadas pelos seus respectivos algoritmos. Somente dois campos são adicionados: o identificador de autenticação *(autentication tag)* (recomendado) e o índice de chave mestre *(master key index - MKI)* (opcional) [8], conforme figura 1. O primeiro campo é utilizado para carregar dados autenticados das mensagens e o segundo é definido, sinalizado e usado pelo gerenciamento de chaves.

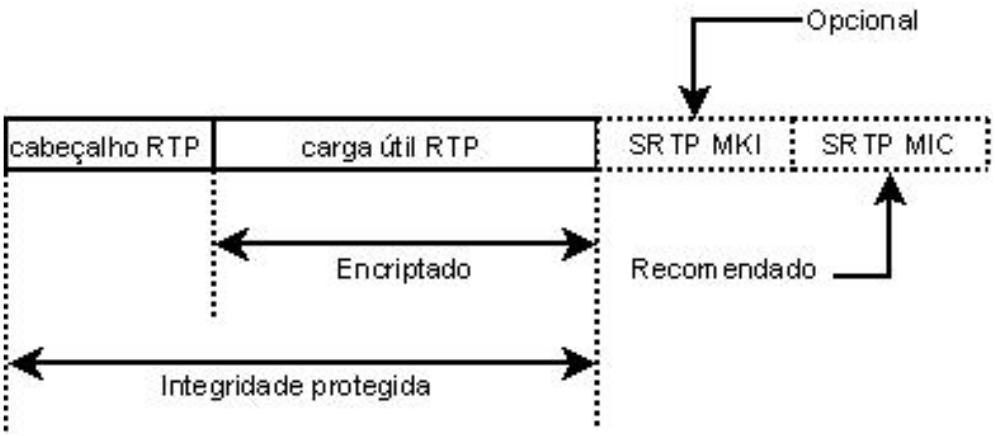

Figura 1: Formato do pacote SRTP.

3.3 ENCRIPTAÇÃO

Para a encriptação do fluxo de dados, o SRTP e o SRTCP padronizam a utilização de um único algoritmo de criptografia, o AES (Advanced Encription Standard), que pode ser utilizado nos modos contador e f8.

O primeiro, AES-CTR ou modo contador, foi desenvolvido para solucionar a dispendiosa proposta de decodificar o acesso a um bloco aleatório de um dado de voz (por exemplo, parte da conversação em uma chamada VoIP) quando a

codificação deste utiliza encadeamento de blocos de cifra, onde exige primeiro a decodificação de todos os blocos situados à frente do bloco escolhido. Neste caso, o vetor de inicialização é acrescido em uma unidade a cada novo bloco, facilitando a decodificação de um bloco em qualquer lugar no dado acessado, não sendo necessário que primeiro sejam decodificados todos os seus predecessores.

Este modo requer que a origem (quem criptografa) gere um único valor por pacote e comunique ao destino (quem descriptografa). Este valor por pacote é chamado de vetor de inicialização (VI). O mesmo VI mais a chave de combinação não devem ser usados mais que uma vez. A origem pode gerar um VI de qualquer maneira que assegure imparidade, ou seja, que não exista outro igual. O AES-CTR tem muitas propriedades que fazem dele um algoritmo de encriptação atrativo para uma rede de alta velocidade [13]. Sua utilização é essencial para tráfego RTP executado sobre redes não confiáveis com possível perda de pacotes, não sendo necessário a recuperação do bloco perdido para decodificar o bloco acessado.

O segundo modo é conhecido como algoritmo f8, AES modo f8, e tem sido desenvolvido para encriptar dados UMTS (por exemplo redes 3G). Em termos gerais, o esquema proposto é uma variante do OFB (Output Feedback, algoritmo de cifras por bloco que provê confidencialidade e integridade da mensagem), com funções de feedback e de inicialização mais elaboradas. Assim como no OFB, o princípio consiste de cifras por bloco. O AES modo f8 deve usar o mesmo tamanho padrão para chave de sessão e de salto usados pelo modo contador do AES [14]. Uma chave de salto é um conjunto aleatório de bits usados como inicialização para a geração de uma chave criptográfica. Outros tipos utilizados como inicializadores são uma senha ou uma frase senha.

O SRTP também permite desabilitar a encriptação usando somente o modo chamado *NULL cipher*. Na verdade o *NULL cipher* é utilizado quando nenhuma confidencialidade é requerida, não desempenhando qualquer encriptação, ou seja, como se o algoritmo de criptografia funcionasse com o fluxo das chaves (*key stream)* contendo somente valores nulos, zeros, copiando o fluxo de entrada para o fluxo de saída sem qualquer alteração. Pode ser usado quando não for requerida a garantia de confidencialidade pelo o SRTP, porém outras funcionalidades como autenticação e integridade podem ser usadas.

É importante ressaltar que o SRTP não criptografa o cabeçalho do RTP, permitindo que mecanismos de compressão de cabeçalho ao nível de enlace ainda funcionem.

3.4 AUTENTICAÇÃO E INTEGRIDADE

Os algoritmos de encriptação não asseguram a integridade da mensagem, permitindo que o atacante possa falsificar os dados ou repetir a transmissão de dados realizada previamente. O padrão SRTP também provê o meio para assegurar a integridade dos dados e a segurança contra ataques de repetição.

Para autenticar a mensagem e assegurar a integridade, o algoritmo HMAC-SHA1 (Hash Message Authentication Code - Secure Hash Algorithm) (definido na RFC 2104) é utilizado, produzindo um resultado de 160 bits que é truncado para 80 bits, tornando-se o identificador de autenticação do pacote. O HMAC é calculado sobre a carga útil do pacote.

A porção de autenticação do pacote SRTP consiste do cabeçalho RTP seguido pela porção encriptada do pacote SRTP (Figura 1). Se ambos, autenticação

e encriptação, forem aplicados, a encriptação deverá ser aplicada antes da autenticação do lado do emissor e inversamente do lado do receptor. O identificador de autenticação provê autenticação do cabeçalho e da carga útil, e indiretamente provê proteção contra ataque do tipo repetição por autenticar o número de sequência [19].

3.5 PROTEÇÃO CONTRA ATAQUES DE REPETIÇÃO

A proteção segura contra ataques de repetição somente é possível quando a proteção de integridade está presente. É recomendável usá-la para proteção contra ataques de repetição, porém somente a utilização da proteção de integridade não garante a segurança contra estes ataques.

Um pacote é repetido quando é armazenado por um atacante e então reenviado na rede. Quando a autenticação de mensagens é configurada, o SRTP protege contra tais ataques através de uma lista de repetição *(replay list)*. Cada receptor SRTP mantém uma lista, que conceitualmente contém o índice de todos os pacotes que têm sido recebidos e autenticados. Na prática esta lista pode utilizar uma abordagem de janela deslizante.

O receptor checa o índice dos pacotes de entrada contra a lista de repetição e a janela. Somente pacotes com índice à frente da janela, ou, do lado de dentro da janela, mas já recebidos, devem ser aceitos.

Depois do pacote ter sido autenticado a lista de repetição deve ser atualizada com o novo índice.

Sem a proteção contra ataques de repetição é possível que um adversário execute manipulações simples no pacote, subvertendo a segurança. Por exemplo,

em uma aplicação de voz, a palavra “sim” pode ser substituída por “não” se a proteção contra ataques repetição não estiver presente.

3.6 CHAVE DE DERIVAÇÃO

A chave de derivação é usada para derivar diferentes chaves no contexto de criptografia (chaves de encriptação, de salto e de autenticação para SRTP e SRTCP) de uma única chave mestre no caminho criptograficamente seguro. Assim, o protocolo de gerenciamento de chaves necessita trocar somente uma chave mestre e todas as chaves de sessão necessárias são geradas através da aplicação da função de derivação de chaves.

A aplicação periódica da função de derivação de chaves resultará em benefícios de segurança. Ela previne que um atacante colete uma larga amostra do texto encriptado com uma única chave de sessão. Certos ataques são mais fáceis de serem executados quando uma quantidade maior de textos encriptados com a mesma chave está disponível. Um atacante terá dificuldades para decifrar uma mensagem em sua íntegra se ele precisar decifrar textos encriptados com chaves de sessão diferentes devido a aplicação da função de derivação de chaves, mesmo que tenham por origem a mesma chave mestre. Cabe ressaltar que a descoberta da chave mestre dará condições de revelar todas as chaves de sessão geradas a partir dela.

O campo índice de chave mestre, Figura 1 (SRTP MKI), identifica a chave mestre a partir da qual as chaves de sessões foram derivadas. O valor deste campo não deve ser identificado pelo contexto criptográfico do SRTP, porém pode ser

utilizado pelo gerenciamento de chaves para o propósito de regerar chaves, identificando uma chave mestre particular dentro de um contexto criptográfico [19].

4 ZRTP (ZIMMERNMANN REAL-TIME TRANSPORT PROTOCOL)

4.1 INTRODUÇÃO

O ZRTP é uma extensão para o RTP que descreve o algoritmo Diffie-Helman (DH) como método de acordo de chave para o SRTP, a fim de estabelecer a chave criptográfica para o fluxo de mídia (voz ou vídeo). A letra 'Z' no acrônimo representa o sobrenome de Phil Zimmermann, um engenheiro bastante conhecido por ter criado o PGP (Pretty Good Privacy). O ZRTP foi submetido ao IETF (Internet Engineering Task Force) por Zimmernmann e é atualmente um Internet-Draft.

O ZRTP é descrito em seu Internet-Draft como sendo um protocolo para acordo de chave que desempenha o algoritmo de troca de chave DH durante o início da chamada no fluxo de mídia e é transportado sobre a mesma porta que o canal da mídia RTP é estabelecido, usando o protocolo de sinalização tal como o SIP. Isto gera um segredo compartilhado *(shared secret)* que é então usado para gerar as chaves de sessão e de salto para o SRTP [16].

Este segredo compartilhado é calculado e usado somente uma vez por sessão, entretanto o protocolo permite que parte dele possa ser armazenado em memória para sessões futuras. O segredo compartilhado é utilizado para gerar a chave mestre *(master key)* das sessões SRTP.

O protocolo ZRTP gera as chaves no fluxo da mídia porque é multiplexado na mesma porta do RTP e não requer suporte ao protocolo de sinalização, por isso pode ser utilizado com qualquer protocolo de sinalização, incluindo SIP, H.323 e outros. O ZRTP é independente da camada de sinalização porque faz todas as negociações de chaves no fluxo de mídia RTP (na conversação da chamada).

Ele utiliza o algoritmo DH para acordar a chave de sessão e parâmetros para o estabelecimentos de sessões SRTP. O ZRTP não substitui o SRTP, mas estende seus recursos.

Este protocolo não conta com uma infra-estrutura de chave pública ou uma autoridade de certificação, na verdade as chaves DH são geradas em cada estabelecimento de sessão, permitindo contornar a complexidade de criação e manutenção de uma estrutura segura de terceiros [9].

Para a sessão da mídia o ZRTP provê confidencialidade, proteção contra ataques de Homem-no-Meio, e, em casos onde o segredo é disponível pelo protocolo de sinalização, provê autenticação. O ZRTP pode utilizar dois atributos do SDP (Session Description Protocol) para prover descobrimento e autenticação através do canal de sinalização [16].

4.2 ESTABELECIMENTO DA SESSÃO RTP

O acordo de chaves é desempenhado por uma sessão RTP que é estabelecida através de um protocolo de sinalização tal como o SIP.

O protocolo de sinalização pode derivar o segredo da sinalização *(sigs)*. Este passa pelo protocolo de sinalização usado para estabelecer a sessão RTP do ZRTP.

O identificador de diálogo para uma sessão SIP segura é uma string composta do identificador da chamada *(call-id)*, identificador do destino ou remoto *(to-tag)* e identificador da origem ou local *(from-tag)*. Os identificadores local e remoto são organizados em ordem ascendente no hash, conforme definições na RFC 3261. O *sigs* é o hash da concatenação do *call-id, to-tag e from-tag*, de acordo com a expressão abaixo:

sigs=hash(call-id | to-tag | from-tag)

Isto pode ser considerado um segredo porque é sempre transportado usando TLS (Time Layer Security), além de ser gerado aleatoriamente para cada chamada SIP.

A sinalização também provê o segredo SRTP (*srtps*), que é o hash da chave mestre SRTP (*chavesrtp*) e chave mestre de salto (*chavesalto*):

srtps=hash(chavesrtp | chavesalto)

4.3 ACORDO DE CHAVES – MODO DIFFIE-HELLMAN

O propósito deste modo é gerar um novo segredo compartilhado, *s0*, para os dois dispositivos ZRTP que realizam a comunicação. Além disso, os dispositivos descobrem se possuem algum segredo compartilhado em comum, quantos são e acordam a ordem para eles: *s1, s2,* etc [16].

O algoritmo de troca de chaves DH por si só não provê proteção contra ataques de Homem-no-Meio *(Man-in-the-Middle)*. Para autenticar a troca de chaves o ZRTP utiliza o método SAS (Short Authentication String) que é essencialmente o valor de hash de uma string derivada dos valores públicos da troca de chaves DH. Os dispositivos farão a comparação deste valor através da sua leitura em “voz alta”. Se os dois resultados coincidirem significa que a probabilidade de nenhum ataque de Homem-no-Meio ter acontecido é alta.

O algoritmo para acordo de chaves pode ser dividido em 4 fases: descobrimento *(discovery)*, compromisso de hash *(hash commitment)*, troca de chaves DH *(DH exchange)*, estabelecimento e confirmação da sessão SRTP *(switch to SRTP and confirmation)* [9].

Durante a fase de descobrimento, de acordo com a Figura 2, Alice e Bob trocam seus identificadores ZRTP *(ZID)*. Além disso, coletam informações sobre a

capacidade do outro, como a versão suportada do ZRTP, funções de hash (SHA-256), algoritmo de criptografia (AES-128 ou AES-256), tamanho do identificador de autorização (HMAC-SHA1), tipos de acordo de chaves (DH3k ou DH4k) e algoritmo SAS (32 ou 256 bits) [9].

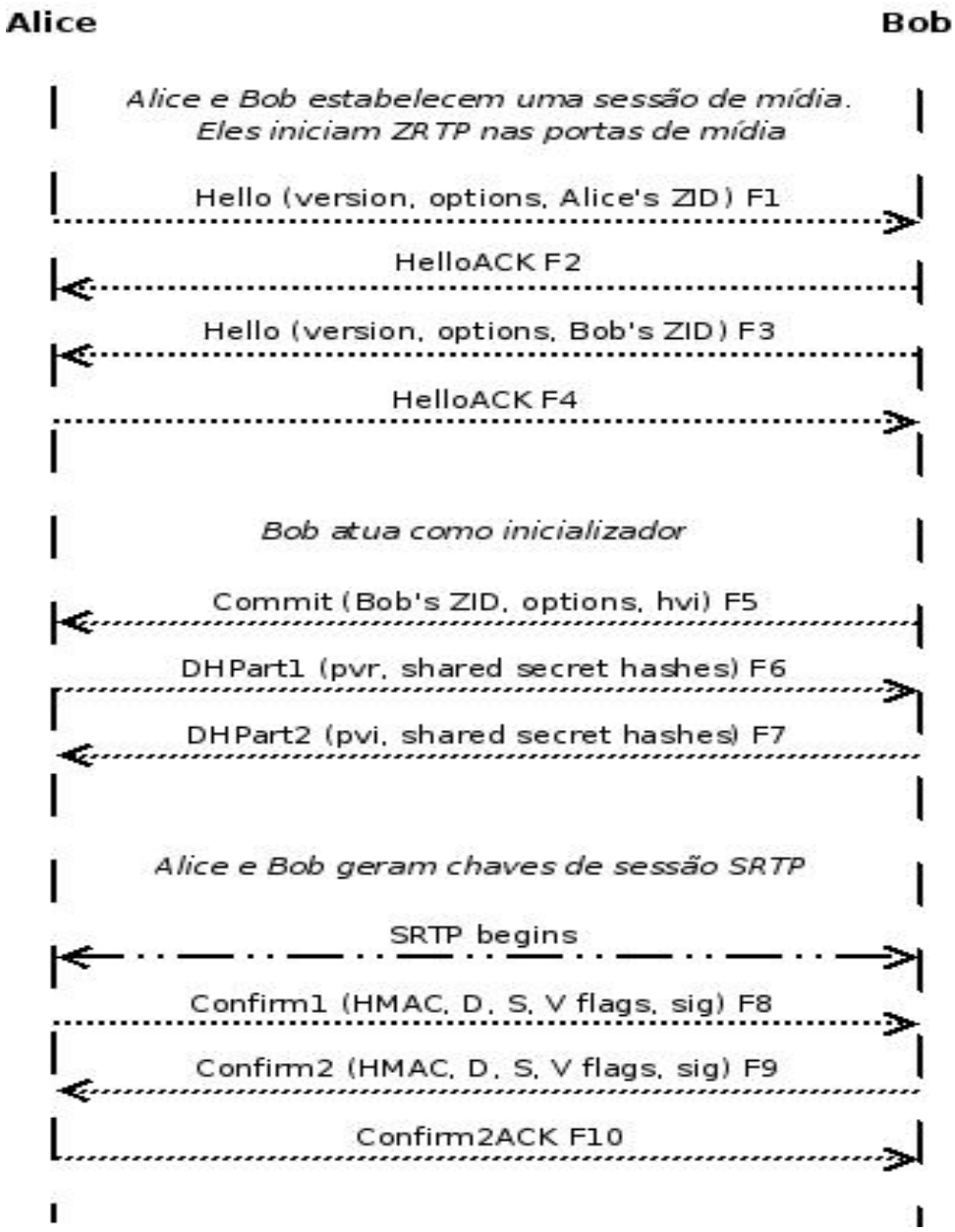

Figura 2: Estabelecimento de uma sessão SRTP usando ZRTP.

A troca de ZID entre Alice e Bob permite que seja determinado se já existe um segredo compartilhado retido *(retained shared secret)* [10] oriundo de sessões anteriores.

As mensagens trocadas durante esta fase são chamadas *Hello* (com tamanho variável), e cada parte responde com uma mensagem *HelloACK* [9].

Depois da fase de descobrimento, Bob inicializa a fase de compromisso de hash escolhendo – entre os parâmetros suportados para ambos os dispositivos – qual função de hash, algoritmo de criptografia, tamanho do identificador de autorização, tipo de acordo de chave e tipo de algoritmo SAS devem ser usados. Após a escolha, Bob envia a mensagem *Commit* com os parâmetros escolhidos [9].

Antes de enviar a mensagem *Commit*, Bob (atuando como inicializador) gera seu valor secreto, *vsB* (valor secreto de Bob), baseado no tipo de acordo de chave, e calcula seu valor público, *vpB* (valor público de Bob) ou *pvi* (public value of initiator) conforme Figura 2, com a expressão: $vpB=g^{vsB} \bmod p$, onde g e p são determinados pelo tipo de acordo de chaves. O *vsB* deve ser duas vezes maior do que o tamanho escolhido para o algoritmo AES, ou seja se o AES for de 128 bits o valor secreto de ter 256 bits.

A mensagem *Commit* possui as opções escolhidas por Bob e um valor de hash, *vhB* (valor de hash de Bob) ou hvi (*hash value of initiator*) conforme Figura 2, que será utilizado na fase de compromisso de hash. O *vhB* é calculado com a seguinte expressão:

vhB=hash(*DHPart2* | mensagem *Hello* de Alice)

Este resultado é o hash da mensagem *DHPart2,* que inclui o *vpB*, concatenada com a mensagem *Hello* de Alice, que é incluída no cálculo com o objetivo de prevenir

contra o ataque de Baixa-Oferta *(Bid-Down)*, onde um intruso pode modificar para baixo a oferta dos parâmetros suportados por um dispositivo. Por exemplo, Alice suporta as opções AES-128 ou AES-256, mas um intruso intercepta e informa somente a opção AES-128. Desta forma a mensagem *Commit* garante que os parâmetros escolhidos são suportados de acordo com as informações da mensagem *Hello* de Alice.

Na mensagem *Commit* também será informado qual o modo será utilizado para estabelecimento das chaves, Acordo de Chaves DH ou o Pré-Compartilhado.

Alice, após o recebimento da mensagem *Commit* vinda de Bob, gera seu valor secreto ou *vsA* (valor secreto de Alice) e calcula seu valor público, *vpA* (valor público de Alice) ou *pvr* (*public value of responder*) conforme Figura 2, com a expressão: $vpA=g^{vsA} \bmod p$. O *vsA* também deve ser duas vezes maior do que o tamanho escolhido para o algoritmo AES, assim como o de Bob.

Alice e Bob devem possuir seus segredos compartilhados retidos e estes podem estar organizados em ordem diferente em cada (ou possuem segredos diferentes ou não possuem nenhum segredo) o que comprometeria o cálculo do *s0*. Para evitar isso Alice ordena seus segredos seguindo a ordem de Bob, designando para *sX*, onde *s*=segredo e *X*=número sequencial, o segredo correto. Em caso de uma não coincidência de um segredo ele será designado como valor nulo.

Para alcançar este objetivo, Alice calcula o valor de hash (HMAC) dos segredos compartilhados usando o primeiro segredo compartilhado retido, *sr1*, como a chave aplicada sobre a string *"Responder"* gerando uma identificação do segredo retido, *sr1-idA*, truncada para 64 bits. Os hashes são calculados de acordo com as expressões abaixo:

sr1-idA=HMAC(sr1, ”Responder”)

sr2-idA=HMAC(sr2, ”Responder”)

sigs-idA=HMAC(sigs, “Responder”)

srtps-idA=HMAC(srtps, “Responder”)

outro_segredo-idA=HMAC(outro_segredo, “Responder”)

Alice envia o hash dos segredos compartilhados retidos (*shared secrets hashes*) e o seu *vpA* para Bob na mensagem *DHPart1.*

Alice calcula o valor HMAC esperado para os segredos compartilhados provenientes de Bob na mensagem *DHPart2*, utilizando a string *“Initiator”* ao invés de *“Responder”* [16].

Ao receber a mensagem *DHPart1* Bob verifica se o *vpA* é diferente de 1 ou p-1, onde ‘p’ é um dos valores públicos gerados pelo próprio DH no início de sua operação. Se o *vpA* for igual a 1 ou p-1 é muito provável que um atacante tenha injetado uma falsa mensagem *DHPart1* o que causa um resultado final desastroso para o algoritmo DH. Neste caso Alice deve ser alertada do ataque e o protocolo deve ser terminado. Se este valor estiver correto, Bob envia a mensagem *DHPart2* contendo o *vpB* e valor HMAC dos possíveis segredos compartilhados retidos (*shared secrets hashes*), onde, assim como Alice, utiliza o *sr1* como chave aplicada sobre a string *“Initiator”* gerando uma identificação do segredo retido, *sr1-idB*, truncada para 64 bits, de acordo com as expressões abaixo:

sr1-idB=HMAC(sr1, ”Initiator”)

sr2-idB=HMAC(sr2, ” Initiator”)

sigs-idB=HMAC(sigs, “Initiator”)

srtps-idB=HMAC(srtps, “Initiator”)

outro_segredo-idB=HMAC(outro_segredo, “Initiator”)

Bob então calcula o HMAC esperado para os segredos recebidos de Alice na mensagem *DHPart1*, utilizando a string *“Responder”* ao invés de *“Initiator”.* Os valores HMACs que coincidirem significam que foram gerados com o mesmo segredo retido e por isso este segredo é guardado. Os que não coincidirem são colocados como nulo, assumindo um tamanho zero de forma que sejam excetuados no cálculo do segredo compartilhado final [16].

Ao receber a mensagem *DHPart2* Alice verifica se *vpB* é diferente de 1 ou p-1. Se o *vpB* for igual a 1 ou p-1 é muito provável que um atacante tenha injetado uma falsa mensagem *DHPart2* o que causa um resultado final desastroso para o algoritmo DH. Neste caso Bob deve ser alertado do ataque e o protocolo deve ser terminado. Posteriormente Alice calcula o seu próprio valor de hash para a fase compromisso de hash, usando o *vpB*, recebido na mensagem *DHPart2*, juntamente com a sua mensagem *Hello* para verificar se este valor irá coincidir com o *vhB* enviado por Bob na mensagem *Commit.* Se os valores forem diferentes provavelmente ocorreu um ataque de Homem-no-Meio, Bob deve ser alertado e o protocolo deve ser terminado.

Alice compara os valores HMACs esperados de Bob, já calculados, com os recebidos na mensagem *DHPart2.* Os valores HMACs que coincidirem significam que foram gerados com o mesmo segredo retido e por isso este segredo é guardado. Os que não coincidirem são colocados como nulo, assumindo um tamanho zero de forma que sejam excetuados no cálculo do segredo compartilhado final. O conjunto de até cinco segredos compartilhados são denominados como *s1*, *s2*, *s3*, *s4* e *s5*, e ordenados pelo inicializador [16].

Como exemplo, considere dois dispositivos ZRTP que compartilham os segredos *sr1*, *sr2* e *outro_segredo*, que é o hash de uma frase senha. Durante a comparação, *sr1-id*, *sr2-id* e *outro_segredo-id* coincidiram, mas *sigs-id* e *srtps-id* não. Como resultado *s1=sr1*, *s2=sr2*, *s5=outro_segredo*, enquanto *s3* e *s4* serão nulos [16].

Na fase de troca de chave DH, Alice e Bob calculam o resultado final do algoritmo DH (*DHResult*). Para Bob o cálculo é:

$$DHResult = vpA^{vsB} \bmod p$$

Para Alice o cálculo é:

$$DHResult = vpB^{vsA} \bmod p$$

Esse cálculo resultará o mesmo valor tanto para Alice quanto para Bob.

Para gerar o segredo compartilhado final *(s0)* é necessário em primeiro lugar calcular o valor de hash das mensagens ZRTP concatenadas *(vhzrtp)*, conforme a expressão:

vhzrtp=hash(*Hello* de Alice | *Commit* | *DHPart1* | *DHPart2)*

Após a obtenção do *vhzrtp*, o *s0* é calculado. O cálculo do *s0* é realizado através da concatenação do *DHResult*, do ZID de Alice e de Bob, do *vhzrtp* e do conjunto dos segredos compartilhados não-nulos já existentes. Desta forma temos:

s0=hash(DHResult | ZID_Alice | ZID_Bob | vhzrtp | s1 | s2 | s3 | s4 | s5).

Após este último cálculo tem-se o segredo compartilhado final.

Um novo segredo compartilhado retido *(sr1)* será gerado a partir do *s0* e para isso o *s0* é a chave aplicada sobre a string *"retained secret":*

sr1=HMAC(s0,"retained secret")

Passando para a fase de estabelecimento e confirmação de sessão SRTP, várias chaves, tal como as usadas pelo SRTP, devem ser derivadas do *s0*. Para isso, o ZRTP usa uma função de derivação de chave baseada em HMAC.

As chaves mestre e mestre de salto SRTP são derivadas do *s0*. Estas chaves são geradas em apenas um sentido do fluxo. Se o fluxo é ida e volta o dobro de chaves devem ser geradas. Estas chaves são geradas com a aplicação da chave *s0* sobre uma determinada string, de acordo com as expressões abaixo. As chaves de Bob são geradas com o seguinte cálculo:

chavesrtpB=HMAC(s0,”Initiator SRTP master key”)

chavesaltoB=HMAC(s0,”Initiator SRTP master salt”)

As chaves de Alice são geradas com o seguinte cálculo:

chavesrtpB=HMAC(s0,”Responder SRTP master key”)

chavesaltoB=HMAC(s0,”Responder SRTP master salt”)

Bob criptografa e Alice descriptografa os pacotes usando as chaves mestre e a mestre de salto de Bob. Estas são truncadas para o tamanho determinado pelo algoritmo SRTP escolhido. Alice criptografa e Bob descriptografa os pacotes usando as chaves mestre e mestre de salto de Alice.

Chaves HMAC são geradas para uso de Alice e Bob. Estas chaves são utilizadas somente pelo ZRTP e não pelo SRTP, são derivadas do segredo compartilhado final e são calculadas, utilizando o *s0* como a chave aplicada sobre uma string, conforme as expressões abaixo:

chavehmacB=HMAC(s0,”Initiator HMAC key”)

chavehmacA=HMAC(s0,”Responder HMAC key”)

As chaves HMAC são necessárias para Bob e Alice assegurarem que as mensagens em cada direção são únicas e não podem ser guardadas em memória por um atacante e refletida de volta para o dispositivo do usuário.

Chaves ZRTP são geradas para Bob e Alice usarem para criptografar as mensagens *Confirm1* e *Confirm2.* Elas são truncadas para o mesmo tamanho que as chaves SRTP foram negociadas. Estas chaves também derivam do segredo *s0* e o utilizam como chave aplicada sobre uma determinada string, conforme as seguintes expressões:

chavezrtpB=HMAC(s0,”Initiator ZRTP key”)

chavezrtpA=HMAC(s0,”Responder ZRTP key”)

Depois do segredo compartilhado final ter sido usado para calcular todas as chaves que foram derivadas dele, ele deve ser apagado da memória. As outras chaves, especialmente as chaves SRTP e de salto, devem ser apagadas da memória quando não forem mais usadas como ao final de uma chamada. A única exceção são os segredos compartilhados retidos ou outros segredos em memória necessários para chamadas futuras.

O valor de SAS (*Short Authentication String*) é resultado do cálculo HMAC de uma string digitada com a aplicação de uma chave HMAC derivada a partir do acordo de chaves. O cálculo ocorre da seguinte forma:

vhsas=HMAC(chavehmacB,”SAS”)

valorsas=vhsas[truncado para 32 bits]

Alice e Bob podem agora enviar as mensagens de confirmação, *Confirm1* e *Confirm2*, que são trocadas essencialmente por duas razões. A primeira, elas confirmam que os procedimentos para o acordo de chaves foram realizados com

sucesso e a encriptação está trabalhando de forma correta, além de habilitarem automaticamente a detecção de ataques do tipo Homem-no-Meio vindo de atacantes que não conhecem o segredo compartilhado [16]. A segunda, elas habilitam o ZRTP a transmitir alguns parâmetros sob criptografia CFB (*Cipher Feedback*, algoritmo de cifras por bloco que provê confidencialidade e integridade da mensagem), tais como identificador de Revelação *(Disclosure flag)*, identificador de Permissão Clara *(Allow Clear flag)* e a mais importante o identificador de verificação do SAS *(SAS Verified flag),* conforme Figura 2, protegendo-os de um observador passivo que quer saber se os usuários estão com o hábito de diligentemente verificar o SAS [16].

A parte criptografada das mensagens *Confirm1* e *Confirm2* contém também o intervalo de expiração de armazenamento em memória para *sr0*. A criptografia desta parte ocorre através do cálculo abaixo, onde as chaves HMAC geradas são aplicadas sobre a parte das mensagens que devem ser criptografadas:

hmacB=HMAC(chavehmacB, parte criptografada da mensagem *Confirm1*)

hmacA=HMAC(chavehmacA, parte criptografada da mensagem *Confirm2*)

A mensagem *Confirm2ACK* é uma confirmação enviada por Alice após o recebimento da mensagem *Confirm2*.

Depois da troca com sucesso das mensagens *Confirm1* e *Confirm2*, ambos os lados descartam o valor do *sr2* e armazenam o *sr1* como *sr2*.

No exemplo de fluxo de chamada com ZRTP exibido na figura 2 é possível notar que a ordem da troca das mensagens *Hello/HelloACK* em F1/F2 e F3/F4 pode ser invertida, isto é, tanto Alice quanto Bob podem enviar a primeira mensagem. Um dispositivo que recebe uma mensagem *Hello* e deseja imediatamente iniciar o acordo de chaves ZRTP pode omitir o *HelloACK* e enviar o *Commit*. Isto deve

resultar na omissão das mensagens F2, F3 e F4. Nota-se que o dispositivo que envia a mensagem de *Commit* é considerado o inicializador da sessão ZRTP e orienta o intercâmbio para o acordo de chaves [16].

4.4 ACORDO DE CHAVES – MODO PRÉ-COMPARTILHADO *(PRESHARED)*

Este modo pode ser usado para gerar chaves mestre e mestre de salto SRTP sem o cálculo DH, sem contar com um ou mais segredos compartilhados originados nos DH entre os dispositivos [16].

Este modo de acordo de chaves é útil para eficientemente adicionar outro fluxo de mídia para uma sessão segura existente, tal como adicionar vídeo para uma sessão que já desempenhou um acordo de chave DH para fluxo de áudio. Também pode ser usado para rapidamente restabelecer uma sessão segura entre duas partes que têm recentemente iniciado e terminado uma sessão segura que já havia desempenhado um acordo de chave DH, sem necessidade de outro cálculo DH, o que talvez seja desejável em processadores de baixo desempenho dentro de ambientes com recursos limitados [16].

A escolha por este modo é definida na opção Tipo de Acordo de Chaves da mensagem *Commit*, assim como no modo DH.

A mensagem *Commit* é enviada pelo inicializador, em nosso caso Bob, assim como no modo DH, e esta mensagem também contém *vhB*, que é o hash da mensagem *DHPart2* inteira (incluindo um valor aleatório ao invés do valor público DH, *vpB*) concatenado com a mensagem *Hello* de quem está respondendo, em nosso caso Alice. O cálculo é o seguinte:

vhB=hash(mensagem *DHPart2* de Bob | mensagem *Hello* de Alice)

A inclusão da mensagem *Hello* de Alice tem o mesmo objetivo do modo DH, ou seja, evitar o ataque de Baixa-Oferta.

As mensagens *DHPart1* e *DHPart2* são trocadas de modo que o segredo compartilhado possa ser determinado. Se for constatado que os dispositivos não possuem segredos compartilhados DH (por exemplo nem *sc1* ou *sc2*) a troca deve ser terminada [16], pois neste caso não existirá nenhum segredo armazenado em memória de onde poderão ser derivadas os novos segredos, sendo assim o modo DH deverá ser realizado.

Como não existe nenhum cálculo DH, o *vpB* e *vpA* são substituídos por valores aleatórios para serem incluídos nas mensagens *DHPart1* e *DHPart2*. Este valor deve ser único para todas sessões ZRTP entre um par de dispositivos desde a última troca de chaves DH. Isso garante que um único par de chave e chave de salto SRTP são gerados para cada sessão de media (isto é, conversação na chamada VoIP). Se essas mensagens forem recebidas com um valor aleatório repetido, o ZRTP deve ser finalizado imediatamente.

Para o cálculo do segredo compartilhado, o *vhzrtp*, deve ser calculado da mesma forma como no modo DH:

vhzrtp=hash(*Hello* de Alice | *Commit* | *DHPart1* | *DHPart2).*

O *s0*, é calculado da mesma forma como no modo DH, com exceção do *DHResult*, pois o algoritmo DH não é desempenhado:

s0=hash(ZID_Alice | ZID_Bob | vh_zrtp | s1 | s2 | s3 | s4 | s5)

Nenhum novo segredo compartilhado retido é derivado, com isso os valores de *sr1* e *sr2* não são alterados neste modo.

Assim como no modo DH, várias chaves, tal como as usadas pelo SRTP, são derivadas da *s0*. Para isto, o ZRTP usa uma função de derivação de chaves baseada em HMAC.

As chaves geradas neste modo são exatamente as mesmas geradas no modo DH (*chavesrtpB*, *chavesaltoB*, *chavesrtpA*, *chavesaltoA*, *chavehmacB*, *chavehmacA*, *chavezrtpB*, *chavezrtpA*, *vhsas* e *valorsas*) e possuem os mesmos propósitos.

As mensagens *Confirm1* e *Confirm2* são trocadas para os mesmos propósitos que são trocadas no modo DH e contêm o mesmo conteúdo. A parte do conteúdo que é criptografada também utiliza as mesmas chaves em relação ao modo DH:

hmacB=HMAC(chavehmacB, parte criptografada da mensagem *Confirm1*)

hmacA=HMAC(chavehmacA, parte criptografada da mensagem *Confirm2*)

A mensagem *Confirm2ACK* é uma confirmação enviada por Alice após o recebimento da mensagem *Confirm2*.

5 IPSEC (INTERNET PROTOCOL SECURITY)

5.1 INTRODUÇÃO

O IP Security é uma coleção de protocolos desenvolvidos pelo IETF com o objetivo de fornecer segurança (autenticidade, integridade e confidencialidade) para um pacote da camada IP. Existem duas RFCs fundamentais que descrevem o IPSec: RFC 2401, que descreve a arquitetura geral de segurança IP; e a RFC 2411, que proporciona uma visão do conjunto de protocolos IPSec e apresenta os documentos que os descrevem.

Diferente dos protocolos abordados anteriormente, que atuam na camada de aplicação, o IPSec atua na camada de rede. Isto significa que ele não interfere diretamente na tecnologia VoIP e pode ser aplicado a outros tipos de mídia.

O IPSec não define o uso de nenhuma técnica de cifragem ou método de autenticação. Na verdade, ele fornece uma estrutura e um mecanismo, deixando a escolha do tipo de cifragem, autenticação e métodos de integridade para o usuário [2]. Por questões de interoperabilidade, todas as implementações IPSec suportam alguns algoritmos pré-definidos. A maioria das implementações devem suportar MD5 (Message-Digest algorithm 5) ou SHA para realizar a autenticação do host de origem, e DES (Data Encryption Standard) ou 3DES (Triplo Data Encryption Standard) para realizar a criptografia.

O IPSec se baseia na criptografia de chave simétrica, e como o transmissor e o receptor negociam uma chave compartilhada antes de instalar uma AS (Associação de Segurança) [1].

O IPSec pode ser utilizado para proteger um canal de comunicação entre dois hosts finais ou entre dois gateways de segurança, que podem ser firewalls. Para isso utiliza dois modos: transporte e túnel. Além disso o IPSec define dois novos cabeçalhos de extensão: AH (Authentication Header) e ESP (Encapsulating Security Payload).

O IPSec trabalha ainda com o conceito de AS, uma espécie de canal virtual de comunicação. O protocolo ISAKMP (Internet Security Association and Key Management Protocol) define procedimentos para o estabelecimento das AS.

Outra característica importante do IPSec é o gerenciamento de chaves de forma dinâmica, utilizando o protocolo IKE (Internet Key Exchange).

O esquema de autenticação do IPSec (tanto para o protocolo AH quanto para o protocolo ESP) usa um sistema denominado HMAC, que é um resumo de mensagem criptografado descrito na RFC 2104. O HMAC usa uma chave secreta compartilhada pelas duas partes, em vez de métodos de chaves públicas para a autenticação de mensagens.

5.2 ASSOCIAÇÃO DE SEGURANÇA

A conexão IPSec é capaz de oferecer autenticidade, integridade e confidencialidade. Para isso é necessário que os dois dispositivos que estão se comunicando determinem quais algoritmos utilizarão, além de compartilharem as chaves criptográficas de sessão, para criarem uma conexão lógica de camada de rede. Esse canal lógico é denominado Associação de Segurança e é o método utilizado pelo IPSec para o gerenciamento de todos os elementos da conexão

segura entre os dois dispositivos, transformando a tradicional camada de rede não orientada a conexão em uma rede com conexões lógicas.

Uma AS é uma conexão em um único sentido entre dois pontos extremos e tem um identificador de segurança associado a ela. Se houver necessidade de tráfego seguro em ambos os sentidos serão exigidas duas associações de segurança [1].

Os serviços de autenticidade, integridade e confidencialidade, citados anteriormente e viabilizados em uma AS, são oferecidos pelos protocolos AH e ESP.

Uma Associação de Segurança é identificada por três elementos: um identificador de protocolo de segurança (AH ou ESP); o endereço IP destino para a conexão; e um identificador de conexão lógica de 32 bits denominado Índice de Parâmetros de Segurança.

Uma AS pode operar em dois modos distintos: transporte e túnel. No modo transporte, em uma AS entre dois dispositivos, somente o cabeçalho IP fica exposto, não recebendo qualquer tipo de segurança. Após o cabeçalho IP aparece o cabeçalho IPSec, sendo seguido pelo cabeçalho de camada TCP/UDP e a carga útil *(payload)* do pacote. (Figura 3).

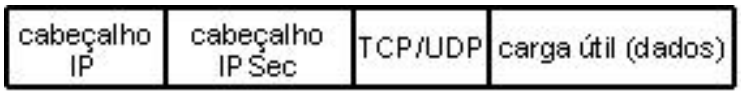

Figura 3: IPSec em modo transporte.

No modo túnel, o tráfego IP transmitido pelos dispositivos é recebido pelo gateway de segurança (roteador, firewall) para então ser protegido. Todo pacote, incluindo o cabeçalho IP, é protegido. Um novo cabeçalho IP é acrescentado, após

este o cabeçalho IPSec, que é seguido de todo o pacote IP com o cabeçalho IP original. (Figura 4).

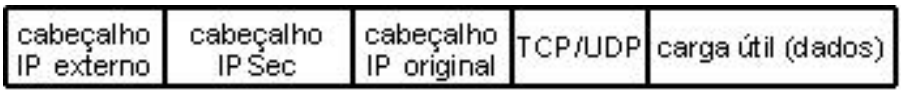

Figura 4: IPSec em modo túnel.

A maior vantagem do modo túnel é que os sistemas finais não precisam ser modificados para utilizar os serviços do IPSec. O modo túnel também protege contra a análise do tráfego, pois é possível apenas determinar os endereços dos gateways de segurança e não os endereços reais de origem e destino dos pacotes dentro dos túneis [11].

5.3 PROTOCOLO AH (AUTHENTICATION HEADER)

O protocolo AH foi desenvolvido com o objetivo de autenticar o dispositivo de origem e assegurar a integridade da carga útil transportada pelo pacote IP, mas não tem o objetivo de assegurar a confidencialidade do pacote. O protocolo calcula a síntese da mensagem, usando uma função de hash e uma chave simétrica, e insere a síntese no cabeçalho do protocolo AH [2].

A posição em que o cabeçalho AH será inserido no pacote IP dependerá do modo de operação da AS (transporte ou túnel), conforme já visto.

Quando o AH é utilizado em modo transporte, o cabeçalho é inserido entre o cabeçalho IP e o TCP/UDP. Neste caso o campo Tipo de Protocolo *(Protocol Type)* do cabeçalho IP, utilizado para identificar o protocolo da camada superior, é substituído pelo valor 51 para indicar que existe um cabeçalho AH em seguida. Um

campo interno ao protocolo AH, chamado Próximo Cabeçalho, define o valor original do campo protocolo (o tipo de carga útil que é transportado pelo datagrama IP) [2].

Quando o AH é utilizado em modo túnel, o cabeçalho AH é inserido entre o cabeçalho IP externo e o cabeçalho IP original. Neste caso o campo Tipo de Protocolo do cabeçalho IP externo é substituído pelo valor 51.

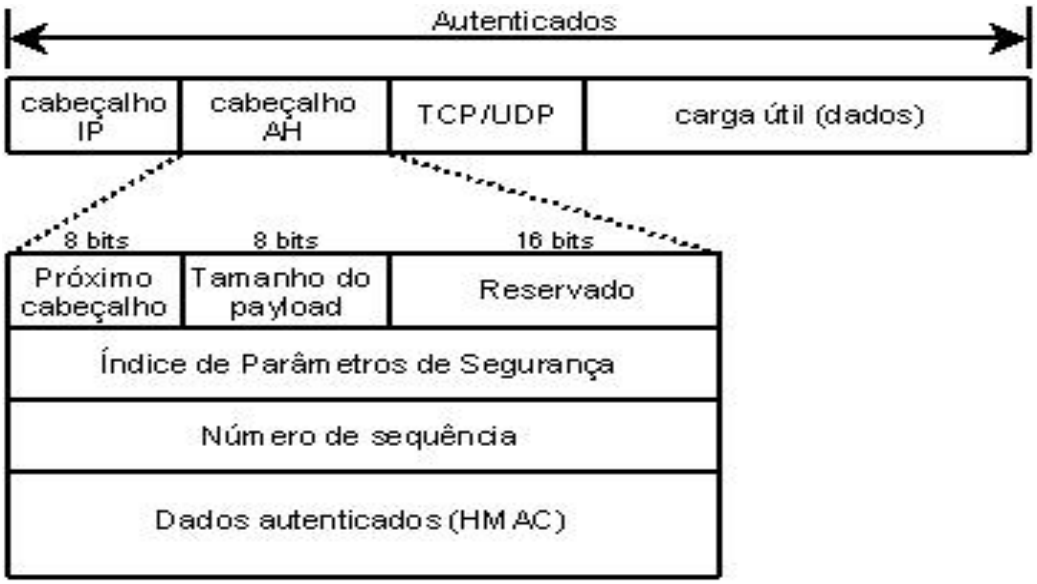

Figura 5: IPSec em modo transporte com cabeçalho de autenticação AH.

Analisando o cabeçalho AH detalhado na figura 5, temos os seguintes campos:

- **Próximo Cabeçalho:** Armazena o tipo de carga útil transportada pelo datagrama IP, ou seja, carrega o valor original do campo Tipo de Protocolo do cabeçalho IP original.
- **Tamanho do Payload:** Não especifica o tamanho da carga útil do datagrama, mas o tamanho do cabeçalho AH.
- **Índice de Parâmetros de Segurança:** Carrega o número identificador de uma Associação de Segurança.

- **Número de Sequência:** É utilizado para numerar todos os pacotes transmitidos em uma AS. Também ajuda a evitar ataques de repetição.
- **Dados autenticados:** Contém a assinatura digital da carga útil. Esta assinatura digital é processada usando o algoritmo de autenticação especificado pela AS, como MD5 ou SHA.

5.4 PROTOCOLO ESP (ENCAPSULATING SECURITY PAYLOAD)

O protocolo ESP é uma versão IPSec posterior ao AH e foi desenvolvido para fornecer autenticação, integridade e privacidade da informação. O ESP adiciona um cabeçalho, um rótulo chamado *trailer* e a autenticação dos dados ao pacote IP original. Os dados autenticados são adicionados ao final do pacote, diferentemente do AH.

O ESP utiliza muitos dos mesmos itens encontrados no cabeçalho de autenticação, mas reorganiza sua ordem [3].

Quando um datagrama IP transporta um cabeçalho e um *trailer* ESP, o valor do campo Tipo de Protocolo do cabeçalho IP original, quando em modo transporte, e do cabeçalho IP externo, quando em modo tunelamento, é igual a 50, pois indica que o próximo cabeçalho é ESP. O campo Próximo Cabeçalho, interno ao *trailer* ESP, mantém o valor do campo Tipo de Protocolo do cabeçalho IP original, identificando assim, o protocolo original da carga útil, como exemplo, TCP/UDP.

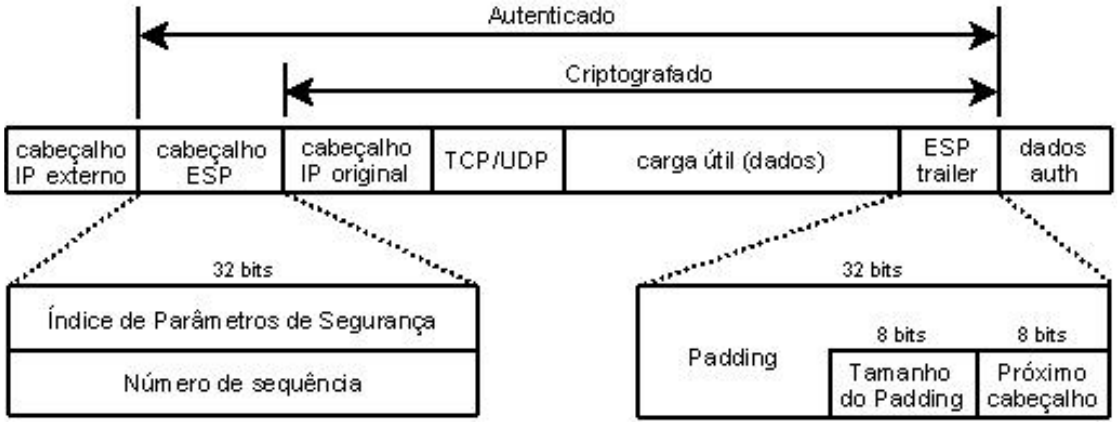

Figura 6: IPSec em modo tunelamento com cabeçalho ESP.

Os campos internos do cabeçalho ESP e do *trailer* ESP são os seguintes:

- **Índice de Parâmetros de Segurança:** Similar ao campo de mesmo nome do protocolo AH. Carrega o número identificador de uma Associação de Segurança.
- **Número de Sequência:** Similar ao campo de mesmo nome do protocolo AH. É utilizado para numerar todos os pacotes transmitidos em uma AS. Também ajuda a evitar ataques de repetição.
- **Enchimento *(Padding)*:** Campo de tamanho variado indo de 0 a 255 bytes. Serve como bits de enchimento para o trailer ESP.
- **Tamanho do Enchimento:** Define a quantidade de bytes do padding, podendo ir de 0 a 255, sendo que o valor máximo dificilmente é usado.
- **Próximo Cabeçalho:** Similar ao campo de mesmo nome do protocolo AH, identificando o valor do campo Tipo de Protocolo do cabeçalho IP original.
- **Dados Autenticados (dados auth):** Este campo também é similar ao de mesmo nome do protocolo AH, porém não fica dentro do cabeçalho,

mas é adicionado ao final do pacote. É o resultado da aplicação do algoritmo para a autenticação dos dados nos campos exibidos na figura 6 do pacote IPSec .

Sistemas em tempo real, como as aplicações VoIP, necessitam de confidencialidade e como geralmente a criptografia não é realizada nos sistemas finais, e sim nos gateways, a melhor escolha para tornar o tráfego de voz seguro é utilizar o cabeçalho ESP em modo túnel [6].

6 ANÁLISE DOS PROTOCOLOS DE SEGURANÇA SRTP, ZRTP E IPSEC

Os capítulos anteriores descreveram os protocolos SRTP, ZRTP e IPSec, com suas características e suas funcionalidades. Com o que foi apresentado sobre eles podemos analisá-los sob o aspecto de segurança e sobrecarga na rede. Estes dois aspectos são importantes, tendo em vista que a implementação dos protocolos deve ser feita em tráfego VoIP.

O SRTP, conforme já visto, oferece confidencialidade, autenticação da mensagem e proteção contra ataques de repetição para o tráfego RTP. Atua na camada de aplicação, portanto é específico para proteger o tráfego VoIP. Tem como objetivo final gerar uma chave simétrica e uma chave de salto para criptografia da sessão de mídia (conversação), porém existe a preocupação de como trocar estas chaves de forma segura entre os dispositivos que realizam a comunicação.

A confidencialidade vem com o algoritmo AES utilizado para geração da chave simétrica da sessão.

A autenticidade e integridade vem através do algoritmo HMAC-SHA1 que calcula o valor de hash da carga útil, do cabeçalho RTP e do número de sequência.

A proteção contra ataques de repetição ocorre com a garantia da integridade do número de sequência somado a uma lista de repetição, pois esta lista é atualizada a medida que os pacotes chegam e são autenticados. Logo um pacote já recebido e autenticado com integridade não deveria ser recebido novamente. Caso aconteça a lista acusará.

A função de derivação de chaves é muito importante quando aplicada periodicamente e resulta em benefícios de segurança. Neste caso, o SRTP pode atualizar a chave de sessão periodicamente e isto é importante para dificultar uma

possível criptoanálise por parte de uma atacante, limitando a quantidade de textos cifrados por uma mesma chave. O protocolo de gerenciamento de chaves troca apenas uma chave mestre e as demais chaves são geradas. A informação sobre a chave mestre encontra-se no campo índice de chave mestre do pacote SRTP.

O SRTP oferece as três características fundamentais para viabilizar a segurança: autenticidade, integridade e confidencialidade, além de outros fatores que agregam para segurança, como proteção contra ataques de repetição e derivação de chaves. Acrescenta dois cabeçalhos ao pacote RTP, código de autenticação da mensagem ou código de integridade da mensagem e índice de chave mestre, sendo que o segundo é opcional, mas fundamental para a função derivação de chaves.

O ZRTP completa o SRTP, pois ele realiza a troca de chaves com segurança. O ZRTP realiza esta troca utilizando o algoritmo DH. O DH é rápido e eficiente, porém não consegue proteger o tráfego contra ataque de Homem-no-Meio. Como o ZRTP não utiliza uma infra-estrutura de chave pública (o que demandaria mais desempenho e atraso) para proteger o desempenho do DH contra este ataque, a fase de compromisso de hash é utilizada para detectar este ataque. Isso é baseado em dois valores de hash, um de cada dispositivo que se comunica, que devem coincidir. Caso contrário o ataque ocorreu e, desta forma, o protocolo será finalizado. Este método de comparação é denominado SAS.

Em termos de desempenho, o modo DH não é melhor do que o modo Pré-Compartilhado do ZRTP, pois, considerando que já existam as chaves mestres vindas de um cálculo DH anterior, o ZRTP desempenha o modo Pré-compartilhado, em que todas as demais chaves derivadas são geradas porém o cálculo DH não é

executado. Isso melhora sensivelmente o desempenho para estabelecer uma sessão segura.

O IPSec atua na camada de rede e, sendo assim, um túnel IPSec protege todo tráfego entre dois dispositivos, não somente o tráfego VoIP. Essa é a grande vantagem do IPSec: proteger todo o tráfego inclusive a sinalização que não era protegida com o ZRTP/SRTP.

O IPSec trabalha com conexões lógicas em um único sentido e para cada uma delas uma chave simétrica para sessão é estabelecida.

Essas conexões lógicas, denominadas AS (Associação de Segurança), possuem dois modos de atuação: transporte e túnel. O primeiro expõe o cabeçalho IP original do pacote e o segundo expõe um novo cabeçalho IP externo, escondendo o original, e protegendo o pacote contra análise de tráfego.

Uma AS pode viabilizar autenticidade, integridade e confidencialidade, dependendo do protocolo utilizado para estabelecê-la. Se o AH for utilizado, a AS garantirá autenticidade da origem e integridade da carga útil, tendo em vista que o AH não garante confidencialidade. Porém, se o protocolo utilizado for o ESP, a AS garantirá as três características e assim a conexão lógica será bem segura.

Com relação a desempenho, o protocolo AH acrescenta somente um cabeçalho ao pacote IP, já o ESP acrescenta um novo cabeçalho e mais dois novos campos ao pacote IP, conforme já visto. Neste caso, o protocolo ESP exige maior desempenho do dispositivo e a sobrecarga na rede é maior em relação ao AH.

A melhor escolha para tornar o tráfego VoIP seguro utilizando IPSec é com o protocolo ESP em modo túnel, pois todo tráfego em tempo real necessita de confidencialidade e normalmente a criptografia não é realizada nos dispositivos

finais por causa da capacidade de processamento. Com isso o modo túnel, que protege os pacotes contra análise de tráfego, deve ser desempenhado nos gateways (roteadores ou firewalls) de forma a viabilizar a máxima proteção IPSec, exigindo um mínimo de desempenho nos dispositivos finais e maior processamento nos equipamentos de maior capacidade (roteadores ou firewalls).

7 CONCLUSÃO

De acordo com o que já foi abordado neste trabalho, é necessário entender por que é tão importante projetar uma solução de segurança para ser aplicada com o tráfego VoIP. Isso se deve às inúmeras vulnerabilidades existentes para o protocolo IPv4 e que podem ser exploradas para o tráfego VoIP, além da existência de ataques que são específicos para este tipo de tráfego.

Os protocolos de segurança abordados neste trabalho oferecem as três características fundamentais da segurança da informação: autenticidade, integridade e confidencialidade. A primeira garante o reconhecimento do dispositivo de origem, ou seja, saber quem ele é; a segunda garante que a informação enviada pela origem é exatamente a mesma que chegou no destino; e a terceira garante que esta informação não seja observada por ninguém que não seja o destino.

Quando os protocolos de segurança apresentados neste trabalho são aplicados ao tráfego VoIP temos a garantia de que a conversação está bem segura devido às funcionalidades oferecidas por estes protocolos, porém precisamos levar em consideração que quanto mais mecanismos de segurança forem implementados sobre o tráfego VoIP, mais cabeçalhos terão os pacotes, maiores ficarão os pacotes e mais processamento será exigido dos equipamentos envolvidos devido ao desempenho dos algoritmos de criptografia. Com isso os pacotes podem demorar mais a serem encaminhados devido ao processamento criptográfico exigido, prejudicando a interatividade da conversa, além de consumirem mais banda, impactando também no atraso total dos pacotes (atrasos de transmissão, de roteadores e de enfileiramento). É claro que se a largura de banda do enlace de

transmissão e o processamento dos equipamentos envolvidos forem bons, todas essas alterações serão imperceptíveis e transparentes para o usuário

O custo-benefício de se trafegar a voz sobre IP com segurança é muito bom, tendo em vista que, tomando os devidos cuidados, a sobrecarga gerada pelos mecanismos pode ser transparente para o usuário final e não afetar a rede.

Segundo [11], que realizou experimentos com IPSec, o impacto do tráfego de voz utilizando IPSec:

> "Não foi substancial para prejudicar o bom fluxo dos dados de voz na rede para experimentos, mas com escalabilidade esses novos aspectos introduzidos pelo IPSec podem se tornar prejudiciais à qualidade do tráfego de voz".

De acordo com [14]:

"SRTP apresenta um framework com baixo custo computacional e de banda", logo, concluímos que sua implementação não causará transtornos à rede, desde que se tomem os devidos cuidados.

Não se pode abrir mão de autenticidade, integridade, confidencialidade, disponibilidade e interatividade quando se fala em VoIP, portanto é fundamental adequar a estrutura da rede de dados a fim de suportar a implementação de segurança sem perder a perfeita interatividade nas chamadas, que, ao final das contas, é a característica que medirá a qualidade do serviço.

8 REFERÊNCIAS BIBLIOGRÁFICAS

[1] TANENBAUM, Andrew. **Redes de Computadores.** Ed. Campus/Elsevier, 4ª edição, 2003.

[2] FOROUZAN, Behrouz. **Comunicação de Dados e Redes de Computadores.** Ed. Bookman, 3ª edição, 2006.

[3] COMER, Douglas. **Interligação de Redes com TCP/IP, vol.1 princípios, protocolos e arquitetura.** Ed. Campus, 5ª edição, 2006.

[4] COLCHER, Sergio. et al. **VoIP: Voz sobre IP**. Ed. Campus, 2005.

[5] DAVIDSON, Jonathan. et al. **Fundamentos de VoIP.** Ed. Bookman, 2008.

[6] BARBIERI, Roberto. et al. **Voice over Ipsec: Analysis and Solutions**. Dipartimento di Scienze dell'Informazione: Università degli Studi di Milano, 2002.

[7] KUHN, Richard. et al. **Security Considerations for Voice Over IP Systems. Recommendations of the National Institute of Standards and Technology.** Gaithersburg, January 2005.

[8] BILIEN, Johan. et al. **Secure VoIP: call establishment and media protection.** Royal Institute of Technology (KTH), Stockolm, Sweden.

[9] BRESCIANI, Riccardo. **The ZRTP Protocol: *Security Considerations.*** Research Report LSV-07-20. Laboratoire Spécification et Vérification. Centre National de la Recherche Scientifique. Ecole Normale Supérieure de Cachan, May 2007.

[10] SOTILLO, Samuel. **Zfone: A new Approach for Securing VoIP Communication.** ICTN 4040. Spring 2006.

[11] PASSITO, Alexandre. et al. **Análise de desempenho de tráfego VoIP utilizando o Protocolo IP Security.** Laboratório de Voz sobre IP, UFAM, 2004.

[12] RODRIGUES, P. H. A. **Mídia Segura**. Programa MOT-CN, NCE/UFRJ, 2007.

[13] HOUSLEY, R. **RFC 3686: Using Advanced Encryption Standard (AES) Counter Mode With IPsec Encapsulating Security Payload (ESP).** January 2004.

[14] BAUGHER, M. et al. **RFC 3711: The Secure Real-Time Transport Protocol (SRTP)**. March 2004.

[15] JACOBSON, V. et al. **RFC 2327: SDP: Session Description Protocol**. April 1998.

[16] ZIMMERMANN, Phil. Internet-Draft: **ZRTP: Media Path Key Agreement for Secure RTP.** Draft-zimmermann-avt-zrtp-04. July 2007. http://tools.ietf.org/html/draft-zimmermann-avt-zrtp-04, acesso em março de 2010.

[17] **Federal Comunication Comission** (tradução – FCC; 2004; pág. 2-3).

[18] http://en.wikipedia.org/wiki/Secure_Real-time_Transport_Protocol, acesso em março de 2010.

[19] http://www.networksorcery.com/enp/protocol/srtp.htm, acesso em março de 2010.

[20] http://en.wikipedia.org/wiki/ZRTP, acesso em março de 2010.

[21] PETRASCHEK, Martin. et al. **Security and Usability Aspects of Man-in-the-Middle Attacks on ZRTP.** Telecommunications Research Center Vienna and University of Vienna, Austria, 2008.