

---

# Media Security in VoIP Systems

Weidong Shao  
Consultant, Security and Software Development

# Agenda

---

## VoIP Security and Media Security

### - Why media is different

- How to make it secure?
- How to set the encryption keys in the right way?

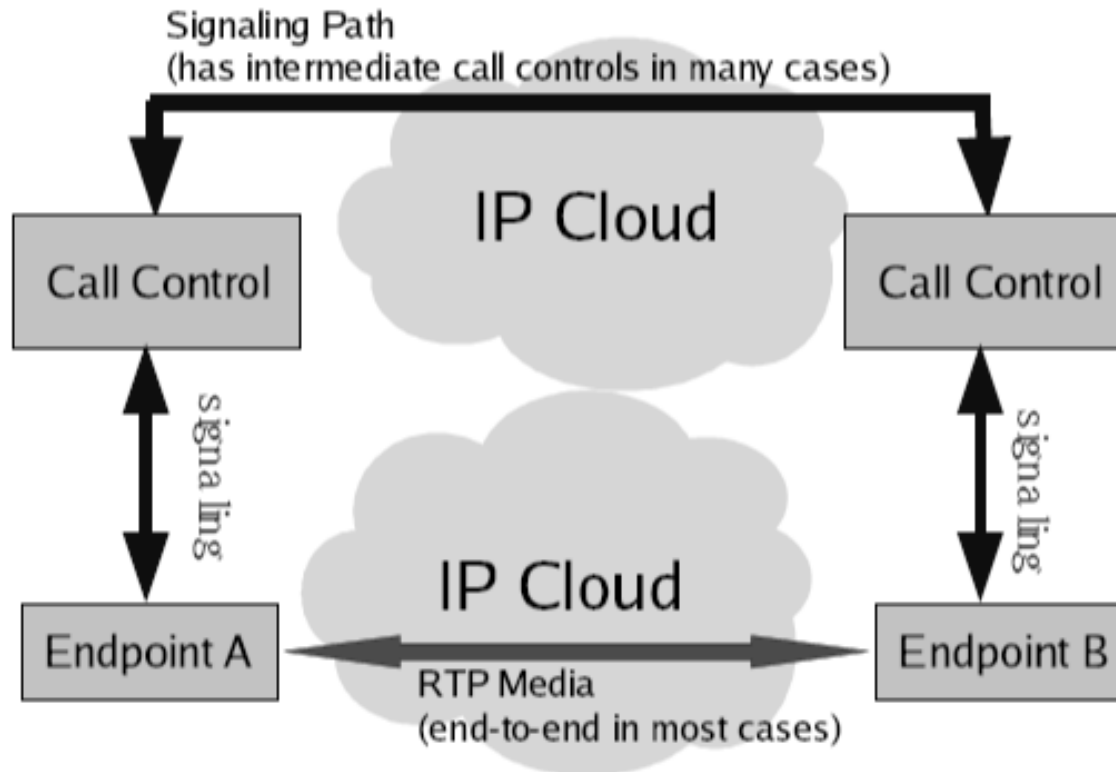
### What is NOT covered

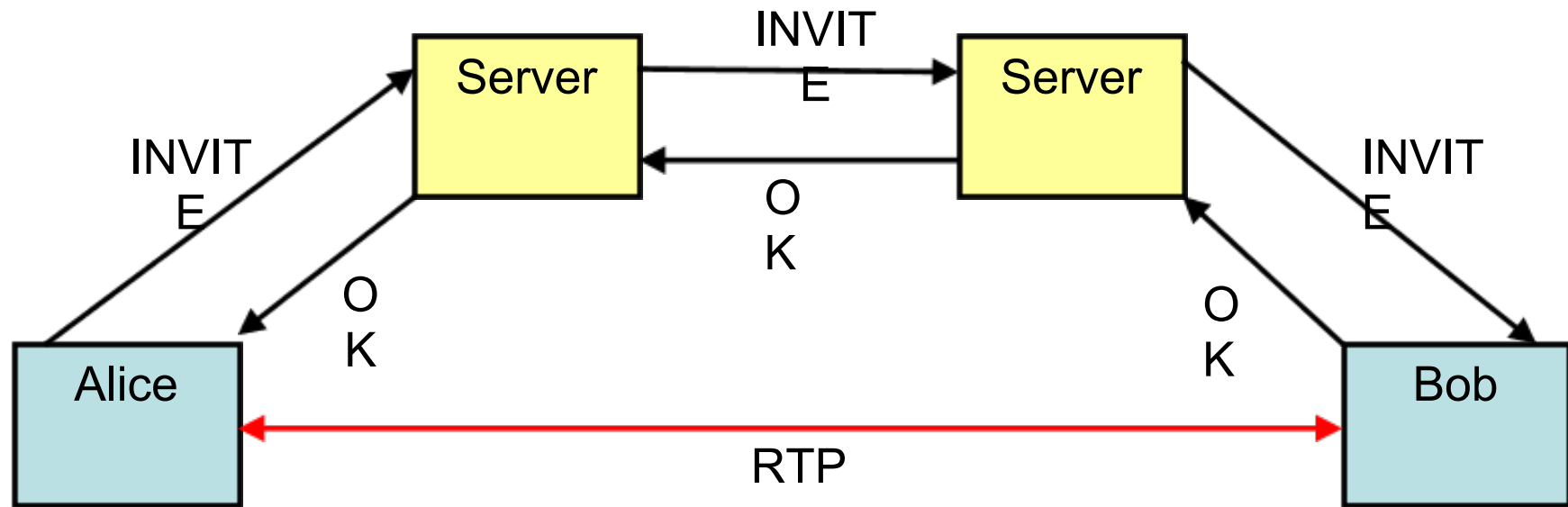
- Introduction to VoIP
- Signaling Protocols
- Signaling security

### Note:

Some slides are based on IETF presentations by draft/RFC authors.

# VoIP: Signaling and Media





Signaling: registration, call setup and termination, accounting etc.

- Often goes through multiple hops

Media: RTP packets.

- Usually end-to-end
- Transport in open network, shared with data

## **Arrests Reveal Vulnerability Of Web Phone Service to Fraud**

By **DIONNE SEARCEY** and **SHAWN YOUNG**

*June 8, 2006; Page B1*

The rise of Internet calling, hailed for its ability to bring consumers inexpensive phone service, is providing an opportunity for hackers, mischief makers and scam artists.

Yesterday, federal authorities arrested the head of two small Miami telecom companies, as well as a Spokane, Wash., computer programmer, for hacking into the networks of as many as 15 other Internet phone providers to fraudulently route customers' calls, according to a federal complaint filed in New Jersey.

The telecom company's owner, Edwin Andres Pena, spearheaded a scheme that scanned the networks of unsuspecting companies, searching for weak spots to exploit and use to route his own customers' calls, the government alleges. He then billed a Newark, N.J., Internet telephone company for more than 500,000 unauthorized calls that he had sold to his customers at deeply discounted rates, the complaint said. Robert Moore, of Spokane, helped Mr. Pena hack into routers to disguise the calls' origins. Mr. Pena collected more than \$1 million in connection fees and used the money to buy a motor boat, luxury cars and Miami real estate, according to the complaint.

- June 8, 2006: Edwin Pena Arrested for VoIP Fraud
  - Half a million calls through other VoIP providers
  - 10 million call minutes
  - 1 million US dollars from the “service”
  - US\$20K paid to a hacker

## VoIP's Security Vulnerabilities

Posted by [Zonk](#) on Tuesday June 13, @11:26AM  
from the [is-your-refridgerator-running](#) dept.

garzpacho writes

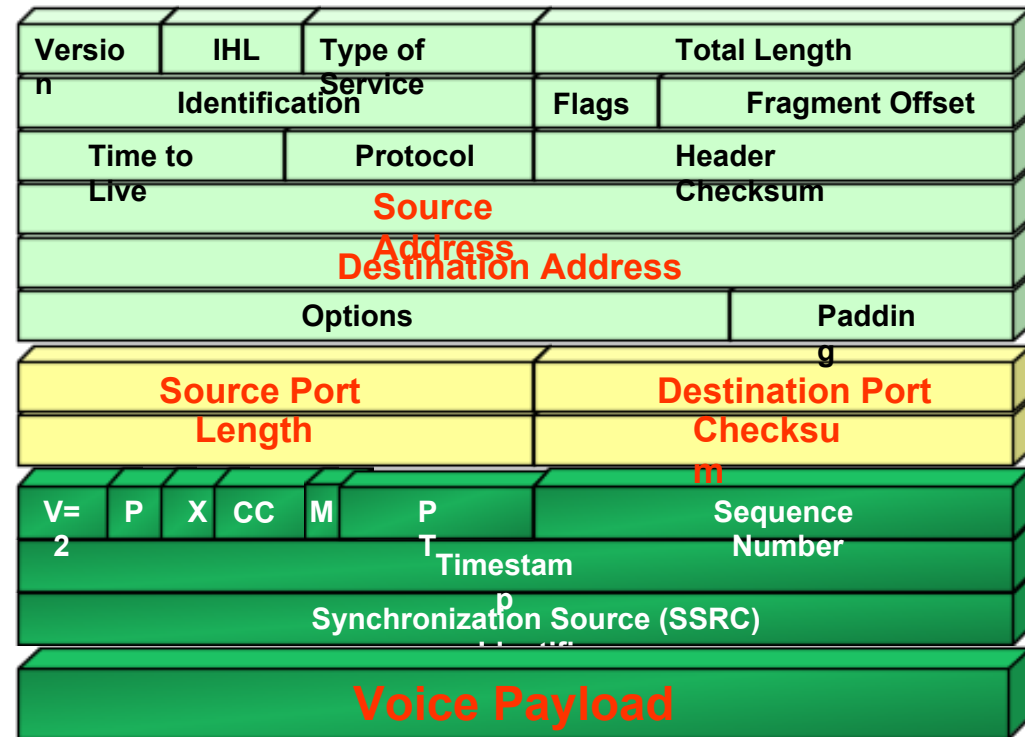
"Experts predict that attacks on VoIP systems could be right around the corner, and are calling for preemptive security measures. The BusinessWeek article compares the current state of voice-over-IP to the pre-spam email era and suggests that spammers could be the first to exploit the system. From the article: 'Here's what VoIP security breaches could mean for consumers. For starters, it's a big channel for spammers. Think of the Viagra ads that flood your e-mail inboxes now. They work because the cost of e-mailing thousands of people at once is so low, only 1% to 3% or so need to respond for it to be worth it, Ingevaldson says. Comparable economics apply to VoIP calls, he says. Then there are potential phishing attacks, where fraudsters posing as banks lead consumers to fake sites. Those and other attempts at identity theft could spring up via VoIP accounts too, experts say. Imagine the messages from relatives of deposed Nigerian dictators -- only this time they're on voice mail, too.'"



- Signaling security
  - client authentication, call admission control, and accounting
  - Digest authentication
  - TLS
  - S/MIME
- Media security
  - Confidentiality,
  - Integrity, SPIT
  - anti-replay protection etc.
- Key management for media encryption

# RTP – Real-time Transport Protocol

- RFC 3550/RFC 1889
- Use UDP
- Dynamically negotiated sessions
  - fields in red



## Observations:

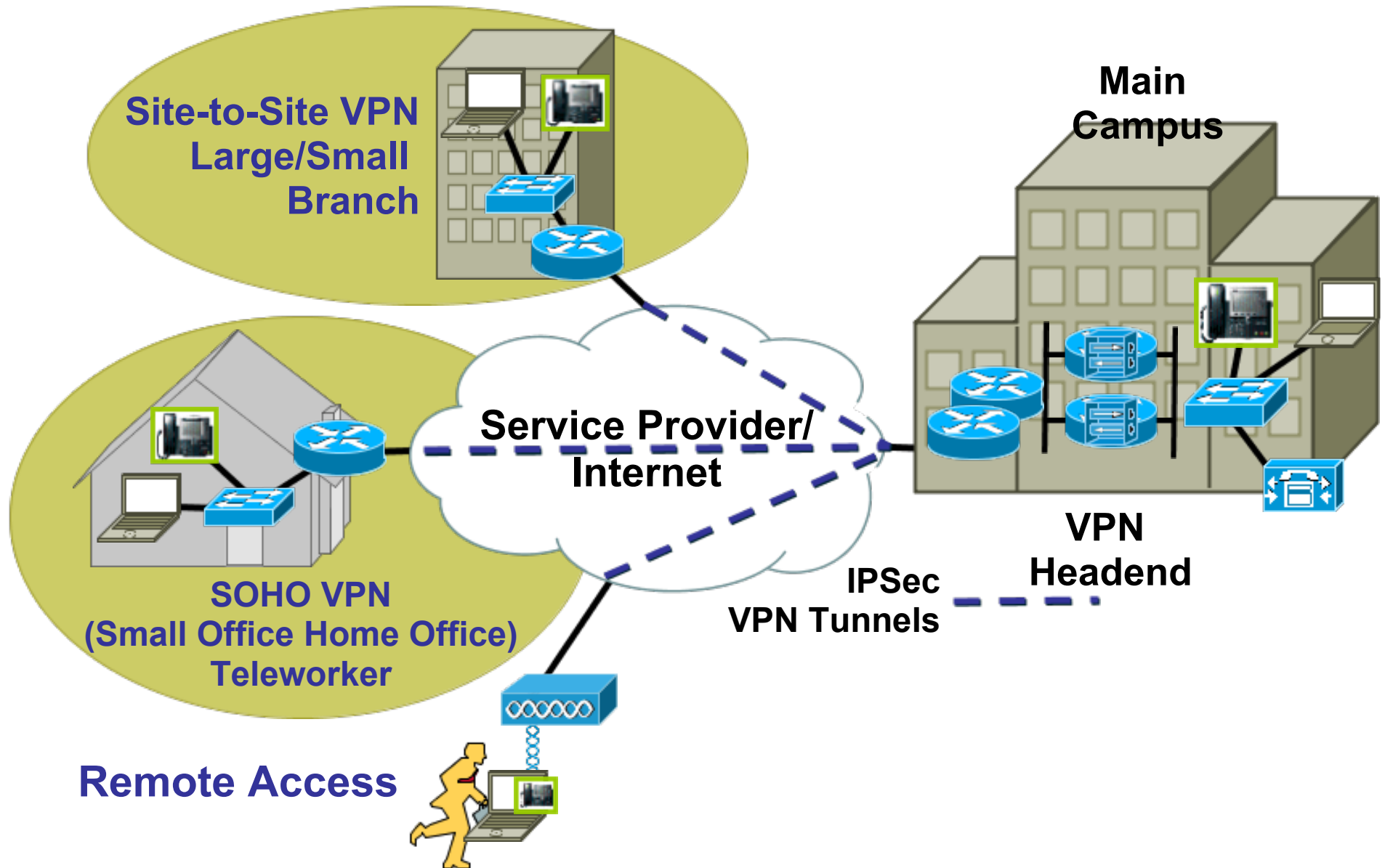
- smaller payload size (compared with data), at constant rate
- typical 10/20/30ms codec size
- relatively large overhead from protocol encapsulations

- Delay sensitive
  - Delayed re-transmission is worse than no transmission
- Delay sources
  - Transmission and propagation delay
    - e.g, 800-byte packet over 64kpbs = 100ms
  - Processing delay
  - Codec processing delay
- Delay variation – jitter
- What can help?
  - CoS, DiffServ, RSVP
  - Link fragmentation and interleaving
  - cRTP (compressed RTP)

## IPSEC

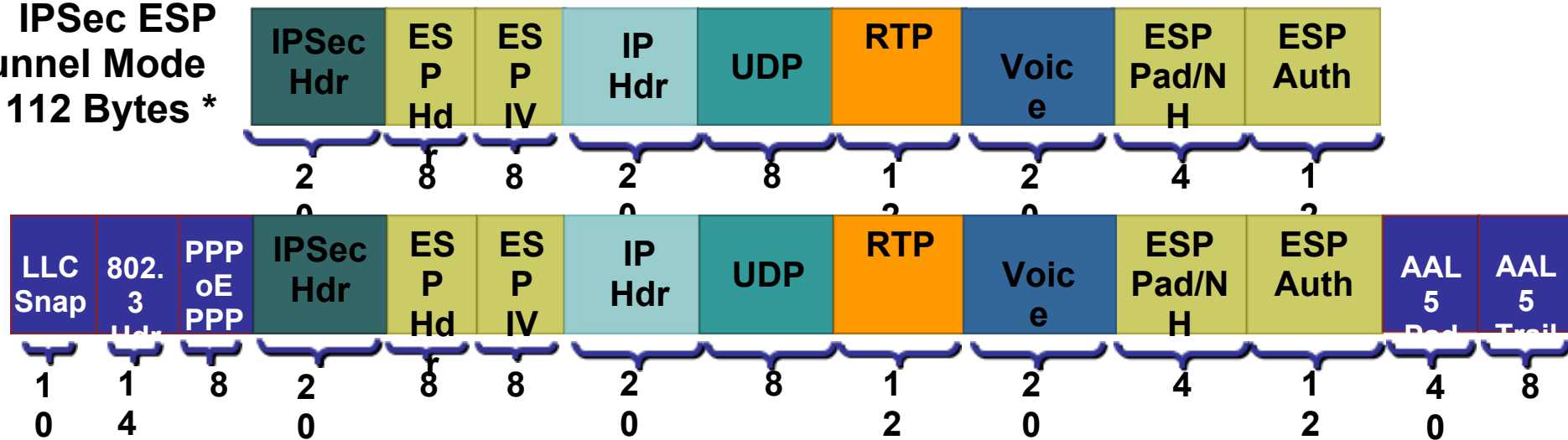
- Rely on network infrastructure.
  - Transparent to applications.
    - No application code changes required
  - Good for site-to-site or remote-access VPN
    - e.g., reuse the enterprise VPN infrastructure
  - Possible but not feasible for peer-to-peer
  - Not scalable

# VoIP Protected by IPSEC-based VPN



# QoS Impacts

- Header expansion  
IPSec ESP  
Tunnel Mode  
112 Bytes \*



- Crypto-engine factors
  - Speed may not be an issue for crypto hardware
  - Queuing/scheduling not QoS aware
- Intermediate devices cannot apply QoS techniques to help

- Transport security is widely used for TCP-based applications.

TLS, HTTPS

- TLS review:

- Handshake protocol

- Use public-key cryptography to establish a shared secret key between the client and the server
- Most applications authenticate servers through certs, client at Layer 7.

- Record protocol

- Use the secret key established in the handshake protocol to protect communication between the client and the server

## DTLS – Datagram TLS

- Leverage the success of TLS
- Generic for UDP applications
- RFC 4743
  
- RTP over DTLS
  - DTLS is inherently client-server, not p2p. e.g. in common handshake, server auth by certs, client auth with password
  - Header overhead (significant in percentage). Breakage of cRTP
  - SRTP compatibility mode (IETF draft, deprecated?)

- RFC 3711 Secure RTP (SRTP)
  - Confidentiality
  - Message authentication
  - Anti-replay
- Specially designed for RTP
  - Low computational overhead
  - Low bandwidth cost (small header expansion)
  - Does not break cRTP
- Key Derivation Function (KDF)
  - From a master key, salt

Works like a stream cipher, e.g. RC4

- XOR key stream with plain text:

$$CT[i] := PT[i] \oplus AES(CTR(i))$$

- Increment Counter

Counter encrypted to generate keystream

- Counter **MUST** never be re-used (with same key)

No harm if counter is public

- But **MUST** be initially unpredictable

## Why media encryption

- How to make it secure?
- **How to set the encryption keys in the right way?**

- Simple: A generates a key and sends to B
  - How to transport the key in a secure manner?
- Is the signaling channel secure?  
e.g., Secured by TLS, IPSEC,
- If not, is there anyway to put a secure envelope?  
e.g, if PKI is in place, just encrypt the key with the other party's public key

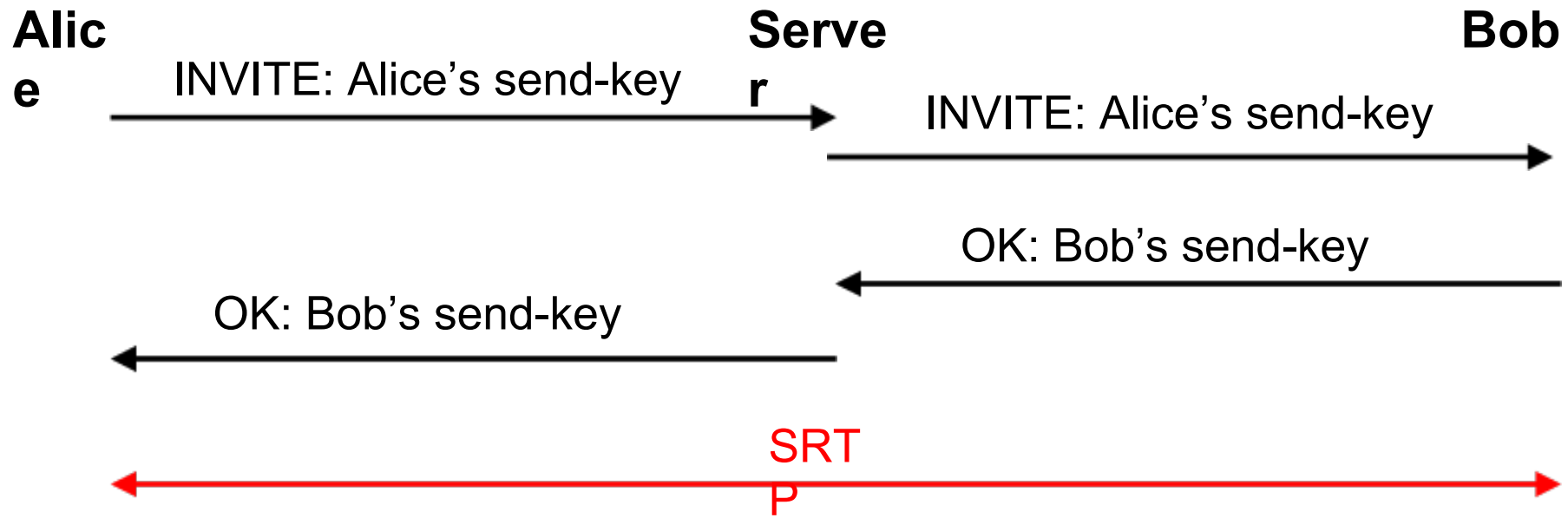
### RFC 4568 – Session Description Protocol Security Description for Media Streams

#### Concepts:

- assume signaling channel is secure
- send the keying material in clear (base64 encoded)

# SDP Security Descriptions

---



Require a secure channel

# SDP Security Description

---

```
v=0
o=jdoe 2890844526 2890842807 IN IP4 10.47.16.5
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.example.com/seminars/sdp.pdf
e=j.doe@example.com (Jane Doe)
c=IN IP4 161.44.17.12/127
t=2873397496 2873404696
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
```

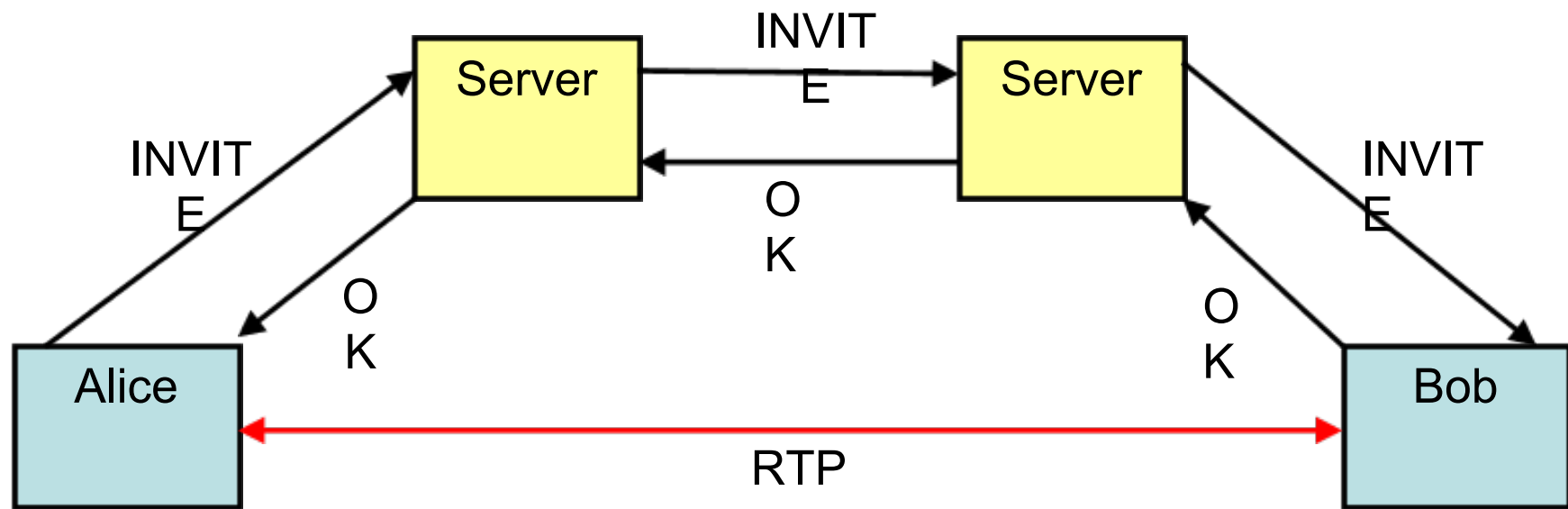
## But, is it really secure?

---

- Signaling usually goes to multiple servers
- Secure signaling end-to-end?
  - Why hop-by-hop TLS?
- All servers have access to keying materials
  - Do you trust servers?
  - Any server can choose to record the information
    - Intentionally
    - Or leaked in debug logs
- Legal interception is easy

- Session keys are generated by the server and distributed in a secure transport
- Clients trust the server
- Pros:
  - Simple to implement
  - Easy for securing voice conference
- Cons
  - Not end-to-end secure
  - Interception is possible through server
  - Secure transport may be difficult across domains (hop-by-hop protection is hard)

## End-to-end v.s. hop-by-hop



- Hop-by-hop security
  - Difficult in configuration and management
  - Require a common trust model
  - Does not necessarily provide end-to-end security

1.  $X = g^x$ , send public value  $X$
  2.  $Y = g^y$ , send public value  $Y$
  3.  $K = X^y = Y^x$
- All communication in clear, an observer can record the exchange
    - Mathematically infeasible to calculate  $K$
    - “Assumption” on Discrete Log Computational Problem
    - No need to secure the message
  - No authentication. Man-in-the-middle attack
  - End-to-end security is possible

PKI solves a lot of problems in key management

Relatively easy to build a self-sufficient PKI model in an enterprise system

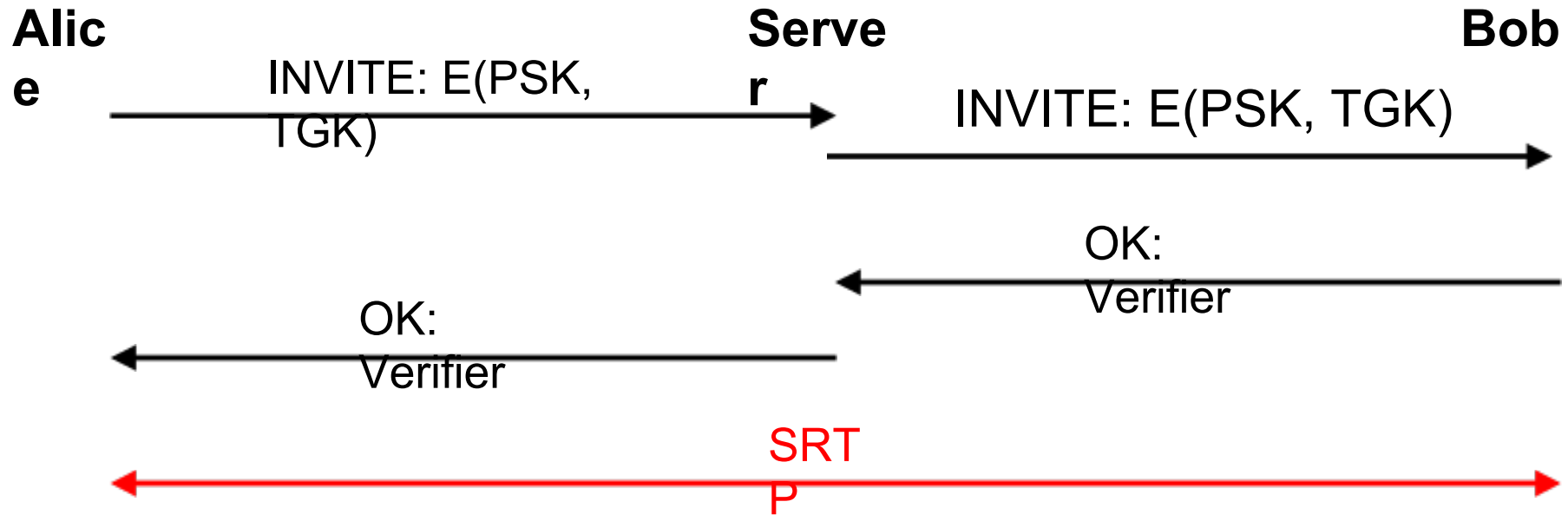
- PKI interoperability is an issue (usually in configuration/management)
- Good for closed systems

Problems:

- Federated users: some manual configuration for trust model
- Certificate storage: easy for hard-phone or security USB token
  - Difficult for soft-client: stored in PC with password protection?
  - Certificate on the fly?
    - Dynamically generate at time of registration

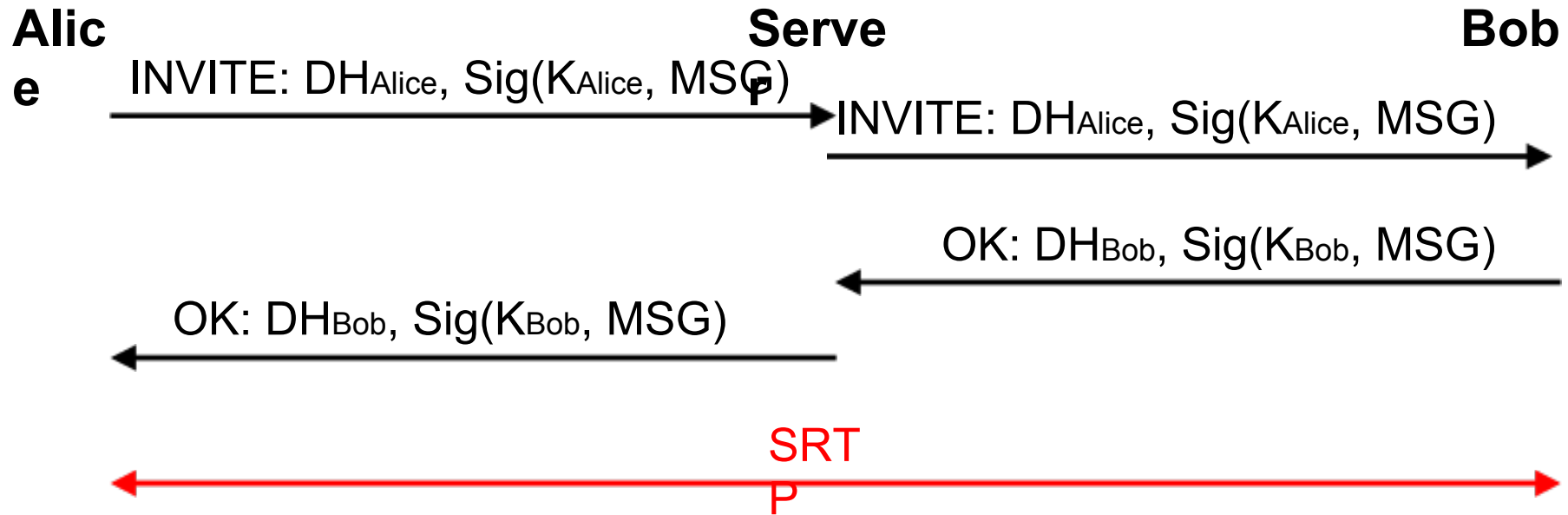
- Do you use authenticated key exchange?
- Where does the exchange happen?
  - Signaling path
  - Media path (in-band)

- RFC 3830
- MIKEY-PSK
  - Use pre-shared key to encrypt the exchange
  - Key transport with secure envelope
- MIKEY – RSA
  - Use public key to encrypt the exchange
  - Key transport with secure envelope
- MIKEY – DH-SIGN
  - Authenticated key exchange
- Other modes are proposed in IETF drafts
- Q: Does this really work?



No need to secure the channel  
Pre-shared key is not scalable

# MIKEY – RSA Sign

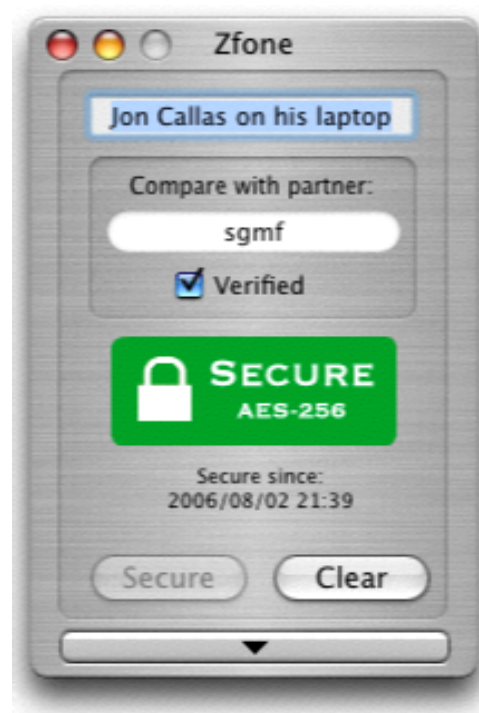


No need to secure the channel  
Require PKI for certificate verification

- RTP encryption but different from SRTP
- Keying materials from D-H exchange
  - H.235 Annex D – similar to MIKEY DH-HMAC
  - H.235 Annex E – similar to MIKEY RSA-SIGN
- H.235 Annex G: added SRTP with MIKEY

- From the creator of PGP
  - “No backdoor” philosophy
  - Trusted 3<sup>rd</sup>-party may not be trustworthy for true security
- Why bother AKE if PKI does not work
  - Use un-authenticated Diffie-Hellman
- What about man-in-the-middle attack?
  - Attack for D-H is possible
  - Useful for active attacks only if there is media relay

- Define an RTP header extension for D-H
- D-H right inside RTP
  - Innovative idea
  - In-band
  - No need to trust your servers
    - No back-doors
- Out-of-band verification
  - Commit string
  - From previous conversation



Prototype from Phil  
First available in June 2006.

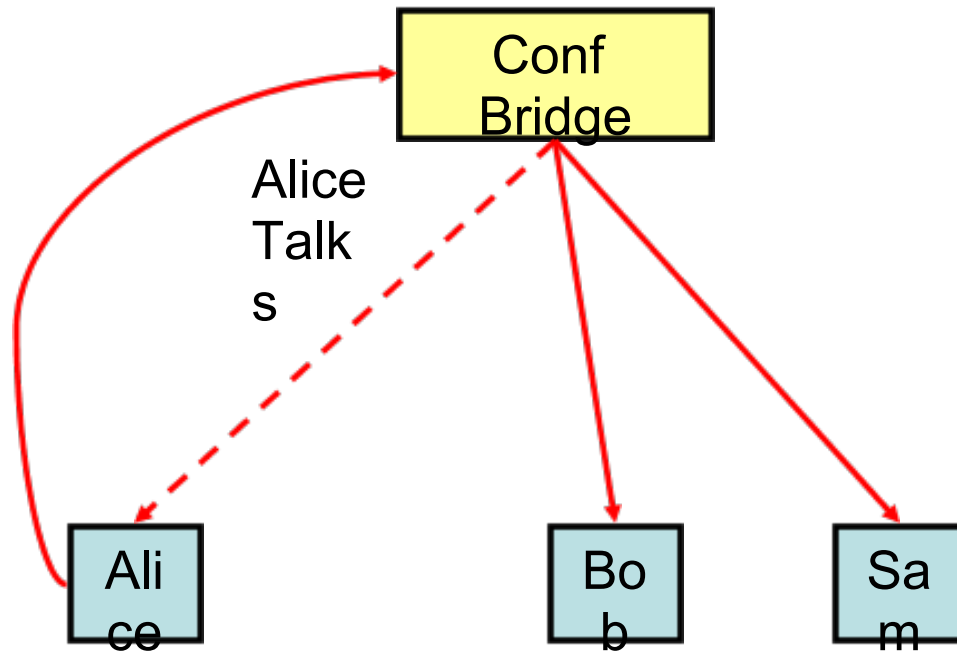
- IETF draft by David McCrew
- Setup in the media path
  - DTLS is directly between the 2 end-points
  - End-to-end secure
- The question is
  - How to authenticate DTLS itself?

Useful in key distribution by a server

- Each client uses D-H to negotiate a key encryption session key.
  - Client-to-server authentication is always easier
    - Trust model is well-defined
- Server transports the key to a client using secure envelope
  - Envelope key derived from D-H exchange
- Recommended for voice conference encryption

# Secure a conference call

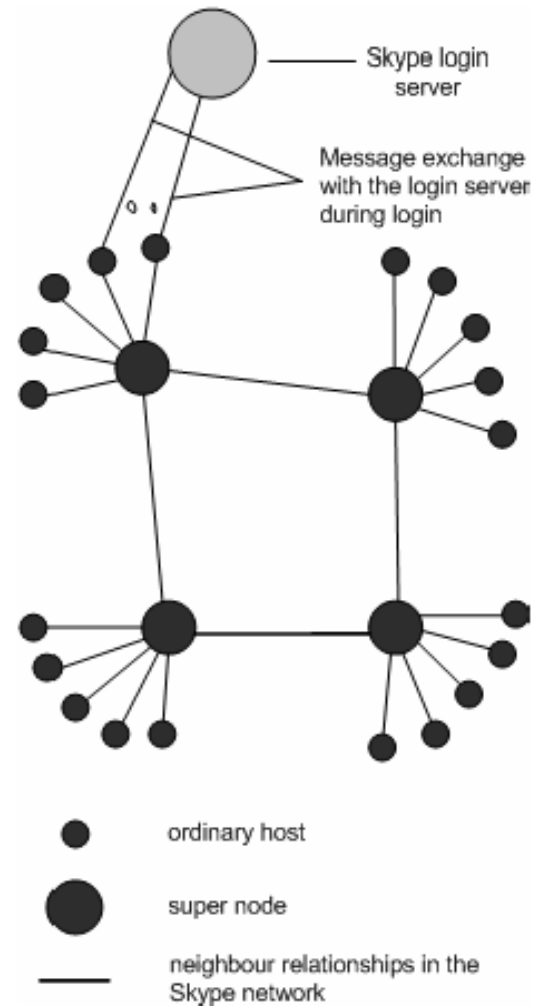
---



Different media stream to each participant

# Skype Security

- Black-box security
- Claims from “external” expert review:
  - End-to-end secure
  - AES-256 encryption for IM and Voice
- Dynamic client certificate
  - Changed if logged in from a new PC
- AKE using D-H?
  - We can only guess



- Most IMs are in plain text
- TLS may be used for client-server communication
- File transfer, video chat
- OTR (Off-The-Record)



1. RFCs: 3830, 4568,
2. DTLS extension for keying of SRTP,  
<http://www.ietf.org/internet-drafts/draft-mcgrew-tls-srtp-00.txt>
3. ZRTP IETF draft