SIP Security

Prof. Dr. Andreas Steffen

Zürcher Hochschule Winterthur

andreas.steffen@zhwin.ch

Agenda

- SIP – The Session Initiation Protocol
- Securing the Session Management
- Securing the Media Streams
- Conclusions
**Session Initiation Protocol (RFC 3261)**

```
sip:alice@atlanta.com  atlanta.com    biloxi.com  sip:bob@biloxi.com
```

**User Agent**  **Proxy**  **Proxy**  **UA**

- **INVITE F1**
- **100 Trying F3**
- **180 Ringing F8**
- **200 OK F11**

- **INVITE F2**
- **100 Trying F5**
- **180 Ringing F7**

- **INVITE F4**
- **180 Ringing F6**

- **200 OK F9**
- **200 OK F10**

- **ACK F12**

**Media Session**

- **BYE F13**
- **200 OK F14**

---

**Basic SIP Trapezoid**

```
sip:alice@atlanta.com
atlanta.com
UA
```

```
sip:bob@biloxi.com
biloxi.com
UA
```

**Hop 1**

**Hop 2**

**Hop 3**

**Direct path**

**Media stream**

- **udp/sip or tcp/sip**  Session Management
- **udp/rtp**  Media Streams
# Securing the Session Management

<table>
<thead>
<tr>
<th>Authentication methods:</th>
<th>Authentication</th>
<th>Data Integrity</th>
<th>Confidentiality</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSK Pre-Shared Keys</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PKI Public Key Infrastructure</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Authentication</th>
<th>Data Integrity</th>
<th>Confidentiality</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP 1.0 Basic Authentication</td>
<td>PSK</td>
<td>-</td>
<td>-</td>
<td>Deprecated by SIPv2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Insecure transmission of password</td>
</tr>
<tr>
<td>HTTP 1.1 Digest Authentication</td>
<td>PSK</td>
<td>-</td>
<td>-</td>
<td>Challenge/response exchange based on MD5 hash of [strong] password</td>
</tr>
<tr>
<td>Pretty Good Privacy (PGP)</td>
<td>PKI</td>
<td>✓</td>
<td>✓</td>
<td>Deprecated by SIPv2</td>
</tr>
<tr>
<td>Secure MIME (S/MIME)</td>
<td>PKI</td>
<td>✓</td>
<td>✓</td>
<td>For encryption the public key of the recipient user agent must be known</td>
</tr>
<tr>
<td>SIPS URI (TLS)</td>
<td>PKI</td>
<td>✓</td>
<td>✓</td>
<td>SIP application and proxies must tightly integrate TLS</td>
</tr>
<tr>
<td>IP Security (IPsec)</td>
<td>PKI</td>
<td>✓</td>
<td>✓</td>
<td>Integration with SIP application not required but proxies must be trusted</td>
</tr>
</tbody>
</table>

# Securing the Media Streams

<table>
<thead>
<tr>
<th>Authentication methods:</th>
<th>Authentication</th>
<th>Data Integrity</th>
<th>Confidentiality</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSK Pre-Shared Keys</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PKI Public Key Infrastructure</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>Authentication</th>
<th>Data Integrity</th>
<th>Confidentiality</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure RTP (SRTP)</td>
<td>PSK</td>
<td>✓</td>
<td>✓</td>
<td>Uses master key which must be distributed by other means</td>
</tr>
<tr>
<td>IP Security (IPsec)</td>
<td>PKI</td>
<td>✓</td>
<td>✓</td>
<td>Integration with SIP application not required but peer must be trusted</td>
</tr>
</tbody>
</table>
# Securing the Media Streams

## Secure RTP Packet Format (RFC 3711)

<table>
<thead>
<tr>
<th>01234567012345670123456701234567</th>
<th>sequence number</th>
</tr>
</thead>
<tbody>
<tr>
<td>V P X CC M PT</td>
<td>timestamp</td>
</tr>
<tr>
<td></td>
<td>synchronization source (SSRC) identifier</td>
</tr>
<tr>
<td></td>
<td>contributing source (CSRC) identifiers</td>
</tr>
<tr>
<td></td>
<td>RTP header extension (optional)</td>
</tr>
<tr>
<td></td>
<td>encrypted</td>
</tr>
<tr>
<td></td>
<td>authenticated</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRTP master key identifier (MKI, optional)</td>
</tr>
<tr>
<td></td>
<td>authentication tag (recommended)</td>
</tr>
<tr>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Secure RTCP Packet Format (RFC 3711)

- Encryption uses AES in Counter Mode (AES-CTR) with 128 bit key
  - IV = f(salt_key, SSRC, packet index)
  - encr_key
  - keystore generator
  - AES-CTR
  - RTP/RTCP payload + XOR
  - encrypted payload

- Authentication uses HMAC-SHA-1 with truncated 80 bit MAC
  - auth_key
  - HMAC SHA-1
  - auth tag
  - 80/32 bits
Session Key Derivation

- Key Derivation uses AES in Counter Mode (AES-CTR)

\[
IV = f(master\_salt, \text{label}, \text{packet index}) \div \text{key derivation rate}
\]

```
master_key
128 bits
192 bits
256 bits
```

```
IV
128 bits
```

```
112 bits
```

```
0x00
0x01
0x02
0x03
0x04
0x05
```

```
encr_key
128 bits
```

```
auth_key
160 bits
```

```
salt_key
112 bits
```

```
0x00
0x01
0x02
0x03
0x04
0x05
```

```
SRTCP
```

```
session keys
```

```
SRTP
```

```
session keys
```

```
key derivation
AES-CTR
```

```
SIP Security
```

Securing the Session Management
SIP INVITE Request

```
INVITE sip:bob@zhwin.ch SIP/2.0
Via: SIP/2.0/UDP 160.85.170.139:5060;branch=z9hG4bK4129d28b8904
To: Bob <sip:bob@zhwin.ch>
From: Alice <sip:alice@zhwin.ch>;tag=daa21162
Call-ID: 392c3f2b568e92a8eb37d448886edd1a@160.85.170.139
CSeq: 1 INVITE
Max-Forwards: 70
Contact: <sip:alice@dskt6816.zhwin.ch:5060>
Content-Type: application/sdp
Content-Length: 239
v=0
o=alice 3157331353 3157331353 IN IP4 160.85.170.139
s=DA SIP Security 2003
c=IN IP4 160.85.170.139
t=0 0
k=clear:910bc4defa71eb6190008762fca6ae2f1d959e87cdf3c0c5c5076ad38ee8
m=audio 10000 RTP/AVP 0
a=ptime:20
a=rtpmap:0 PCMU/8000
```

S/MIME based Authentication and Encryption

```
INVITE sip:bob@zhwin.ch SIP/2.0
Via: SIP/2.0/UDP 160.85.170.139:5060;branch=z9hG4bK4129d28b8904
To: Bob <sip:bob@zhwin.ch>
From: Alice <sip:alice@zhwin.ch>;tag=daa21162
Call-ID: 392c3f2b568e92a8eb37d448886edd1a@160.85.170.139
CSeq: 1 INVITE
Max-Forwards: 70
Contact: <sip:alice@dskt6816.zhwin.ch:5060>
Content-Type: multipart/signed;boundary=992d915fef419824;
micalg=sha1;protocol=application/pkcs7-signature
Content-Length: 3088
--992d915fef419824
Content-Type: application/pkcs7-mime;smime-type=envelopeddata; name=smime.p7m
Content-Disposition: attachment; handling=required; filename=smime.p7m
Content-Transfer-Encoding: binary
<envelopedData object encapsulating encrypted SDP attachment not shown>
--992d915fef419824
Content-Type: application/pkcs7-signature;name=smime.p7s
Content-Disposition: attachment; handling=required; filename=smime.p7s
Content-Transfer-Encoding: binary
<signedData object containing signature not shown>
--992d915fef419824--
```
Practical Results and Conclusions

• Practical Results
  • ZHW diploma thesis in 2003 demonstrated feasibility of S/MIME protected session management and SRTP secured media streams.
  • reSIPprocate available from www.resiprocate.org implements a SIPv2 stack and offers basic S/MIME support using OpenSSL.
  • TinyCA available from tinyca.sm-zone.net was used as a graphical interface on top of OpenSSL for X.509 certificate generation.
  • libsrtp library available from srtp.sourceforge.net implements SRTP.

• Conclusions
  • S/MIME encrypted and/or signed attachments in SIP messages are an attractive alternative to the hop-by-hop security offered by TLS and allow the secure transfer of secret SRTP master keys via end-to-end encryption.
  • Similar to S/MIME protected email, the verification of peer certificates on a global scale remains one of the open problems yet to be solved.