Introduction to Multicast Security
Beyond SSL/TLS

Chaos Communication Camp 2011

Frank Rehberger
software architect  +49 173 205 7118
frank.rehberger@sked.net
Sked.net – Middleware and security consulting
Overview

- Use cases
- Multicast Fundamentals
- Comparing security solutions
- Introduction to SRTP
- Group Key Exchange
Multicast Transport Use Cases

Scaling, Fault-tolerant Distributed Apps

- Multi-participant multimedia conferences
- Replicated Databases/Filesystems
- Distributed simulation
- Active badge
- Control and measurement systems
- Event systems
- Data Distribution Services (DDS)
Multicast Transport Use Cases

- One message reaching all

```
Source -> 239.240.0.1:7400
```

```
Message
```

```
1  2  3  N
Receiver
Receiver
Receiver
Receiver
```
Fault-tolerant Realtime Middleware

DDS: Data Distribution Service for RT-Systems
- Open OMG standard since 2004
- Data Centric Publish/Subscribe, >1000 nodes
- High volume of events/data

Field of applications
- Avionics
- Automotive
- Finance-IT
- Public transport
Multicast Fundamentals

- UDP providing multicast functionality
- IPv4 multicast address:
  - Address range 224.0.0.0 to 239.255.255.255, e.g. 239.240.0.1
  - Requires Internet Group Management Protocol (IGMP)
- IPv6 multicast address:
  - Have the prefix ff00::/8, e.g. ff15::1
  - Requires Multicast Listener Discovery Protocol (MLD)
Multicast Fundamentals

- Listeners:
  - Binding to local address and port, eg 0.0.0.0:7400 and join a mcast group 239.240.0.1
  - Setsockopt: IP_ADD_MEMBERSHIP, IP_DROP_MEMBERSHIP

- Senders:
  - Binding to any address, any port
  - Setsockopt: IP_MULTICAST_LOOP, IP_MULTICAST_TTL, IP_MULTICAST_IF

http://www.linuxjunkies.org/html/Multicast-HOWTO.html
Multicast Fundamentals

Problem

- Any node can join or leave a multicast group
- Any node can send multicast messages

Requirements

- Authentication
- Confidentiality
- Integrity
- Non-repudiation
SRTP: Secure Real Time Protocol
- Protects RTP over UDP
- Support for Multicast

TLS: Transport Layer Security
- Popular for Email, web, etc.
- Usually over TCP, but also UDP
- No support for multicast

IPsec: network security protocol
- Multicasting not supported
- Layer 3, host2host
Why does SSL/TLS not match

- 1:1 / client-server oriented
- In-Band/blocking Key-Exchange/Handshake
  - On re-keying critical interruption of data flow
- Connection-/Stream-oriented
  - Not robust against packet loss
- Variation: Datagram-TLS (RFC4347)
  - Robust against packet loss (except handshake phase)
  - Handshake phase requires reliable transport
  - still 1:1
SSL/TLS Handshake

TCP three-way handshake

Client Hello

<Client Random, Proposed algorithms>

Server Hello

<Server random, Selected algorithm>

CA Certificate

Server done

Client Key Exchange

Enc (Pub(s),<Pre-Master secret>)

Both sides perform a known calculation to derive the Master Key
Requirements for Alternatives

- 1:N communication pattern (e.g. UDP-Multicast)
- Skaling
- Non-blocking Key-exchange
  - 1) Out-of-band or 2) using separate signaling channel
- Robust against packet loss
- Auth. & Confidentiality & Integrity
- Optional: Robust against replay

Possible Solutions: SRTP and IPsec
Alternative: Ipsec (Network Layer)

- Network layer/host2host
- Out-Of-band key-exchange (IKE)
- Cisco: GRE (Generic Route Encapsulation) tunnels
  - Tunneling Mcast messages thru Ipsec tunnel
  - Each tunnel is encrypted separately
- Requires full mesh
- Not scaling
- Requires administrative rights to configure IPsec
S-RTP as Alternative

- Underlying RTP/RTCP: Unicast & Multicast
  Packet oriented multimedia protocol

- Protection fine grained: application/user/port

- Session oriented

- Out-Of-band key-exchange
  - Application/use-case specific

- Protecting Auth & Confidentiality & Integrity

- Optional: Protects against reply attacks

- Can be configured on application layer
  - no admin rights
RTP – Use Scenarios

- Primarily for multi-participant multimedia conferences

But also suitable for

- Storage of continuous data
- Interactive distributed simulation
- Active badge
- Control and measurement
RTP – Real Time Protocol

- RFC-3550 in 2003
- Suitable for appl. transmitting real time data end2end
  - audio, video or simulation data,
  - over multicast or unicast network services
- Over unreliable transport (UDP)
  - Packet loss
  - Re-ordering
- Augmented by a control protocol (RTCP) to allow
  - Monitoring of the data delivery in a manner scalable to large multicast networks,
  - To provide minimal control and identification functionality.
- RTP and RTCP designed to be independent of the underlying transport and network layers.
RTP/RTCP

- RTP: to carry data that has real-time properties.

- RTP control protocol (RTCP): to monitor the quality of service and to convey information about the participants in an ongoing session.
RTP Communication

- RTP provides
  - payload type identification
  - sequence numbering
  - timestamping
  - delivery monitoring

- RTP supports multicast distribution if provided by the underlying network
RTP Message

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>V=2</td>
<td>P</td>
<td>X</td>
<td>CC</td>
</tr>
<tr>
<td>timestamp</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>synchronization source (SSRC) identifier</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>contributing source (CSRC) identifiers</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>payload ...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RTP padding</td>
<td>RTP pad count</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Secure RTP Communication

- RFC-3711 in 2004
- Symmetric ciphers (AES, etc.)
- Using stream-ciphers
  - calculation parallelizable
  - pre-calculation possible
  - AES counter mode (CTR)
  - AES Counter with CBC-MAC (CCM)
- Robust against packet loss
- No key in-band exchange as with TLS
Secure RTP Message

| V=2 | P | X | CC | M | PT | | sequence number |
|-----|---|---|----|---|----| | |
| +---------------------------------------------------------------+ |
| +---------------------------------------------------------------+ |
| | timestamp |
| +---------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
| | synchronization source (SSRC) identifier |
| +-------------------------------------------------------------------------------------------------+ |
| | contributing source (CSRC) identifiers |
| | .... |
| +-----------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
| | RTP extension (OPTIONAL) |
| +-----------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
| | payload ... |
| +---------------------------------------------------------------+ |
| | |
| | |
| | |
| | RTP padding | RTP pad count |
| +-----------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
| ~ | SRTP MKI (OPTIONAL) |
| +-----------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
| : authentication tag (RECOMMENDED) : |
| +-----------------------------------------------------------------+ |
| +-----------------------------------------------------------------+ |
Secure RTP/RTCP Overview

- **SRTP Provides Protection**
  - Replay protection (windowing)
  - Confidentiality (AES)
  - Authentication/Integrity (HMAC-SHA1)

- **Cryptographic Context formed by sender + receiver**

- **Key Derivation: Session keys from single Master Key**
  - Preventing attacker from collecting large amounts of cipher text with one single session key
  - Derivation rate in relation to number of sent packets

- **Short commings:**
  - Sequence counter only 16bit: Synced roll-over counter required – out-of-band or optional message attribute
  - Little documentation regarding Key Exchange for multicast sessions

- **Implementation: libSRTP (C language)**
  - BSD-based license
Group Key-Exchange

- SRTP relies on an external key management protocol to set up
  - Initial master key
  - Initial sequence number
  - Current roll-over-counter (ROC)

- Key exchange should scale and be robust

- No standard Key-Exchanges:
  - ZRTP (in-band, 1:1, DH)
  - EKT: Encrypted Key Transport for Secure RTP (in-band, decentralized, key-encr.cipher AES)
  - SIP/SDES (off-band, overhead)
  - MIKEY (various modes)
    - RSA-R: reverse RSA
Mcast Challenges

- Not all devices or interfaces support Multicast
  - Most Android devices do not support Mcast (2010)
  - Some network-interfaces configured without Mcast
  - Works within LAN, but limited in WAN cross routers
- Slightly differences per OS (Windows vs Linux)
- Routing of Mcast traffic
  - Default network-interface for Mcast messages
  - Multi-homed hosts require out-bound Mcast socket per NIC with explicit routing settings
Group Key-Exchange: Mikey

MIKEY: Multimedia Internet Keying Protocol [RFC3830]

Modes:
- Pre-shared key mode (PSK),
- Public-key (RSA) mode,
- Diffie-Hellman exchange (DHE) mode.

Primary motivation of protocol design
- low-latency requirements of real-time communication,
- exchanges finish in one-half to 1 roundtrip;
Group Key-Exchange: Mikey

MIKEY Scenarios (RFC3830)

- **Peer-to-Peer: simple one-to-many**

- **Many-to-Many (distributed)**

- **Many-to-Many (centralized)**
Questions?

Contact: Frank.Rehberger@sked.net
Fon: +49-173-205-7118

Further reading
http://datatracker.ietf.org/wg/msec/charter/