WebRTC

The technology

The WebRTC technology uses three main specifications in networking:
ICE and STUN
DTLS
SRTP

To establish a WebRTC connection, ICE is used. The web client sends STUN-requests to the WCS server, the WCS server responds to these requests and hence confirms it is ready to establish connection.

On the next stage, parties exchanges SSL certificates via DTLS and establish an encrypted channel between the web client and the WCS server. When the connection is established, SRTP traffic is transmitted.

.

Possible problems

In most cases problems are related to UDP traffic of ICE, STUN, DTLS, SRTP not flowing between parts of the system.

Troubleshooting

Make sure all the traffic that takes part in establishing a WebRTC sessions and sending media data is unhindered and passes freely between call participants. Media ports of the WCS server in the range of [31000-32000] by default must be open to receive the incoming UDP traffic. If the WCS server is behind NAT and has an external IP address, make sure UDP packets sent to this external address are correctly routed to the corresponding ports of the WCS server behind NAT.

ICE and STUN traffic

Here is how ICE transactions exchange looks like before establishing connection. In the dunp these are usual STUN requests and answers. After the connection is established, ICE keeps working to monitor the established connection. If monitoring indicates ICE does not pass, the call is interrupted.

.

DTLS traffic

DTLS starts working directly after ICE has established connection. Exchange of SSL certificates is several Handshake messages resulting in an established secure connection to transfer media data.

.

SRTP traffic

Recognizing SRTP packets

When the data are sent via SRTP, Wireshark cannot recognize SRTP packets. Finding this packets is easy enough. By default, WCS uses the following port range to transmit media traffic [31000-32000], including WebRTC. In the dump we can see two unrecognized UDP packets, one of which is sent through the port 31030, and the second one is received to that port. For such packets, you should explicitly specify the protocol used.

.

Decoding SRTP packets

Wireshark can decode the UDP packets it found if we explicitly specify the protocol. In the packet properties select 'Decode As..', then select the RTP protocol for all packets that run between the browser (port 31030) and the WCS server (port 58732). These ports are reserved dynamically, so in your case the values might be different.

.

Decoded SRTP traffic

As a result of decoding the protocol, Wireshark will display the decoded SRTP traffic:

.

SRTP packet header

SRTP traffic is encrypted. This means if you try to play it, you will hear noises instead of normal speech. But only the content traffic is encrypted. The main RTP headers remain non-encrypted and they can be seen in the RTP packet. This is handy to analyze SRTP traffic parameters. The below example shows an SRTP packet with Payload Type, Sequence Number, Timestamp, SSRC.

.

The list of SRTP and RTP streams taking part in a WebRTC session

SRTP and RTP streams can be analyzed using Wireshark. To do this, use the 'Telephony - RTP - Show All Streams' menu.

.

In this case, streams with SSRC 0x3B359CA0 and 0x1EC9BA4B are SRTP streams between the web browser and WCS, because the source and destination address is the IP address of the web client (we know it beforehand). The other two streams, specifically, the first and the third ones from the top, are RTP streams between WCS and the SIP server (we know addresses of the WCS server and the SIP server too).

SRTP stream analysis

As described above, SRTP packet headers are not encrypted, so the SRTP stream is available for analysis of quality, losses, jitter, latency just like a conventional RTP stream:

.

Attachments:

- web-phone-call-server-webrtc-srtp-stream-analysing-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-srtp-streams-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-srtp-encrypted-packet-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-srtp-decoded-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-srtp-decode-as-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-srtp-decode-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-dtls-wireshark.jpg (image/jpeg)
- web-phone-call-server-webrtc-ice-wireshark.jpg (image/jpeg)
- WebRTC-ICE-DTLS-SRTP.jpg (image/jpeg)