



Advanced Internet Technologies

Google Meet

Project made by Lucas Jakin, Francesca Craievich & Lorenzo Valentino

University of Trieste

Department of Engineering and Architecture



What is Google Meet?

Google Meet is a video communication platform developed by Google.

Start a call

- Video and audio meetings up to 500 people
- Screen sharing possibility
- Record meetings in Google Drive

Create a group link

- Meetings are easy to schedule/join
- Share the link via email, chat, or other platforms
- Optional integration with Google Calendar



Core Functionalities

Purpose

- Secure and user-friendly
- Desktop, tablet, laptop and mobile devices
- Online classes

Key Highlights

- Available on web browsers or mobile app
- Messages sent even during meetings
- Ideal for presenting slides, documents during meetings

Testbed

Tools used

- **Wireshark:** for sniffing network packets and analyzing traffic
- **Network Tab:** to observe HTTP requests and responses
- **WebRTC-internals:** for in-depth analysis of WebRTC-related data

Testing Environment

- Device: VM with Debian
- Browser: Firefox

Wireshark

Initial phase

TLS authenticates the user;
DTLS secures the WebRTC
session

Intermediate phase

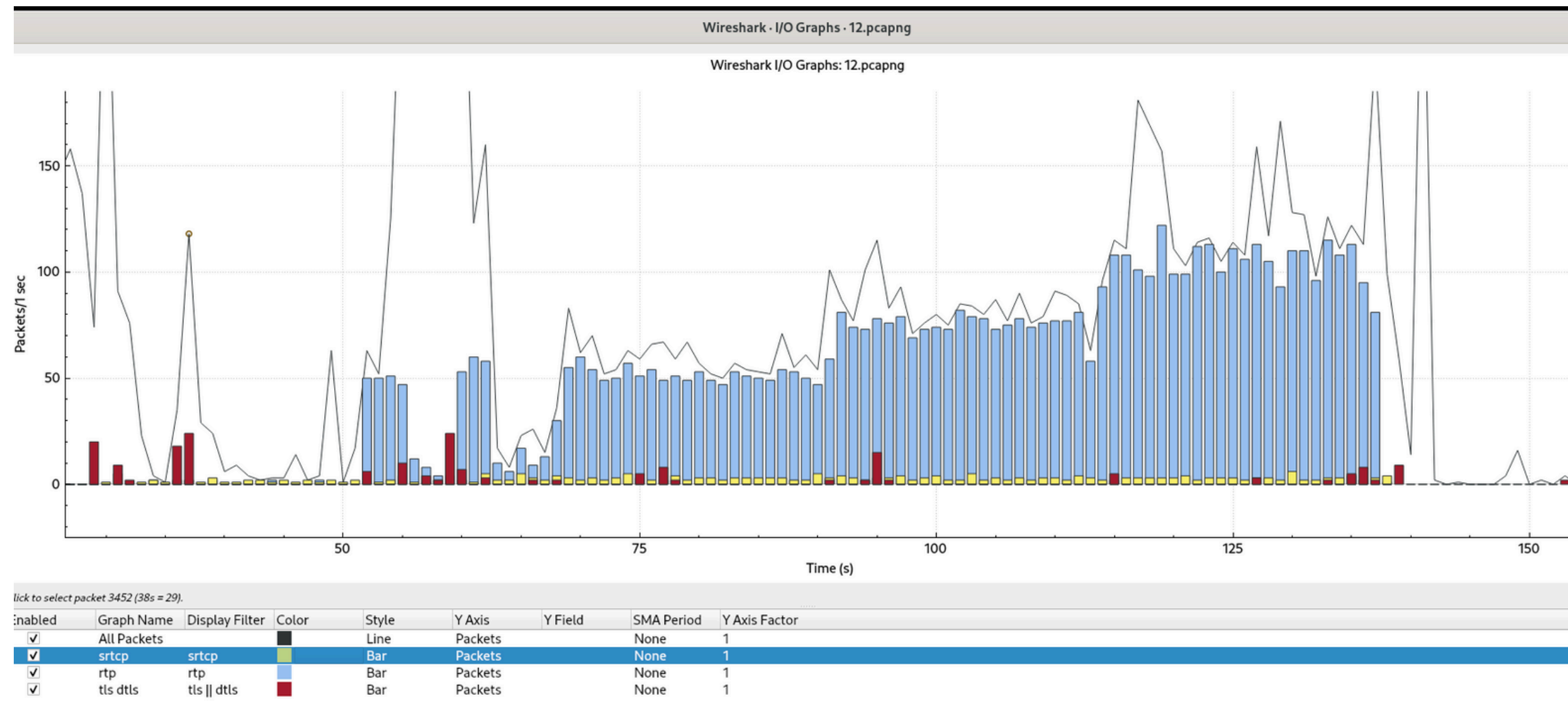
SRTCP begins controlling RTP,
monitoring quality and
synchronizing streams

Active phase

RTP dominates, transmitting
audio/video

Conclusion

SRTCP and RTP finish together



Wireshark

Protocol	Percent Packets	Packets	Percent Bytes	Bytes	Bits/s	End Packets	End Bytes	End Bits/s	PDUs
▼ Frame	100.0	13375	100.0	11557084	570 k	0	0	0	13375
▼ Ethernet	100.0	13375	1.6	188028	9,274	0	0	0	13375
▼ Internet Protocol Version 6	0.0	3	0.0	120	5	0	0	0	3
▼ User Datagram Protocol	0.0	2	0.0	16	0	0	0	0	2
Multicast Domain Name System	0.0	2	0.0	84	4	2	84	4	2
Internet Control Message Protocol v6	0.0	1	0.0	8	0	1	8	0	1
▼ Internet Protocol Version 4	100.0	13370	2.3	267400	13 k	0	0	0	13370
▼ User Datagram Protocol	96.6	12914	0.9	103312	5,096	0	0	0	12914
Session Traversal Utilities for NAT	0.4	47	0.0	4264	210	47	4264	210	47
▼ Secure Real-time Transport Control Protocol	2.1	279	0.1	8672	427	253	7900	389	285
Malformed Packet	0.2	26	0.0	0	0	26	0	0	26
Real-Time Transport Protocol	43.8	5852	41.9	4847691	239 k	5809	4843795	238 k	5852
QUIC IETF	49.0	6554	48.5	5600616	276 k	6554	5481570	270 k	6719
Multicast Domain Name System	0.0	2	0.0	84	4	2	84	4	2
Domain Name System	0.9	122	0.1	7635	376	122	7635	376	122
Datagram Transport Layer Security	0.6	77	0.1	12453	614	77	12453	614	77
Data	0.2	24	0.0	4966	244	24	4966	244	24
▼ Transmission Control Protocol	3.4	456	4.6	532490	26 k	261	14004	690	456
Transport Layer Security	1.4	193	4.8	556331	27 k	193	513666	25 k	224
▼ Hypertext Transfer Protocol	0.0	2	0.0	1128	55	0	0	0	2
Online Certificate Status Protocol	0.0	2	0.0	554	27	2	554	27	2
Address Resolution Protocol	0.0	2	0.0	74	3	2	74	3	2

No display filter.

Help Copy Close

Network Tab

Filter: ☐ Invert | More filters ▼ | All | Fetch/XHR | Doc | CSS | JS | Font | Img | Media | Manifest | WS | Wasm | Other

Timeline: 5000 ms, 10000 ms, 15000 ms, 20000 ms, 25000 ms, 30000 ms, 35000 ms, 40000 ms, 45000 ms, 50000 ms, 55000 ms, 60000 ms

Name

- landing
- rs=AA2YrTvDtorsWuiBHYzP5-IS7pwgoAa95g
- manifest.json
- browserinfo?f.sid=-5714420442263182409&bl=boq_meet...=1&soc-pl...
- browserinfo?f.sid=-5714420442263182409&bl=boq_meet...=1&soc-pl...
- ResolveMeetingSpace
- aqx-vqxb-nry?ijlm=1736333890580&hs=185**
- aqx-vqxb-nry?ijlm=1736333890580&hs=185
- m=_b,_tp?wli=MeetingsUi.LYTc9oXDotA.loadAudioAnaly...er.O;Meetin...
- m=_b,_tp?wli=MeetingsUi.LYTc9oXDotA.loadAudioAnaly...er.O;Meetin...
- log?format=json&hasfast=true&authuser=0
- log?format=json&hasfast=true&authuser=0
- log?format=json&hasfast=true&authuser=0
- lazy.min.js
- lazy.min.js
- log?format=json&hasfast=true&authuser=0
- log?format=json&hasfast=true&authuser=0
- m=yDXup,pA3VNb?wli=MeetingsUi.LYTc9oXDotA.loadAudi...3BMeet...
- join_call_6a6a67d6bcc7a4e373ed40fdeff3930a.ogg**
- log?format=json&hasfast=true&authuser=0
- log?format=json&hasfast=true&authuser=0
- SyncMeetingSpaceCollections
- m=lz4Mu?wli=MeetingsUi.LYTc9oXDotA.loadAudioAnalyz...3BMeeting...

Details for **aqx-vqxb-nry?ijlm=1736333890580&hs=185**:

General

Request URL: https://meet.google.com/aqx-vqxb-nry?ijlm=1736333890580&hs=1

Request Method: GET

Status Code: 200 OK (from service worker)

Referrer Policy: no-referrer-when-downgrade

Response Headers (33)

Request Headers

Header	Status	Protocol	Type
:authority:	200	h2	preflight
:method:	200	h2	preflight
	200	h2	preflight
	200	h2	script
	200	h2	fetch
	200	h2	fetch
	200	h2	fetch

Details for **join_call_6a6a67d6bcc7a4e373ed40fdeff3930a.ogg**:

General

Request URL: https://www.gstatic.com/meet/sounds/join_call_6a6a67d6bcc7a4e373ed40fdeff3930a.ogg

Request Method: GET

Status Code: 206 Partial Content

Remote Address: [2a00:1450:4002:403::2003]:443

Referrer Policy: origin-when-cross-origin

- **Initial Configurations:** A POST request loads UI libraries and settings.
- **Meeting Code:** A unique code identifies the session.
- **Preflight:** The browser verifies network and security policies.
- **Join Call:** A GET request with 206 Partial Content starts the meeting.

Network Tab

The screenshot displays the Chrome DevTools Network Tab. The top toolbar includes filters for 'All', 'Fetch/XHR', 'Doc', 'CSS', 'JS', 'Font', 'Img', 'Media', 'Manifest', 'WS', 'Wasm', and 'Other'. The network request list on the left shows several requests, with 'UpdateMeetingDevice' and 'WriteConferenceSessionLog' highlighted by black arrows. The 'Headers' panel for the 'UpdateMeetingDevice' request is expanded, showing the following details:

- Request URL: `https://meet.google.com/$rpc/google.rtc.meetings.v1.MeetingDevice`
- Request Method: `POST`
- Status Code: `200 OK`
- Remote Address: `[2a00:1450:4002:411::200e]:443`
- Referrer Policy: `origin-when-cross-origin`

The 'Response Headers' panel for the 'leave_call_bfab46cf473a2e5d4...' request is expanded, showing the following details:

- Request URL: `https://www.gstatic.com/meet/sounds/leave_call_bfab46cf473a2e5d474c1b71ccf843a1.ogg`
- Request Method: `GET`
- Status Code: `206 Partial Content`
- Remote Address: `[2a00:1450:4002:403::2003]:443`
- Referrer Policy: `origin-when-cross-origin`

The 'Request Headers' panel for the 'leave_call_bfab46cf473a2e5d4...' request is expanded, showing the following details:

- :authority: `www.gstatic.com`

UpdateMeetingDevice:
Updates and synchronizes device settings

SyncMeetingSpaceCollections:
Keeps the meeting state synchronized

ResolveMeetingSpace:
Permissions for the meeting when joining the session

WriteConferenceLog: Logs network conditions, user actions, and call quality for diagnostics

WebRTC internals

```
16/12/2024, 16:07:48 ▶ transceiverAdded [0], audio
▼ transceiverAdded [1], video
Caused by: addTransceiver

getTransceivers()[1]:{
  mid:null,
  kind:'video',
  sender:{
    track:null,
    streams:['e14f1ca6-585d-41a2-9577-55d5b4c4d02d'],
    encodings: [
      {active: true, },
    ],
  },
  receiver:{
    track:'5fe6662d-3064-43f4-9169-fa5968fe3f39',
    streams:[],
  },
  direction:'sendonly',
  currentDirection:null,
}

▼ createDataChannel
label: ignored, ordered: true, negotiated: false

▼ createOffer
options: {offerToReceiveVideo: -1, offerToReceiveAudio: -1,
negotiationneeded

16/12/2024, 16:07:48 ▶ createOfferOnSuccess (type: "offer", 4 sections)
16/12/2024, 16:07:48 ▶ createDataChannel
16/12/2024, 16:07:49 ▶ setLocalDescription (munged) (type: "offer", 4 sections)
16/12/2024, 16:07:49 ▶ createDataChannel
16/12/2024, 16:07:49 ▶ createDataChannel
16/12/2024, 16:07:49 setLocalDescriptionOnSuccess
▼ signalingstatechange
have-local-offer

16/12/2024, 16:07:49 ▶ transceiverModified [0], audio
16/12/2024, 16:07:49 ▶ transceiverModified [1], video
```

Media Identifier: null because the transceiver has not yet negotiated a value with the remote peer

transceiver is configured to only send video data and not receive any.

name of data channel

messages will be received in order in which they were sent

data channel has not been negotiated manually, but will be handled automatically by WebRTC

SDP offer with 4 sections:

- Session-level metadata
- Audio section
- Video section
- Data section

Conclusions

Which protocols at L4 and L7 are involved?

L4: TCP, UDP, DTLS, QUIC

L7: SRTCP, RTP, mDNS, TLS, STUN

Which servers are contacted?

The IP ranges [216.58.192.0 - 216.58.223.255] and [142.250.0.0 - 142.251.255.255] belong to **Google LLC** (AS15169), located at 1600 Amphitheatre Parkway, Mountain View, CA 94043, USA

Conclusions

What are the key functions of the different servers?

- **Authentication:** Handled by **clients.l.google.com**(216.58.204.129) for managing user sessions
- **Video & Audio:** Handled by **meeting.googleapis.com**(142.251.209.10, 142.250.180.170, 216.58.204.138, etc.) for video/audio data transmission during the call
- **UI Elements:** Served by **gstatic.com**(216.58.205.35) and **fonts.gstatic.com**(216.58.204.131), providing static assets used in the interface
- **Access:** The core interface and joining mechanisms are provided by **meet.google.com**(216.58.204.142)

Advanced Internet Technologies

**Thank you for
the attention!**

University of Trieste

Department of Engineering and Architecture