



A MAGYARORSZÁGI **DIGITALIZÁCIÓ** SZOLGÁLATÁBAN

WebRTC & NAT and Firewall Traversal Update

GÉANT STUN/TURN Pilot infrastructure

Mihály Mészáros

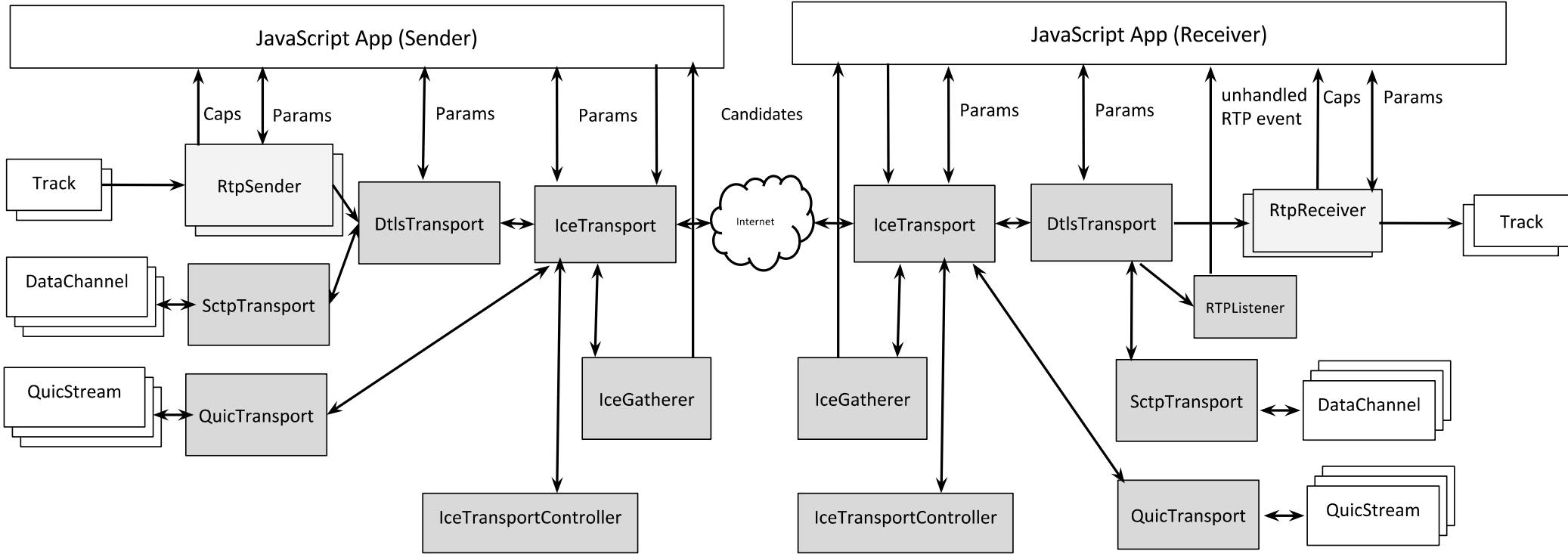
2019.10.09.

Standardization

W3C WebRTC WG update

- Finalize 1.0
 - More and more ORTC
- WebRTC-NV moving target
- WG Charis
 - Jan-Ivar Bruaroey (Mozilla)
 - Bernard Aboba (Microsoft)
 - Harald Alvestrand (Google)
- Latest news:
 - https://webrtc.internaut.com/demos/slides/Leveraging_WebRTC.pdf
 - https://www.w3.org/2011/04/webrtc/wiki/September_19-20_2019

- Higher Level API
 - Easy to learn, but gives less control
 - Many years discussion (ORTC)
 - Bigger responsibility on the Browser side. (heavy lifting!)
 - To serve all use cases is impossible
 - Even coordinating developments is very difficult
 - Worldwide consensus building is hard question!
- More Detailed API
 - Don't use SDP anymore, to avoid it's limitations
 - Less shared media engine code (More flexibility, New use cases!)
 - Bigger responsibility on the user side



Source: <http://draft.ortc.org/>



- Developers are demanding for more control
 - Deeper API requested
 - even deeper than ORTC
 - hand in hand with more responsibility
 - Vendors could differentiate themselves with implementation.
 - We could go back in time again to the closed source incompatibility hell? :-)
 - More control in developers hand, and more complexity!
 - A web developer usually is not a telecommunication expert
 - The first idea was that WebRTC 1.0 should hide complexity from developers, and give only moderate control on the multimedia session establishment.
 - Fast learning curve, avoid frustrated developers

With Great Power ..



Source: <https://flic.kr/p/TEEaFq>

How deep is not too deep?

A. PeerConnection: Mix of direct control and SDP (☹️)



Simple things simple
Less control

B. ORTC: all direct control; no SDP (🐜)



C. Split out encoder/decoder from RtpSender/RtpReceiver (pluggable transport)



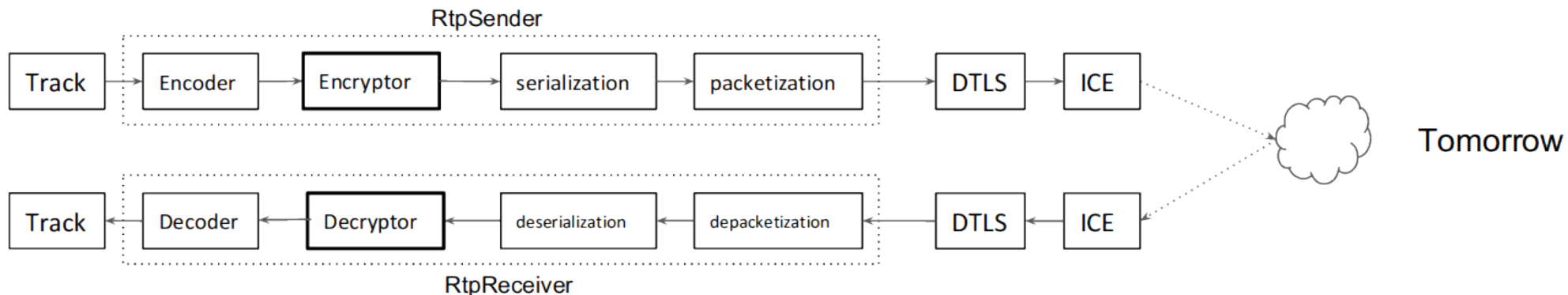
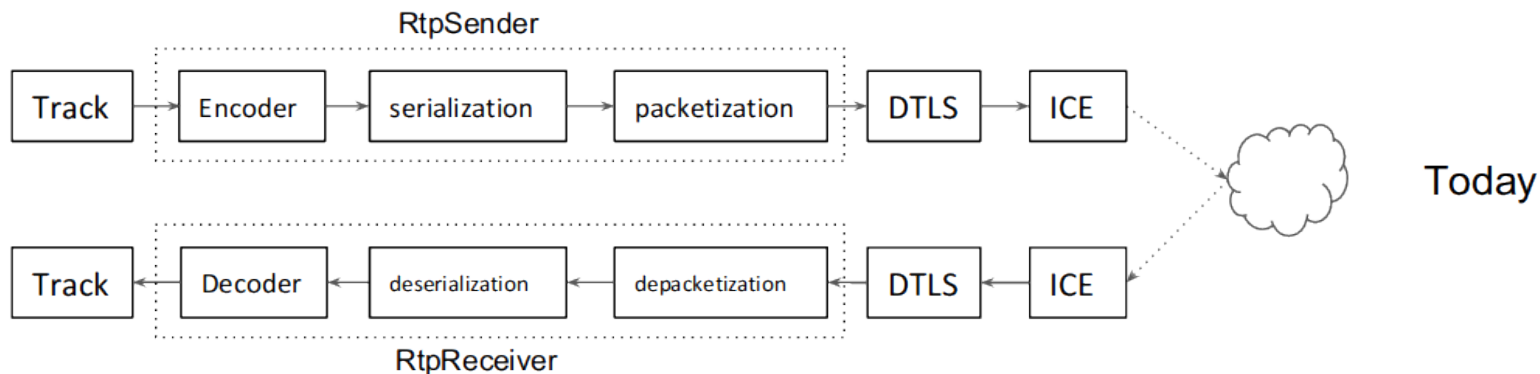
D. Low-level RTP control (bring your own RTP packetization, FEC, RTX, e2e crypto, ...)



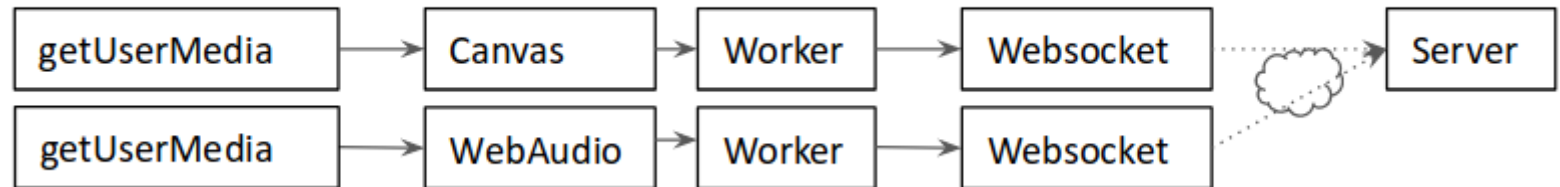
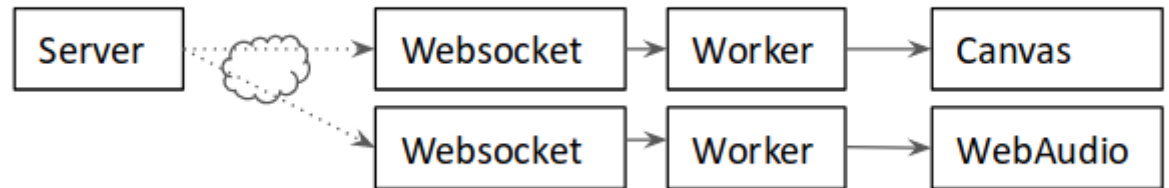
E. Max wasm (bring your own codec, jitter buffer): need raw media access



More detailed API



- WebRTC hacks article:
 - [https://webrtchacks.com/zoom-avoids-using-webrtc/](https://webrtc hacks.com/zoom-avoids-using-webrtc/)
- WASM = Web Assembly
- Zoom
 - video/audio
 - encode/decode



ICE

A. PeerConnection: Not standalone, uses SDP (☹)



B. WebRTC-ICE extension spec: standalone, no SDP (🤖)



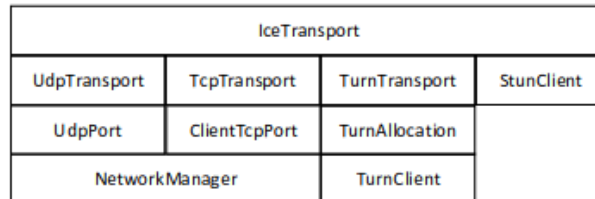
C. ORTC: split gatherer/transport (supports forking)



D. FlexICE



E. SLICE

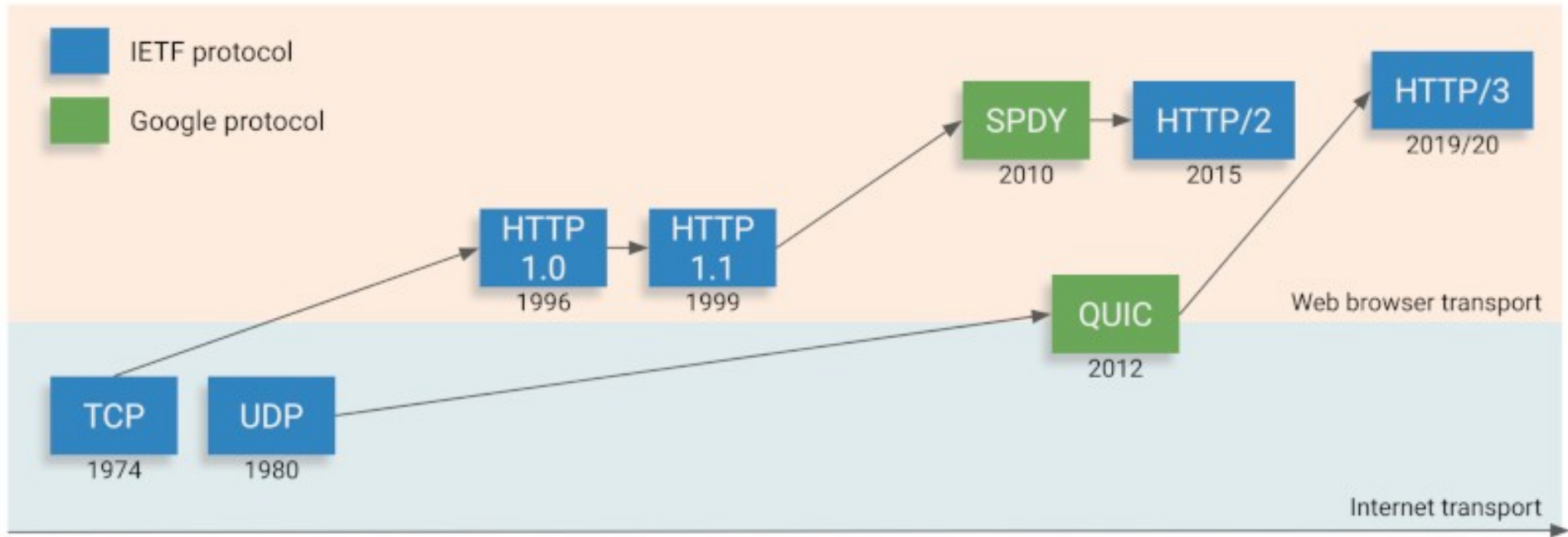


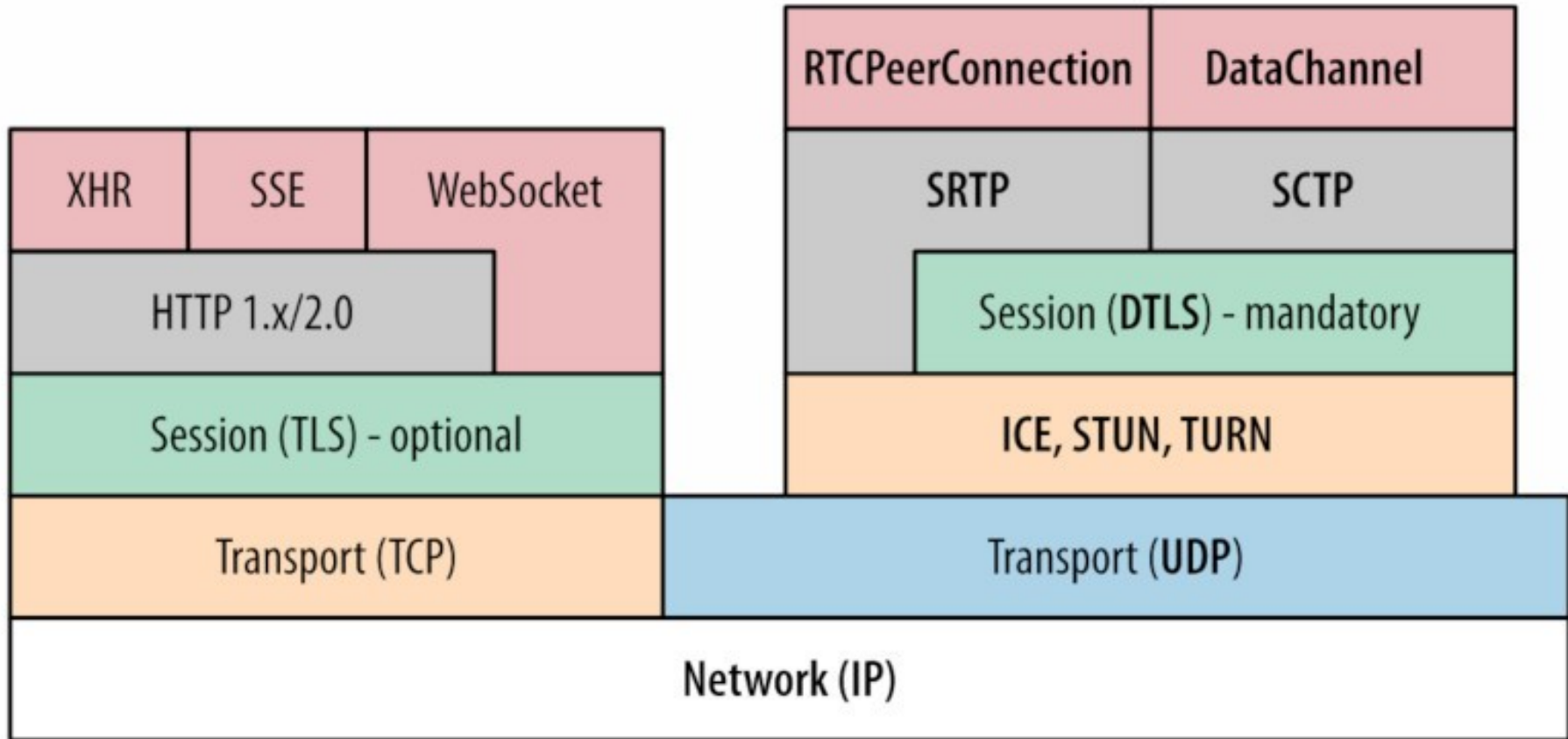
Simple things simple
Less control

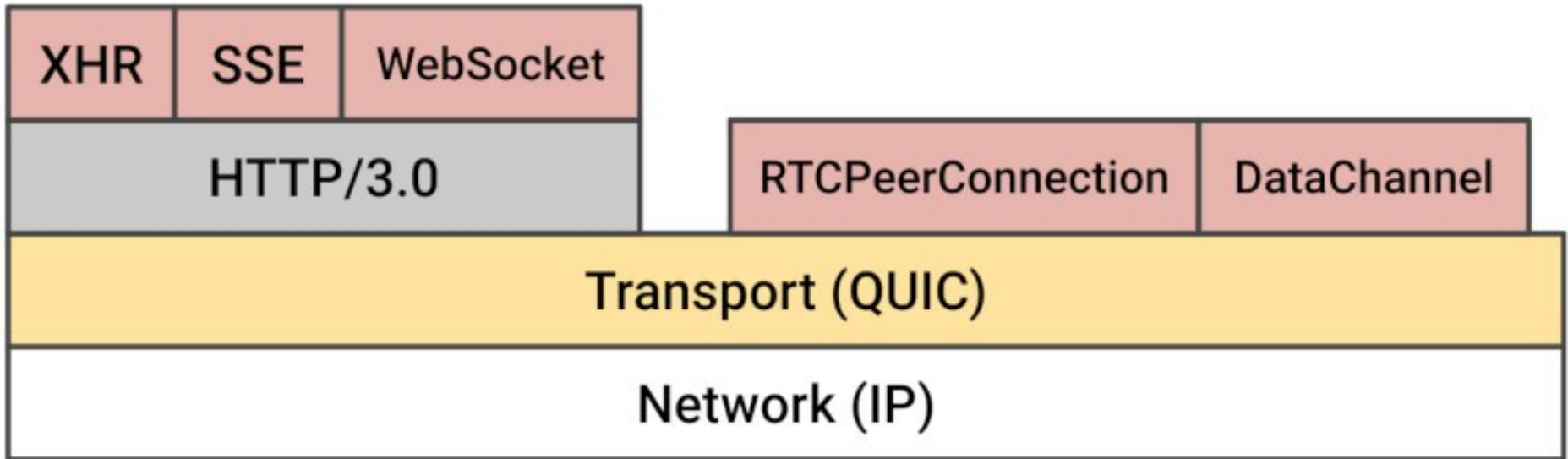
More control
More complex

- WebRTC-ICE
 - allows for constructing an ICE agent separate from a PeerConnection, without SDP, suitable for use with other transports (QuicTransport).
- ORTC
 - also allows parallel forking. But it also requires more objects to be managed by the app (IceGatherer)
- FlexICE
 - also allows for control over ICE checking frequency/pausing, local candidate lifetimes, continual gathering, and timeouts for disconnect. But it requires the app to do more to get the advanced uses (you just get normal behaviour otherwise). New controls can be added incrementally.
- SLICE
 - allows the app to implement a custom ICE stack with the lowest-level primitives possible. But it requires the app use a library, because no one will be implementing their own ICE stack. Apps that don't need advanced things can just use a high-level IceTransport. Those that do are capable of tweaking everything

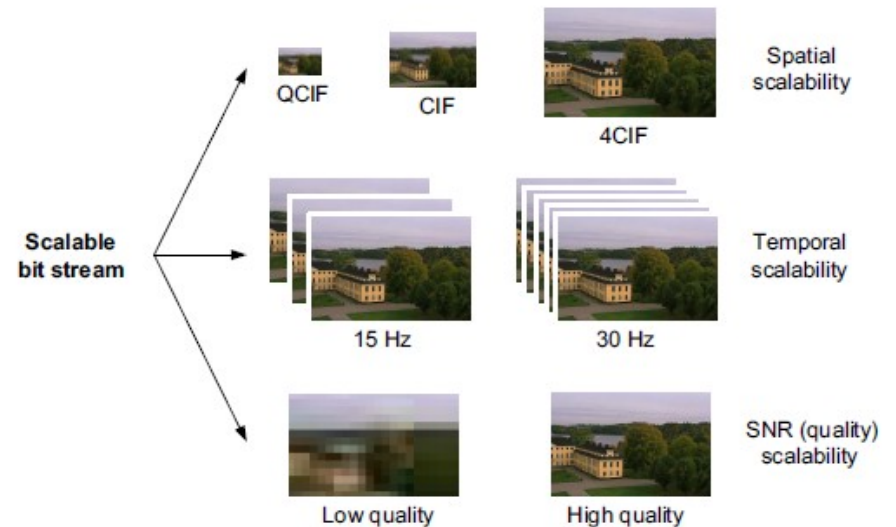
- A QUIC developed as HTTP/3 transport protocol
 - Currently only supports reliable transport
 - Built-in encryption, and congestion control
- DataChannel
 - SCTP => QUIC
- Bring Your Own (BYO) Transport. Replacement of RTP/RTCP?
 - RTP designed in the 1990-s. Maybe it's time to redesign it?
 - QUIC-R transport (Colin Perkins, Jörg Ott)
 - R as realtime extension
 - Replacement of both MPEG-DASH(TCP) and RTP/RTCP(UDP)
 - Converged streaming and videoconferencing





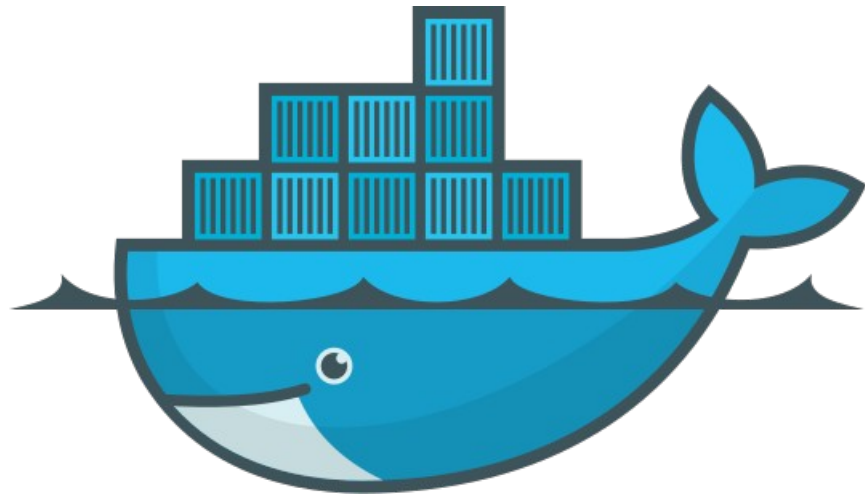


- AV1 is in the pipe
 - Better quality
 - Less bandwidth
 - Optimised for high resolution (4k, 8k)
 - Little bit more CPU
- Scalable Video Coding SVC
 - Spatial
 - Temporal
 - Quality



Docker

Containers are coming!

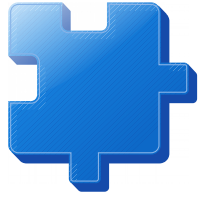


docker

The text "coTURN" in a bold, white, sans-serif font, centered on the page. The background features a blue gradient with a yellow wave at the bottom and faint binary code (0s and 1s) in the upper left.

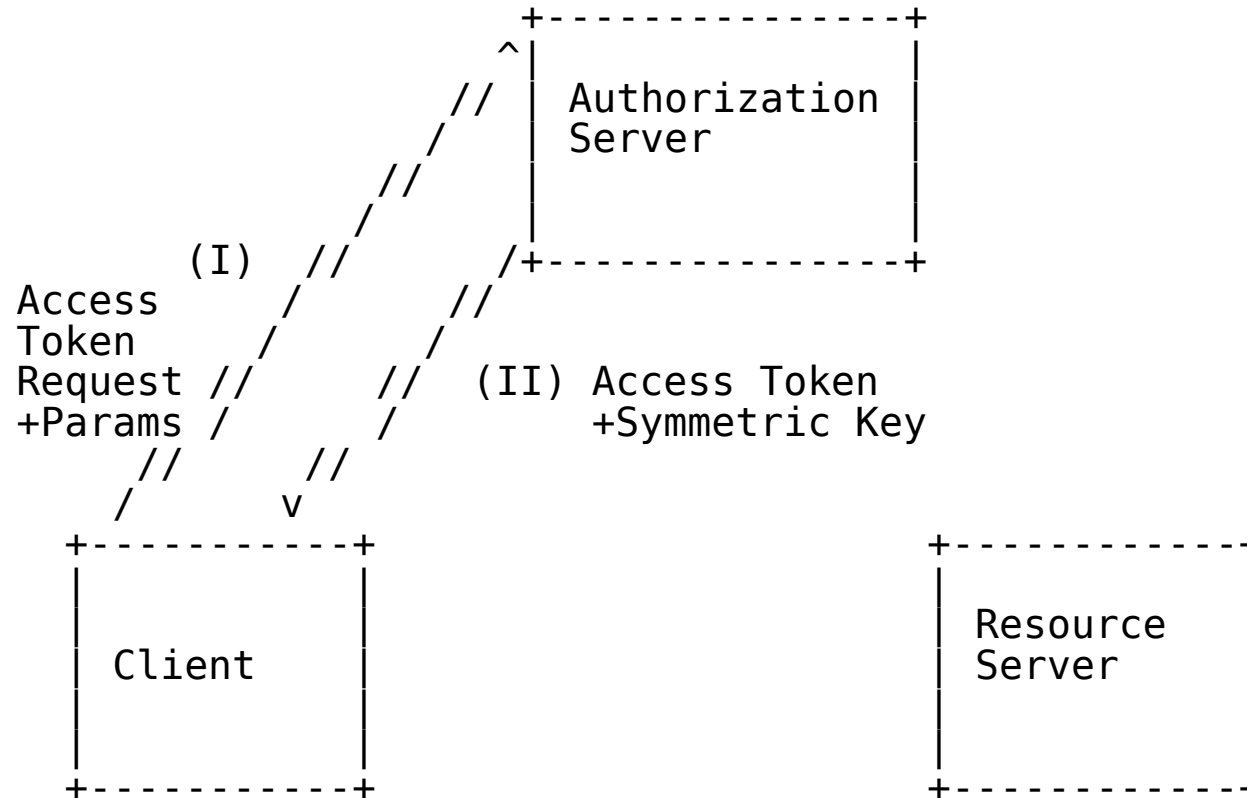
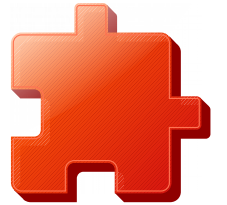
coTURN

- Involved in development
 - Merged many PRs and handled issues
 - Security Fix
 - 2 CVE coordinated with Cisco
 - Still many open issues :(
 - Debian packaging
 - Thanks to Ferenc Wágner support
- Docker
 - Host networking (Docker bug)
 - NAT could also work with some limitations
 - Docker-compose for different backends
 - MongoDB, MySQL, Postgres, Redis, SQLite

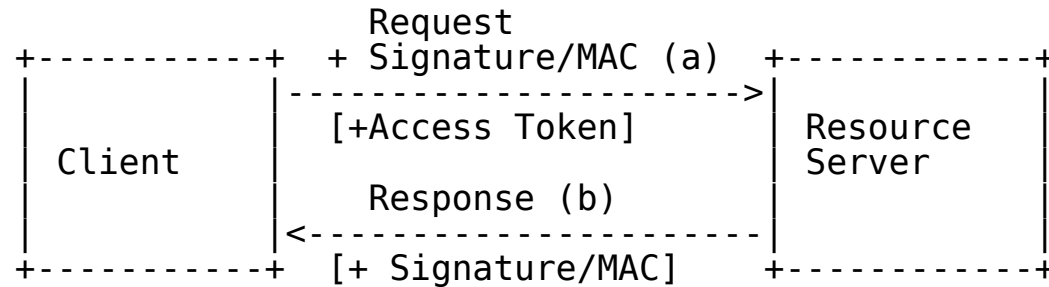


- Scoped credential
 - Time limited
 - Server limited
- Multiple Auth Server
 - co-located TURN
- It is an IETF Standard
 - Compared to other auth methods
 - REST is a discontinued draft
 - Classic LTC is not suitable for Web
 - JS app can't keep credential in secret.





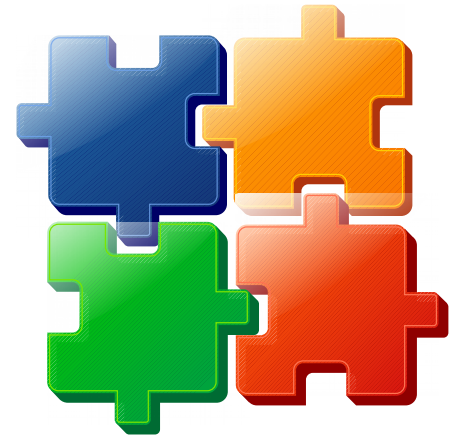
TURN OAuth PoP



^
Symmetric Key
+
Parameters

^
Symmetric Key
+
Parameters

- OAuth Proof of Possession key distribution
- coTURN issue SHA1 length different for
 - Long Term Credential SHA1(MD5())
 - MD5 output 16 byte
 - OAuth SHA1 default input 20byte
- https://bugzilla.mozilla.org/show_bug.cgi?id=1247616
 - I am working implementation in Firefox TURN + OAUTH.
 - I have successfully authenticated against a coTURN server
 - We need in second phase to update token and credential periodically
 - iceRestart





"We believe in rough consensus and running code."

turn_outh.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

stun && udp.port == 41642

No.	Time	Source	Destination	Protocol	Length	Info
357	10.97056...	193.6.222.35	193.224.22.66	STUN	90	Allocate Request UDP lifetime: 3600
359	10.97964...	193.224.22.66	193.6.222.35	STUN	198	Allocate Error Response error-code: 401 (Unauthorized) Unauthorized with nonce realm: mycompany.org
360	10.98086...	193.6.222.35	193.224.22.66	STUN	230	Allocate Request UDP lifetime: 3600 user: 1234 realm: mycompany.org with nonce
361	10.99017...	193.224.22.66	193.6.222.35	STUN	162	Allocate Success Response XOR-RELAYED-ADDRESS: 193.224.22.66:51093 XOR-MAPPED-ADDRESS: 193.6.222.35:41642 lifetime: 3600
387	11.06083...	193.6.222.35	193.224.22.66	STUN	226	CreatePermission Request XOR-PEER-ADDRESS: 193.224.22.66:50282 user: 1234 realm: mycompany.org with nonce
388	11.06111...	193.6.222.35	193.224.22.66	STUN	190	Send Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
390	11.06959...	193.224.22.66	193.6.222.35	STUN	130	CreatePermission Success Response
394	11.07932...	193.224.22.66	193.6.222.35	STUN	186	Data Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
395	11.08225...	193.224.22.66	193.6.222.35	STUN	218	Data Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
396	11.08334...	193.6.222.35	193.224.22.66	STUN	154	Send Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
399	11.10475...	193.224.22.66	193.6.222.35	STUN	306	Data Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
400	11.10637...	193.6.222.35	193.224.22.66	STUN	810	Send Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
404	11.12719...	193.224.22.66	193.6.222.35	STUN	718	Data Indication XOR-PEER-ADDRESS: 193.224.22.66:50282
405	11.13336...	193.6.222.35	193.224.22.66	STUN	166	Send Indication XOR-PEER-ADDRESS: 193.224.22.66:50282

Message Type: 0x0003 (Allocate Request)
Message Length: 164
Message Cookie: 2112a442
Message Transaction ID: df67abecd8c5258159c77f24

Attributes

- REQUESTED-TRANSPORT: UDP
- LIFETIME: 3600
- USERNAME: 1234
- REALM: mycompany.org
- NONCE: 647ef6832c6c5bcf

Unknown

- Attribute Type: Unknown (0x001b)
Attribute Length: 64
Value: 000cee62d347c6eb4c9ebff59b17afb18ed601afbb79697...

MESSAGE-INTEGRITY

FINGERPRINT

```
0010 08 00 45 00 00 d4 8f d8 40 00 40 11 32 f4 c1 06 ..E....@@.2...
0020 de 23 c1 e0 16 42 a2 aa 0d 96 00 c0 c5 39 00 03 .#.B.....9...
0030 00 a4 21 12 a4 42 df 67 ab ec d8 c5 25 81 59 c7 .!.B.g....%.Y.
0040 7f 24 00 19 00 04 11 00 00 00 00 0d 00 04 00 00 $......
0050 0e 10 00 06 00 04 31 32 33 34 00 14 00 0d 6d 79 .....12 34...my
0060 63 6f 6d 70 61 6e 79 2e 6f 72 67 00 00 00 00 15 company.org....
0070 00 10 36 34 37 65 66 36 38 33 32 63 36 63 35 62 .647ef6 832c6c5b
0080 63 66 00 1b 00 40 00 0c ee 62 d3 47 c6 eb 4c 9e cf...@...b.G..L.
0090 bf f5 9b 17 af bd 18 ed 60 1a fb b7 96 97 b7 51 .....Q
00a0 5c a1 5c 0e af 01 cb 1d 1b 33 f7 78 aa 8c cd 5f \.....3.x...
00b0 25 1e 74 a2 7e b4 ae 06 8f f6 7e 53 66 98 99 76 %t~.....Sf.v
00c0 a7 71 73 57 44 3e 00 08 00 14 67 44 72 cf ac 7f .qswD>...gDr...
00d0 82 74 0a 9b 40 96 6e c7 9a a3 b5 38 a9 c2 80 28 .t@.n...8...()
00e0 00 04 84 95 12 63 .....c
```

Attribute Type (stun.attribute), 68 bytes

Packets: 177948 - Displayed: 83697 (47.0%)

Profile: sip



Advancing MultiPartyMeeting

- Open Source
 - <https://github.com/misi/mm>
- <https://hub.docker.com/r/misi/mm/>
- MultipartyMeeting Listens Multiple ports
 - Web Server (express), Signaling(socket.io), Selective Forwarding Unit (MediaSoup)
- Docker large port range NAT issue
 - <https://success.docker.com/article/docker-compose-and-docker-run-hang-when-binding-a-large-port-range>
 - Host networking (workaround)



Enhance Multiparty-Meeting Auth (OIDC)

- Enhance AAI integration
 - replace passport-dataporten to more generic
- Open ID Connect (OIDC)
- openid-client
 - <https://www.npmjs.com/package/openid-client>
 - passport
- Claim
 - Name, Picture
- SAML Satosa SAML \Leftrightarrow OIDC gateway
 - <https://satosa.aai.niif.hu/.well-known/openid-configuration>



MultiPartyMeeting

Easy to use, easy to deploy

- Easy Ansible based Setup
 - Requirements
 - Debian VM
 - a DNS name
 - modify config files
 - The script request Let's Encrypt certificate, and configures, docker, run image customize logo, etc.
- Demo setup in 10 min!
- <https://github.com/misi/mm-ansible>

Try Multiparty Meeting!



E-learning integration

MM in Moodle module

- Loose integration
 - <https://github.com/misi/multipartymeeting-moodle>
- Good feedback from pilot schools..
- Knockplop



 Join to the Meeting

- Follow the Standardization (IETF, W3C)
- Containers are coming (Docker, K8S)
 - RTC in container is not a low hanging fruit
- **GÉANT: STUN/TURN infrastructure pilot**
 - Try it: <https://turn.geant.org>
 - OAuth is in the pipe
- Try MultipartyMeeting, it is free, open, extendable!
 - OIDC, E-learning integration, etc.
- Goal: Serve the community communication needs with advanced reliable, extendable open source tools!

- Standardization
 - how to contribute to the standardisation?
- Containers are coming (Docker, K8S)
 - Are there better and easier to be integrated than containers?
- GÉANT: STUN/TURN infrastructure pilot
 - how to deploy and maintain at GÉANT level?
- OAuth / OIDC is in the pipe
 - GÉANT T&I /EduGAIN confederation supportive enough for OIDC?
- MultipartyMeeting
 - What is needed for a better service?

Many thanks for Your attention!

Questions?