PION

Thank You

Aaron France
Adam Kiss
Aleksandr Razumov
Alex Browne
Alex Harford
AlexWoo(武杰)
Antoine Baché
Ato Araki
Atsushi Watanabe
Brendan Rius
Cecylia Bocovich
Chao Yuan
Chris Hiszpanski
Christopher Fry
CloudWebRTC
Daniele Sluijters
Denis
Gareth Hayes
Guilherme
Hongchao Ma
Hugo Arregui
Igor German
Ingmar Wittkau
Jadon Bennett
Jake B
Jason Maldonis
John Bradley
Justin Okamoto
Konstantin Itskov
Lander Noterman
Lukas Herman
Lukas Rezek
Luke Curley
Marouane
Max Hawkins
Michael MacDonald
Michiel De Backker
Mike Coleman
Mészáros Mihály
Oleg Kovalov
Rafael Viscarra
Raphael Derosso Pereira
Raphael Randschau
Robert Eperjesi
RunningMan
Ryan Gordon
Sean DuBois
Sebastian Waisbrot
Slugalisk
Stefan Tatschner
Steve
Suhas Gaddam
Tobias Fridén
Vicken Simonian
Will Watson
Woodrow Douglass
Yutaka Takeda
Zizheng Tai
adwpc
chenkaiC4
earle
feixiao
frank
igolaizola
imalic3
jinleileiking
jose nazario
mchlrhw
mxmCherry
notedit
rob-deutsch
ronan
simonacca-fotokite
songjiayang
spaceCh1mp
sylba2050
wattanakorn495
xsbchen
zigazeljko
Clayton McCray
Hendrik Hofstadt
Luke S
Bao Nguyen
Konstantin Chugalinskiy

WHAT IS WEBRTC?

1

Browser APIs

E2E Secure Connection between Peers

Multiple Audio/Video Tracks

Binary Data

- Can be lossy
- Can be unordered
- Multiple distinct DataChannels

2

What we need to solve

Connectivity
- Multiple Possible Routes
  - Not in the same network!
  - UDP?

Security
- E2E Secure

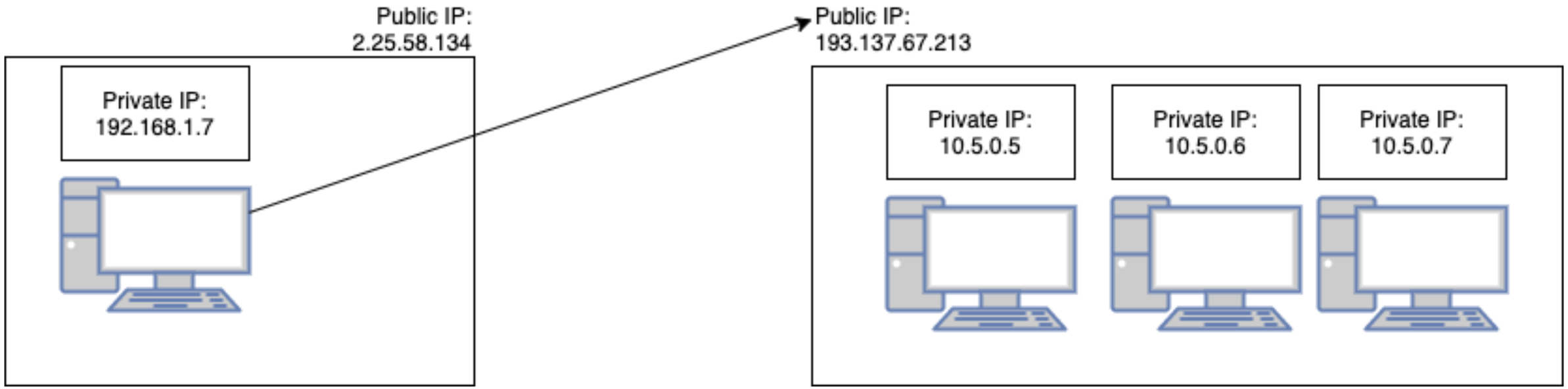
Latency
- 400 milliseconds max
- Unreliable Transport (UDP)

Ease of Use

WHAT DOES IT SOLVE

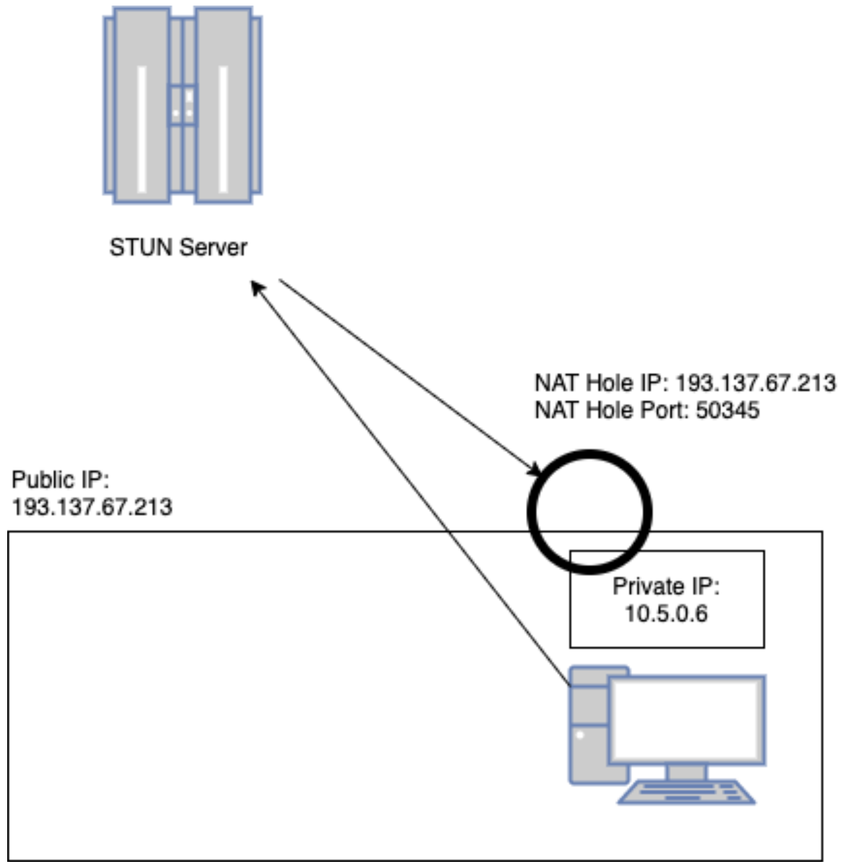
3

Problem: Connect two users with no Public IP

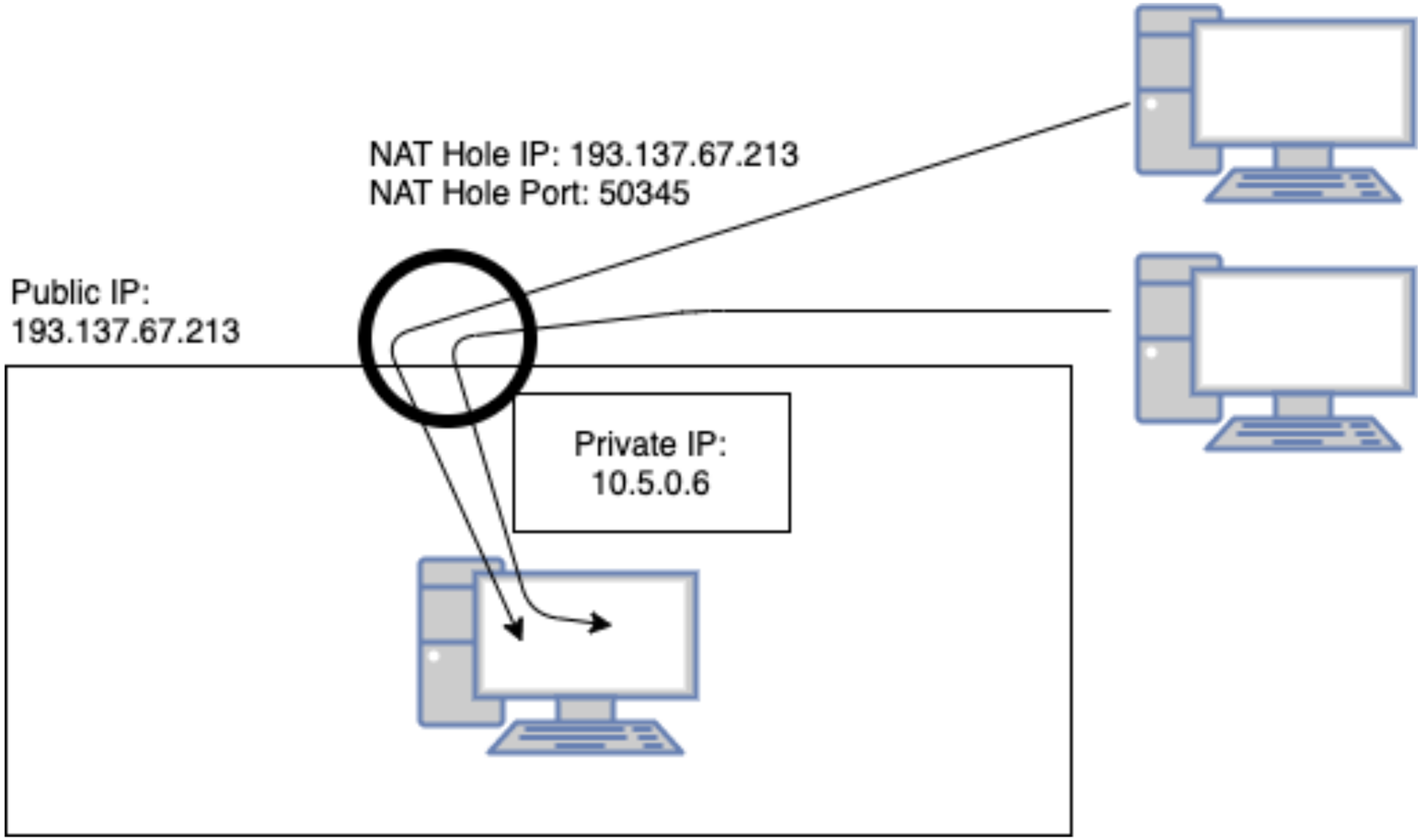


4

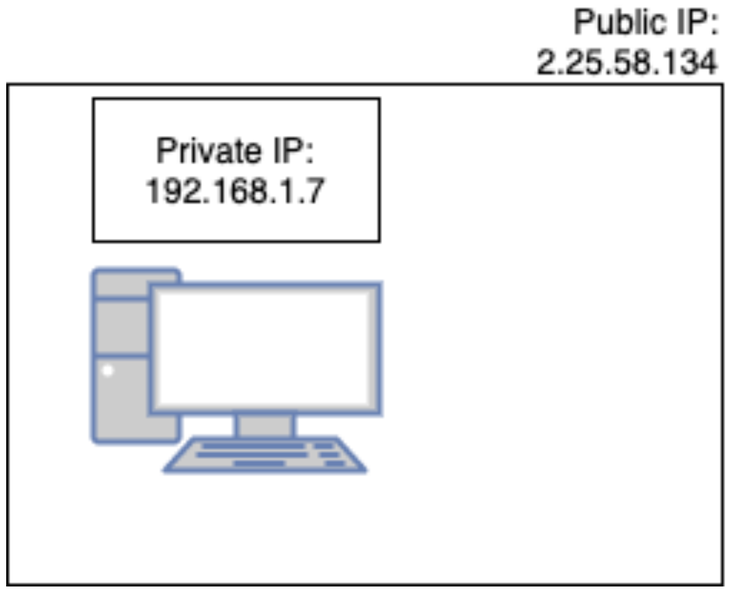
Solution: NAT Traversal

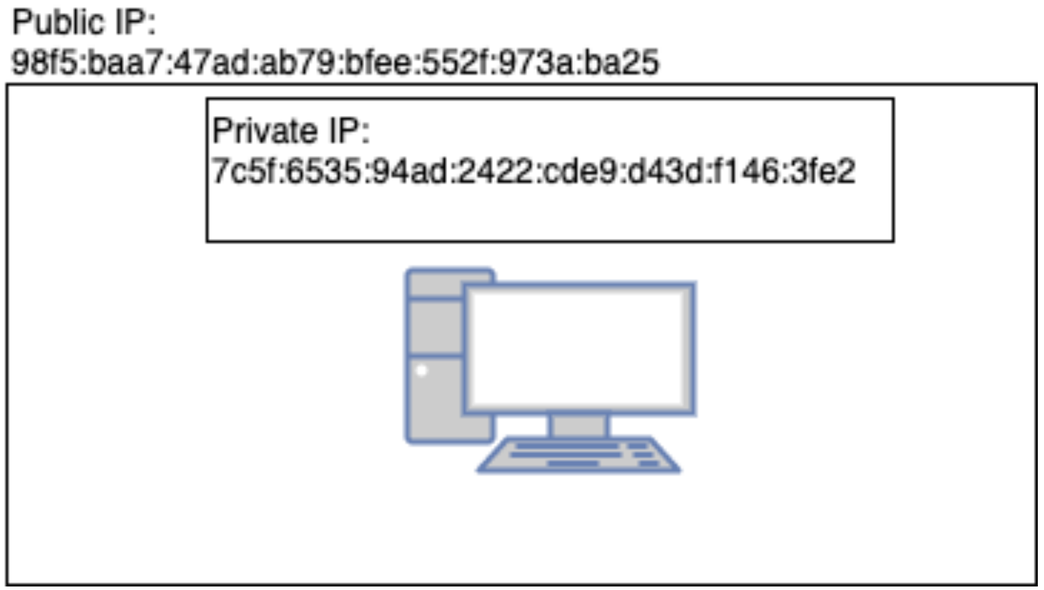


Solution: NAT Traversal

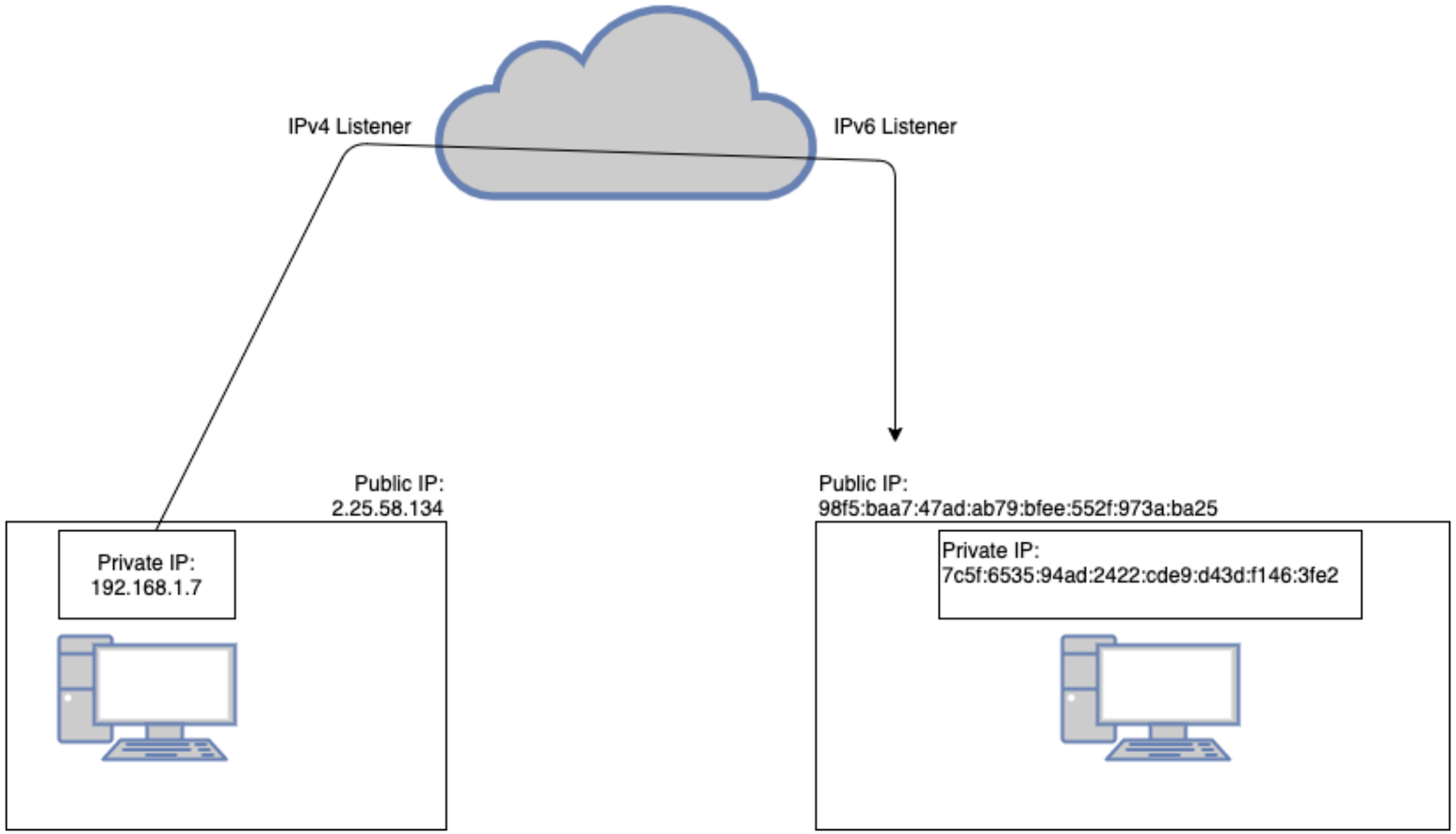


Problem: One is IPv4, One is IPv6

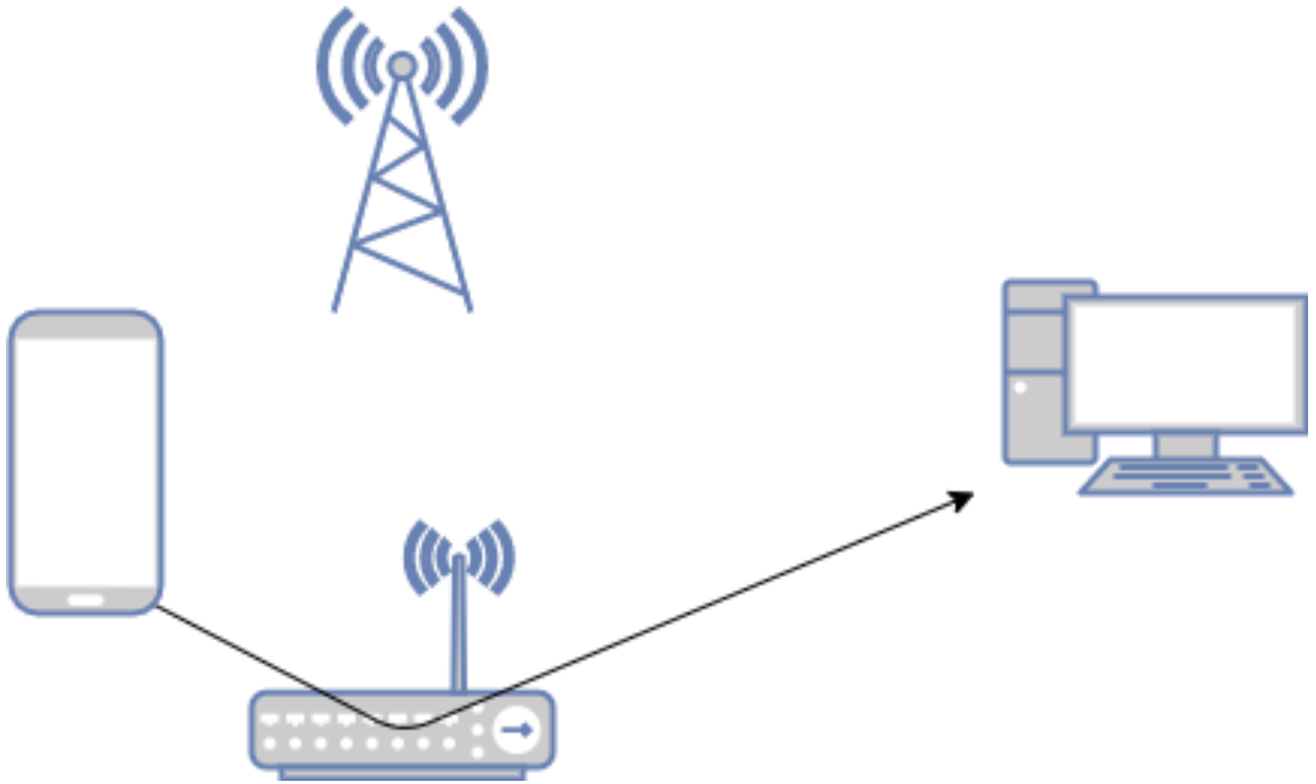




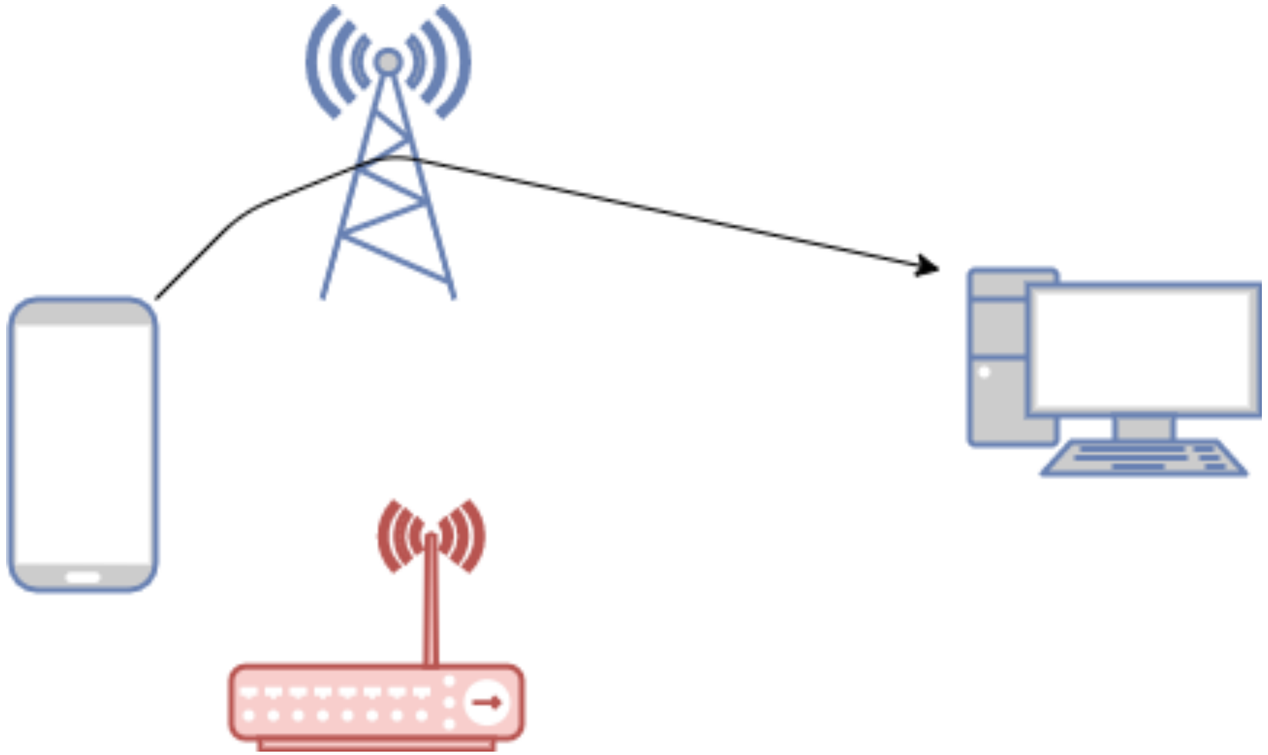
Solution: TURN



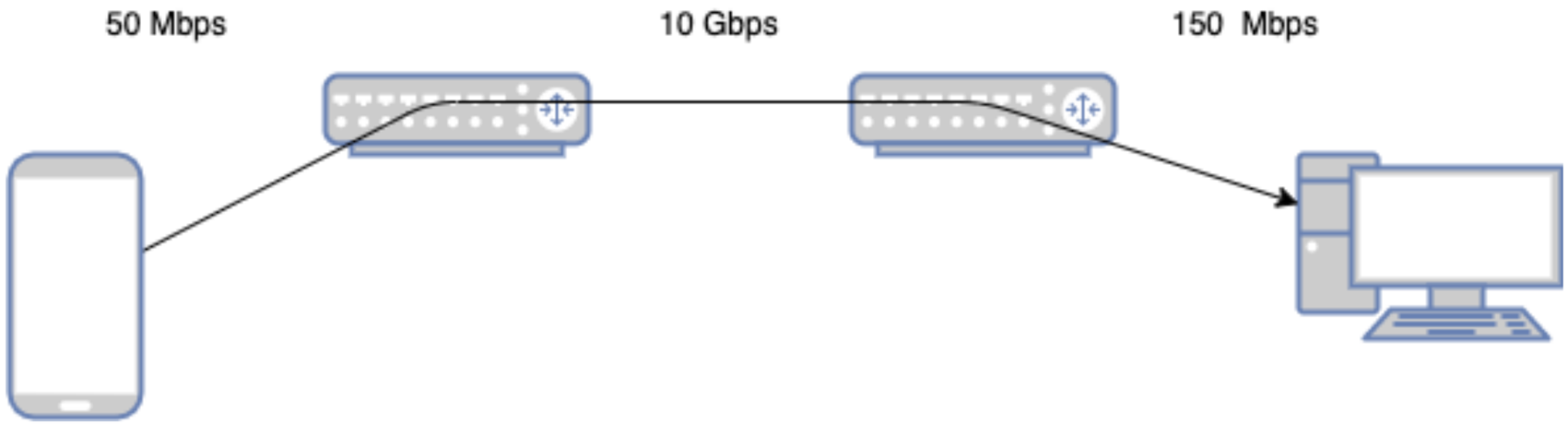
Problem: Wi-Fi on Fire



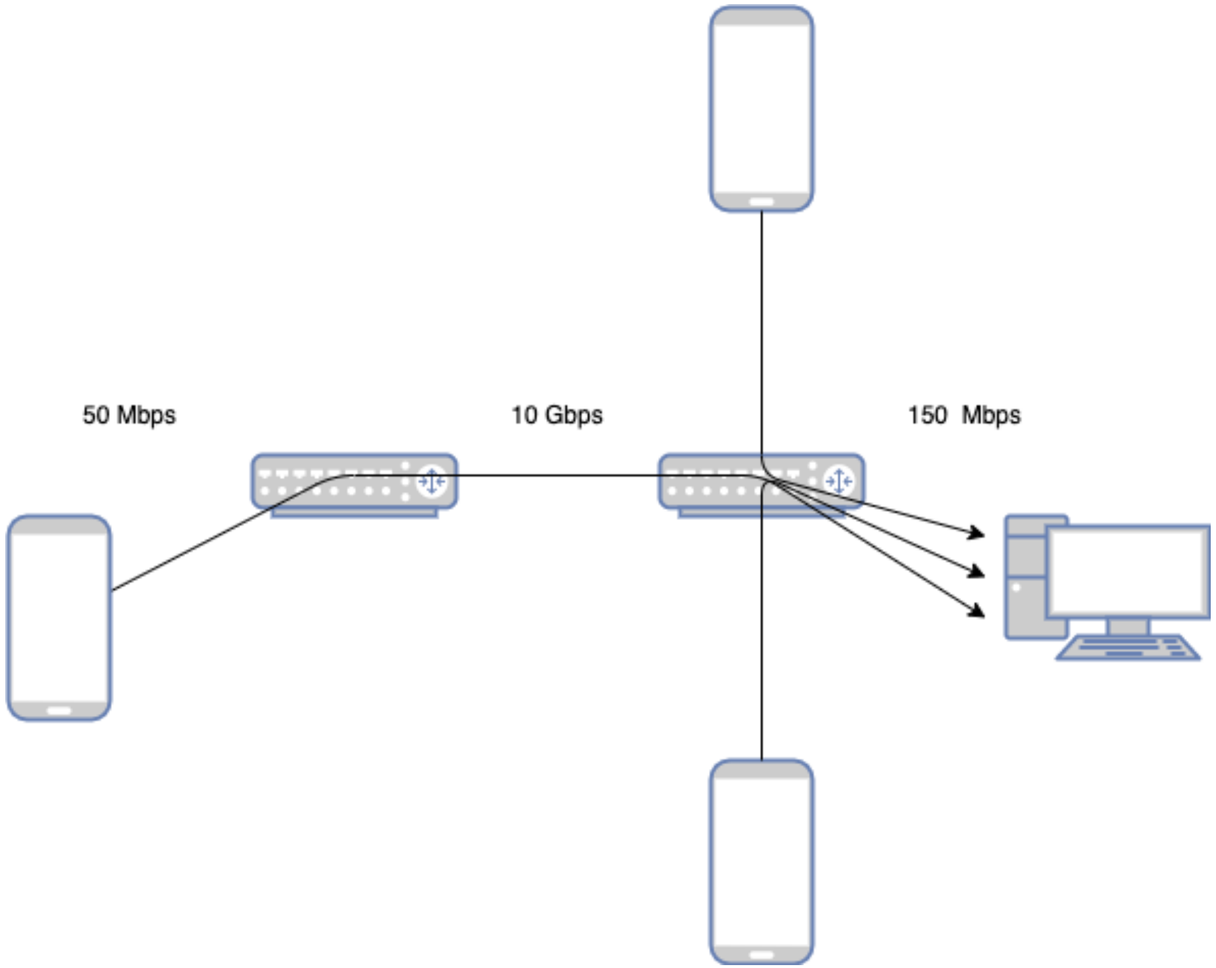
Solution: ICE Renomination



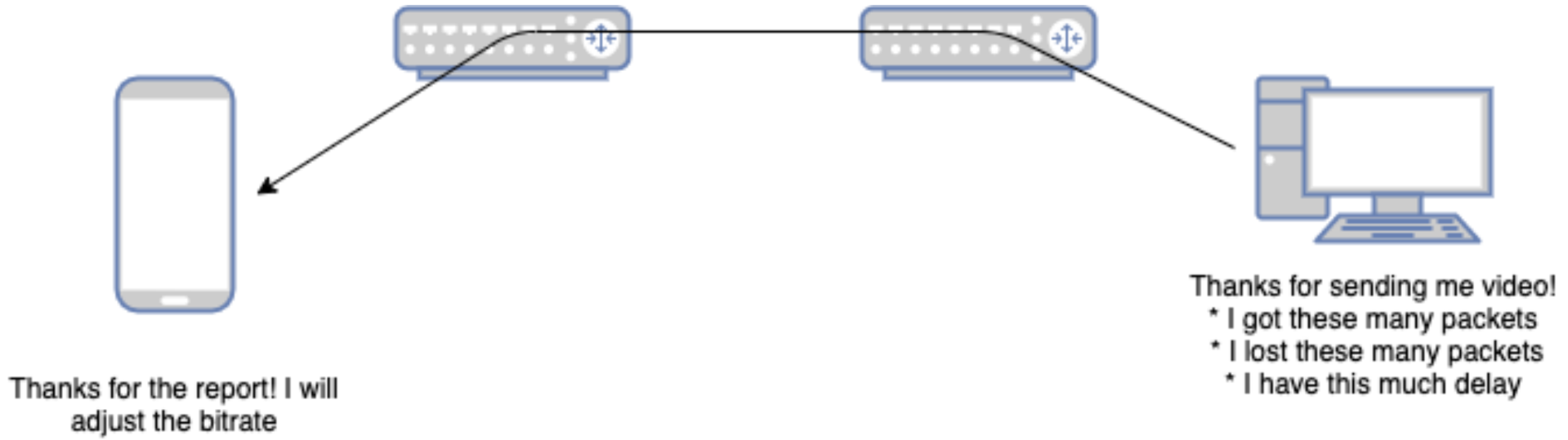
Problem: What bitrate do I upload?



Problem: What bitrate do I upload?



Solution: Congestion Control

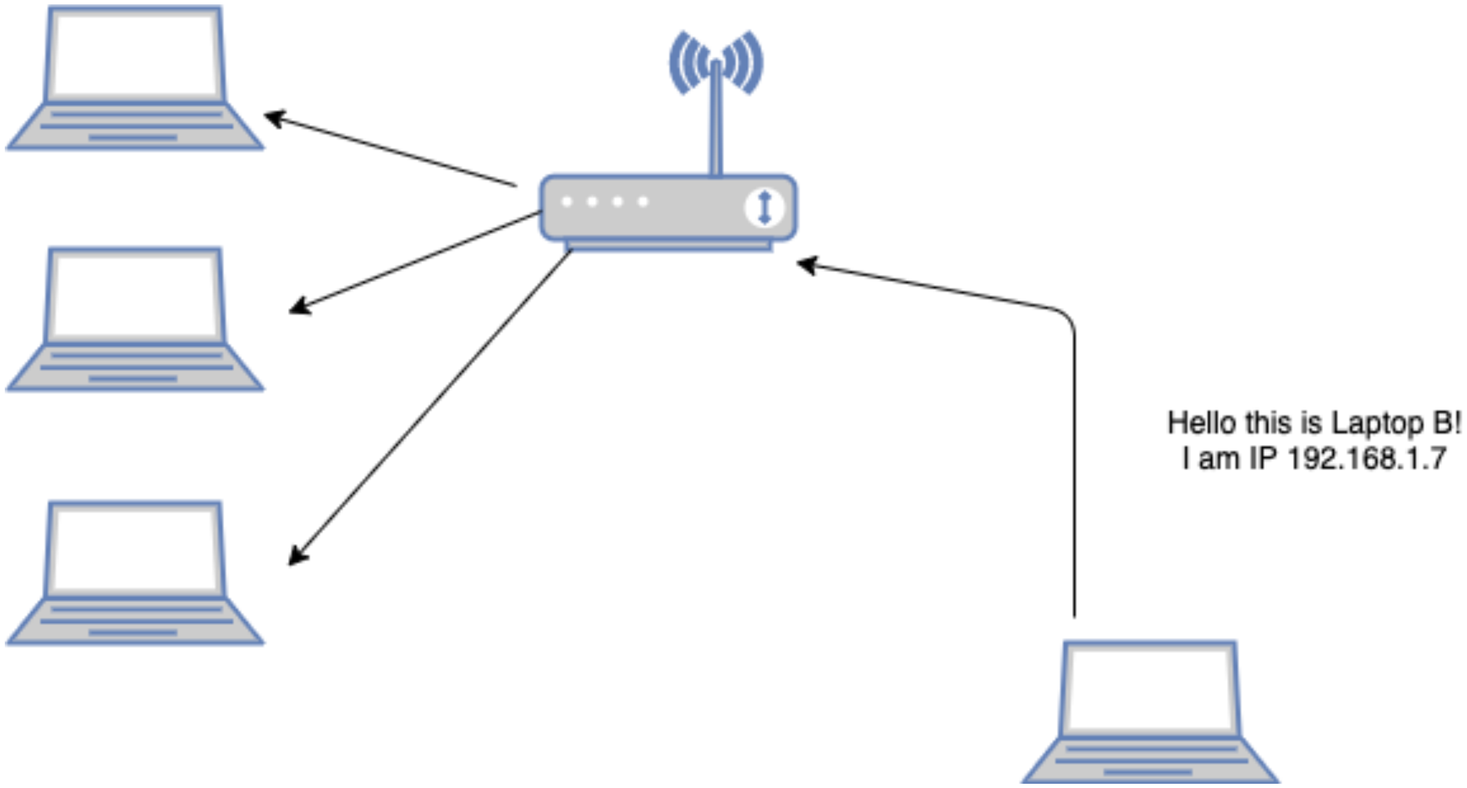


Problem: connect without knowing IP?

I want to connect to Laptop A

I want to connect to Laptop B

Solution: Multicast DNS



TIME TO BUILD

15

Connecting (Signaling Offer/Answer)

```go
package main

import (
    "github.com/pion/webrtc/v2"
)

func main() {
    peerConnection, err := webrtc.NewPeerConnection(webrtc.Configuration{})
    if err != nil {
        panic(err)
    }

    offer, err := peerConnection.CreateOffer(nil)
    if err != nil {
        panic(err)
    }

    err = peerConnection.SetLocalDescription(offer)
    if err != nil {
        panic(err)
    }

    // send Offer to remote PeerConnection via any protocol
    // receive Answer from remote PeerConnectian
    answer := webrtc.SessionDescription{}
    err = peerConnection.SetRemoteDescription(answer)
    if err != nil {
        panic(err)
    }

    // You are connected
}
```

Sending Data (DataChannels)

```
    datachannel, err := peerConnection.CreateDataChannel("my-fun-channel", nil)
    if err != nil {
        panic(err)
    }

    datachannel.OnOpen(func() {
        err = datachannel.SendText("Hello World!")
        if err != nil {
            panic(err)
        }
    })
}
```

Receiving Data (DataChannels)

```
peerConnection.OnDataChannel(func(datachannel *webrtc.DataChannel) {
    datachannel.OnOpen(func() {
        fmt.Printf("New stream %s \n", datachannel.Label())
    })

    datachannel.OnMessage(func(msg webrtc.DataChannelMessage) {
        fmt.Printf("%s \n", msg.Data)
    })
})
```

Deploy to the browser!

```
seaduboi@38f9d359441f:~/go/src/github.com/Sean-Der/kranky$ GOOS=js GOARCH=wasm go build -o main.wasm
seaduboi@38f9d359441f:~/go/src/github.com/Sean-Der/kranky$ cat index.html
<html>
        <head>
                <meta charset="utf-8"/>
                <script src="wasm_exec.js"></script>
                <script>
                        const go = new Go();
                        WebAssembly.instantiateStreaming(fetch("main.wasm"), go.importObject).then((result) => {
                                go.run(result.instance);
                        });
                </script>
        </head>
        <body></body>
</html>
```

19

Send Video

```
videoTrack, err := peerConnection.NewTrack(webrtc.DefaultPayloadTypeVP8, 50000, "video", "pion")
if err != nil {
    panic(err)
}

_, err = peerConnection.AddTrack(videoTrack)
if err != nil {
    panic(err)
}

for {
    frame, _, err := ivf.ParseNextFrame()
    if err != nil {
        panic(err)
    }

    err = videoTrack.WriteSample(media.Sample{Data: frame, Samples: 90000})
    if err != nil {
        panic(err)
    }
}
```

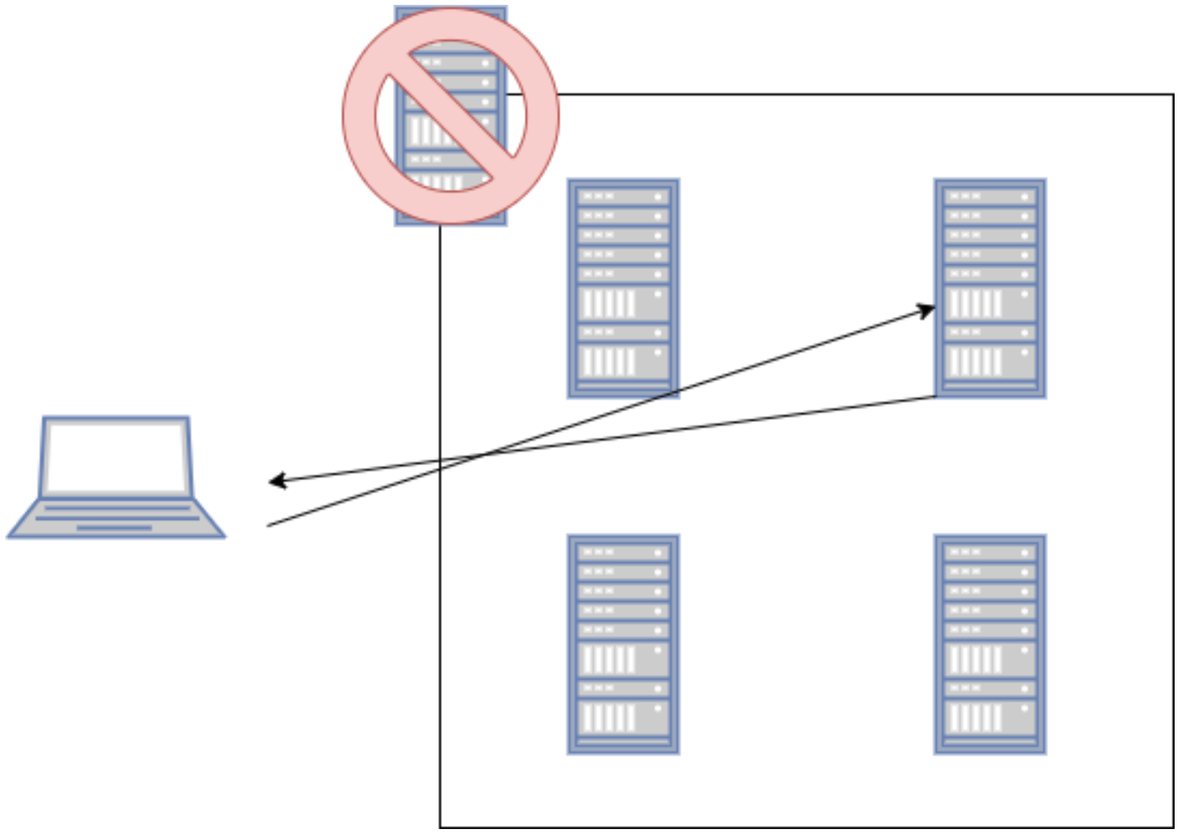
Receive Video

```
peerConnection.OnTrack(func(track *webrtc.Track, receiver *webrtc.RTPReceiver) {
    if track.Codec().Name == webrtc.Opus {
        for {
            packet, err = track.ReadRTP()
            if err != nil {
                panic(err)
            }
            // Playback using Media library of your choice
        }
    }
})
```

PION IN ACTION

ssh-p2p

github.com/nobonobo/ssh-p2p



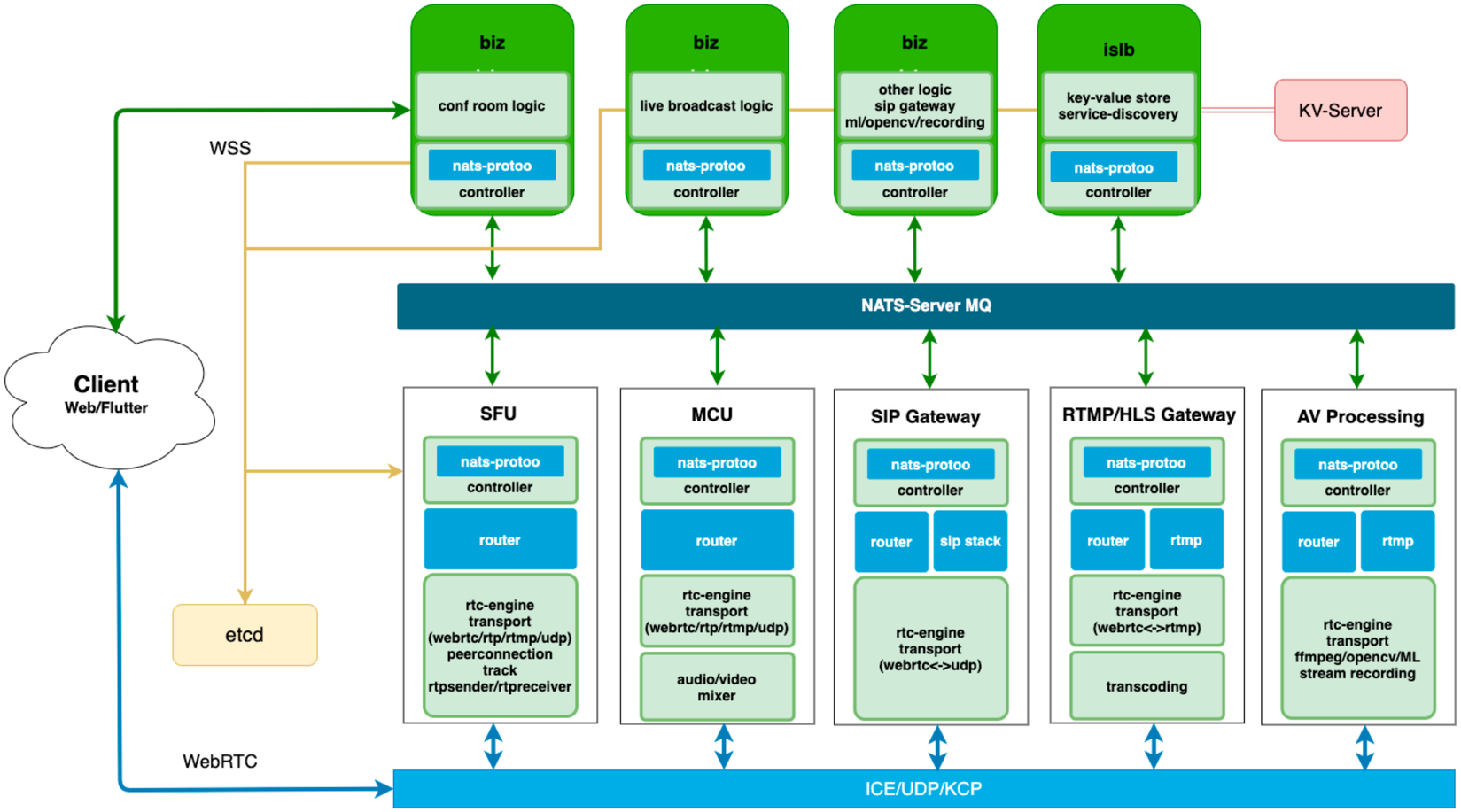
CloudRetro.io

RTCTunnel

github.com/rtctunnel

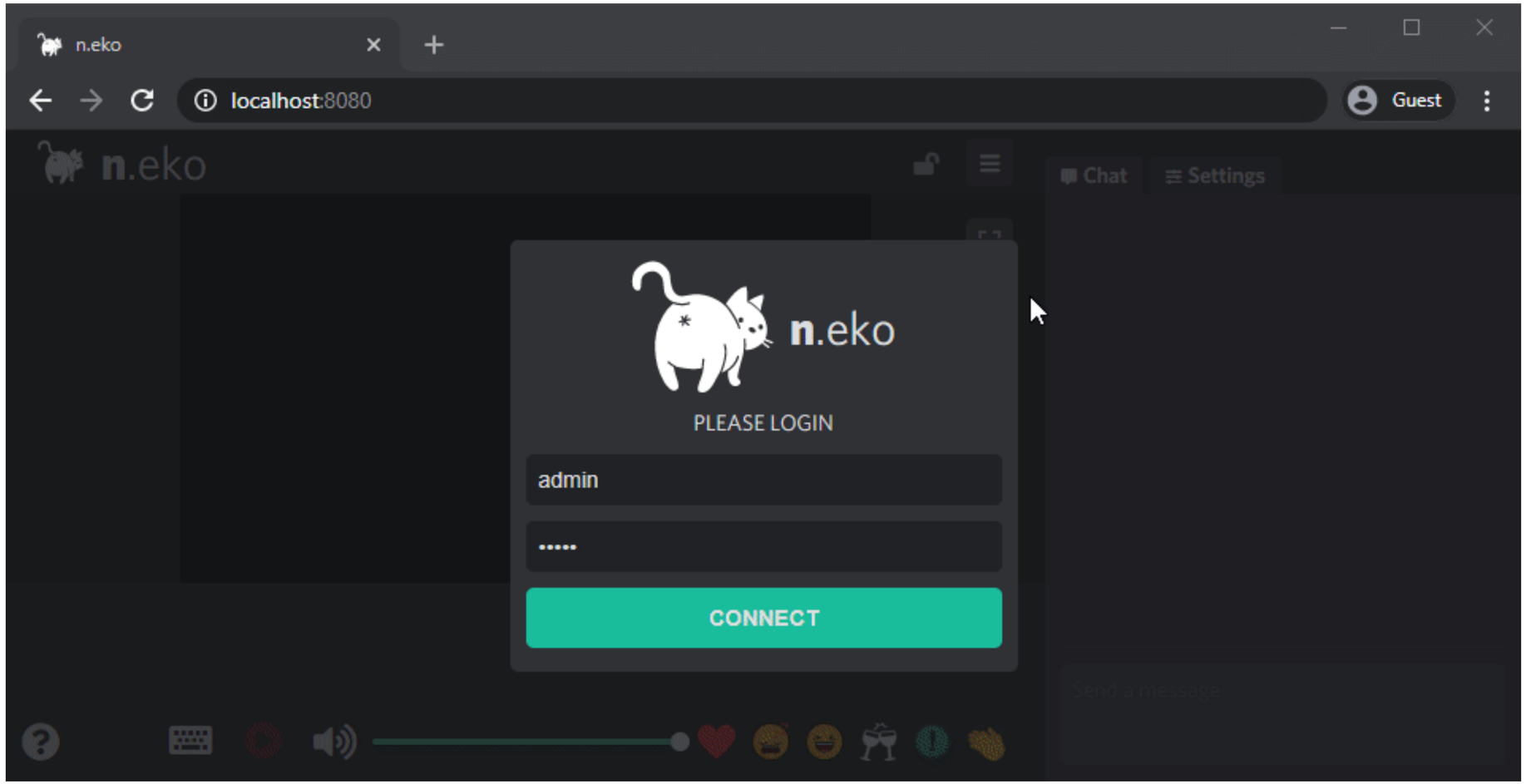
Ion

github.com/pion/ion



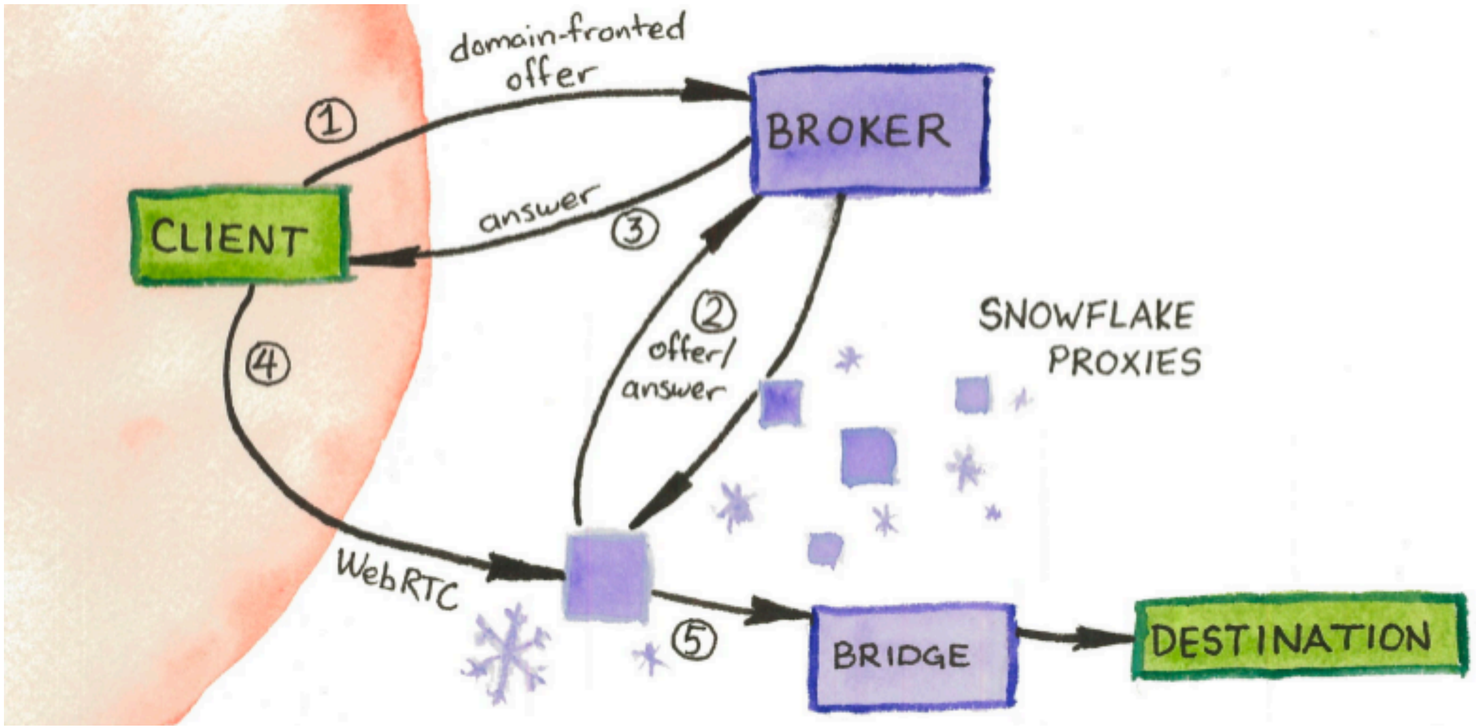
Neko

github.com/nurdism/neko

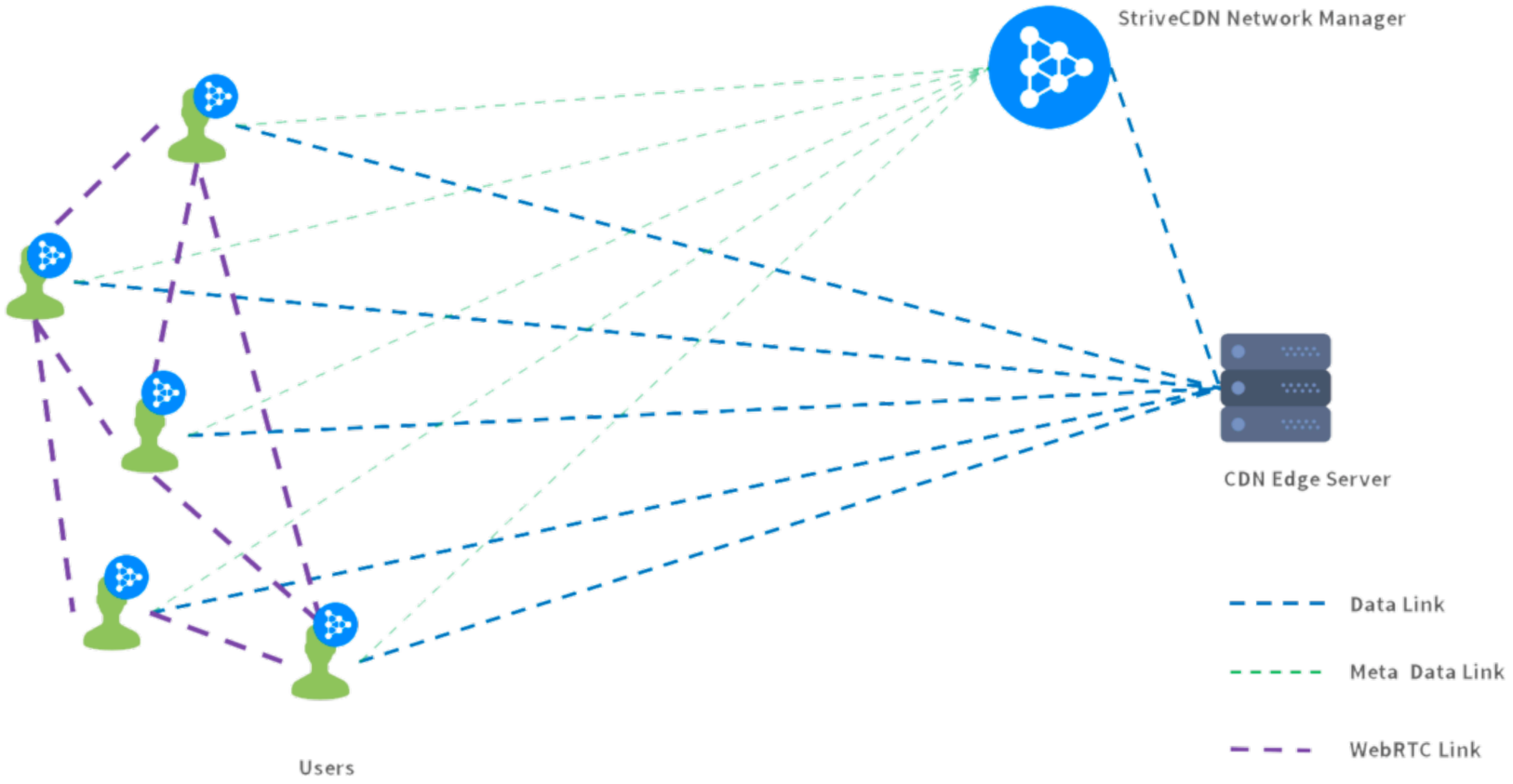


Snowflake

snowflake.torproject.org

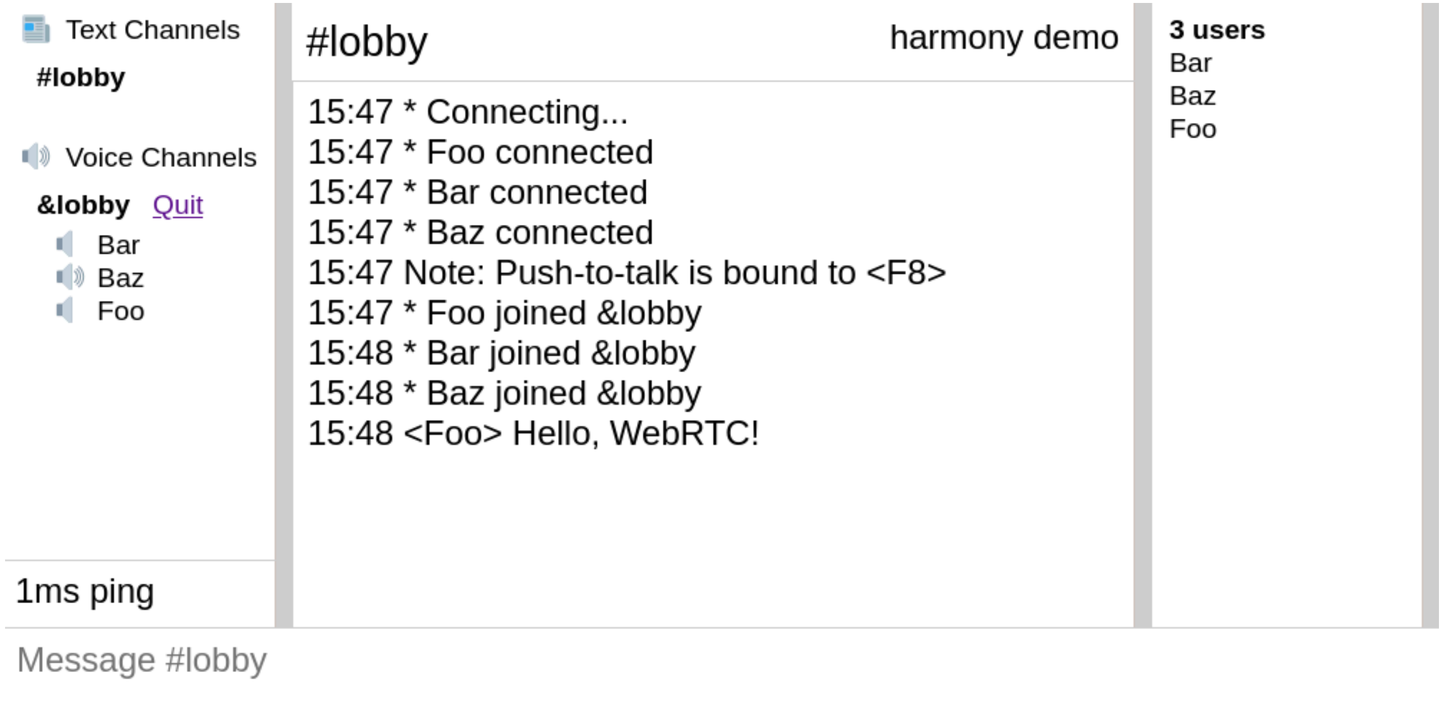


Strive CDN


StriveCDN Network Manager
CDN Edge Server
Users
Data Link
Meta Data Link
WebRTC Link

Harmony

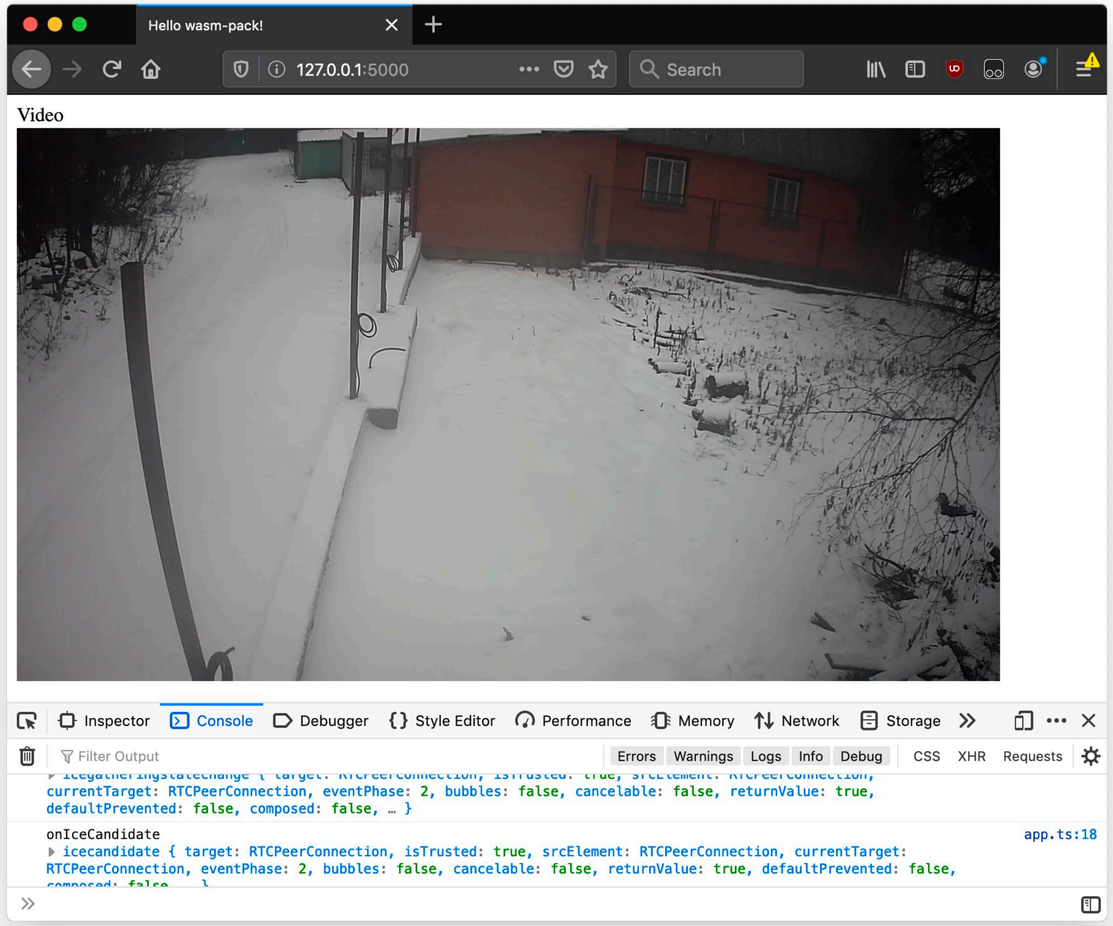
gitlab.com/tslocum/harmony



TelloGo

https://github.com/Ragnar-H/TelloGo

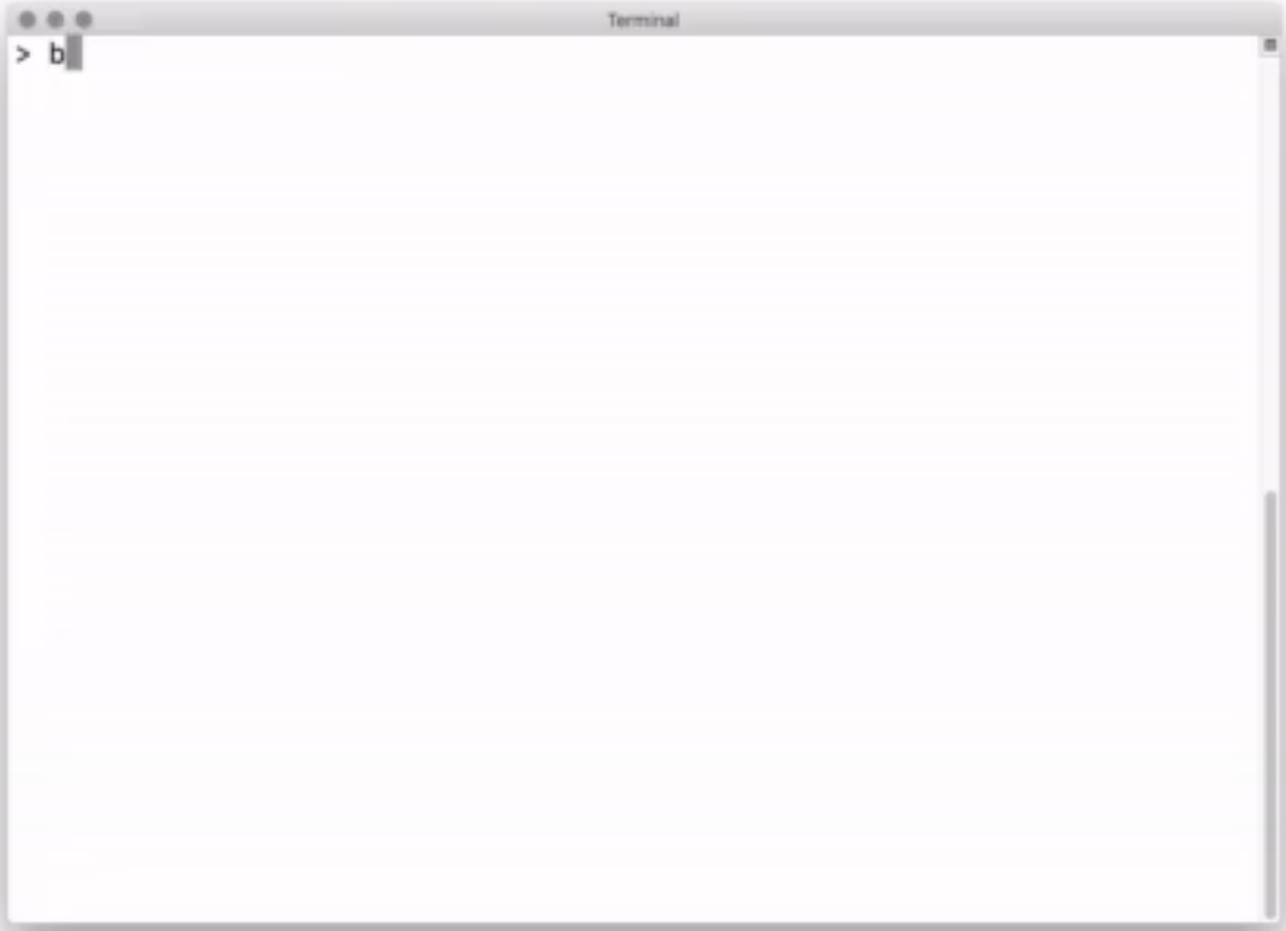
IoT Camera


Hello wasm-pack!
127.0.0.1:5000
Search
Video
Inspector
Console
Debugger
Style Editor
Performance
Memory
Network
Storage
Filter Output
Errors
Warnings
Logs
Info
Debug
CSS
XHR
Requests
currentTarget: RTCPeerConnection, eventPhase: 2, bubbles: false, cancelable: false, returnValue: true,
defaultPrevented: false, composed: false, … }
onIceCandidate
app.ts:18
icecandidate { target: RTCPeerConnection, isTrusted: true, srcElement: RTCPeerConnection, currentTarget:
RTCPeerConnection, eventPhase: 2, bubbles: false, cancelable: false, returnValue: true, defaultPrevented: false,

31

ascii

https://github.com/dialup-inc/ascii



Pion needs you

Empower those helping the internet!

Gain deep WebRTC knowledge

A fun challenge where you pick the goals

github.com/pion

pion.ly/slack

twitter.com/_pion