Privacy and Decentralisation with Multicast

Brett Sheffield — Librecast Project

@brett_sheffield — @librecast

Before we begin...

Multicast

"IP Multicast will play a prominent role on the Internet in the coming years. It is a requirement, not an option, if the Internet is going to scale. Multicast allows application developers to add more functionality without significantly impacting the network."
– RFC3170, Sep 2001

Efficient

Scalable

Real-World

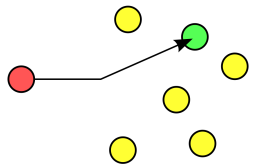
Privacy

Decentralisation

What is Multicast?

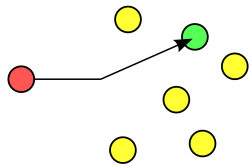
Definition

Definition

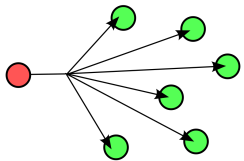


Unicast

Definition

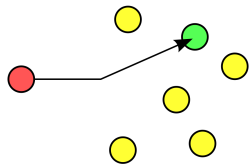


Unicast

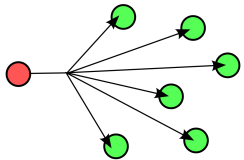


Broadcast

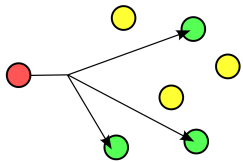
Definition



Unicast

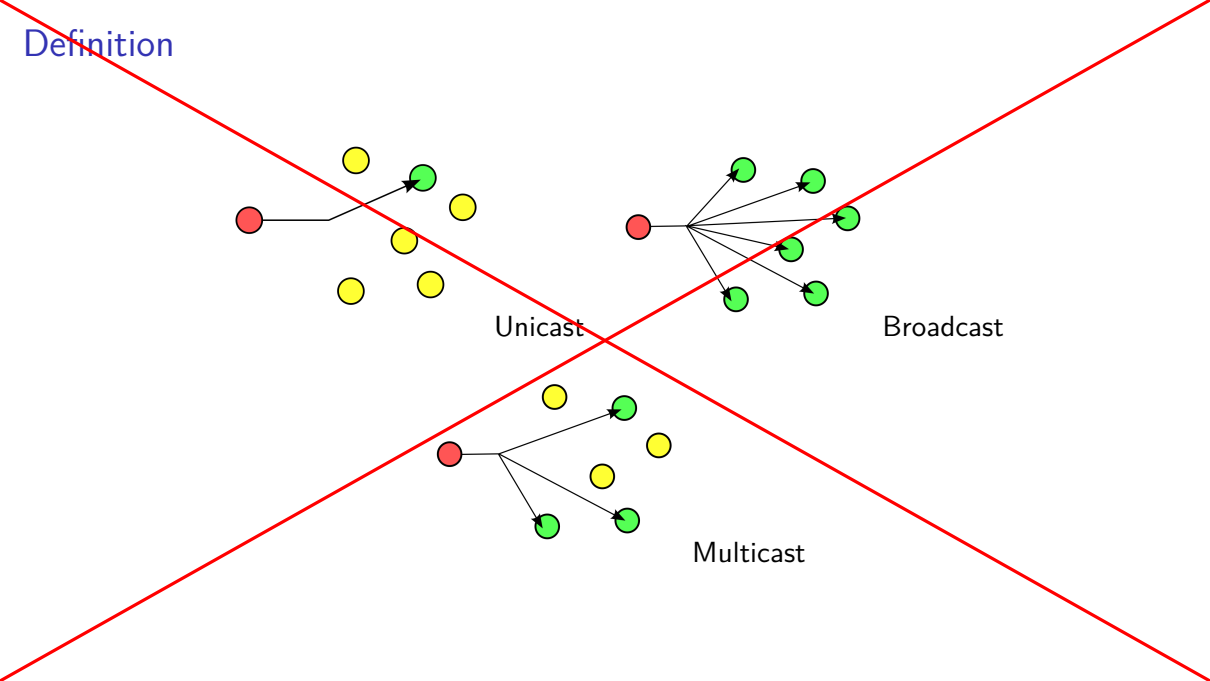


Broadcast



Multicast

Definition



What is Multicast?

Unicast, Broadcast

Multicast

What is Multicast?

Unicast, Broadcast PUSH

Multicast

What is Multicast?

Unicast, Broadcast PUSH

Multicast PULL

Multicast Misconceptions

Multicast Misconceptions

- only for streaming

Multicast Misconceptions

- only for streaming
- no use for video on demand

Multicast Misconceptions

- only for streaming
- no use for video on demand
- unreliable

Multicast Misconceptions

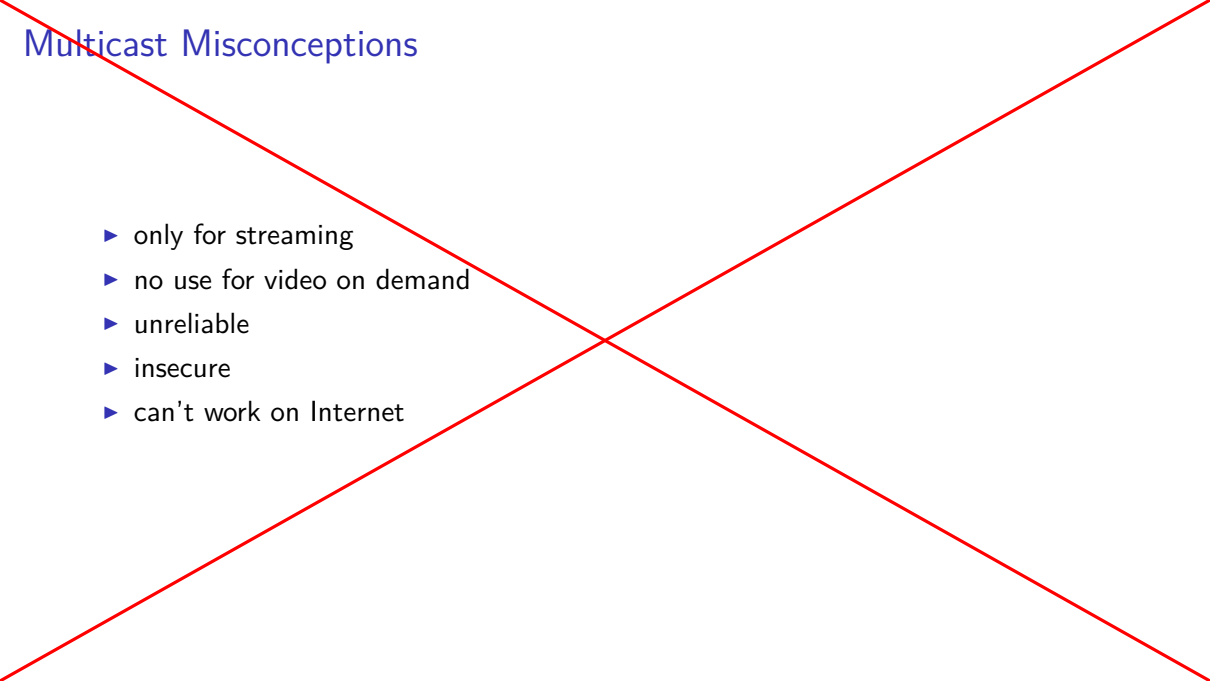
- only for streaming
- no use for video on demand
- unreliable
- insecure

Multicast Misconceptions

- only for streaming
- no use for video on demand
- unreliable
- insecure
- can't work on Internet

Multicast Misconceptions

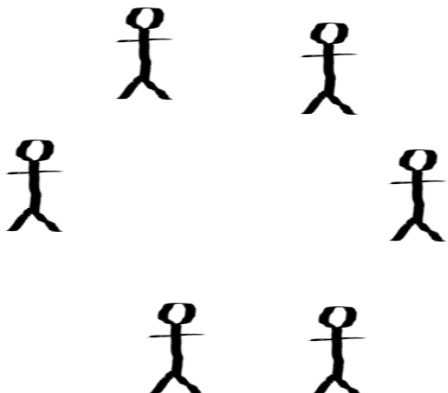
- only for streaming
- no use for video on demand
- unreliable
- insecure
- can't work on Internet

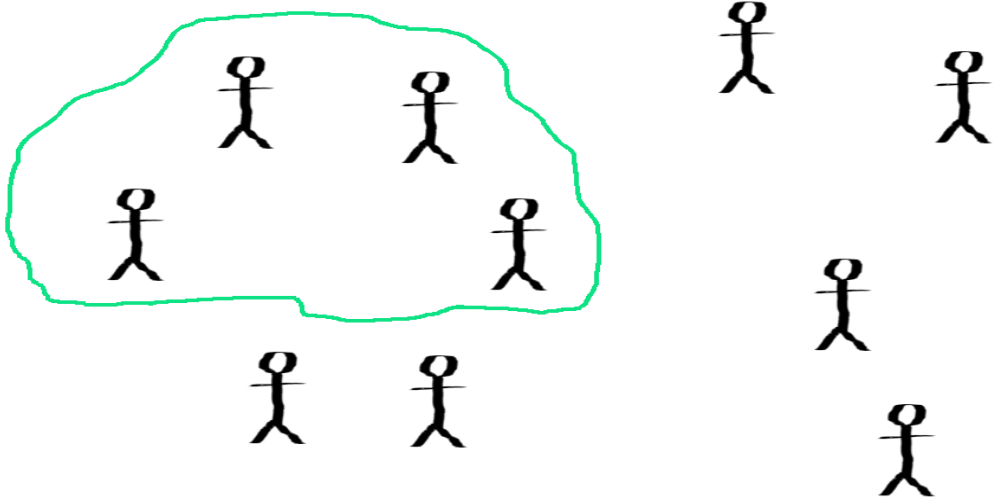


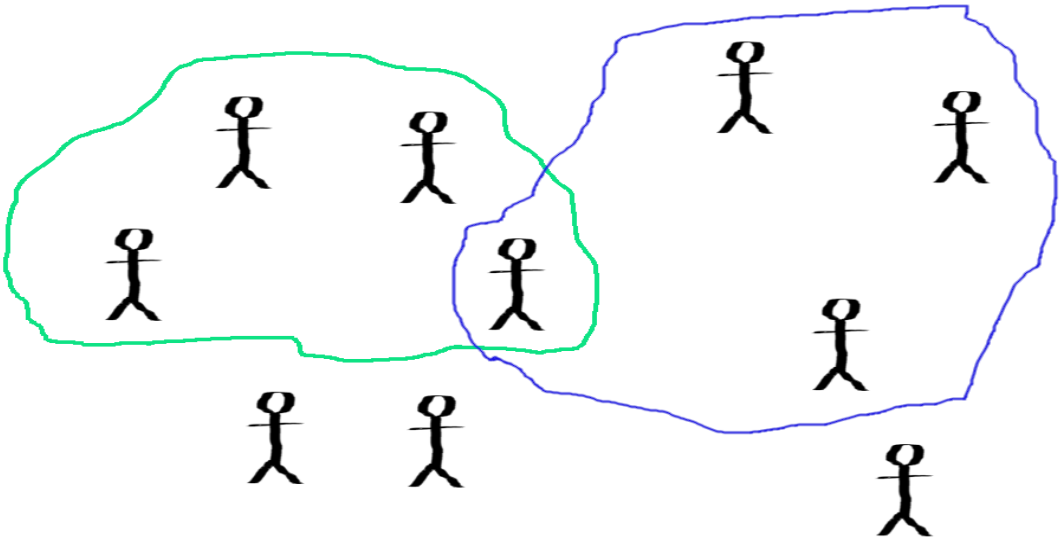
Multicast is ...

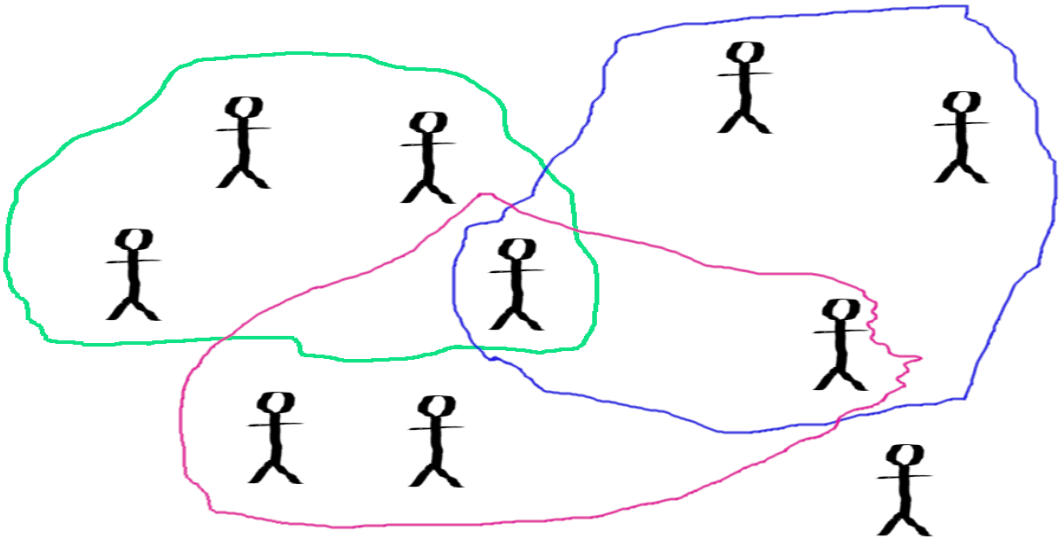
Multicast is ...
Group Communication

All Communication is
Group Communication









The Problem with Unicast

$$\text{Unicast} \implies \text{Centralized}$$

Multicast $\Longrightarrow$ Decentralized

Why Does it Matter?

WARNING:

The approval of the Ministry of Manpower is required if speaker is a foreigner and is giving a talk on racial, communal, religious, caused-related or political topics. The applicant has to submit the letter of approval for a Miscellaneous Work Pass (for foreign speaker) to the Police Officer before a permit can be issued.


THE UNIVERSAL DECLARATION
OF Human Rights

Internet as a Tool

Internet Under Threat


LinkedIn
foursquare
LIVEJOURNAL
vimeo
YouTube
facebook
Blogger
twitter
flickr
Like


PARLIAMENT
SINGAPORE

Efficiency Matters

Privacy and Security: Design Goals

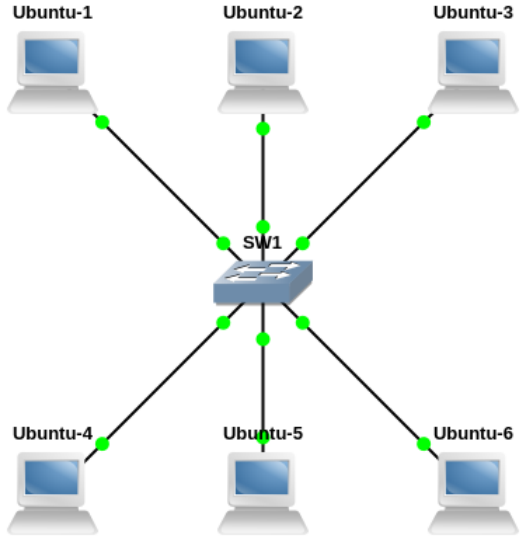
A Brief History of IP Multicast

In the beginning...

PIM
(Protocol Independent Multicast)

PIM
(Unicast Dependent Multicast)

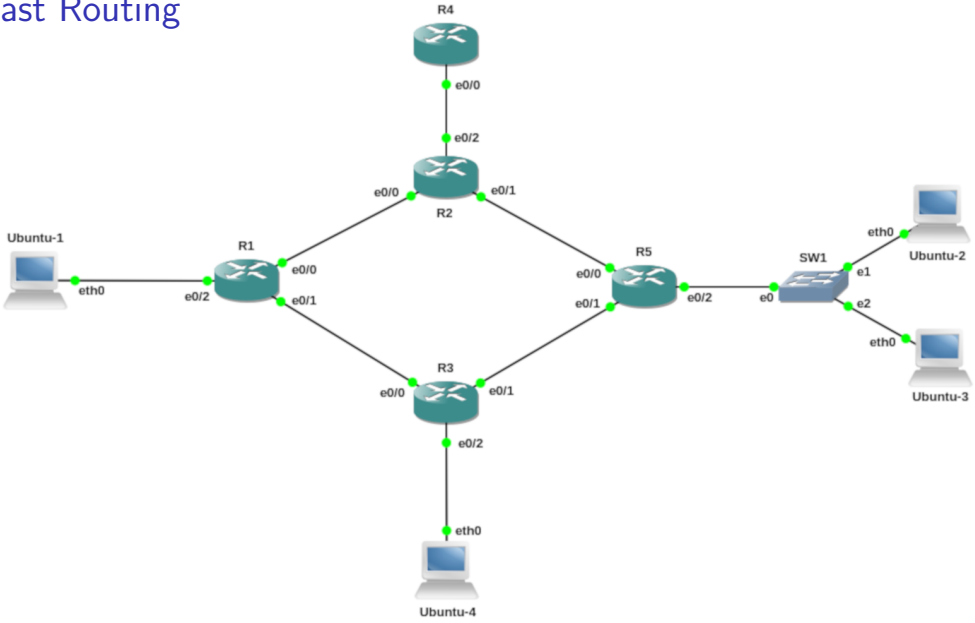
LAN multicast (MLD Snooping)


Ubuntu-1
Ubuntu-2
Ubuntu-3
SW1
Ubuntu-4
Ubuntu-5
Ubuntu-6

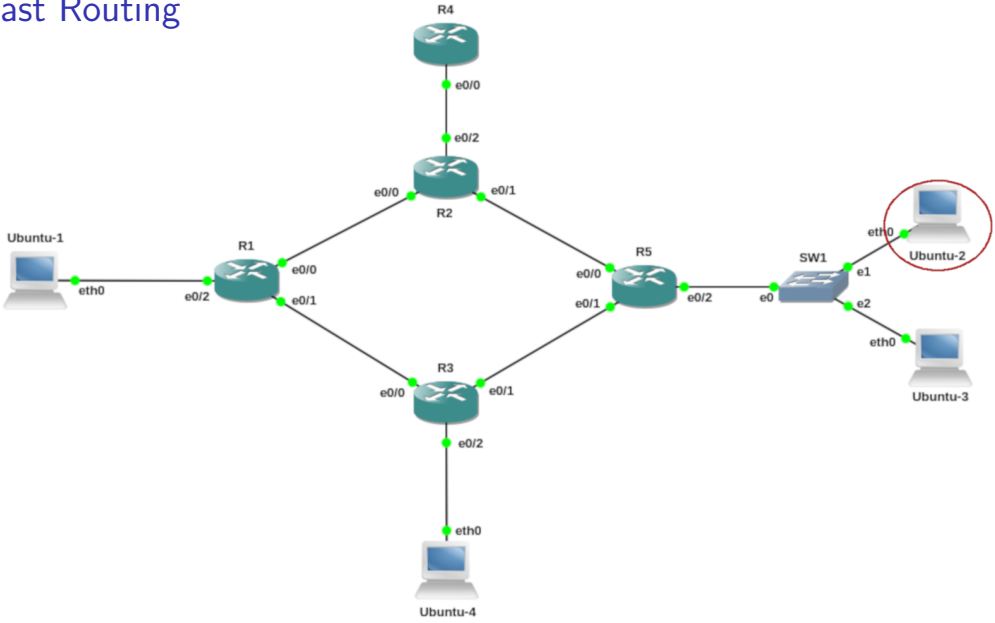
Multicast Routing

Rendezvous Points

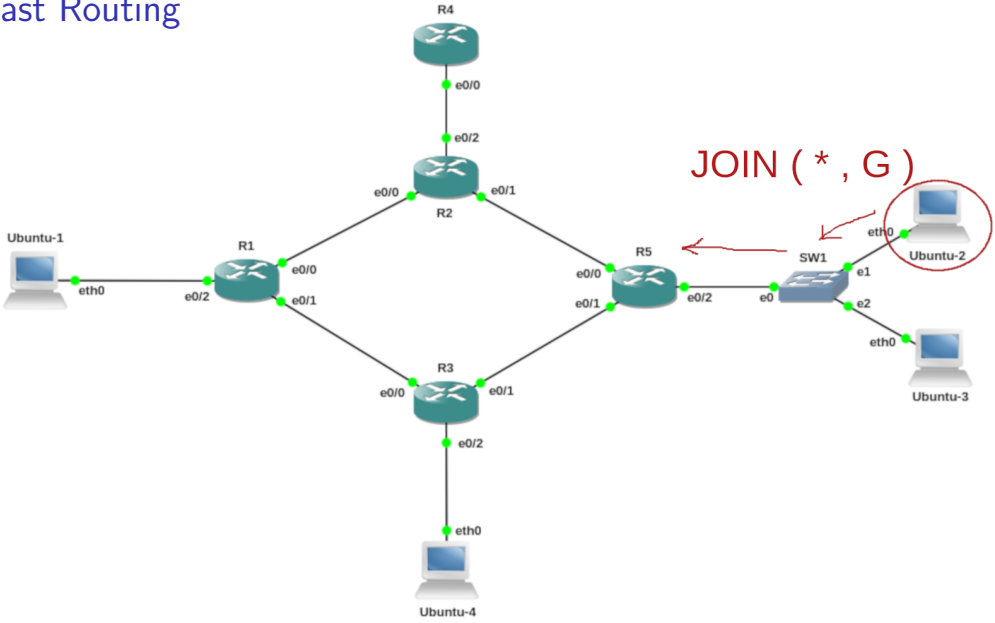
Multicast Routing



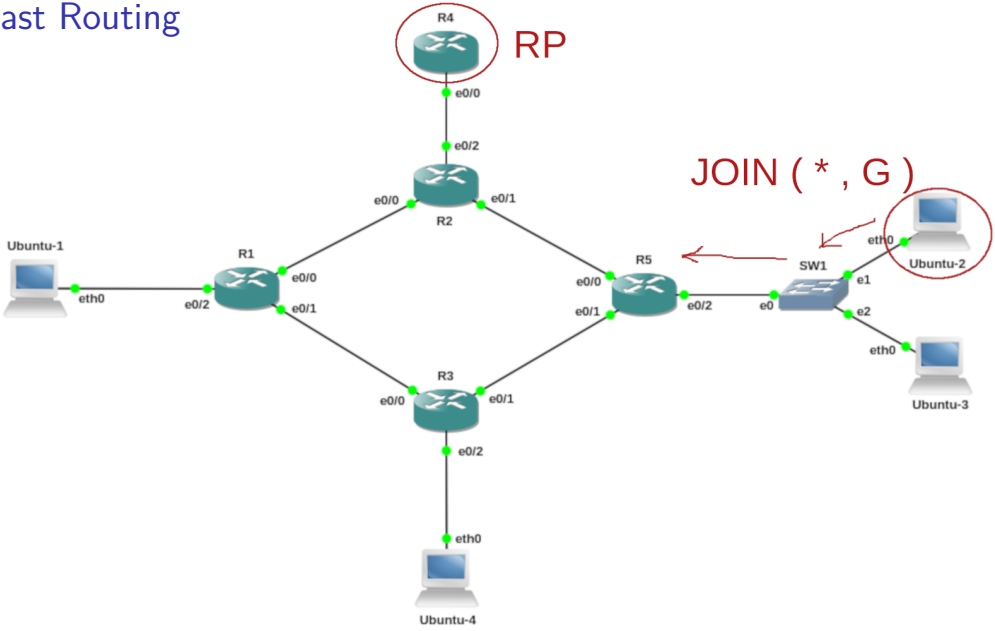
Multicast Routing



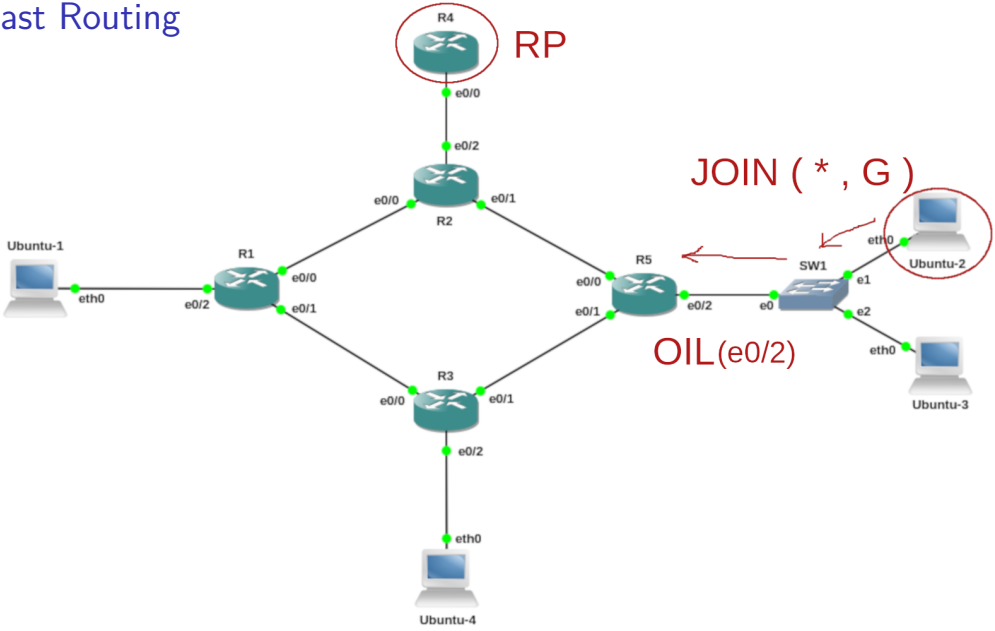
Multicast Routing



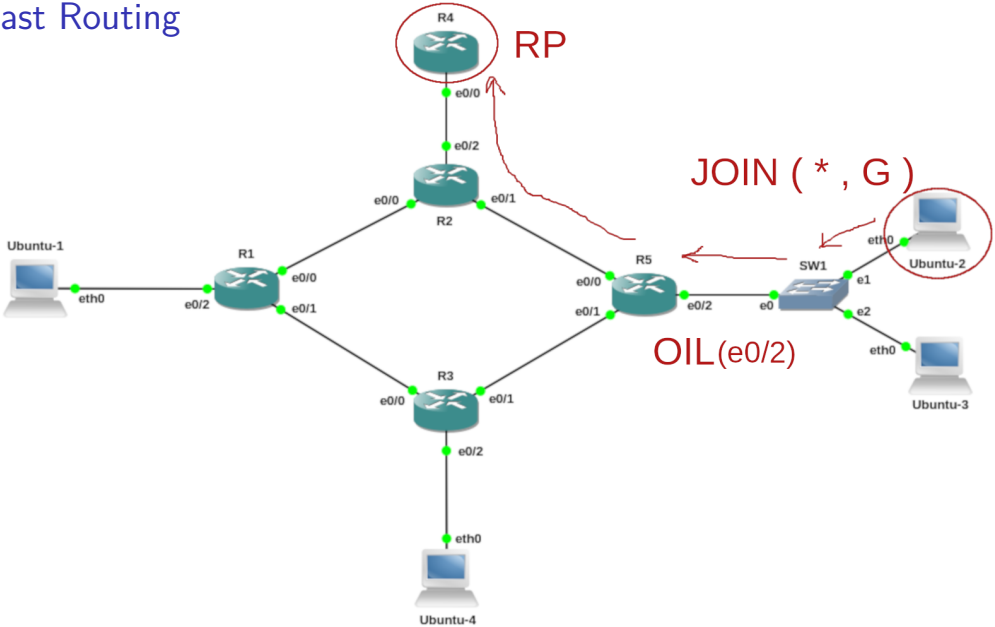
Multicast Routing



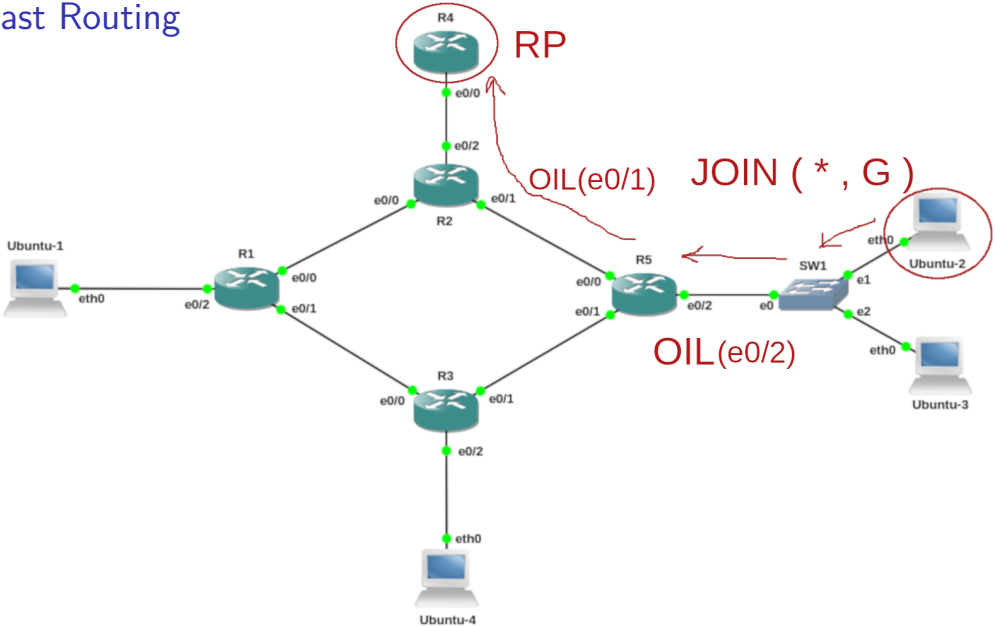
Multicast Routing



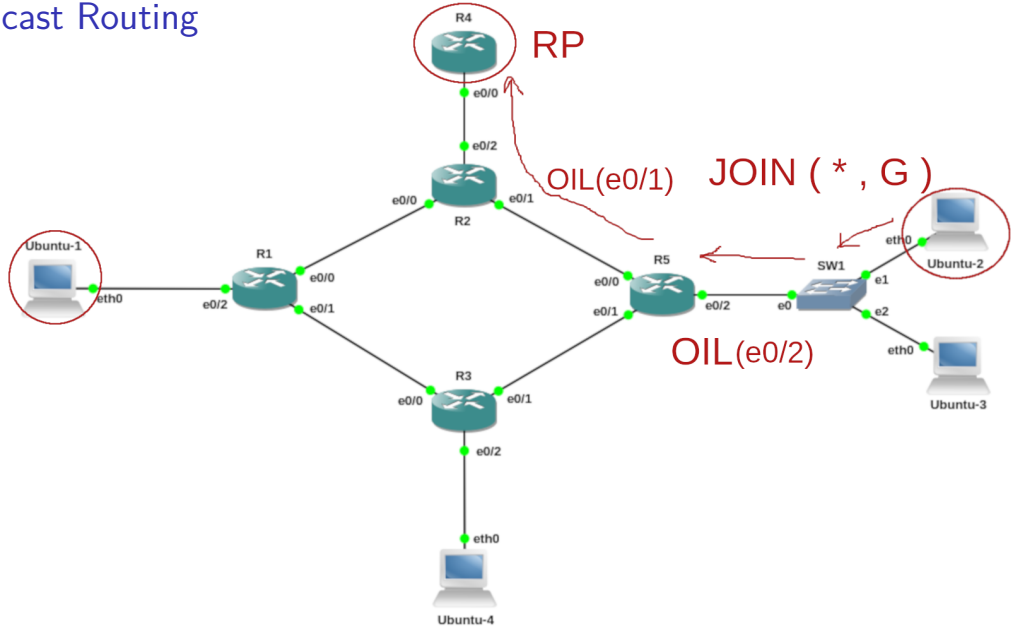
Multicast Routing



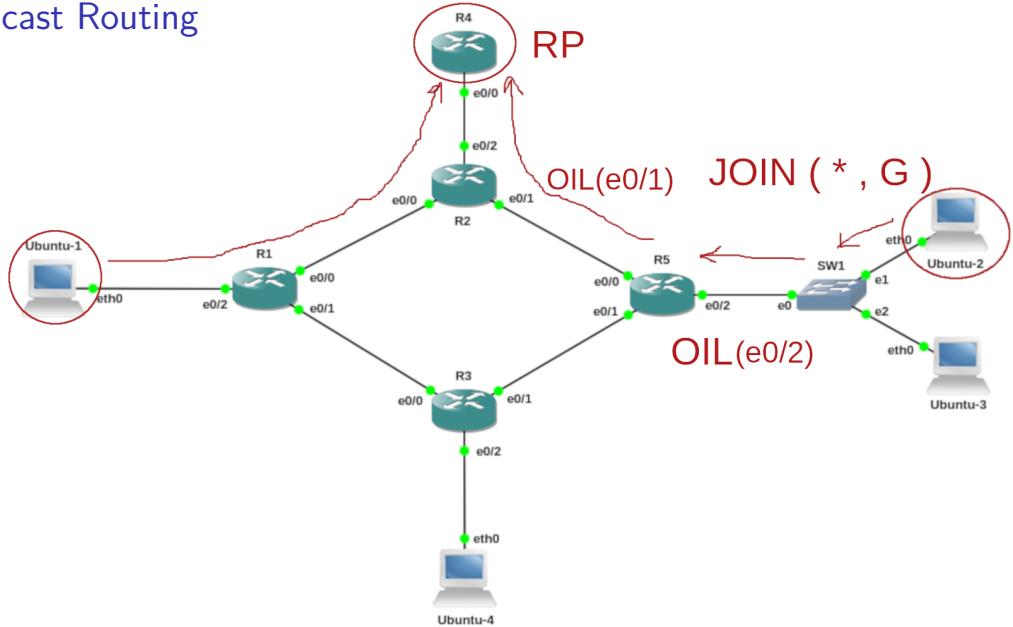
Multicast Routing



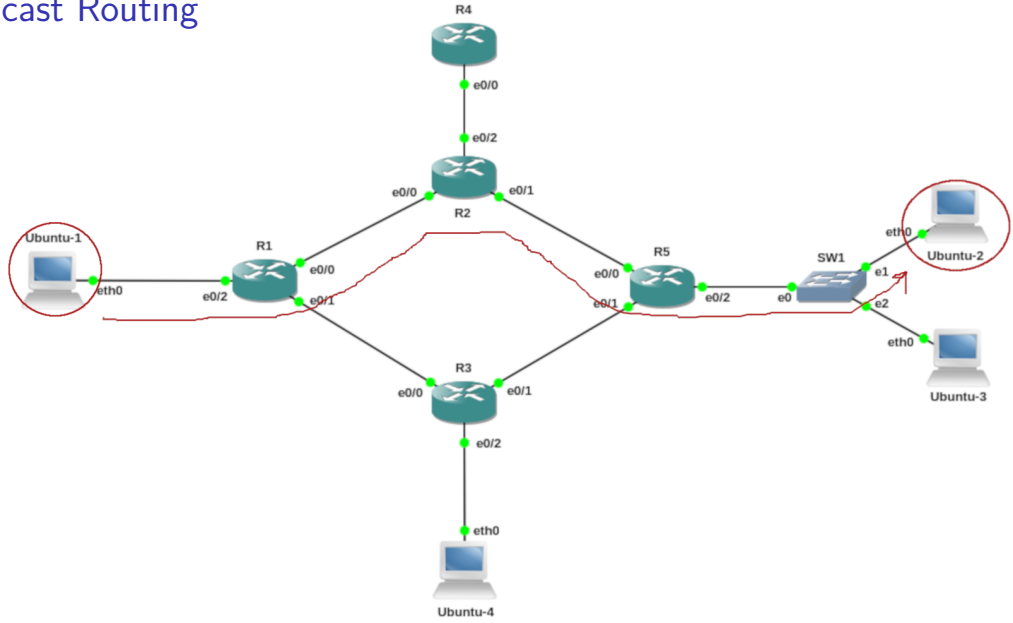
Multicast Routing



Multicast Routing



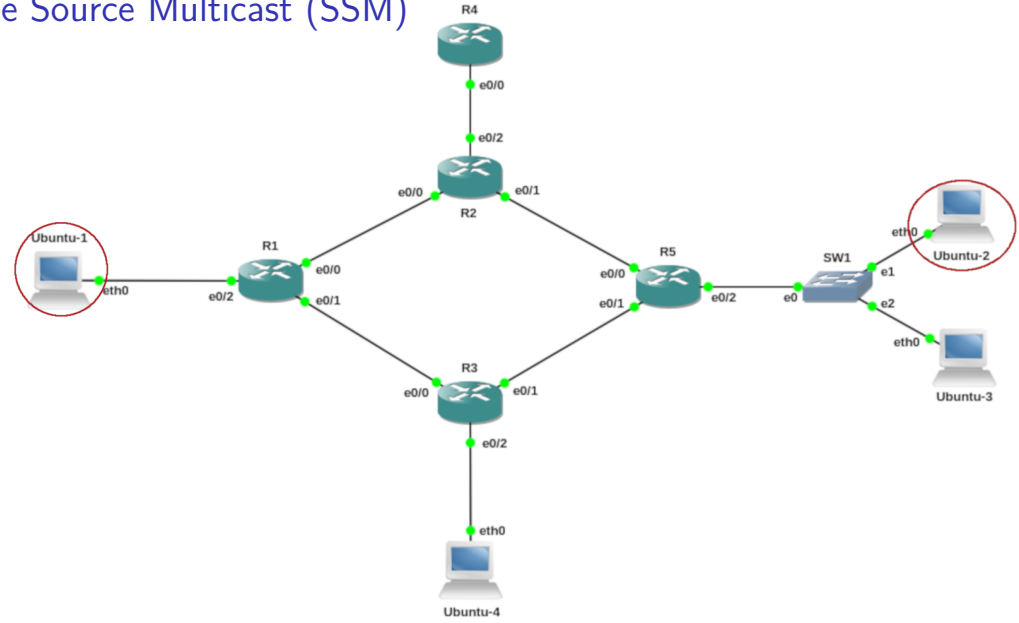
Multicast Routing



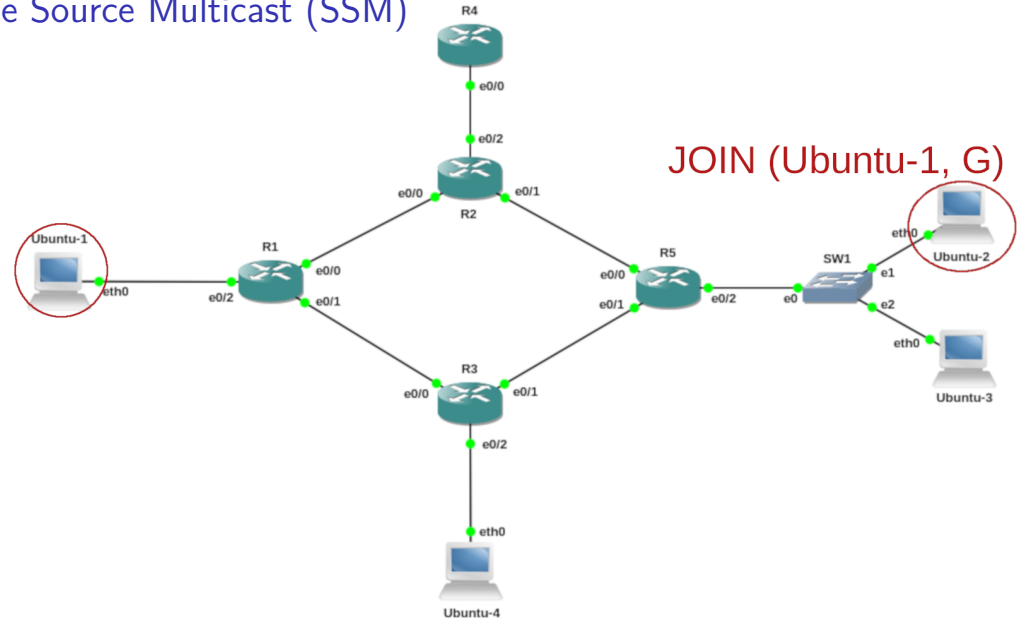
Any Source Multicast (ASM)
(*,G)

Single Source Multicast (SSM)
(S,G)

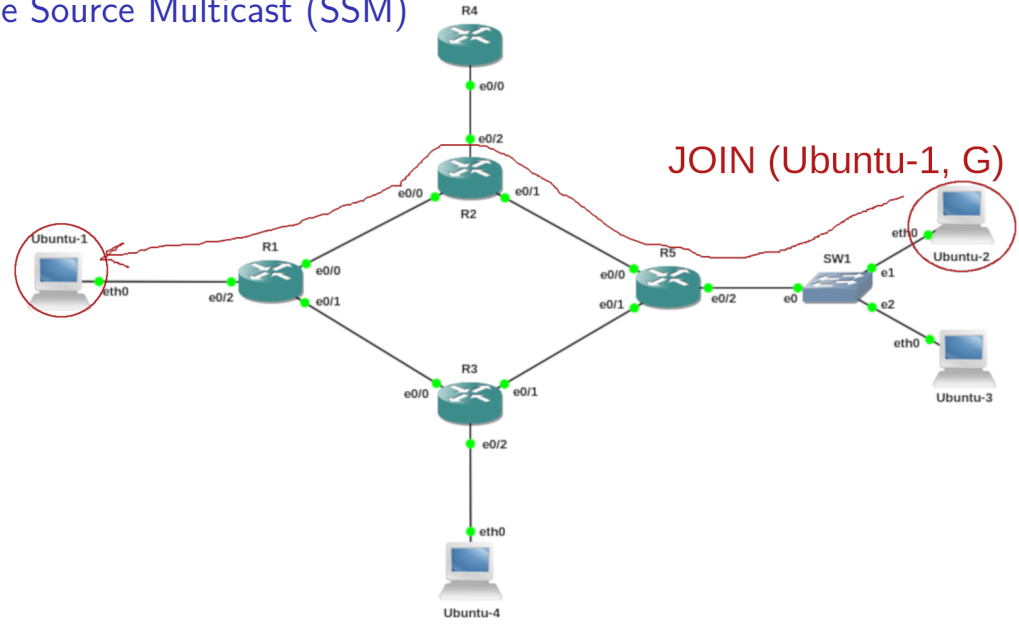
Single Source Multicast (SSM)



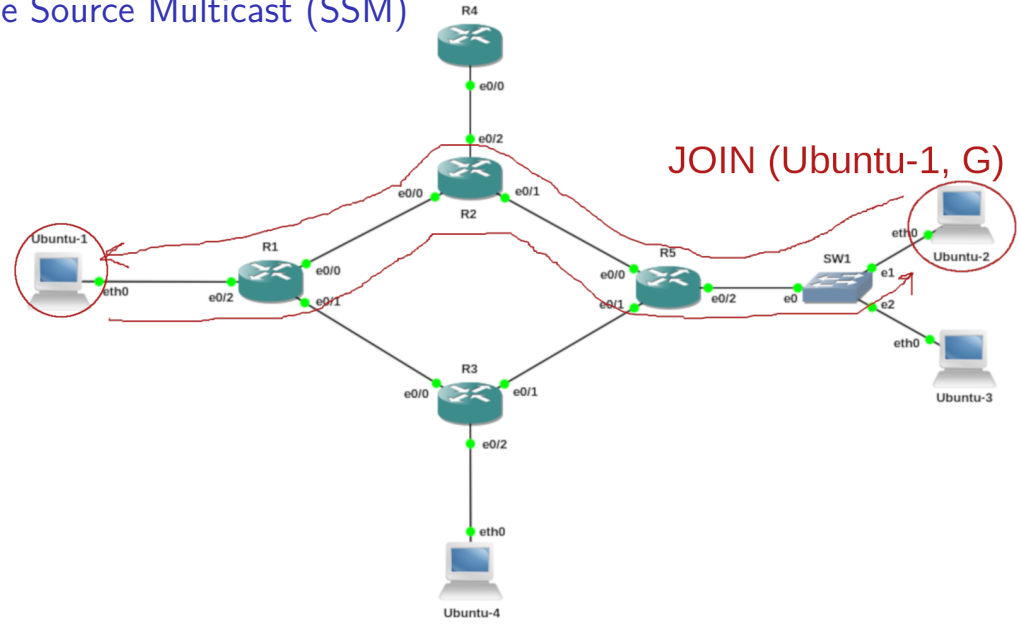
Single Source Multicast (SSM)



Single Source Multicast (SSM)



Single Source Multicast (SSM)



IPv6
(2^{112})

Mbone

Castgate

RFC 7450
Automatic Multicast Tunneling (AMT)

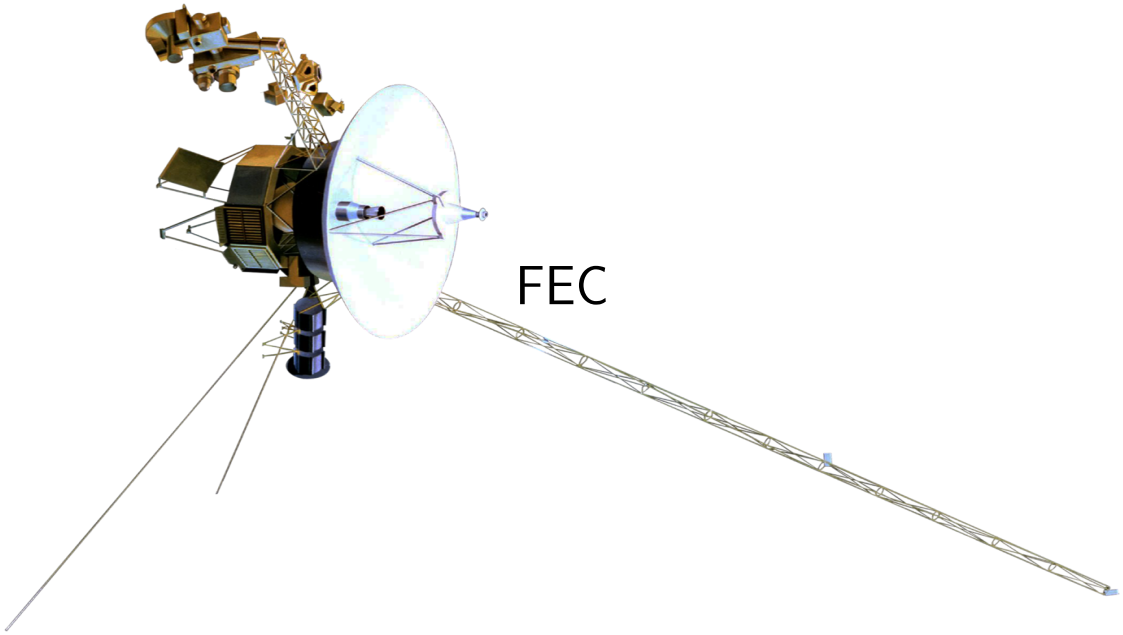
TCP/IP

Are there other ways we can achieve TCP/IP-like reliability?

PGM (RFC 3208 - Experimental)

NACKs, Replay

Loop and Repeat


FEC

Multicast Applications

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)
- Email

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)
- Email
- WWW

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)
- Email
- WWW
- Facebook + All social media

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)
- Email
- WWW
- Facebook + All social media
- File sharing

Multicast Applications

- Chat (IRC / Jabber / Slack / RocketChat / ChatOps)
- Email
- WWW
- Facebook + All social media
- File sharing

(ALL BUILT ON UNICAST)

OSI Layer

Layer			Protocol data unit (PDU)
Host layers	7	Application	Data
	6	Presentation	
	5	Session	
	4	Transport	Segment, Datagram
Media layers	3	Network	Packet
	2	Data link	Frame
	1	Physical	Symbol

Multicast Party Tricks

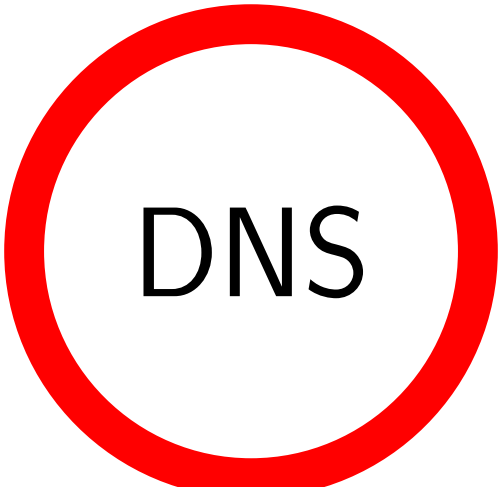
Video Streaming
(one to many)

Video Conferencing
(many to many)

Replication

Consensus (Paxos)

Syslog


DNS

Anatomy of an IPv6 Multicast Address

Anatomy of an IPv6 Multicast Address

ff

Anatomy of an IPv6 Multicast Address

ff1

Anatomy of an IPv6 Multicast Address

ff1e

Anatomy of an IPv6 Multicast Address

ff1e: + group address (112 bits)

Multicast "DNS"

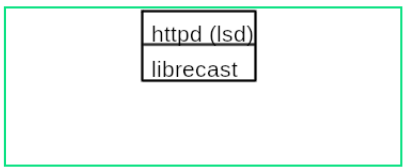
ff1e: + HASH("example.com")

Multicast "DNS"

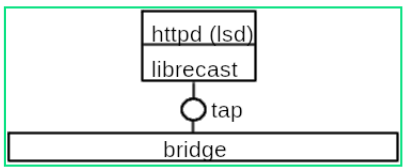
example.com $\Longrightarrow$ ff1e:873e:378f:f6a5:a1f6:fa49:95f1:0faf

Chat

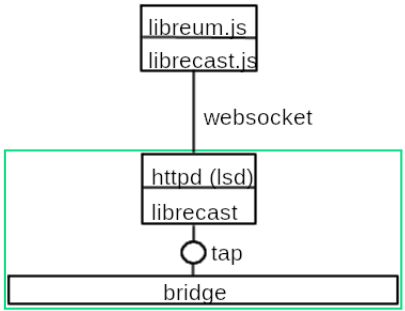
Chat Server



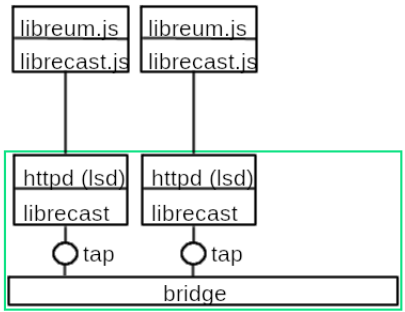
Chat Server



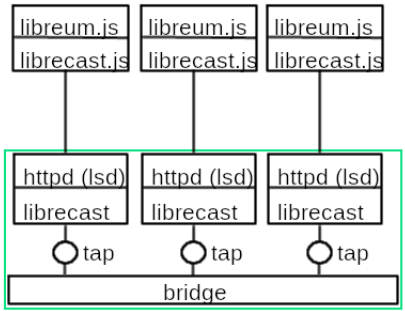
Chat Server


libreum.js
librecast.js
websocket
httpd (lsd)
librecast
tap
bridge

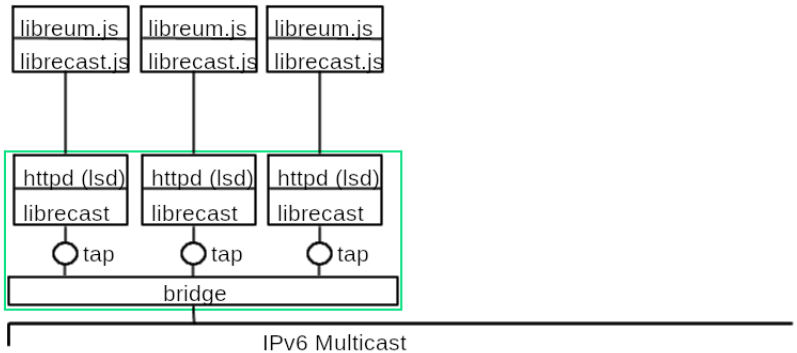
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
bridge

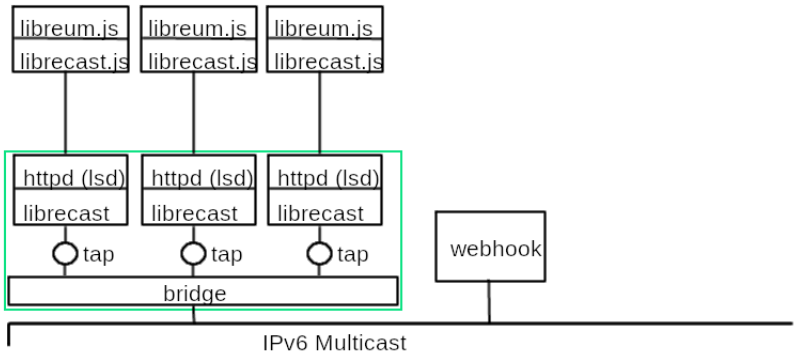
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
bridge

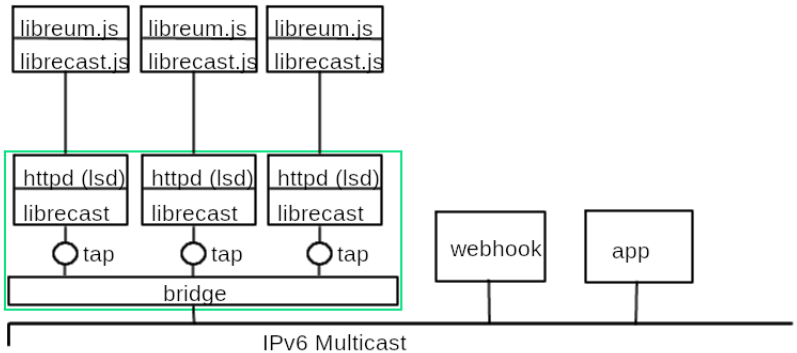
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
bridge
IPv6 Multicast

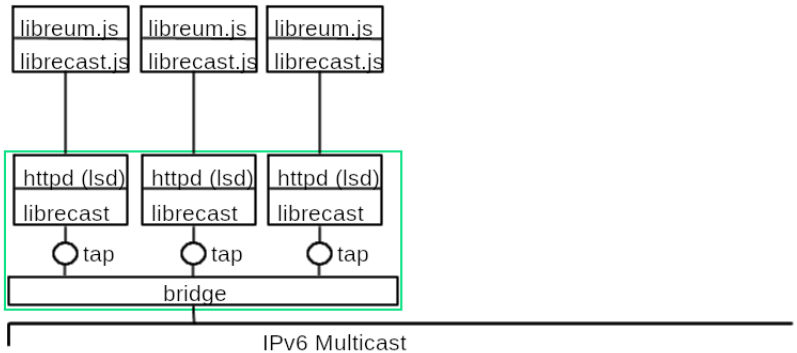
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
bridge
webhook
IPv6 Multicast

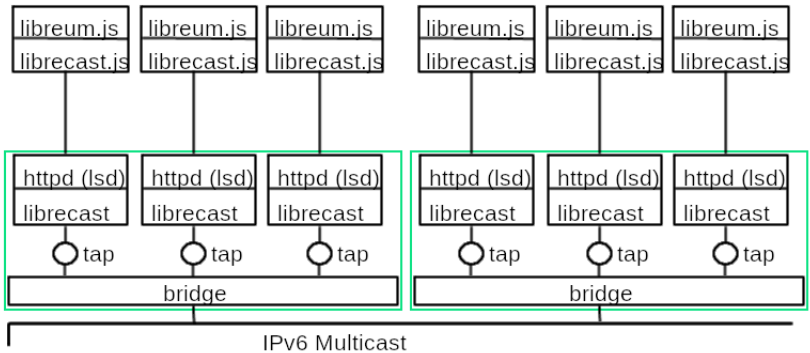
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
bridge
webhook
app
IPv6 Multicast

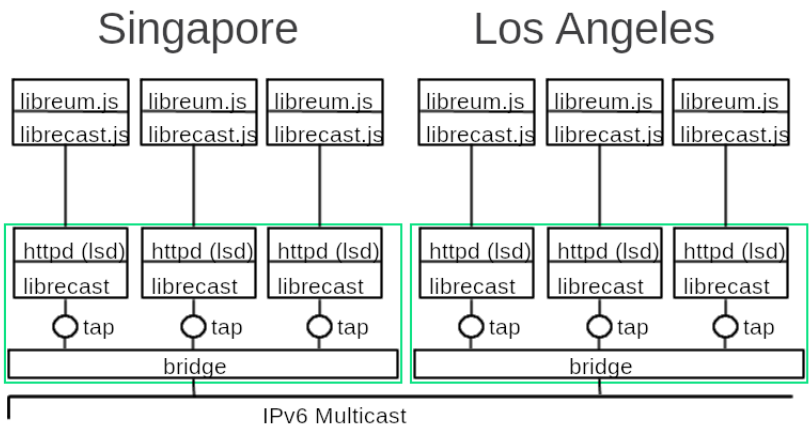
Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
bridge
IPv6 Multicast

Chat Server


libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
tap
tap
tap
bridge
bridge
IPv6 Multicast

Chat Server


Singapore
Los Angeles
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
libreum.js
librecast.js
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
httpd (lsd)
librecast
tap
tap
tap
tap
tap
tap
bridge
bridge
IPv6 Multicast

```
lc_ctx_t *ctx;
lc_socket_t *sock;
lc_channel_t *chan;
lc_message_t msg;

ctx = lc_ctx_new();
sock = lc_socket_new(ctx);
chan = lc_channel_new(ctx, channelName);
lc_channel_bind(sock, chan);

lc_msg_init_size(&msg, strlen(msgtext) - 1);
lc_msg_send(chan, &msg);

/* clean up */
lc_socket_close(sock);
lc_channel_free(chan);
lc_ctx_free(ctx);
```

IOT Updates

```
	signal(SIGINT, sigint_handler);

	ctx = lc_ctx_new();
	sock = lc_socket_new(ctx);
	chan = lc_channel_new(ctx, MY_HARDCODED_CHANNEL);
	lc_channel_bind(sock, chan);

	memset(&f, 0, sizeof(iot_frame_t));

	/* calculate file hash */
	hash(f.hash, map, sb.st_size);

	while (running) {
		for (int i = 0; i <= sb.st_size && running; i += MTU_FIXED) {
			f.op = 0; /* TODO: data opcode */
			f.size = sb.st_size;
			f.off = i;

			if ((i + MTU_FIXED) > sb.st_size)
				f.len = sb.st_size - i;
			else
				f.len = MTU_FIXED;

			logmsg(LOG_DEBUG, "sending %i - %i", i, (int)(i+f.len));

			memcpy(f.data, map + i, f.len);

			lc_msg_init_data(&msg, &f, sizeof(f), NULL, NULL);
			lc_msg_send(chan, &msg);
#ifdef PKT_DELAY
			usleep(PKT_DELAY);
#endif
		}
	}
	terminate();

	return 0;
```

Datagram:

Datagram:

- checksum

Datagram:

- checksum
- size of file

Datagram:

- checksum
- size of file
- size of chunk

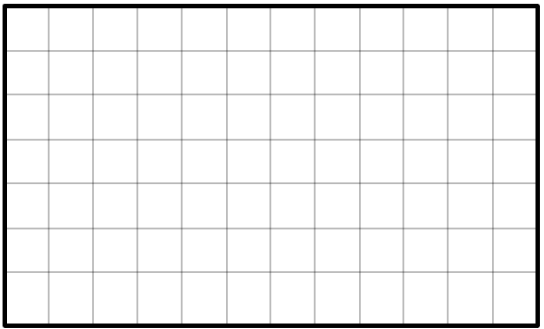
Datagram:

- checksum
- size of file
- size of chunk
- offset

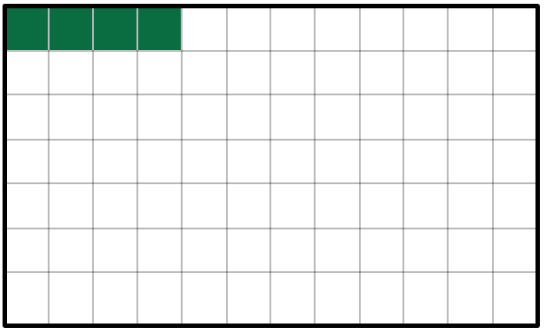
Datagram:

- checksum
- size of file
- size of chunk
- offset
- data

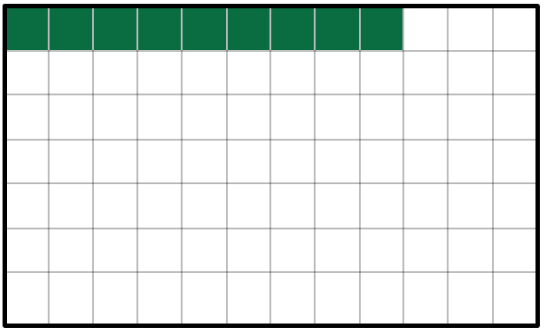
Receiving a File



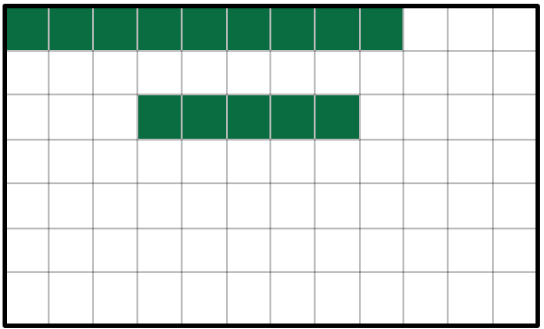
Receiving a File



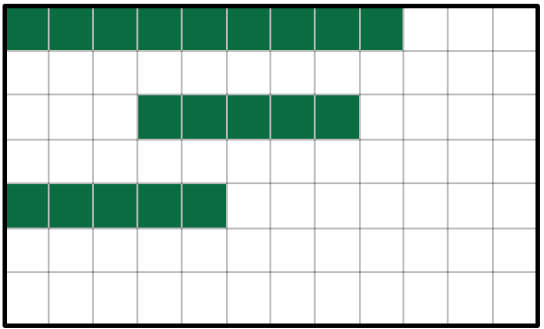
Receiving a File



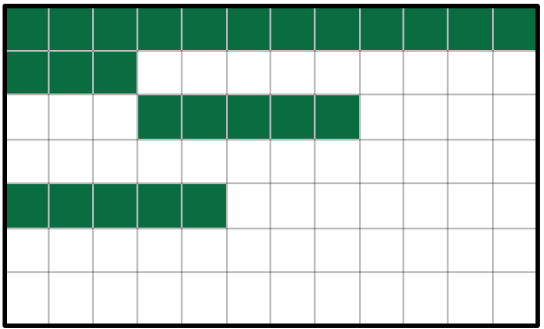
Receiving a File



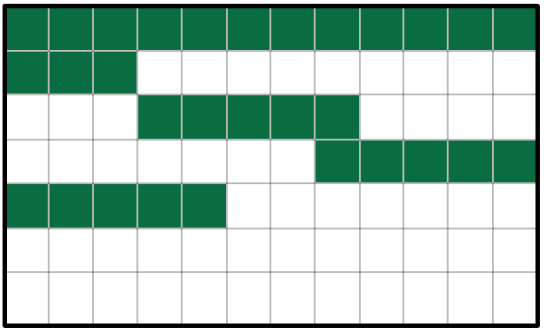
Receiving a File



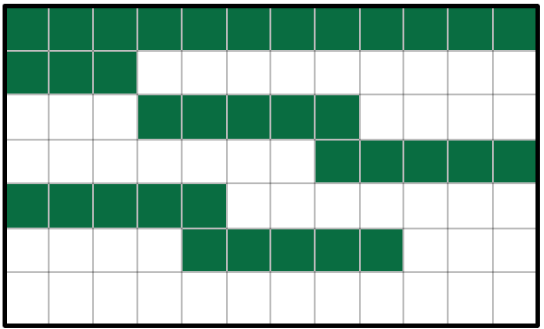
Receiving a File



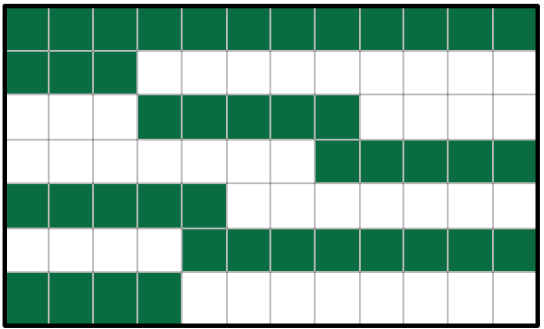
Receiving a File



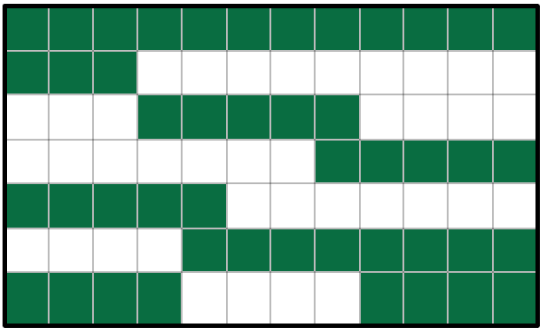
Receiving a File



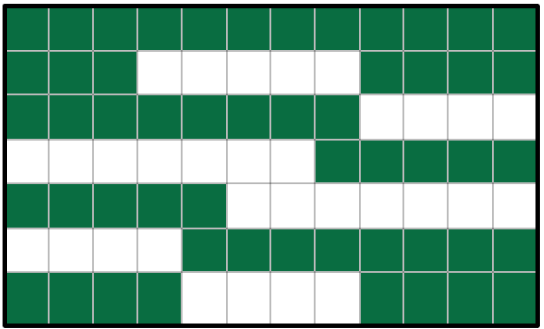
Receiving a File



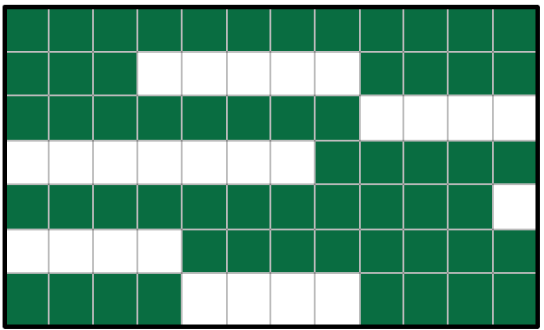
Receiving a File



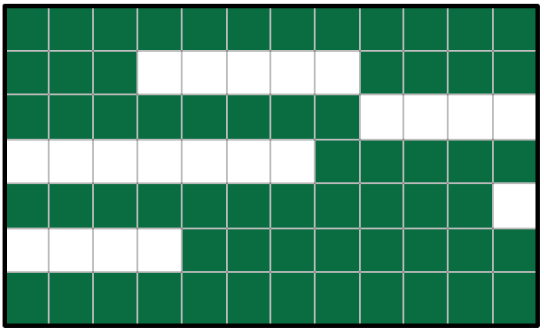
Receiving a File



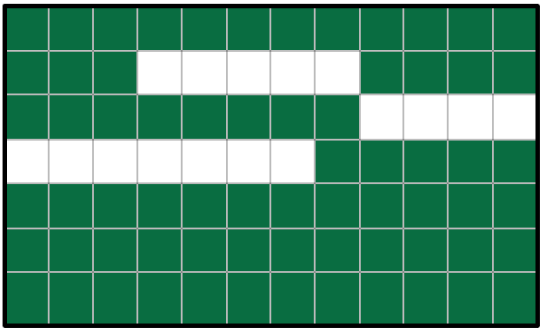
Receiving a File



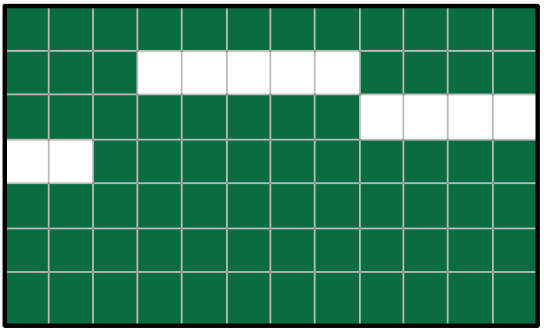
Receiving a File



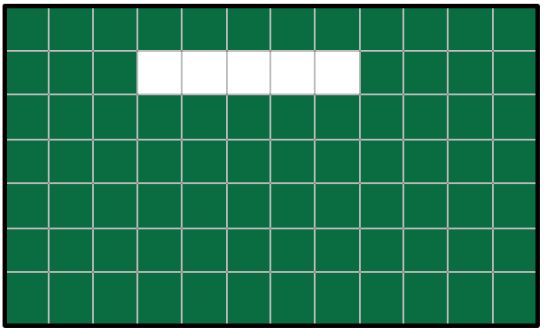
Receiving a File



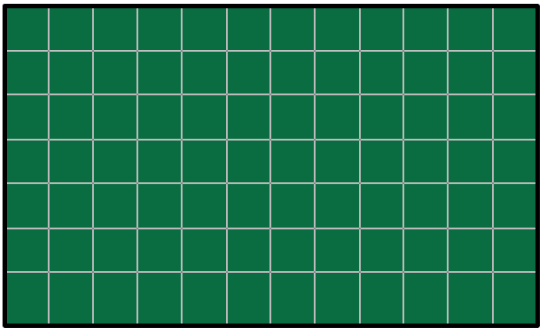
Receiving a File



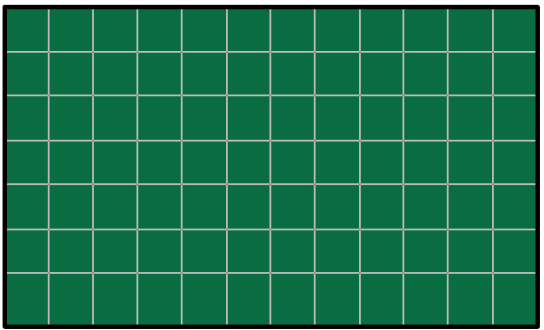
Receiving a File



Receiving a File

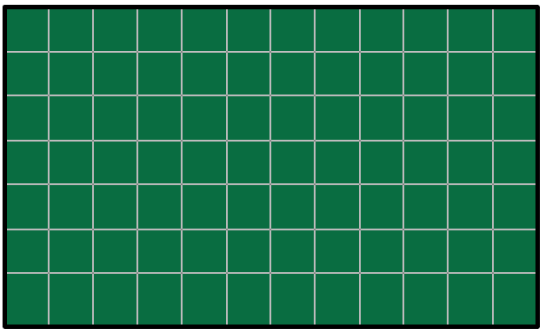


Receiving a File



Data received: 100%

Receiving a File

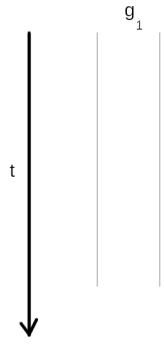


Data received: 100%

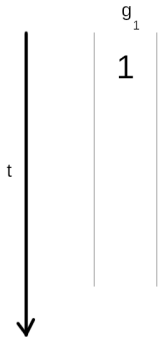
Checksum match

Reliability and Flow Control

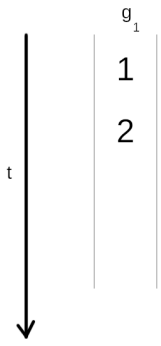
Flow Control



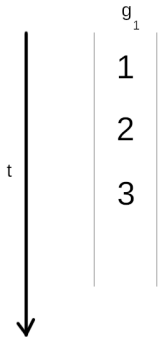
Flow Control



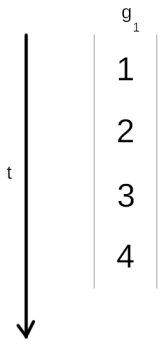
Flow Control



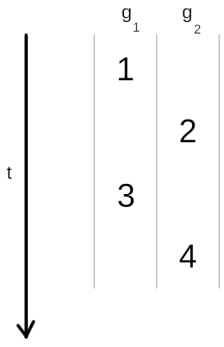
Flow Control



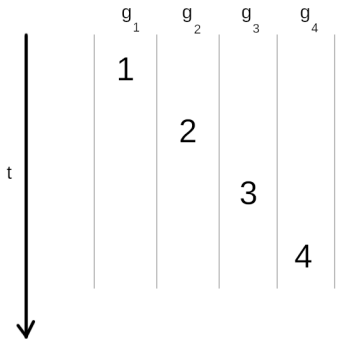
Flow Control



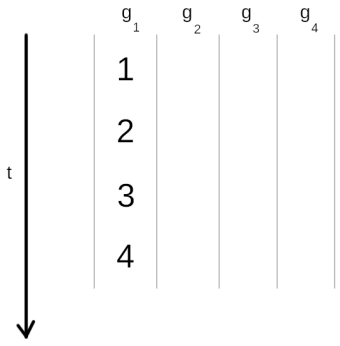
Flow Control



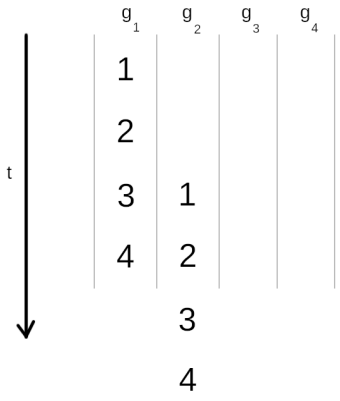
Flow Control



Reliability

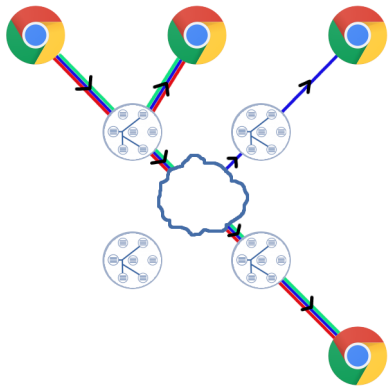


Reliability



WebRTC

WebRTC Simulcasting



HTTP/3 & QUIC

HTTP over multicast QUIC

Network Working Group L. Pardue
Internet-Draft
Intended status: Informational R. Bradbury
Expires: February 9, 2020 S. Hurst
BBC Research & Development
August 8, 2019

Hypertext Transfer Protocol (HTTP) over multicast QUIC
draft-pardue-quic-http-mcast-05

Abstract

This document specifies a profile of the QUIC protocol and the HTTP/3 mapping that facilitates the transfer of HTTP resources over multicast IP using the QUIC transport as its framing and packetisation layer. Compatibility with the QUIC protocol's syntax and semantics is maintained as far as practical and additional features are specified where this is not possible.

Librecast

Librecast

Librecast

- Developers Developers Developers

Librecast

- Developers Developers Developers
- Messaging Library

Librecast

- Developers Developers Developers
- Messaging Library
- Transitional Technology

Librecast

- Developers Developers Developers
- Messaging Library
- Transitional Technology
- Improved Routing Protocol

Librecast

- Developers Developers Developers
- Messaging Library
- Transitional Technology
- Improved Routing Protocol
- Work with FOSS projects to enable multicast everywhere

Librecast

- Developers Developers Developers
- Messaging Library
- Transitional Technology
- Improved Routing Protocol
- Work with FOSS projects to enable multicast everywhere
- Ensure new standards (eg. WebRTC, QUIC) support multicast

Brett Sheffield — Librecast Project

http://brettsheffield.com — Email: brett@librecast.net
Freenode: bacs — Twitter: @brett_sheffield / @librecast
https://librecast.net/ — github.com/librestack