DEZEMBER
2018

SCIENCE-HACK

inspired by: TUM: Junge Akademie

T · · ·

ERLEBEN, WAS VERBINDET.


TEAM MAGENTA
Setup a webserver with QUIC
and benchmark it
ERLEBEN, WAS VERBINDET.

AGENDA

01 What is Quic?

02 Chromium

03 Proto-Quic and Lib-Quic

04 Quick-Go

05 Quic-Go with Caddy

01. WHAT IS QUIC?

- New (web) communication protocol
- Developed by Google using UDP
- Good alternative to TLS, TCP, HTTP
- Doesn‘t need support from the OS
- Allways encrypted
- Might become HTTP/3.0

02. CHROMIUM

- very large and complex
- server only for integration testing, not performant
- takes very long to build (didn’t finish yesterday)

- Alternatively: tried to extract quic_server and quic_client from chromium
- Given example client and server weren’t able to connect

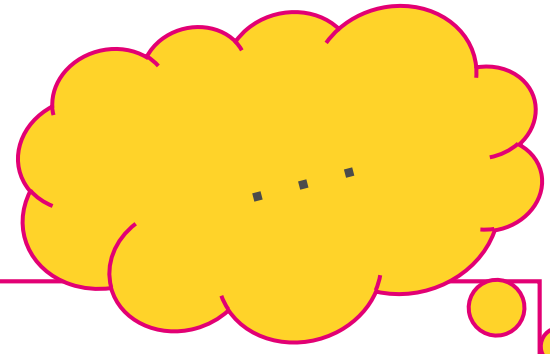
Not suitable for a first try

03. PROTO-QUIC AND LIB-QUIC

- Extracted from chromium
- Only one dependency
- Not supported anymore
- No documentation about how to use it

No real alternatives to the original

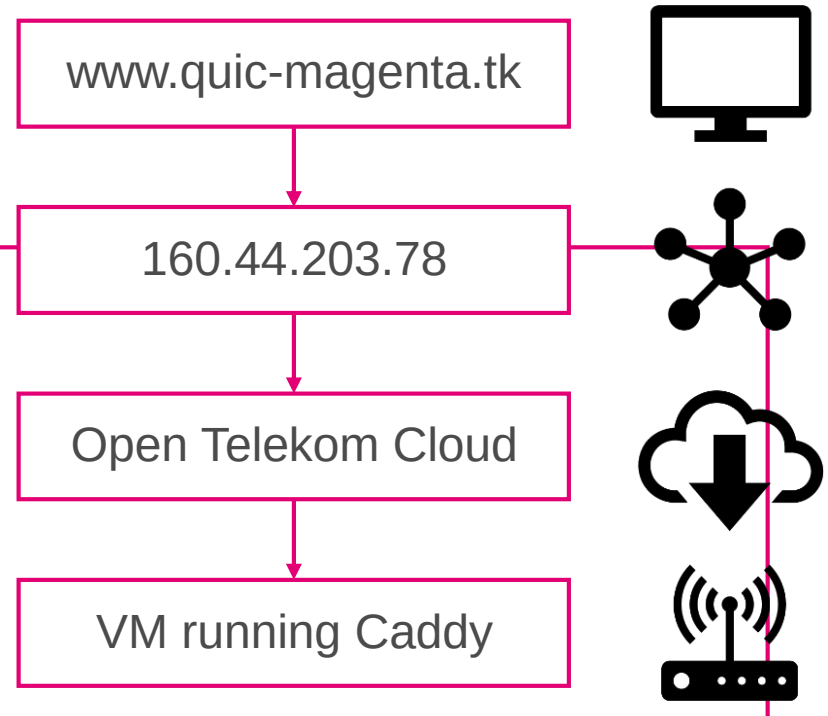
04. QUIC-GO



- Way less dependencies than chromium
- Written in Go
  - New language for us
  - Executing Go-packages is difficult
- Got the given example server running but not with Chrome
- Modification not possible due to language barriers

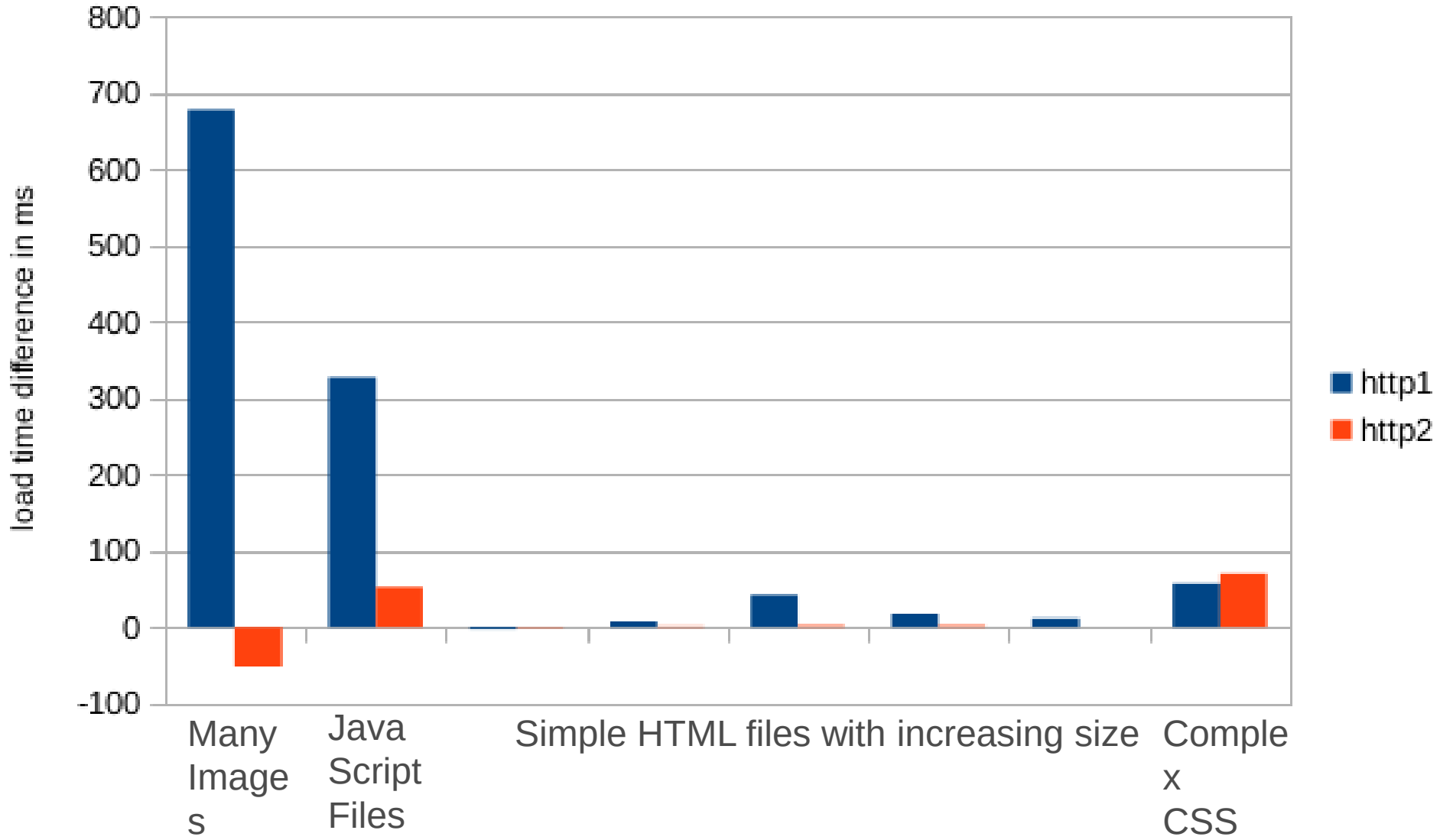
Works, but more time to learn Go needed

05. QUIC-GO WITH CADDY



- Out of the box webserver
  - Supportes Quic using Quic-Go
- Next problems: Quic only runs encrypted
  - Certificat nedded
    - Fix IP and Domain needed
- Solution:
  - Server provided by Open Telekom Cloud
  - Free domain from freenom.com for one year
  - Certificates from letsencrypt.com

It works!


800
700
600
500
400
300
200
100
0
-100
load time difference in ms
Many Images
Java Script Files
Simple HTML files with increasing size
Complex CSS and JS-Libs
http1
http2

Demo