Communication Networks

Prof. Laurent Vanbever

Online/COVID-19 Edition

Communication Networks

Spring 2020

Laurent Vanbever

nsg.ee.ethz.ch

ETH Zürich (D-ITET)

May 25 2020

Last Monday on

Communication Networks

E-mail

MX, SMTP, POP, IMAP

IPv6

the (very) beginning

E-mail

MX, SMTP, POP, IMAP

IPv6

We studied e-mail from three different perspectives

Content

Format: Header/Content

Encoding: MIME

Infrastructure/ Transmission

SMTP: Simple Mail Transfer Protocol

Infrastructure mail servers

Retrieval

POP: Post Office Protocol

IMAP:Internet Message Access Protocol

Content

Format: Header/Content

Encoding: MIME

Infrastructure/ Transmission

Retrieval

A header, in 7-bit U.S. ASCII text

Header

From: Laurent Vanbever <lvanbever@ethz.ch>

To: Tobias Buehler <buehlert@ethz.ch>

Subject: [comm-net] Exam questions

A body, also in 7-bit U.S. ASCII text

From: Laurent Vanbever <lvanbever@ethz.ch>
To: Tobias Buehler <buehlert@ethz.ch>
Subject: [comm-net] Exam questions

Body

Hi Tobias,

Here are some interesting questions…

Best,
Laurent

Email relies on 7-bit U.S. ASCII… How do you send non-English text? Binary files?

Solution

Multipurpose Internet Mail Extensions

commonly known as MIME, standardized in RFC 822

MIME defines

- additional headers for the email body
- a set of content types and subtypes
- base64 to encode binary data in ASCII

MIME relies on Base64 as binary-to-text encoding scheme

Relies on 64 characters out of the 128 ASCII characters
the most common *and* printable ones, i.e. A-Z, a-z, 0-9, +, /

Divides the bytes to be encoded into sequences of 3 bytes
each group of 3 bytes is then encoded using 4 characters

Uses padding if the last sequence is partially filled
i.e. if the |sequence| to be encoded is not a multiple of 3

Content

Infrastructure/ Transmission

Retrieval

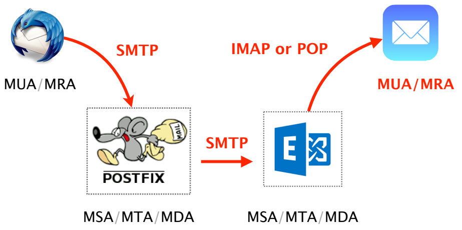
SMTP: Simple Mail Transfer Protocol

Infrastructure
mail servers

We can divide the e-mail infrastructure into five functions

Mail	User	Agent	Use to read/write emails (mail client)
Mail	Submission	Agent	Process email and forward to local MTA
Mail	Transmission	Agent	Queues, receives, sends mail to other MTAs
Mail	Delivery	Agent	Deliver email to user mailbox
Mail	Retrieval	Agent	Fetches email from user mailbox

MSA/MTA/MDA and MRA/MUA are often packaged together leading to simpler workflows



Simple Mail Transfer Protocol (SMTP) is the current standard for transmitting e-mails

SMTP is a text-based, client-server protocol
client sends the e-mail, server receives it

SMTP uses reliable data transfer
built on top of TCP (port 25 and 465 for SSL/TLS)

SMTP is a push-based protocol
sender pushes the file to the receiving server

E-mails typically go through at least 2 SMTP servers, but often way more

Separate SMTP servers for separate functions

SPAM filtering, virus scanning, data leak prevention, etc.

Separate SMTP servers that redirect messages

e.g. from lvanbever@tik.ee.ethz.ch to lvanbever@ethz.ch

Separate SMTP servers to handle mailing-list

mail is delivered to the list server and then expanded

As with most of the key Internet protocols, security is an afterthought

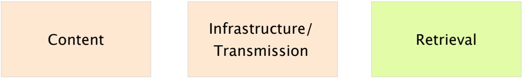
SMTP Headers

MAIL FROM:	no checks are done to verify that the sending MTA is authorized to send e-mails on behalf of that address

Email content (DATA)

From:	no checks are done to verify that the sending system is authorized to send e-mail on behalf of that address
Reply-to:	ditto

In short, *none* of the addresses in an email are typically reliable


Content

Infrastructure/ Transmission

Retrieval

POP: Post Office Protocol

IMAP: Internet Message Access Protocol

POP is a simple protocol which was designed to support users with intermittent network connectivity

POP enables e-mail users to

- retrieve e-mails locally — when connected
- view/manipulate e-mails — when disconnected

and that's pretty much it…

Unlike POP, Internet Message Access Protocol (IMAP) was designed with multiple clients in mind

Support multiple mailboxes and searches on the server

client can create, rename, move mailboxes & search on server

Access to individual MIME parts and partial fetch

client can download only the text content of an e-mail

Support multiple clients connected to one mailbox

server keep state about each message (e.g. read, replied to)

Today on Communication Networks

IPv6

Course recap

the end

So… **What?!**

IPv6

Course recap

the end

(see last week's slides)

IPv6

Course recap

So… **What?!**

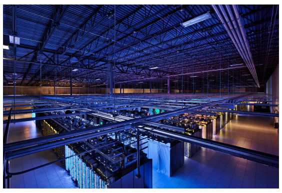
Communication Networks

So what?!

Knowledge

Understand how the Internet works and why

from your network plug…



…to Google's data-center

List any

technologies, principles, applications…

used after typing in:

> www.google.ch

and pressing enter in your browser

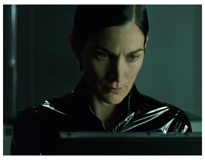
Insight

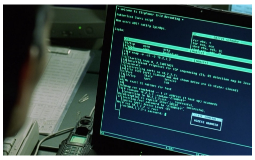
Key concepts and problems in Networking

Naming Layering Routing Reliability Sharing

Skill

Build, operate and configure networks

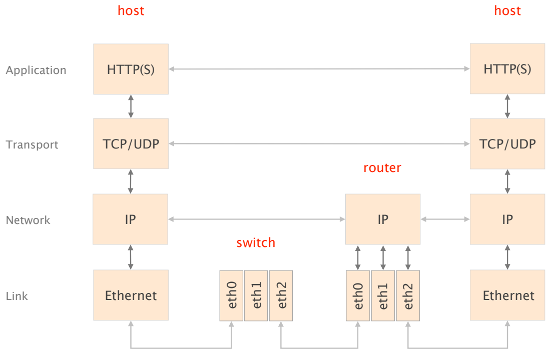




Trinity using a port scanner (`nmap`) in Matrix Reloaded™

The Internet is organized as layers, providing a set of services

	layer	service provided
L5	Application	network access
L4	Transport	end-to-end delivery (reliable or not)
L3	Network	global best-effort delivery
L2	Link	local best-effort delivery
L1	Physical	physical transfer of bits



We started with the fundamentals of routing and reliable transport

	Application	network access
L4	Transport	end-to-end delivery (reliable or not)
L3	Network	global best-effort delivery
	Link	local best-effort delivery
	Physical	physical transfer of bits

We saw three ways to compute valid routing state

	Intuition	Example
#1	Use tree-like topologies	Spanning-tree
#2	Rely on a global network view	Link-State SDN
#3	Rely on distributed computation	Distance-Vector BGP

We saw how to design a reliable transport protocol and you implemented one yourself

goals

correctness	ensure data is delivered, in order, and untouched
timeliness	minimize time until data is transferred
efficiency	optimal use of bandwidth
fairness	play well with other concurrent communications

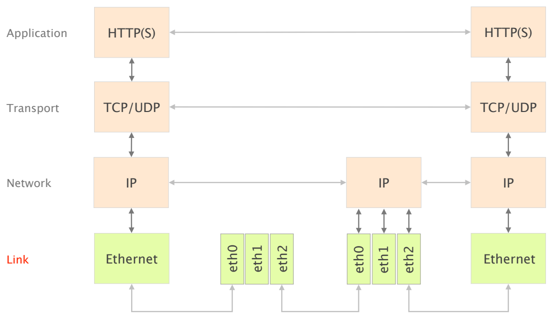
In each case, we explored the rationale behind each protocol and why they came to be

Why did the protocols end up looking like this?
minimum set of features required

What tradeoffs do they achieve?
efficiency, cost,…

When is one design more adapted than another?
packet switching *vs* circuit switching, DV *vs* LS,…

We then climbed up the layers, starting from layer 2


Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
IP
IP
IP
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet

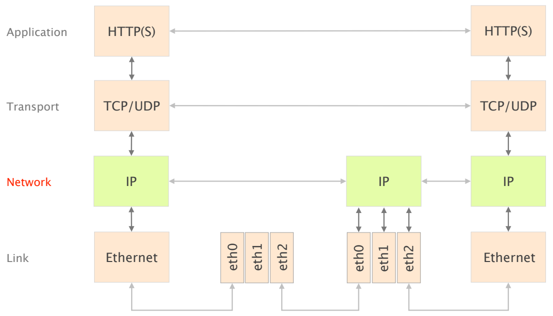
Communication Networks

Part 2: The Link Layer

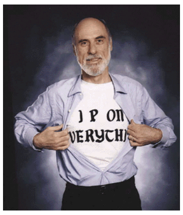
ETH

#1	What is a link?
#2	How do we identify link adapters?
#3	How do we share a network medium?
#4	What is Ethernet?
#5	How do we interconnect segments at the link layer?

We then spent multiple weeks on layer 3


Application
Transport
Network
Link
HTTP(S)
HTTP(S)
TCP/UDP
TCP/UDP
IP
IP
IP
Ethernet
eth0
eth1
eth2
eth0
eth1
eth2
Ethernet

Internet Protocol and Forwarding



source: Boardwatch Magazine

1 IP addresses
use, structure, allocation

2 IP forwarding
longest prefix match rule

3 IP header
IPv4 and IPv6, wire format

We also talked about **IPv6**

Internet routing

from here to there, and back

1 Intra-domain routing

Link-state protocols

Distance-vector protocols

2 Inter-domain routing

Path-vector protocols

Internet routing comes into two flavors:
intra- and *inter-domain* routing

inter-domain routing

intra-domain routing

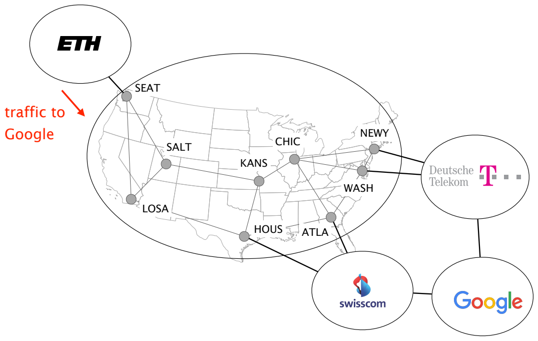
Find paths between networks

Find paths within a network

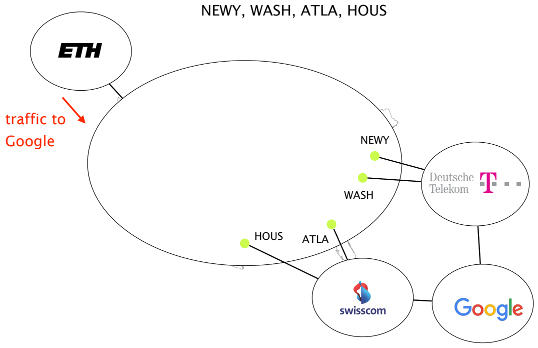
inter-domain routing

intra-domain routing

Find paths between networks



Google can be reached via
NEWY, WASH, ATLA, HOUS

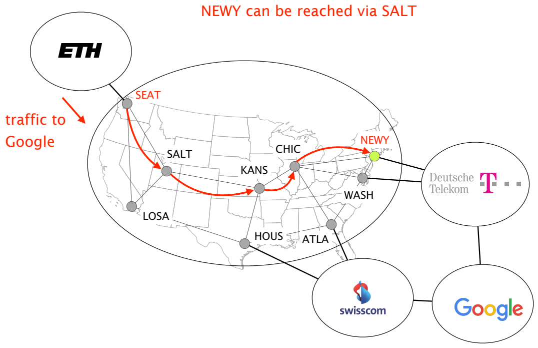


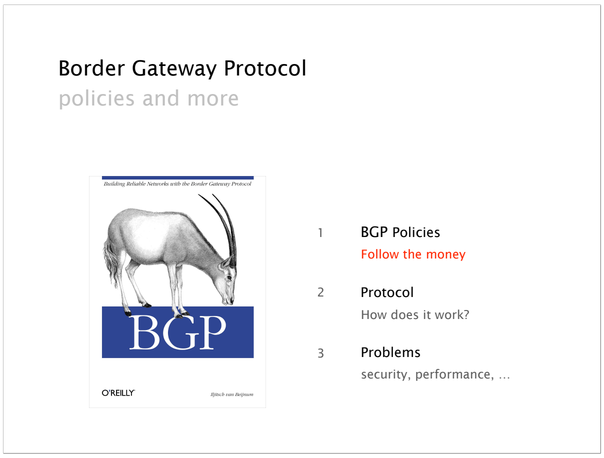
inter-domain routing

intra-domain routing

Find paths within a network

NEWY can be reached via SALT





Border Gateway Protocol
policies and more

Building Reliable Networks with the Border Gateway Protocol

BGP

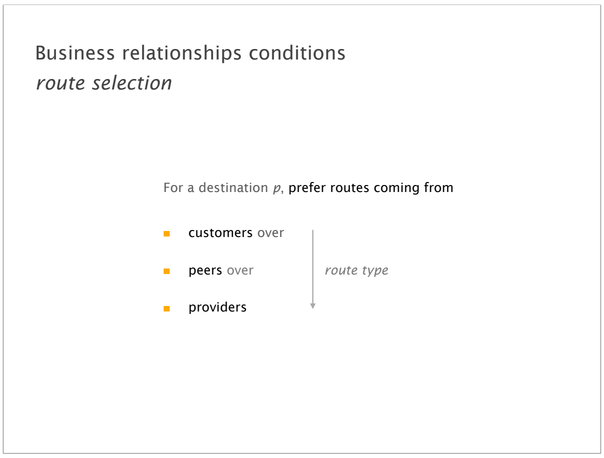
O'REILLY

Iljitsch van Beijnum

1 BGP Policies
Follow the money

2 Protocol
How does it work?

3 Problems
security, performance, …

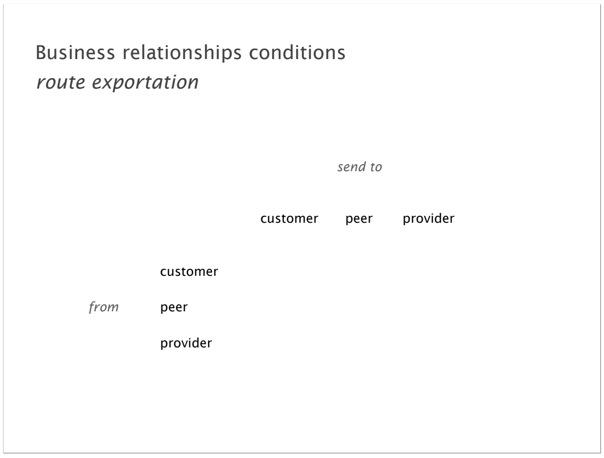


Business relationships conditions
route selection

For a destination *p*, prefer routes coming from

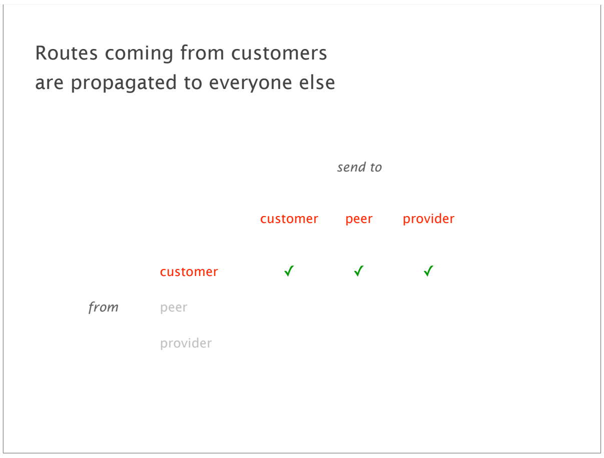
- customers over
- peers over
- providers

route type



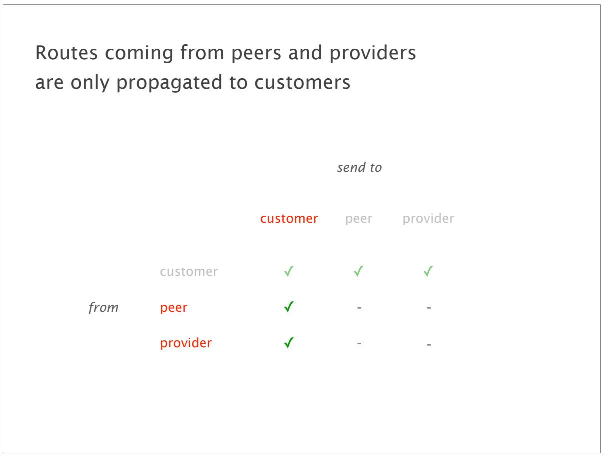
Business relationships conditions
route exportation

		send to		
		customer	peer	provider
	customer			
from	peer			
	provider			



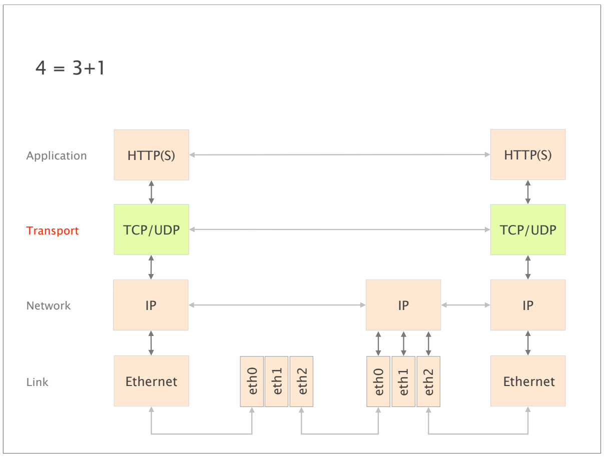
Routes coming from customers
are propagated to everyone else

		send to		
		customer	peer	provider
	customer	✓	✓	✓
from	peer			
	provider			



Routes coming from peers and providers
are only propagated to customers

		send to		
		customer	peer	provider
	customer	✓	✓	✓
from	peer	✓	-	-
	provider	✓	-	-



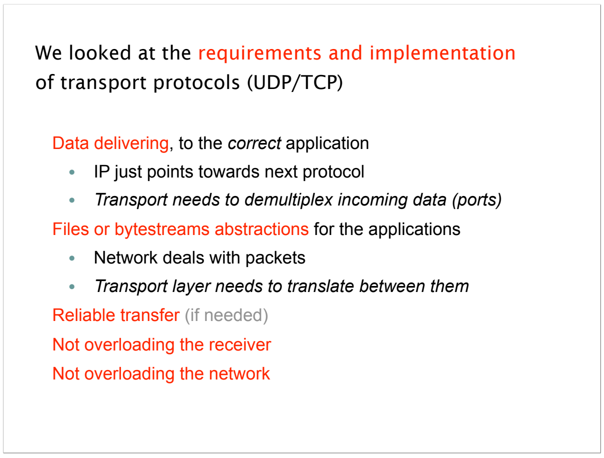
4 = 3+1

Application HTTP(S) HTTP(S)

Transport TCP/UDP TCP/UDP

Network IP IP IP

Link Ethernet eth0 eth1 eth2 eth0 eth1 eth2 Ethernet



We looked at the requirements and implementation of transport protocols (UDP/TCP)

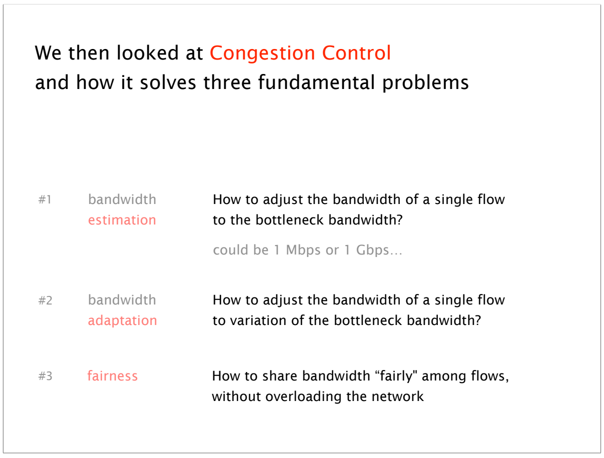
Data delivering, to the *correct* application
- IP just points towards next protocol
- *Transport needs to demultiplex incoming data (ports)*

Files or bytestreams abstractions for the applications
- Network deals with packets
- *Transport layer needs to translate between them*

Reliable transfer (if needed)

Not overloading the receiver

Not overloading the network



We then looked at Congestion Control and how it solves three fundamental problems

#1	bandwidth estimation	How to adjust the bandwidth of a single flow to the bottleneck bandwidth? could be 1 Mbps or 1 Gbps…
#2	bandwidth adaptation	How to adjust the bandwidth of a single flow to variation of the bottleneck bandwidth?
#3	fairness	How to share bandwidth "fairly" among flows, without overloading the network

… by combining two key mechanisms

detecting congestion

reacting to congestion

We then looked at what's running on top of all this …

DNS

google.ch ⟷ 172.217.16.131

Web

http://www.google.ch

We then looked at what's running on top of all this …

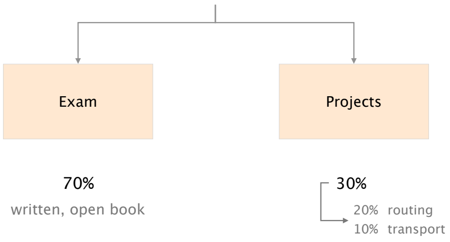
Video Streaming

HTTP-based

E-mail

MX, SMTP, POP, IMAP

Your final grade



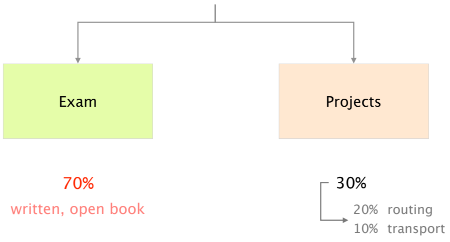
Exam

70%
written, open book

Projects

30%
20% routing
10% transport

Your final grade



Exam

70%
written, open book

Projects

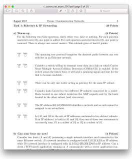
30%
20% routing
10% transport

The exam will be open book, most of the questions will be open-ended, with some multiple choices

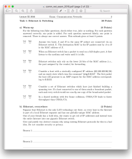
verify your understanding
of the material

Make sure you can do *all* the exercises, especially the ones in previous exams

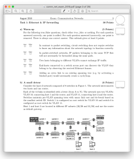
Millesime 2017



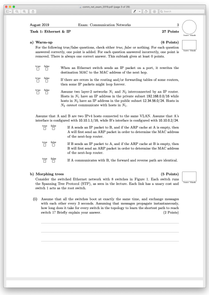
Millesime 2016



Millesime 2018



Millesime 2019



https://comm-net.ethz.ch/#tab-exam

Don't forget the assignments, they matter

No programming question — no Python at the exam

but we could ask you to describe a procedure in English

What would you change in your solution to achieve *X*?

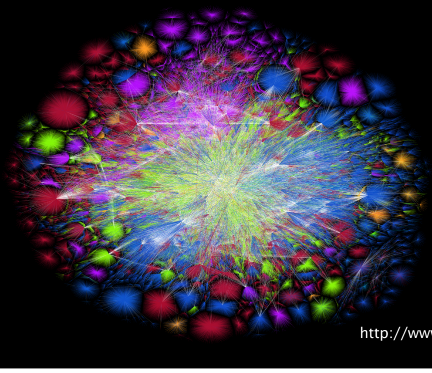
No configuration question — no Quagga at the exam

but we could ask you to describe a configuration in English

How would you enforce policy *X*?

We'll organize **another remote Q&A session closer to the exam** (details to follow)

Now you (better) understand this!



http://www.opte.org

Communication Networks

What's next?

Master-level lecture, every Fall semester

Advanced Topics in Communication Networks

Topics (not exhaustive)

- Tunneling
- Hierarchical routing
- Traffic Engineering
- Virtual Private Networks
- Quality of Service/Scheduling
- IP Multicast
- Fast Convergence
- Network virtualization
- Network programmability
- Network measurements

+ labs & a project

if you liked the routing project, you will like this lecture as well

`https://adv-net.ethz.ch/`

Master-level lecture, every semester (with D-INFK)

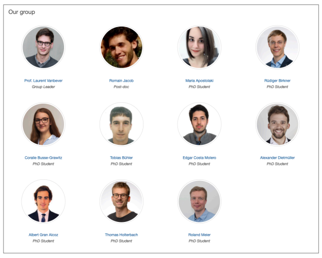
Seminar in Communication Networks

- Understand recent research result
- Read, present, and critique research papers
- Identify new research opportunities

`https://nsg.ee.ethz.ch/courses/`

Consider doing one of your theses with our group!

bachelor, semester or master



Our group

Prof. Laurent Vanbever
Group Leader

Romain Jacob
Post-doc

Maria Apostolaki
PhD Student

Rüdiger Birkner
PhD Student

Coralie Busse-Grawitz
PhD Student

Tobias Bühler
PhD Student

Edgar Costa Molero
PhD Student

Alexander Dietmüller
PhD Student

Albert Gran Alcoz
PhD Student

Thomas Holterbach
PhD Student

Roland Meier
PhD Student

`https://nsg.ee.ethz.ch/theses/`

Enjoy a **well-deserved** break!

Communication Networks

Spring 2020

Laurent Vanbever
nsg.ee.ethz.ch

ETH Zürich (D-ITET)
May 25 2020