NTNU – Trondheim
Norwegian University of
Science and Technology

Security in Computer Networks (2)

TTM4200 – Computer Networks

individual Readiness Assurance Test

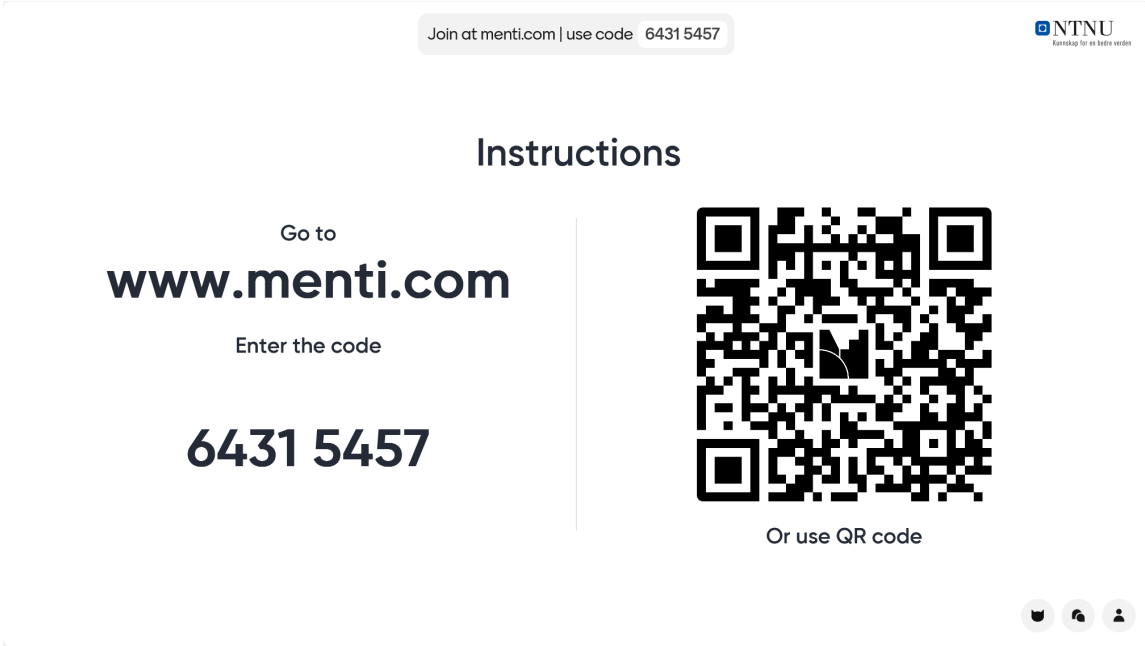
- Find the right answer **alone.**
- Choose the answer that fits **best.**
- **One** handwritten A4 page as helping material.

team Readiness Assurance Test

- Find the right answer **in teams**.
- Choose the answer that fits **best.**
- **One** handwritten A4 page as helping material.

Time for a break

we restart at 11:15



Exam question #1

1. Name three different aspects / goals of security in computer networks and briefly explain their meaning.

2. Name the two main classes of cryptographic algorithms that were covered in the course. Briefly discuss their differences and name one algorithm from each class.

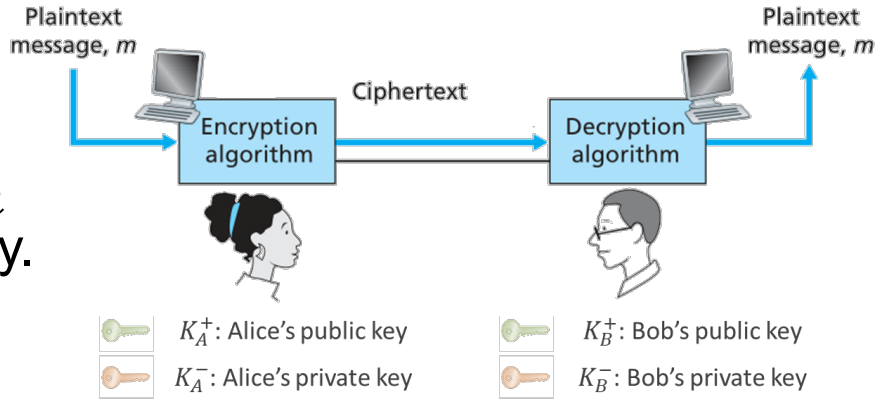
10m

Solving Exam question #1

1. Name three different aspects / goals of security in computer networks and briefly explain their meaning.
   - Confidentiality / secrecy: only sender and intended receiver should be able to read message.
   - Message integrity: sender and receiver want to be sure that message has not been altered during delivery without detection.
   - Authentication: mutual identity confirmation of sender and receiver.
   - Access / availability of services.

2. Name the two main classes of cryptographic algorithms that were covered in the course. Briefly discuss their differences and name one algorithm from each class.
   - Symmetric: stream / substitution ciphers, block ciphers, DES, AES. Requires sender and receiver(s) to know shared secret.
   - Asymmetric / public-key: DH, RSA. Communication parties have public and private keys. Knowing public key of recipient is enough to negotiate shared secret.

Exam question #1 (cntd.)



3. Consider the graphic. Suppose Alice wants to encrypt a message m to Bob using public key cryptography. Assume that Alice only cares about confidentiality.

   a) How does Alice generate the ciphertext?

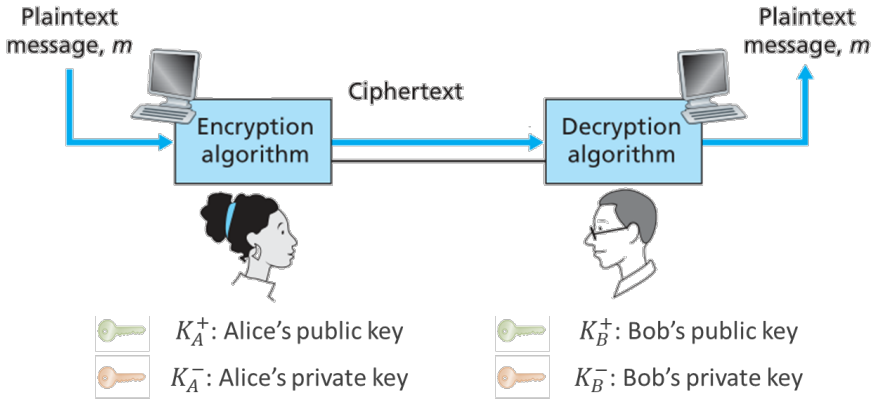
   b) How does Bob recover the original message?

   c) What kind of attack does Alice prevent by encrypting the message?

   d) Name and outline one kind of attack that could still be performed by an attacker who sits between Alice and Bob.

10m

Solving Exam q. #1 (cntd.)



3. Consider the graphic. Suppose Alice wants to encrypt a message m to Bob using public key cryptography. Assume that Alice only cares about confidentiality.

   a) How does Alice generate the ciphertext?
   - $c = K_B^+(m)$.

   b) How does Bob recover the original message?
   - $m = K_B^-(c)$.

   c) What kind of attack does Alice prevent by encrypting the message?
   - Eavesdropping: no one except Bob can recover the original message within a reasonable time frame.

   d) Name and outline one kind of attack that could still be performed by an attacker who sits between Alice and Bob.
   - MITM / impersonation: delete message, forge alternative message.

Exam question #2

1. Name three different types of attacks a malicious actor can perform and briefly explain them.

2. Name the two main classes of cryptographic algorithms that were covered in the course. Briefly discuss their differences in complexity and name one algorithm from each class.

10m

Solving Exam question #2

1. Name three different types of attacks a malicious actor can perform and briefly explain them.
   - Eavesdrop: intercept / listen in on messages.
   - Man-in-the-middle: insert, modify, or delete messages.
   - Impersonation: fake header fields / craft packets.
   - Hijacking of connections: replace one end of communication with themselves.
   - Denial of service: disrupt availability of service, e.g., by consuming all its resources.

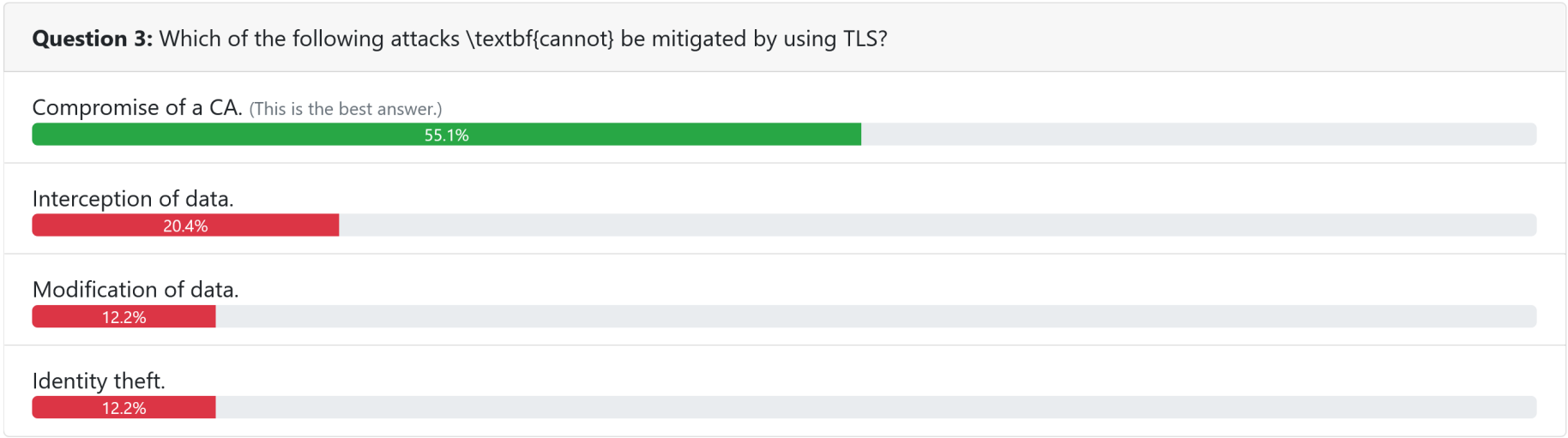
2. Name the two main classes of cryptographic algorithms that were covered in the course. Briefly discuss their differences in terms of complexity and name one algorithm from each class.
   - Symmetric: stream / substitution ciphers, block ciphers, DES, AES. Fast encryption and decryption but more keys are needed (N*(N-1)/2.
   - Asymmetric / public-key: DH, RSA. Exponentiation is expensive compared to symmetric encryption, but fewer keys are needed (2N).

Enough for today

Welcome

We start at 10:15

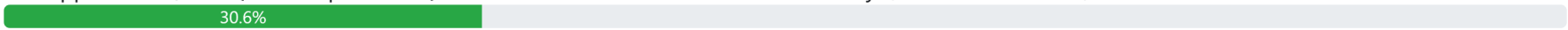
RAT revision



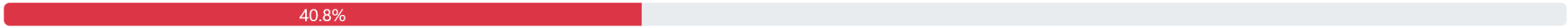
RAT revision

Question 5: Suppose that an application which uses standard TCP sockets for network-based communication is used with a VPN. Which of the following statements is \textbf{false}?

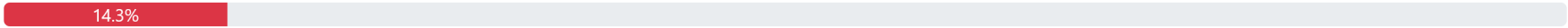
The application \textbf{must import IPsec} libraries in order to communicate securely. (This is the best answer.)



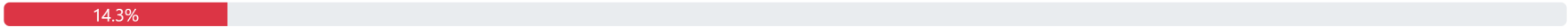
The application \textbf{does not need to implement IPsec} libraries in order to communicate securely.



The application can be used \textbf{without changes}, and devices on the public Internet are \textbf{not able} to easily determine that the application uses TCP.



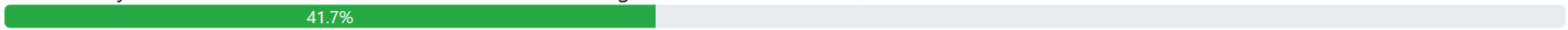
The application \textbf{does not} require TLS sockets in order to work securely.



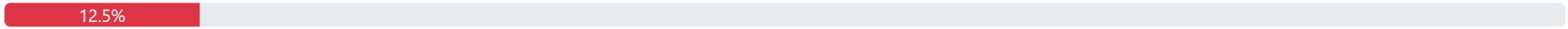
RAT revision

Question 9: Suppose Bob initiates a TLS connection to Trudy who is pretending to be Alice. During the handshake, Trudy sends Bob Alice's certificate. In what step of the TLS handshake algorithm will Bob discover that he is not communicating with Alice? Assume that Trudy \textbf{does not} have Alice's private key.

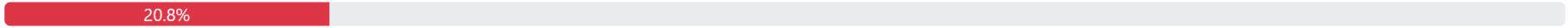
When Trudy sends Bob a HMAC of all the handshake messages. (This is the best answer.)



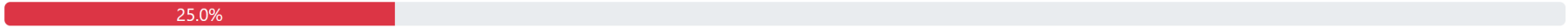
When Bob computes a HMAC of all the handshake messages to send to Trudy.



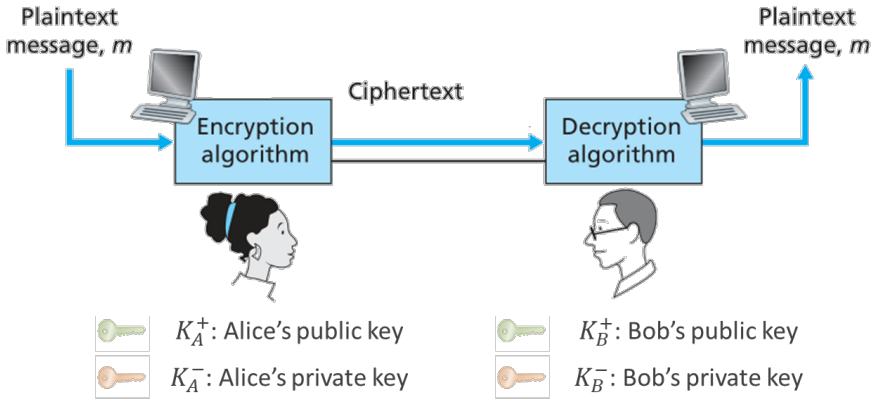
When Trudy sends to Bob Alice's certificate because it will be invalid.



When Bob tries to generate and encrypt the Pre-Master Secret (PMS) with the extracted public key.



Exam question #2 (cntd.)



3. Consider the graphic. Suppose Alice wants to digitally sign a message m, so that Bob can be sure that it comes from Alice and has not been altered on the way to him.

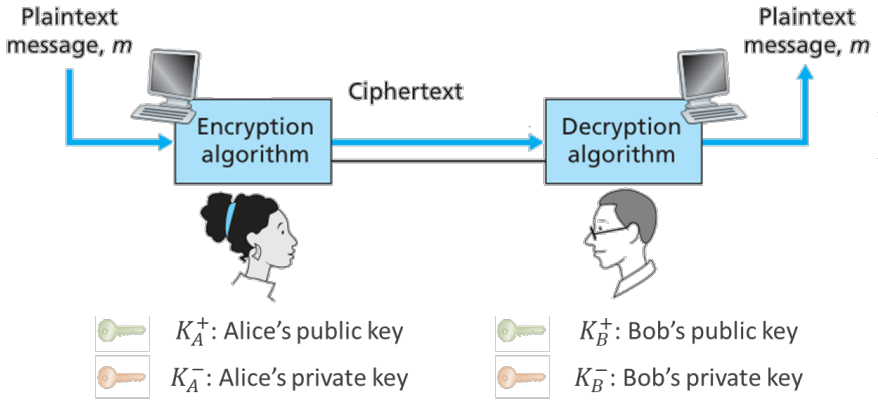
   a) What are the two properties that Alice would like to achieve called?

   b) How does Alice generate the signed message?

   c) How can Bob verify that the message has not been altered?

10m

Solving Exam q. #2 (cntd.)



3. Consider the graphic. Suppose Alice wants to digitally sign a message m, so that Bob can be sure that it comes from Alice and has not been altered on the way to him.

   a) What are the two properties that Alice would like to achieve called?
   - Message integrity and sender authentication.

   b) How does Alice generate the signed message?
   - $s = (m,\ K_A^-(H(m)))$. (hash mandatory; otherwise, MITM can truncate encrypted version and use public key to generate corresponding m').

   c) How can Bob verify that the message has not been altered?
   - Check $\mathrm{H}(m) == K_A^+(s[1])$.

Review Problem 18 (*pg. 715)*

Suppose Alice wants to send an e-mail to Bob. Bob has a public-private key pair $(\mathrm{K}_B^+, \mathrm{K}_B^-)$, and Alice has Bob's certificate. However, **Alice does not have a public-private key pair**. Alice and Bob (and the entire world) share the same hash function $\mathrm{H}(\cdot)$.

a) In this situation, is it possible to design a scheme so that Bob can verify that Alice created the message? If so, show how with a block diagram for Alice and Bob.

b) Is it possible to design a scheme that provides confidentiality for sending the message from Alice to Bob? If so, show how with a block diagram for Alice and Bob.

10m

Solving Problem 18 (*pg. 715*)

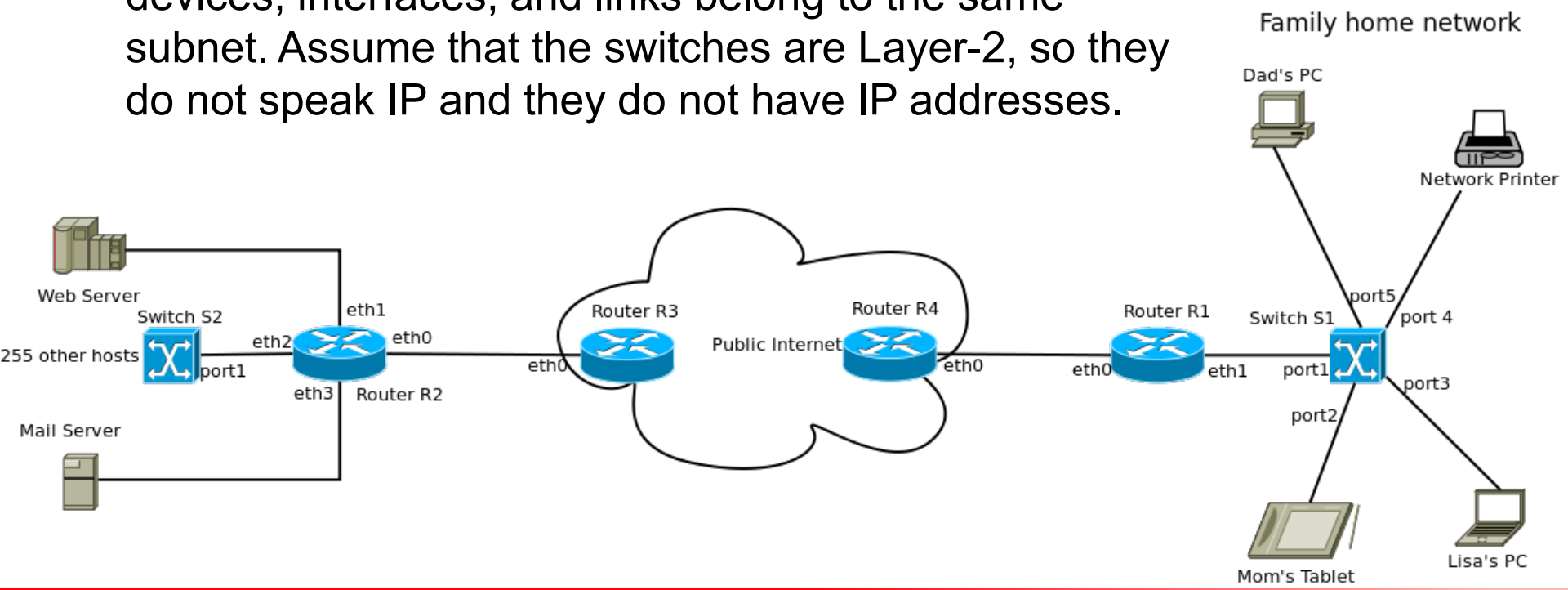
a) No, without a public-private key pair or a pre-shared secret, Bob cannot verify that Alice created the message.

b) Yes, Alice simply encrypts the message with Bob's public key and sends the encrypted message to Bob.

(Alice) $\mathrm{m} \rightarrow \mathrm{K}_B^+(\mathrm{m}) \rightarrow\rightarrow\rightarrow$ (Bob) $\mathrm{K}_B^-\left(\mathrm{K}_B^+(\mathrm{m})\right) \rightarrow \mathrm{m}$

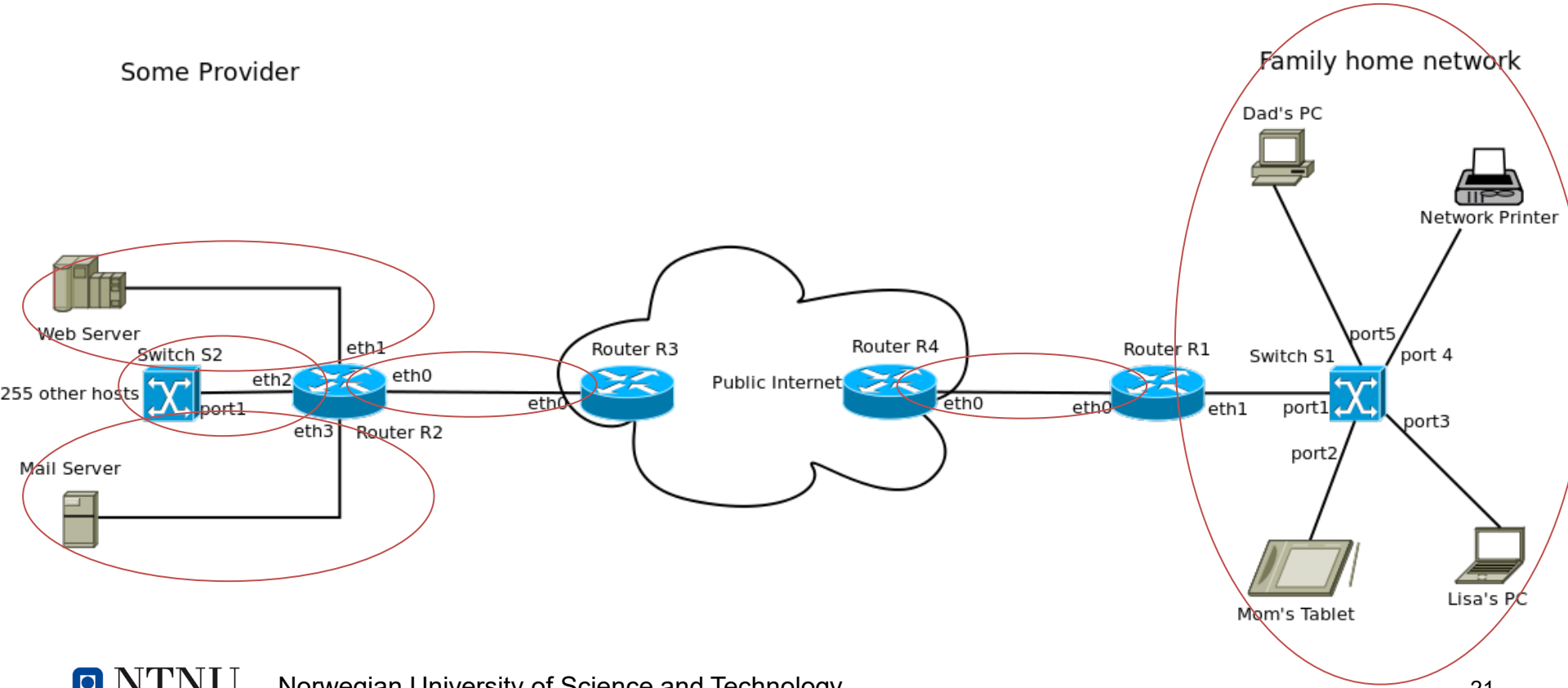
Networks and subnets pt. 1

- **Identify each sub-network (subnet)**, i.e., mark which devices, interfaces, and links belong to the same subnet. Assume that the switches are Layer-2, so they do not speak IP and they do not have IP addresses.



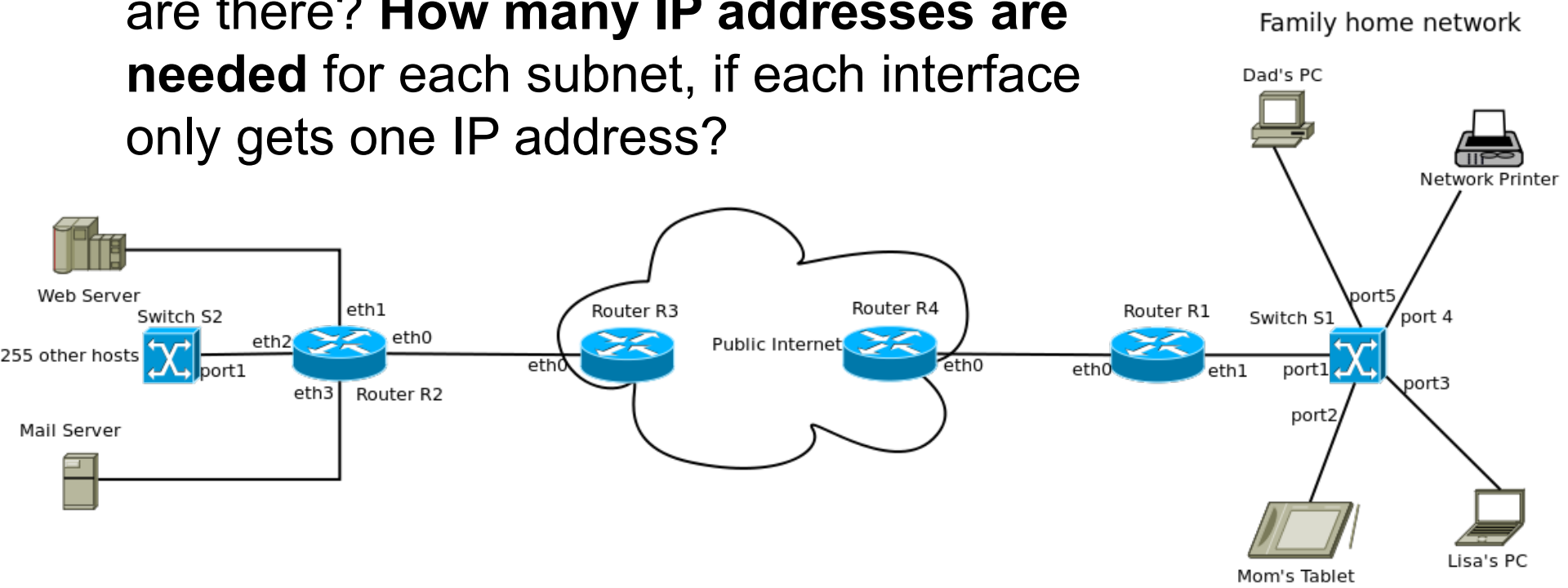
5m

Solution


Some Provider
Family home network
Dad's PC
Network Printer
Web Server
Switch S2
255 other hosts
port1
eth2
eth1
eth3
Router R2
eth0
eth0
Router R3
Public Internet
Router R4
eth0
eth0
Router R1
eth1
Switch S1
port5
port 4
port1
port3
port2
Mail Server
Mom's Tablet
Lisa's PC

Networks and subnets pt. 2

- Within each subnet, **how many interfaces** are there? **How many IP addresses are needed** for each subnet, if each interface only gets one IP address?



5m

Solution

- Subnet 1: 2 interfaces (Web server, Router R2 eth1)
- Subnet 2: 256 interfaces (each of the 255 hosts, and Router R2 eth2)
- Subnet 3: 2 interfaces (Mail server, Router R2 eth3)
- Subnet 4: 2 interfaces (R2 eth0, R3 eth0)
- Subnet 5: 2 interfaces (R4 eth0, R1 eth0)
- Subnet 6: 5 interfaces (Router R1 eth1, Dad's PC, Mom's Tablet, Lisa's PC, Network Printer)

- For each subnet: need an additional 2 addresses, one to address the entire network, and one as broadcast address

- Subnet 1: need 4 IP addresses
- Subnet 2: need 258 IP addresses
- Subnet 3: need 4 IP addresses
- Subnet 4: Need 4 IP addresses
- Subnet 5: Need 4 IP addresses
- Subnet 6: Need 7 IP addresses

Break

We continue at 11:15

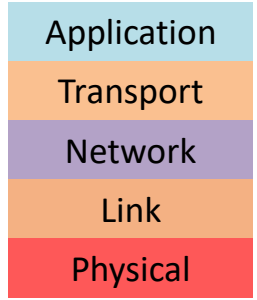
VIRTUALIZATION

Virtualization of Networks

- Virtualization of resources
  - Powerful abstraction in systems engineering
- Computing examples
  - Virtual memory, virtual devices, virtual servers, cloud service
  - Virtual machines: e.g., Java
- Layering of abstractions
  - Don’t get lost in the details of lower layers, only deal with lower layers abstractly
- The internet itself uses this principle on many occasions
  - E.g. network layers

What is an Overlay?

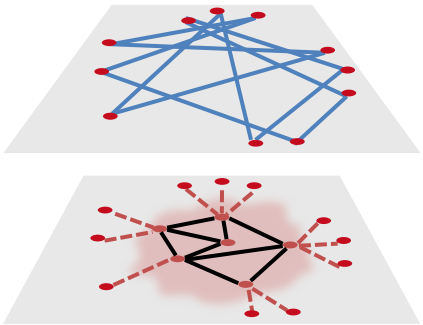
- Dedicated virtual network
  - On top of a (physical) network
  - Consists of overlay nodes (end systems)
  - Nodes are connected by virtual/logical links
    ➔Tunnels
  - Nodes communicate via logical paths
    ➔Single overlay hop may traverse several physical links
  - Mostly focused on application level



Slides adapted from Tobias Hosfeld

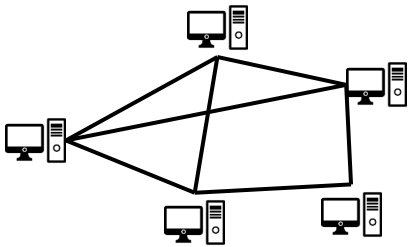
What is an Overlay?

- Overlay
  - **Logical network** above the **physical network** structure
  - All P2P networks are overlay networks
- Different semantics than the underlying transport network
  - E.g. neighborhood, address space, and routing

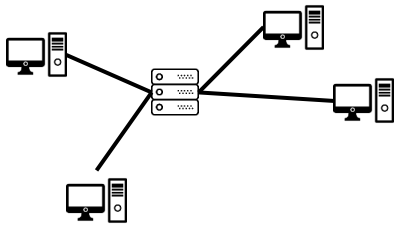


Example: P2P Networks

- “Peer”
  - Peer means *equal*, from Latin word *par*
  - Peer-to-Peer (P2P): Relationship between equal partners
- P2P Network
  - Communication network between (end user) devices in the Internet
  - Peers share resources, like hard disk or network bandwidth
  - P2P network is **not** client-server network



Peer-to-Peer



Client-Server

Use Cases for P2P

- Content distribution platforms/ file sharing
  - Index realized with structured overlays
  - E.g. BitTorrent...
  - Server is not a Bottleneck anymore
  - ➔ Cooperative download can be faster than download from server
- Providing QoS edge-based applications
  - Re-routing on application layer
  - E.g. the original version of Skype
- Streaming
  - To decrease delay and load on main server

P2P: Pro and Cons

- Advantages of P2P
  - Low costs: No powerful, expensive server required
  - High reliability: No single point of failure
  - High availability: Multiple copies of contents available
  - Scalability: The more peers join the P2P network, the more resources are available

  ➔ **P2P network scales**
  - Ease of administration: Self-organization, no control instance

P2P: Pro and Cons

- Disadvantages of P2P
  - Churn: Peers may arbitrarily join or leave the network
  - Higher efforts: Searching for contents, maintenance of logical network structure, security, routing, etc.

  ➔ Impact on performance
  - Programming: Debugging in case of errors
  - Trust: Sharing resources with anonymous users

Real-World Overlay

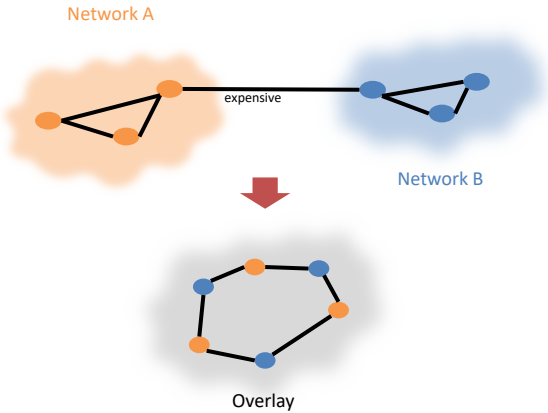
- Social experiment in 1967 by Milgram
- Human society as network of social connections
  - How large is the distance between any two people?
- Experiment
  - Random people are selected to relay a letter to someone they do not know
  - Only allowed to send letter to people they know
- Results
  - Random people in the US are only separated by 6 hops from any other person

Functionality and Requirements of Overlays

- Functionality
  - Connectivity maintenance
  - Different kinds of routing: Semantic, group-based, topology-aware, mobility supporting

- Main requirements
  - Efficiency
  - Scalability
  - Fault-tolerance
  - Load-balancing

Features of Overlays



- Enables deployment of distributed services
  - No need to know the physical network
- Typical:
  - Large-scale environment
  - No administration
  - ➔ Self-organizing structures
- Problems:
  - Is the shortest connection in overlay not shortest connection in physical layer
  - ➔ Leads to inefficient usage of network resources
  - ➔ Expensive links are used often
  - Solution: Topology-aware overlays

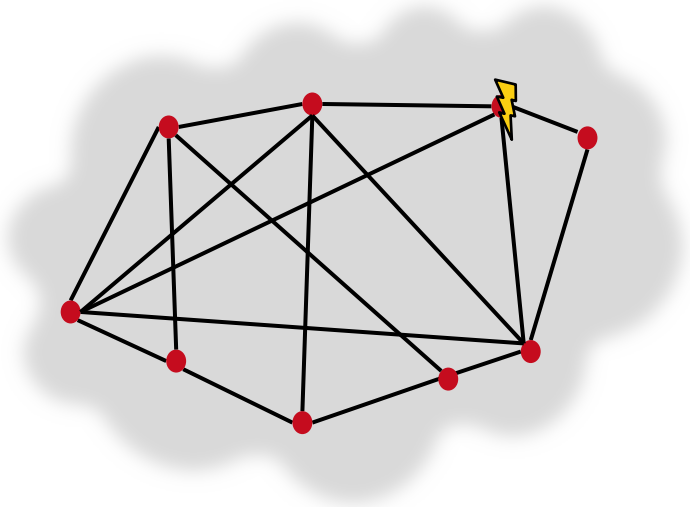
Overlays: Self-Organizing Structures

- Node failure/leave

➔ Self-repairing structure

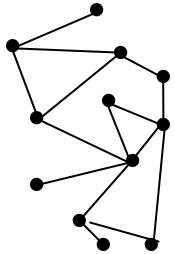
- Node joins

➔ Self-adapting structure

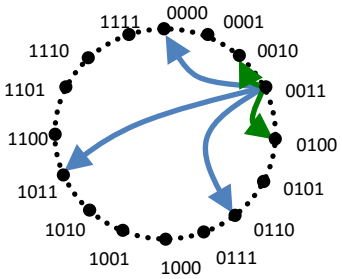


Overlay Structure

- Unstructured overlays
  - Arbitrary topology
  - Randomly connected peers (in P2P)
  - May comprise inherently self-organization
  - Automatic re-wiring of upon link or node failure
  - *Preferential attachment / “Small World”*
  - Mesh topologies and multiple meshes



- Structured overlays
  - Predefined data structures specify in a generic way the communication relationship among peers
  - Detailed wiring is determined during operation
  - Trees and rings (multiple instances possible)



Why use Overlays?

- Anonymization (“mix”) networks, e.g., Tor

- Need to obtain improved services from the Internet
  - Not supported services (multicast, anycast)
  - Not efficiently supported services (QoS, routing)

- To evolve the Internet
  - Deploy new technologies

Overlays: Pros and Cons

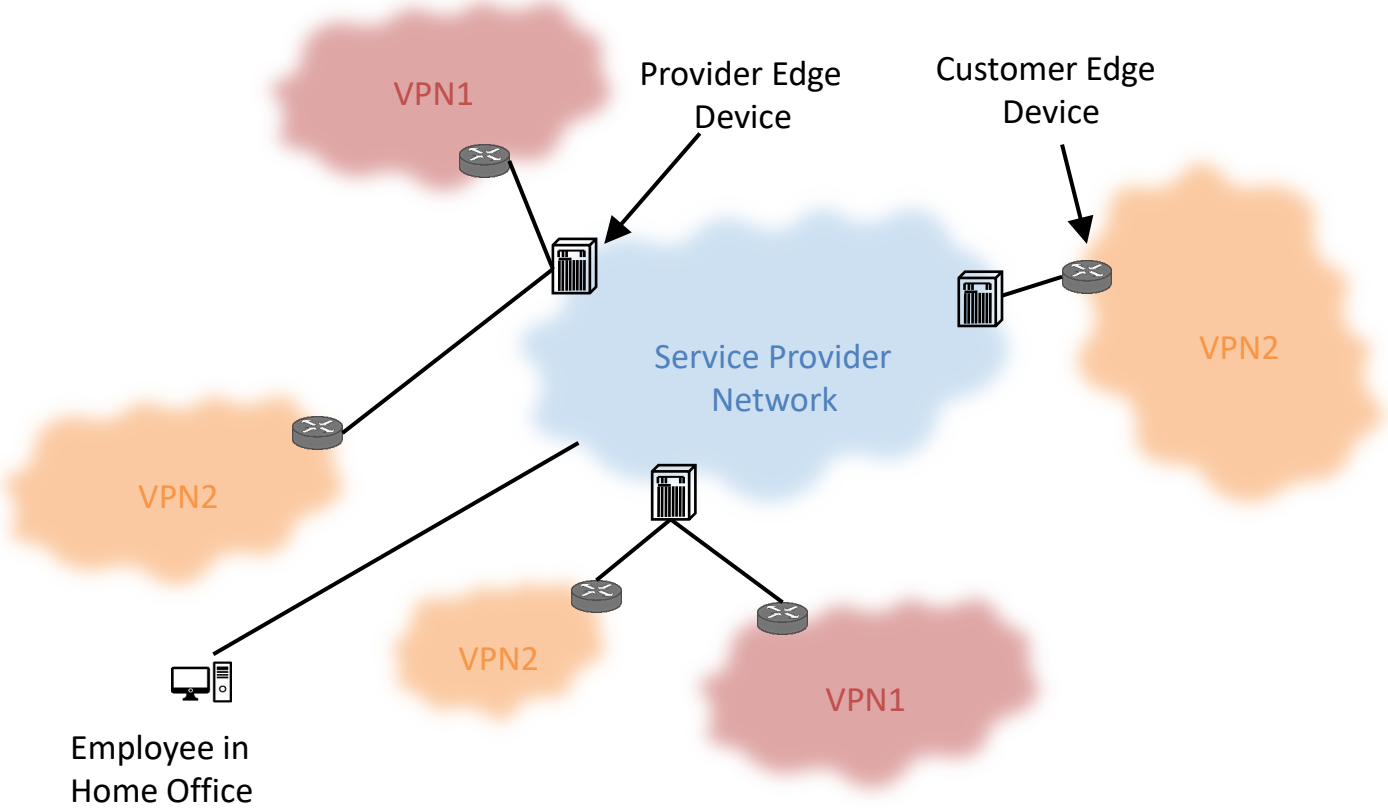
- Advantages
  - Flexible design with respect to topology, maintenance, protocol
  - Alternative solution to overcome the limitations of the Internet
  - ➔ i.e. the lack of extensibility
  - ➔ No costly modifications necessary
  - Certain protocols have a low acceptance because they require modification of all routers in the network (e.g. multicast)
  - ➔ “Good” alternative
  - May perform better than underlying physical network (e.g. Skype)

- Disadvantages
  - Longer latency
  - Inefficient usage of network resources
  - Setup and maintenance traffic

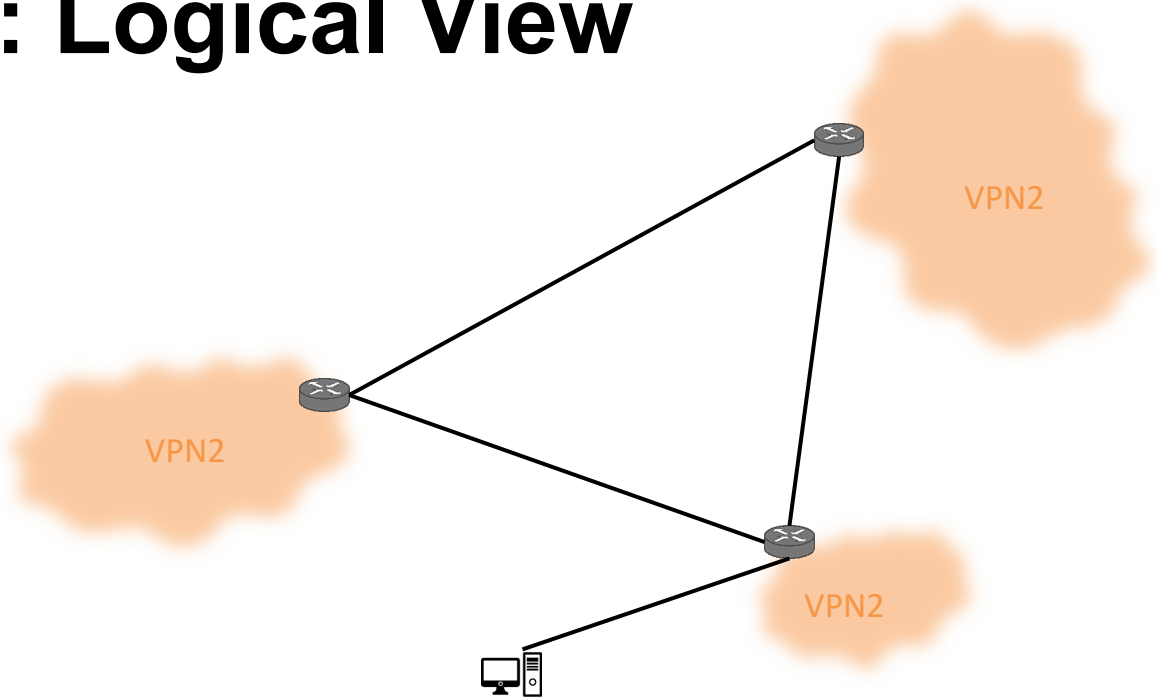
Virtual Private Networks

- Virtual Private Networks (VPN)
  - Networks perceived as being private networks by customers using them…
  - …, but built over shared infrastructure owned by service provider
- Service provider (SP) infrastructure
  - Backbone
  - Provider edge devices
- Customer
  - Customer edge devices
  - Communicate over shared backbone

Virtual Private Networks



VPN: Logical View



Types of VPN

- Leased-line VPN
  - PEs interconnected via static virtual channels
  - Connection is permanent, service is always available
  - Drawbacks: Configuration costs, maintenance by SP (long time, much manpower)

- Network based VPN
  - Customer's routers connect to SP routers
  - SP routers maintain separate (independent) IP contexts for each VPN
  - Sites can use private addressing
  - Traffic from one VPN can not be injected into another
  - Single provider edge device contains multiple virtual routers
    → Can handle different VPNs separately
  - Tunnel encapsulation/decapsulation performed in provider edge equipment

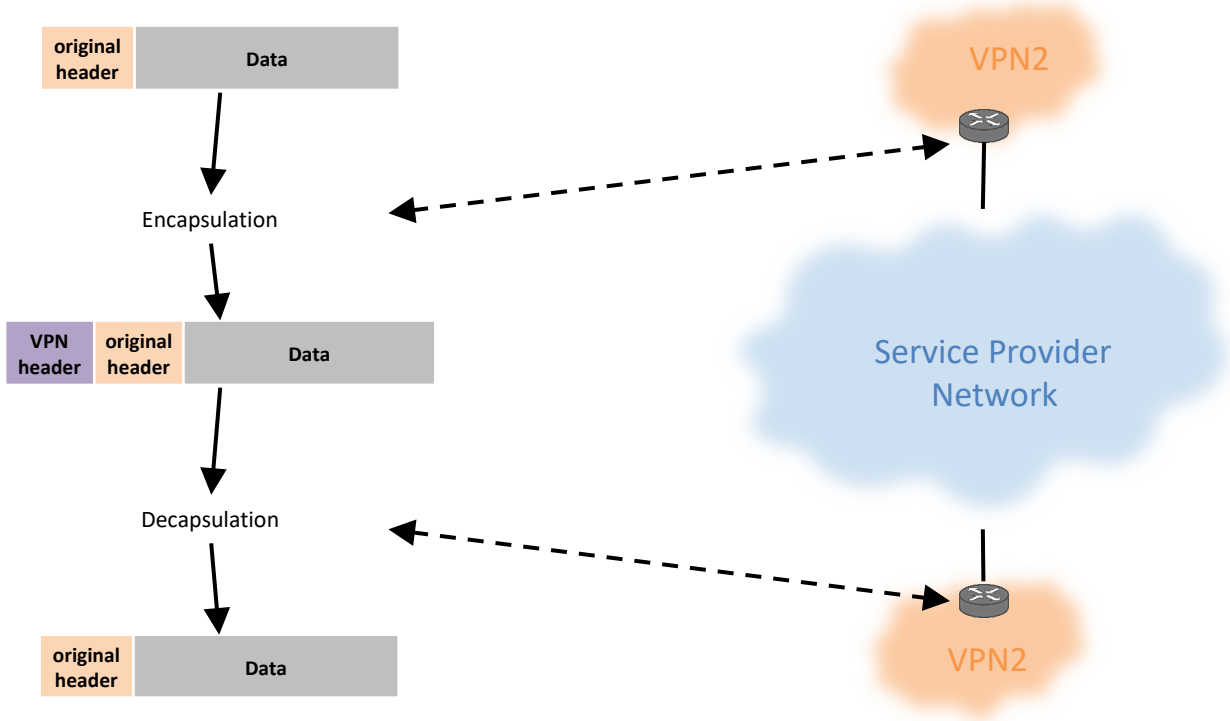
Types of VPN

- Customer edge based VPN
  - Customer sites interconnected via tunnels
  - Tunnels encrypted typically
  - SP treats VPN packets like all other packets
  - All VPN functions implemented by customer
  - Drawbacks: Expertise by customer to acquire, configure, manage VPN

Customer Edge Based VPN

- Packets for a different part of the private network packaged in VPN packets
  - Responsibility of router
- Payload of VPN Packets is encrypted
- VPN packets are send over the Internet to other part of private network
- Router there decrypts packet and forwards in on a lower layer
  - E.g. if payload is an IP Packet: VPN packet is received, Payload forwarded to destination on layer 3

Tunneling


original header
Data
Encapsulation
VPN header
original header
Data
Decapsulation
original header
Data
VPN2
Service Provider Network
VPN2

Summary

- Virtualization of Networks
- Overlays
  - Definition
  - Example P2P: Definition, use cases, pro and cons
  - Functionality, requirements, features
  - Different structures
  - Pro and cons
- Virtual private networks
  - Types of VPN
  - Tunneling

Enabling Alternative Architectures

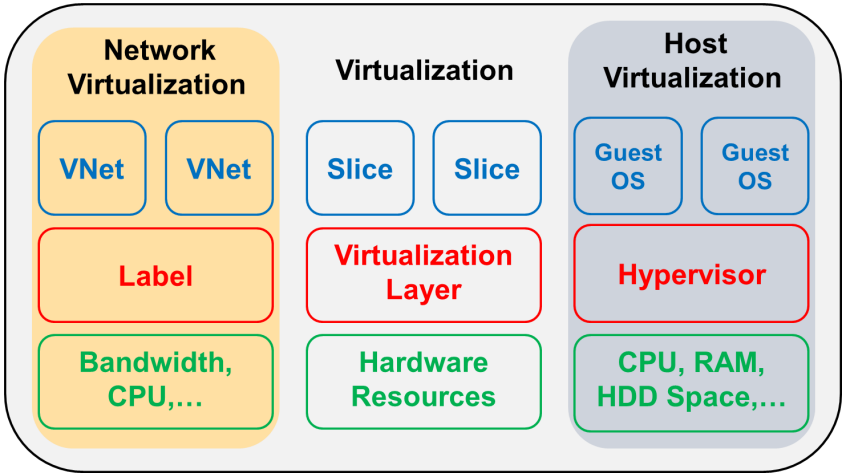
NETWORK VIRTUALIZATION & SDN

NTNU Norwegian University of Science and Technology

Network Virtualization

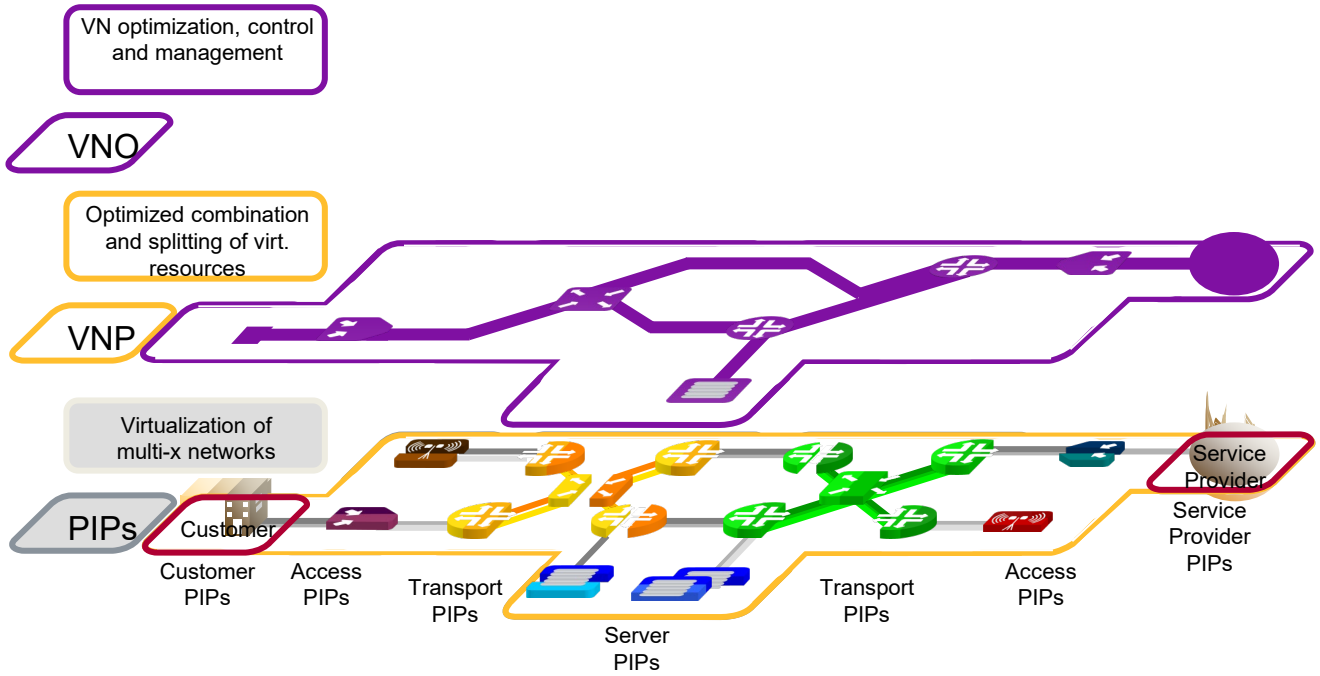
- Virtual networks offer
  - Resource isolation
  - Multiple networks in parallel == diversity
    - Different architecture/protocol per virtual network
  - Dynamic
    - New ones will come and old ones will go
    - Migration / Expansion / Contraction
- Simplify network management and service offerings
  - Virtual networks != VPN – VPN is just a service!
  - Virtual networks != P2P network – P2P is just an overlay
- Advantages
  - Overcome ossification of network core
  - Efficient utilization of resources
  - New business opportunities

Virtualization



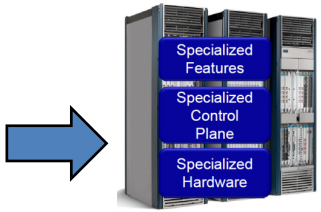
- Virtualization of intermediary devices corresponds to host virtualization concept with a focus on I/O
- Network Virtualization focusses on the separation of traffic flows using virtual paths often defined by labels, e.g. VLAN, MPLS, VXLAN, NVGRE

Service-Based Network Virtualization



SDN: A Computing Analogy

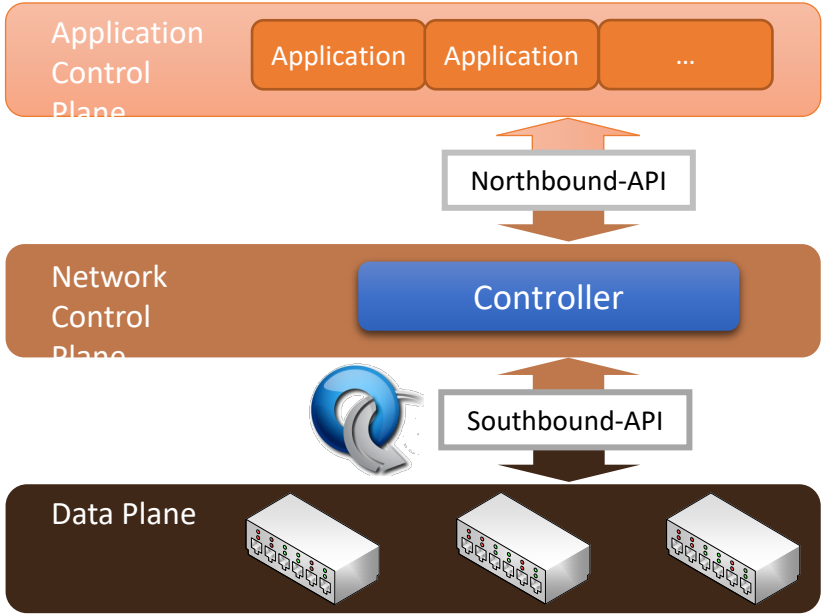
Vertically integrated
Closed, proprietary
Slow innovation
Small industry



Vertically integrated
Closed, proprietary
Slow innovation

Source: Nick McKeown

Software Defined Networking



Key Ideas of SDN

- Separation of control plane and data plane
  - Commoditization of Networking Hardware
  - Enable “mix-and-match” networking
- Open Interfaces
  - North- and southbound: Lower market-entry hurdle
  - East- and westbound: Increase interoperability
- Programmability
  - Tailored to fit solutions for companies with expertise in networking development and/or manpower
  - Flexible networking for everybody else

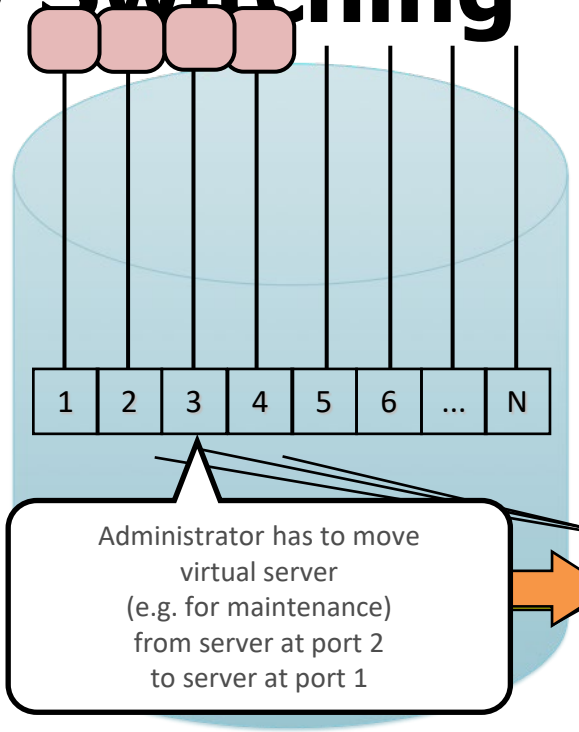
OpenFlow

- The OpenFlow standard specifies a communication protocol between the data plane of a networking element and a remote control plane

- OpenFlow **enables** Software Defined Networking
- OpenFlow introduced by the McKeown group at Stanford University (2008)

- Since version 1.2 the standardization body for OpenFlow is the Open Networking Foundation (ONF)

Flow Switching

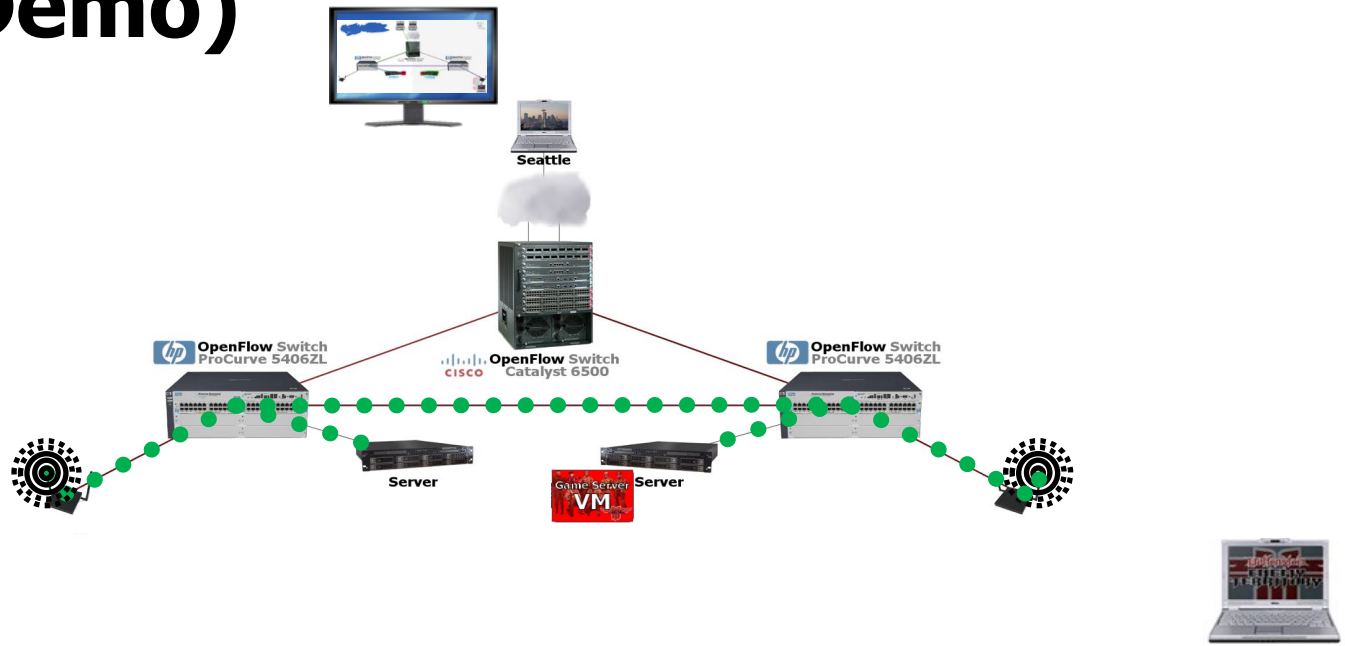


Flow Switching

- Considers source and destination
- IP : Port → IP : Port (TCP/UDP)
- Is able to drop flows / packets
- Enables rerouting of flows during runtime
- Speed up using hash tables

Flow	Src IP	Src Port	Dst IP	Dst Port	Ziel
1	X	5050	Y	80	3
2	Y	80	X	5050	1
3	Z	4433	U	22	6
4	V	4763	Y	80	4

Example: Server Migration (SIGCOMM 2008 Demo)



- Migration of virtual servers to save network resources and optimize the quality of experience

Network Functions

Today's networks consist of …

- Intermediary devices

Routers

Switches

- … and many different middle boxes, e.g. …

DPI

WAN Accelerator

Intrusion Detector

Firewall

DDoS Protector

QoE Monitor

Load Balancer

Network Function Virtualization Concept

- **Central idea:**
  Run network functions as virtual machines on commodity hardware

- **Goals:**
  - Cost reduction through use of commodity hardware
  - Leveraging the cloud
    - ➔ Flexible function instantiation
    - ➔ Dynamic function relocation
    - ➔ Resource-efficient operation
    - ➔ Resilient operation
  - Interchangeability of hardware/software

Source: NFV White Paper