



Welcome to

# Networking, TCP/IP and Security for beginners

PROSA Nov 2024

Henrik Kramselund he/him han/ham hlk@zencurity.com @kramse

Slides are available as PDF, kramse@Codeberg  
basic-tcpip-and-security.tex in the repo security-courses

# Contact information



- Henrik Kramselund, he/him internet samurai mostly networks and infosec
- Network and security consultant Zencurity, teach at KEA and activist
- Master from the Computer Science Department at the University of Copenhagen (DIKU)
- Email: [hlk@zencurity.com](mailto:hlk@zencurity.com)      Mobile: +45 2026 6000
- Run a small network for fun AS57860

You are welcome to drop me an email

# Goals for today



- Introduce basic TCP/IP terminology
- Show various network configurations with common protocols
- Describe how you can connect a router or switch to the network
- Describe basics of TCP/IP in 30 minutes
- Let you get some hands on with IP protocols

Photo is NWWC camp at BornHack 2024, next year 16-23rd of July 2025



## Time schedule

- 17:00 - 18:15  
Introduction and basics – with my network
- 18:45 - 21:00  
Connect to the network, play with TCP/IP, switches and routers.  
Mix of presentation and exercises

Note: even though I talk a lot about Unix and Linux, you can definitely run a lot of tools on Windows and Mac OS X. The basic tools are available like the built-in ones and Nmap

Command line tools are sometimes used in the slides, as they only show text where a GUI screenshot can be cluttered with a lot of information, feel free to find GUI tools and web sites with same functionality

# Exercises



Exercises are completely optional

- Try ping and traceroute
- See your own IP settings
- Borrow a USB Ethernet and connect to a switch or router

Linux is a toolbox I will use and participants are recommended to research virtual machines

# Course Materials



- This material is in multiple parts:
- Slide show - presentation - this file
- Exercises - PDF which is used for this and other workshops
- Additional resources from the internet are linked throughout
- Wikipedia has a LOT of nice pages about IP protocols, for example:

Transport Layer Security (TLS) is a cryptographic protocol designed to provide communications security over a computer network. The protocol is widely used in applications such as email, instant messaging, and voice over IP, but its use in securing HTTPS remains the most publicly visible.

Source: [https://en.wikipedia.org/wiki/Transport\\_Layer\\_Security](https://en.wikipedia.org/wiki/Transport_Layer_Security)

# Prerequisites



If you are interested in TCP/IP you are welcome

If you want to be an expert in IP and network security I recommend doing exercises

It is recommended to use virtual machines for the exercises

Network security and most internet related security work has the following requirements:

- Network experience
- TCP/IP principles - often in more detail than a common user
- Programming is an advantage, for automating things
- Some Linux and Unix knowledge is in my opinion a **necessary skill** for infosec work
  - too many new tools to ignore, and lots found at sites like Github and Open Source written for Linux

## Wifi Hardware



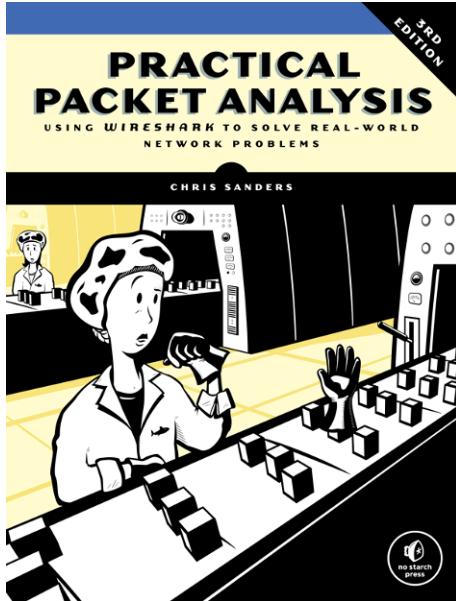
If you want to do sniffing of wireless it will be an advantage to have a wireless USB network card. Make sure to play nice, and dont abuse knowledge!

- The following are two recommended models:
- TP-link TL-WN722N hardware version 2.0 cheap but only support 2.4GHz
- Alfa AWUS036ACH 2.4GHz + 5GHz Dual-Band and high performing
- Both work great in Kali Linux for our purposes, but are older models by now

Unfortunately the vendors change models often enough that the above are hard to find. I recommend using your favourite search engine and research which cards work with Kali Linux and airodump-ng.

I have some available you can borrow

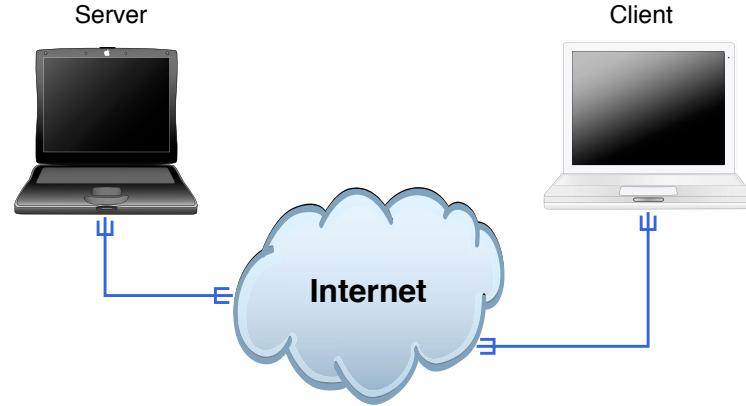
# Book: Practical Packet Analysis (PPA)



*Practical Packet Analysis, Using Wireshark to Solve Real-World Network Problems* by Chris Sanders, 3rd Edition April 2017, 368 pp. ISBN-13: 978-1-59327-802-1 <https://nostarch.com/packetanalysis3>

I recommend this book for people new to networking, it has been in HumbleBundle book bundles multiple times

# Internet Today



Clients and servers, roots in the academic world

Protocols are old, some more than 20 years

Very few protocols were encrypted, today a lot has switched to HTTPS and TLS

# Internet is Open Standards!



We reject kings, presidents, and voting.  
We believe in rough consensus and running code.  
– The IETF credo Dave Clark, 1992.

Request for comments (RFC) – a series of documents spanning decades  
RFC, BCP, FYI, informational – first ones from 1969!  
Are not updated but status is changed to Obsoleted when new versions are published  
Standards track:  
Proposed Standard → Draft Standard → Standard

# Internetworking: history



- 1961 L. Kleinrock, MIT packet-switching theory
- 1962 J. C. R. Licklider, MIT - notes
- 1964 Paul Baran: On Distributed Communications
- 1969 ARPANET 4 nodes
- 1971 14 nodes
- 1973 Design of Internet Protocols started
- 1973 Email is about 75% of all ARPANET traffic
- 1974 TCP/IP: Cerf/Kahn: A protocol for Packet Network Interconnection
- 1983 EUUG → DKUUG/DIKU Denmark
- 1988 About 60.000 systems on the internet - The Morris Worm hits about 10%
- 2002 About 130 million Internet hosts
- 2010 IANA reserved blocks 7% (Maj 2010) - <http://www.potaroo.net/tools/ipv4/>

# What is the Internet



Communication between humans - currently!

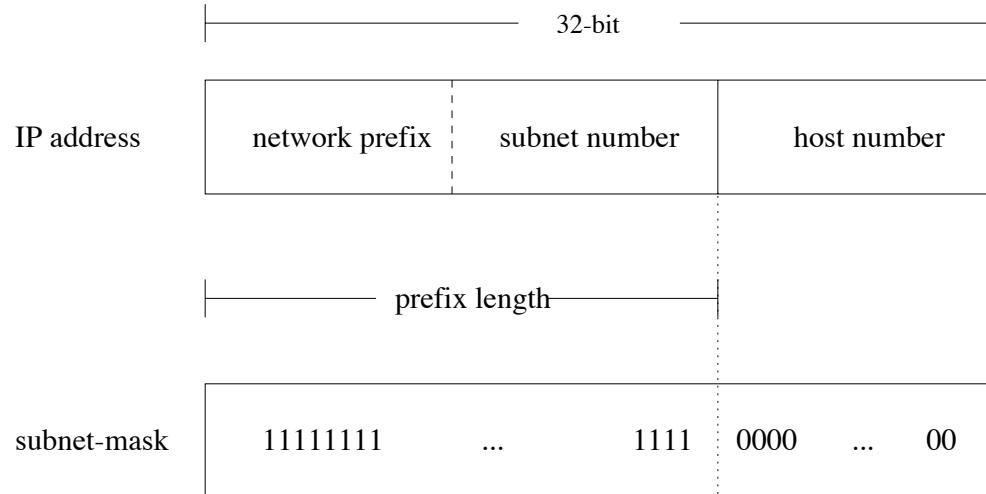
Based on TCP/IP

- best effort
- packet switching (IPv6 calls it packets, not datagram)
- *connection-oriented* TCP
- *connection-less* UDP

RFC-1958:

A good analogy for the development of the Internet is that of constantly renewing the individual streets and buildings of a city, rather than razing the city and rebuilding it. The architectural principles therefore aim to provide a framework for creating cooperation and standards, as a small "spanning set" of rules that generates a large, varied and evolving space of technology.

# Common Address Space



- Internet is defined by the address space
- IPv4 based on 32-bit addresses, example dotted decimal format 10.0.0.1
- IPv6 very similar to IPv4 without NAT, 128-bit addresses in hex ::1, 2a06:d380:0:101::80

# How to use the Internet Protocols (IP)



Names are used by humans

[www.kramse.org](http://www.kramse.org)

[hik@kramse.org](mailto:hik@kramse.org)

Computers use the addresses

www	IN	A	185.129.63.130
	IN	AAAA	2a06:d380:0:102::80
mail	IN	A	217.157.63.115
	IN	AAAA	2a06:d380:0:102::25



## Documentation Prefix, IPv6 updates etc.

Even documentation has its own prefix, RFC5737:

The blocks 192.0.2.0/24 (TEST-NET-1) , 198.51.100.0/24 (TEST-NET-2) ,  
and 203.0.113.0/24 (TEST-NET-3) are provided for use in  
documentation.

IPv6 listed in RFC3849 2001:DB8::/32

See RFC3330 *Special-Use IPv4 Addresses* which is updated by RFC6890 *Special-Purpose IP Address Registries* which in turn is updated by RFC8190

Use the web version of RFCs to surf back and forth <https://www.rfc-editor.org/rfc/rfc8190>

# CIDR Classless Inter-Domain Routing



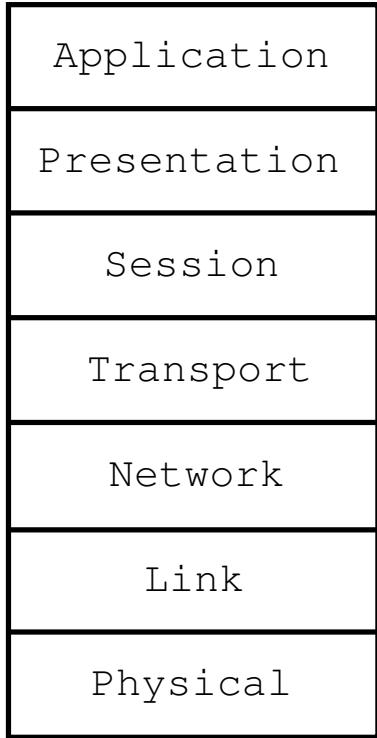
Classful routing		Classless routing CIDR	
4 class C networks	Inherent subnet mask	Supernet	Subnet mask
192.0.8.0	255.255.255.0	192.0.8.0	255.255.252.0
192.0.9.0	255.255.255.0		252d=11111100b
192.0.10.0	255.255.255.0		
192.0.11.0	255.255.255.0	Base network/prefix	
			192.10.8.0/22

- Subnet mask originally inferred by the class
- Started to allocate multiple C-class networks - save remaining B-class  
Resulted in routing table explosion - btw Stop using A, B, C
- A subnet mask today is a row of 1-bit
- Supernet, supernetting
- 10.0.0.0/24 means the network 10.0.0.0 with 24 subnet bits (mask 255.255.255.0)
- 2a06:d380:0:101::80/64 means the network with 64-bit prefix length

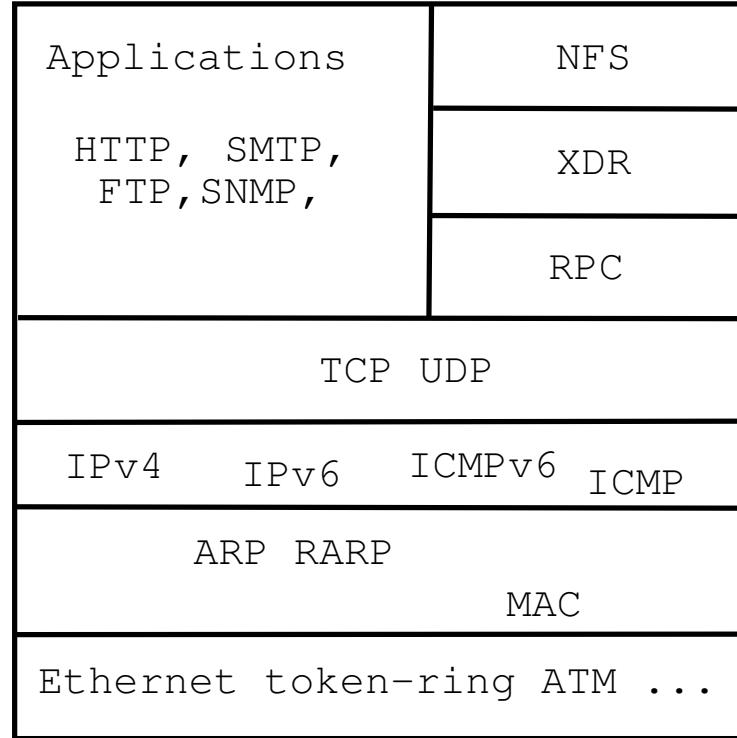
# Protocols: OSI and Internet models



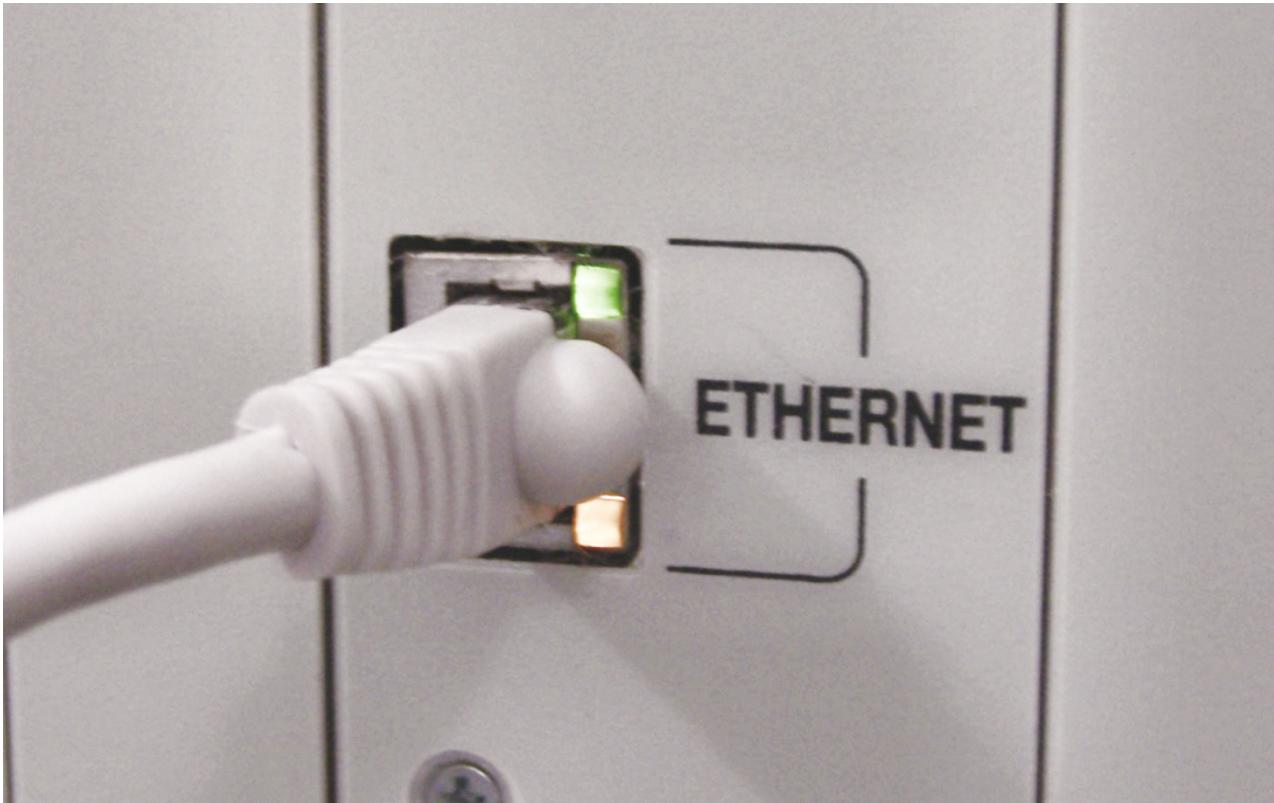
OSI Reference Model



Internet protocol suite



## Ethernet, cables



Show link, and activity – blinkenlights

# MAC address



00-03-93	(hex)	Apple Computer, Inc.
000393	(base 16)	Apple Computer, Inc.
		20650 Valley Green Dr.
		Cupertino CA 95014
		UNITED STATES

Network technologies use a layer 2 hardware address

Typically using 48-bit MAC addresses known from Ethernet MAC-48/EUI-48

First half is assigned to companies – Organizationally Unique Identifier (OUI)

Using the OUI you can see which producer and roughly when a network chip was produced

<http://standards.ieee.org/regauth/oui/index.shtml>

# Bridges



Ethernet is a broadcast technology data transmitted into the ether – a cable

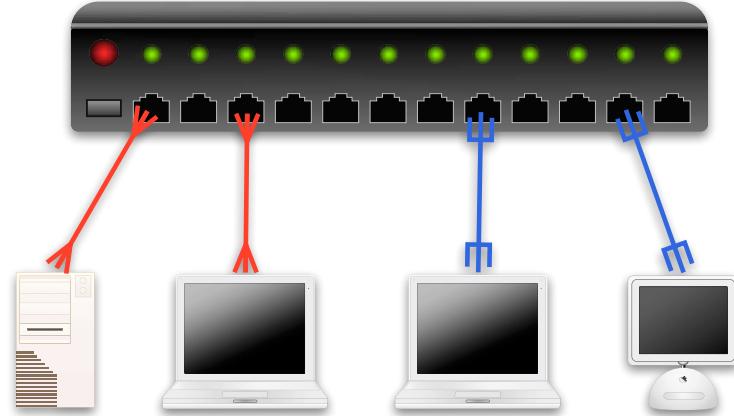
This limits how many devices to connect

Using bridges we can connect segments – which copy between them if needed

It learns the devices on each side (MAC address)

See also [http://en.wikipedia.org/wiki/ALOHA\\_Net](http://en.wikipedia.org/wiki/ALOHA_Net)

# A switch



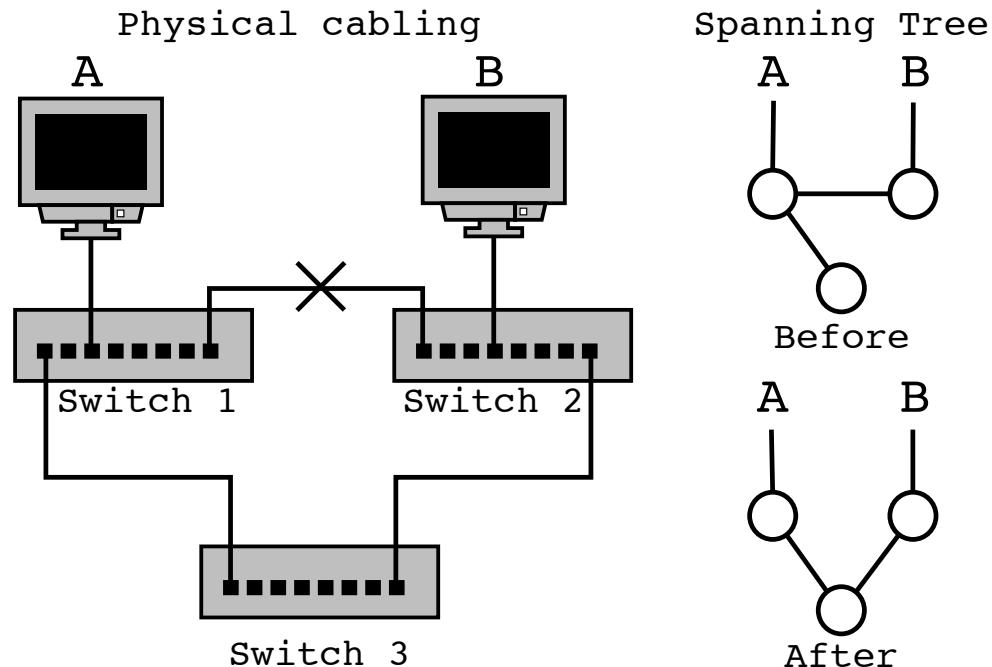
Today we use switches, Don't buy a hub, not even for experimenting or sniffing  
A switch can receive and send data on multiple ports at the same time  
Performance only limited by the backplane and switching chips  
Can also often route with the same speed and mirror packets

## Wireless



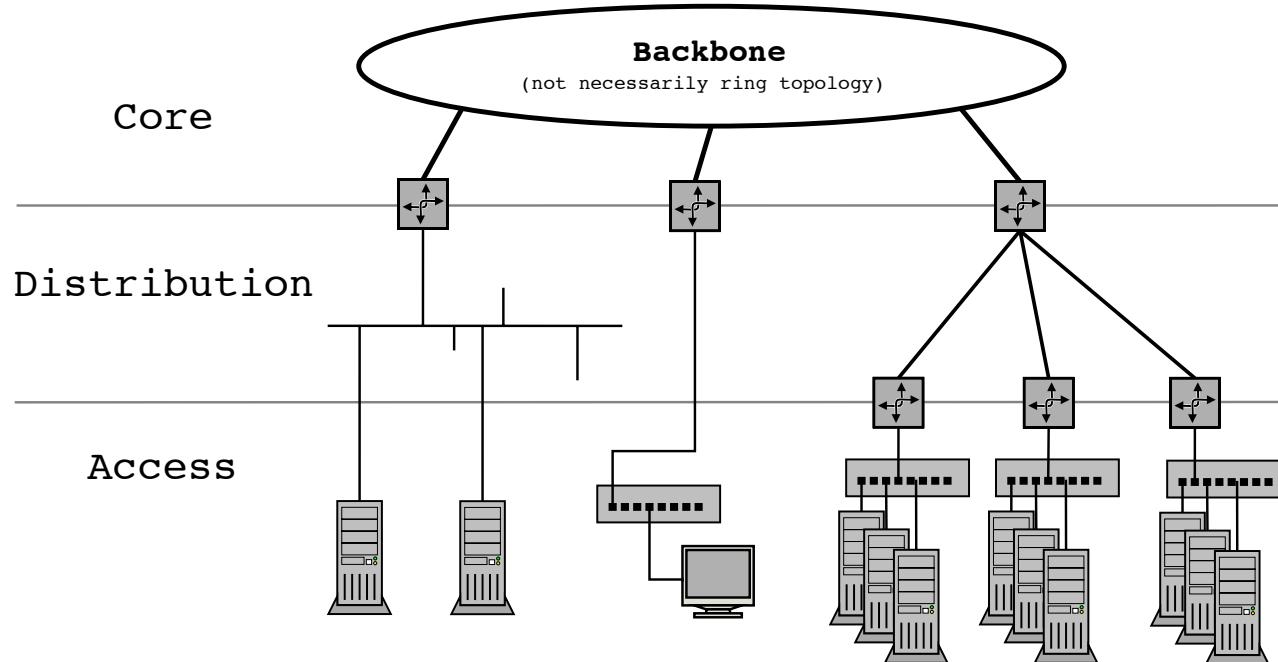
A typical home router would have built-in 802.11 Access-Point (AP) and some Ethernet LAN ports

# Topologier og Spanning Tree Protocol



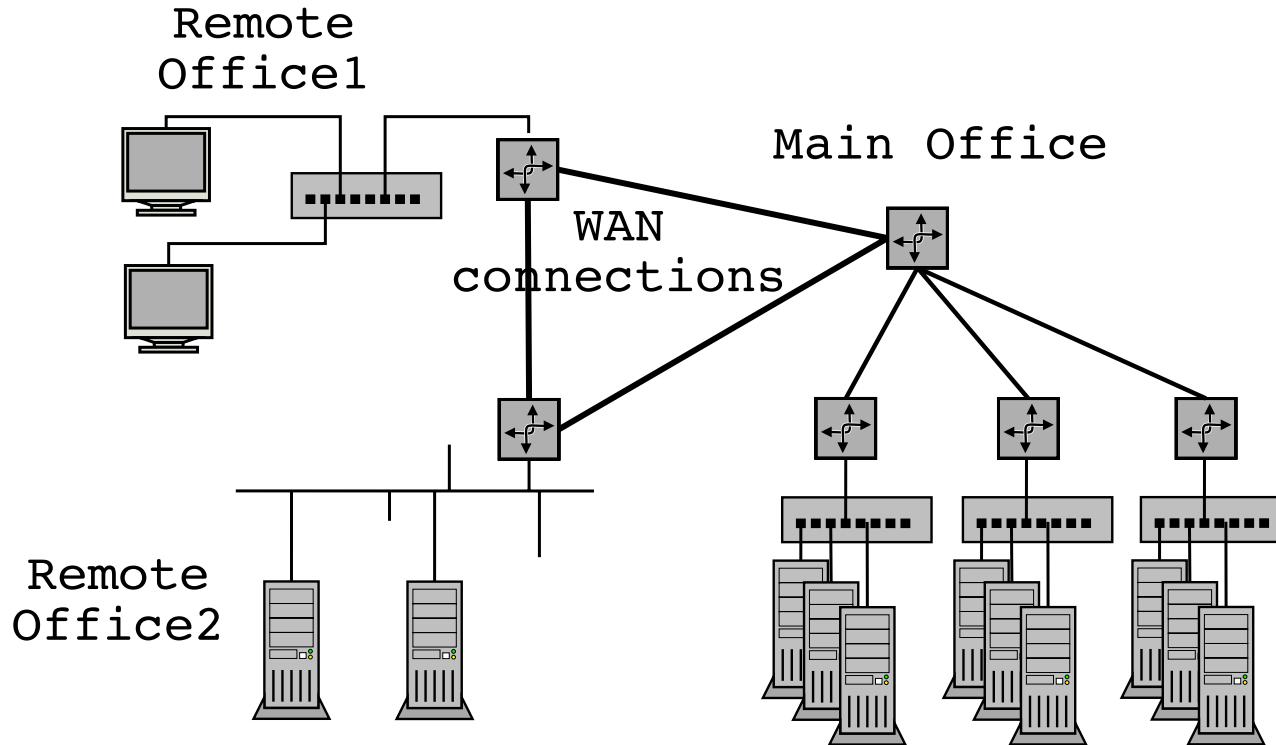
Se mere i bogen af Radia Perlman, *Interconnections: Bridges, Routers, Switches, and Internetworking Protocols*

# Core, Distribution og Access net

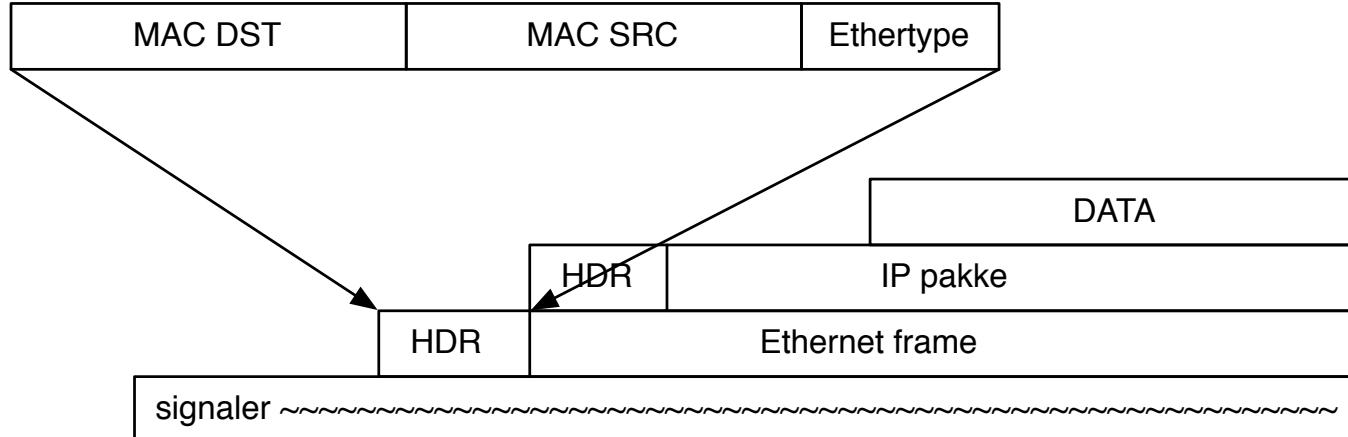


Det er ikke altid man har præcis denne opdeling, men den er ofte brugt

## Bridges and routers



# Packets of data

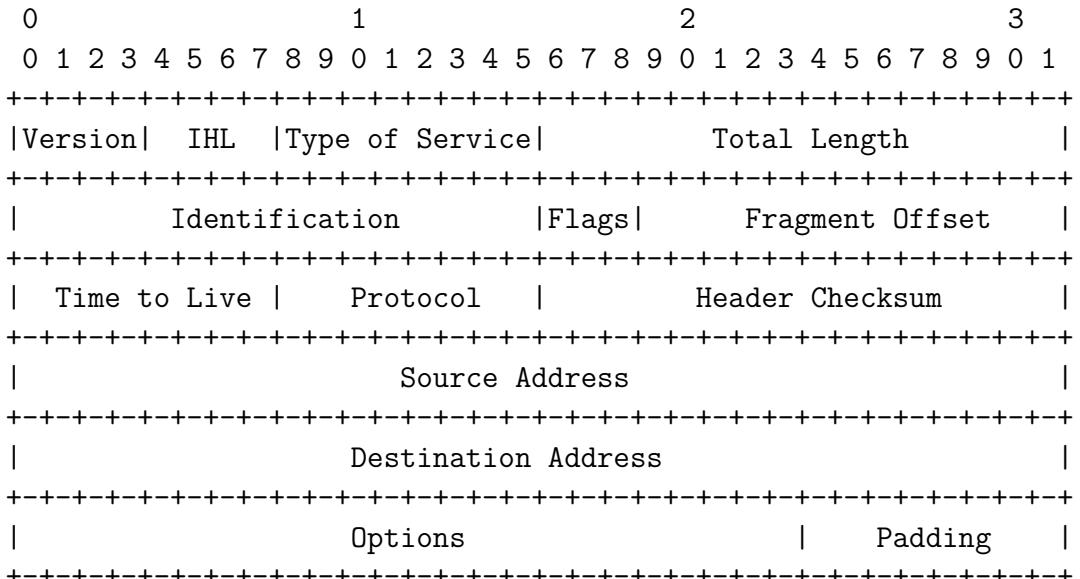


Looking into the hardware we see that data is laid out according to a structure – frames and packets

Often a start and end signal of a frame – like Ethernet

Today we talk about packets of 1500 bytes which is common in Ethernet

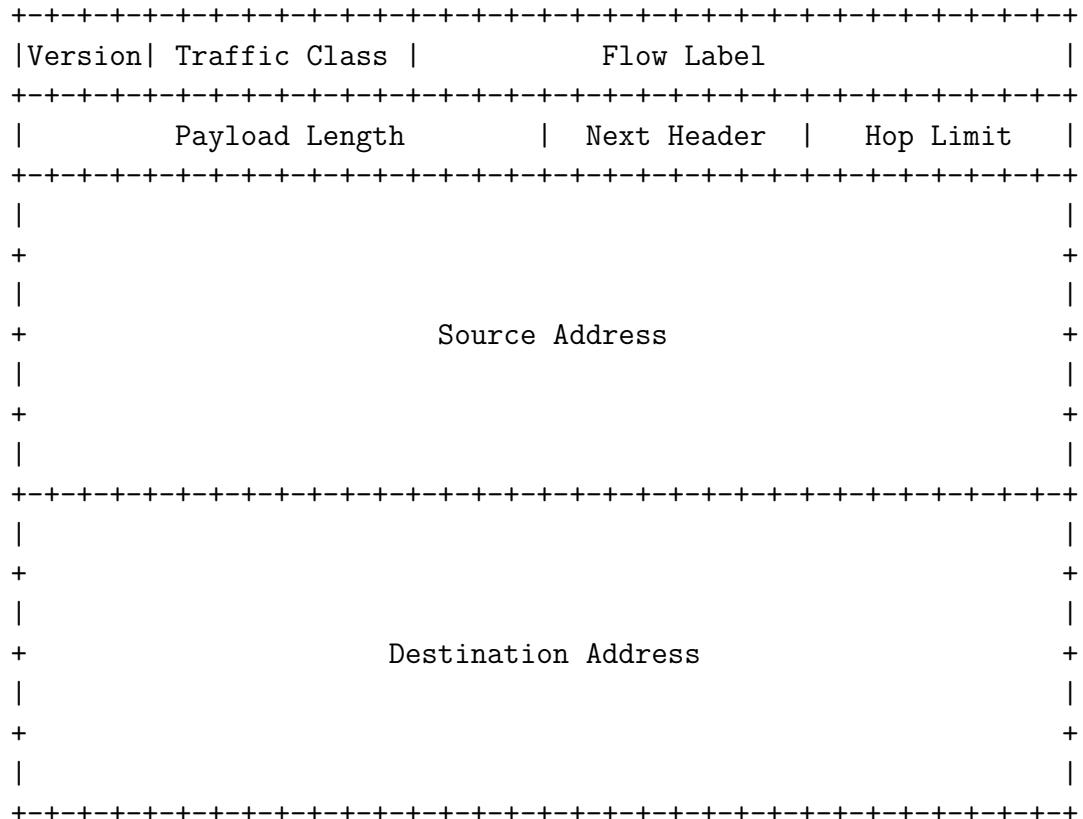
# IPv4 header - RFC-791 september 1981



Example Internet Datagram Header

Source: <https://datatracker.ietf.org/doc/html/rfc791> and updated later

# IPv6 header - RFC-1883 December 1995

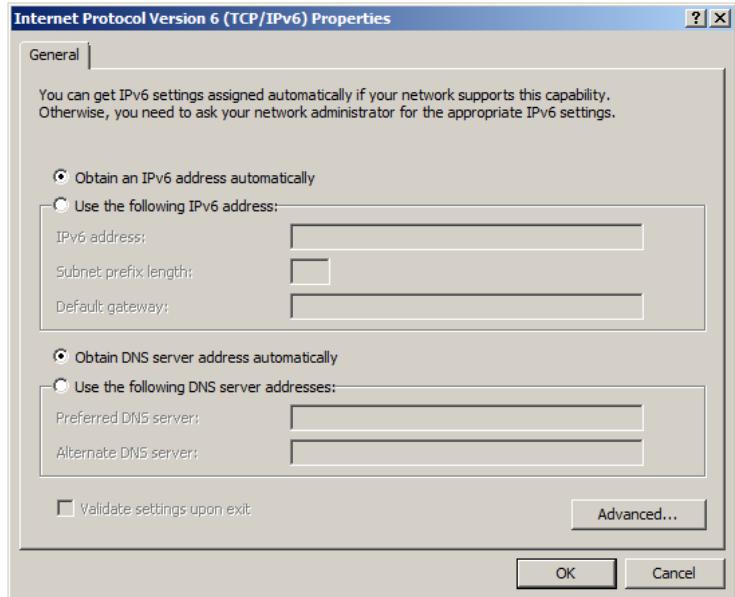


# Windows - ipconfig



```
C:\ Command Prompt  
Microsoft Windows [Version 6.1.7600]  
Copyright <c> 2009 Microsoft Corporation. All rights reserved.  
  
C:\Users\Henrik Kramhoej>ipconfig  
  
Windows IP Configuration  
  
Ethernet adapter Local Area Connection:  
  
Connection-specific DNS Suffix . : kramse.dk  
IPv6 Address . . . . . : 2001:16d8:dd0f:cf0f:f049:94d0:75d8:683e  
Temporary IPv6 Address . . . . . : 2001:16d8:dd0f:cf0f:84bd:adea:fb61:8960  
Link-local IPv6 Address . . . . . : fe80::f049:94d0:75d8:683e%11  
IPv4 Address . . . . . : 10.0.42.107  
Subnet Mask . . . . . : 255.255.255.0  
Default Gateway . . . . . : fe80::200:24ff:fec8:b24c%11  
10.0.42.1  
  
Tunnel adapter isatap.kramse.dk:  
  
Media State . . . . . : Media disconnected  
Connection-specific DNS Suffix . : kramse.dk  
  
Tunnel adapter Local Area Connection* 11:  
  
Connection-specific DNS Suffix . :  
IPv6 Address . . . . . : 2001:0:5ef5:73b8:1000:322b:f5ff:d594  
Link-local IPv6 Address . . . . . : fe80::1000:322b:f5ff:d594%13  
Default Gateway . . . . . :  
  
C:\Users\Henrik Kramhoej>_
```

# Windows – control panel with DHCP



DHCP is responsible for giving you a dynamic address

# Unix - practical examples ifconfig and ping



```
$ ifconfig en0
en0: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
      inet6 fe80::216:cbff:feac:1d9f%en0 prefixlen 64 scopeid 0x4
          inet 10.0.42.15 netmask 0xffffffff broadcast 10.0.42.255
          inet6 2001:16d8:dd0f:cf0f:216:cbff:feac:1d9f prefixlen 64 autoconf
              ether 00:16:cb:ac:1d:9f
              media: autoselect (1000baseT <full-duplex>) status: active

$ ping6 ::1
PING6(56=40+8+8 bytes) ::1 --> ::1
16 bytes from ::1, icmp_seq=0 hlim=64 time=0.089 ms
16 bytes from ::1, icmp_seq=1 hlim=64 time=0.155 ms

$ traceroute6 2001:16d8:dd0f:cf0f::1
traceroute6 to 2001:16d8:dd0f:cf0f::1 (2001:16d8:dd0f:cf0f::1)
from 2001:16d8:dd0f:cf0f:216:cbff:feac:1d9f, 64 hops max, 12 byte packets
 1  2001:16d8:dd0f:cf0f::1  0.399 ms  0.371 ms  0.294 ms
```

# The basic tools for countering threats



## Knowledge and insight

- Networks have end-points and conversations on multiple layers
- Wireshark is advanced, try right-clicking different places
- Name resolution includes low level MAC addresses, and IP - names
- Tcpdump format, built-in to many network devices
- Remote packet dumps, like `tcpdump -i eth0 -w packets.pcap`
- Story: tcpdump was originally written in 1988 by Van Jacobson, Sally Floyd, Vern Paxson and Steven McCanne who were, at the time, working in the Lawrence Berkeley Laboratory Network Research Group  
<https://en.wikipedia.org/wiki/Tcpdump>

Great network security comes from knowing networks!

# Network Knowledge Needed

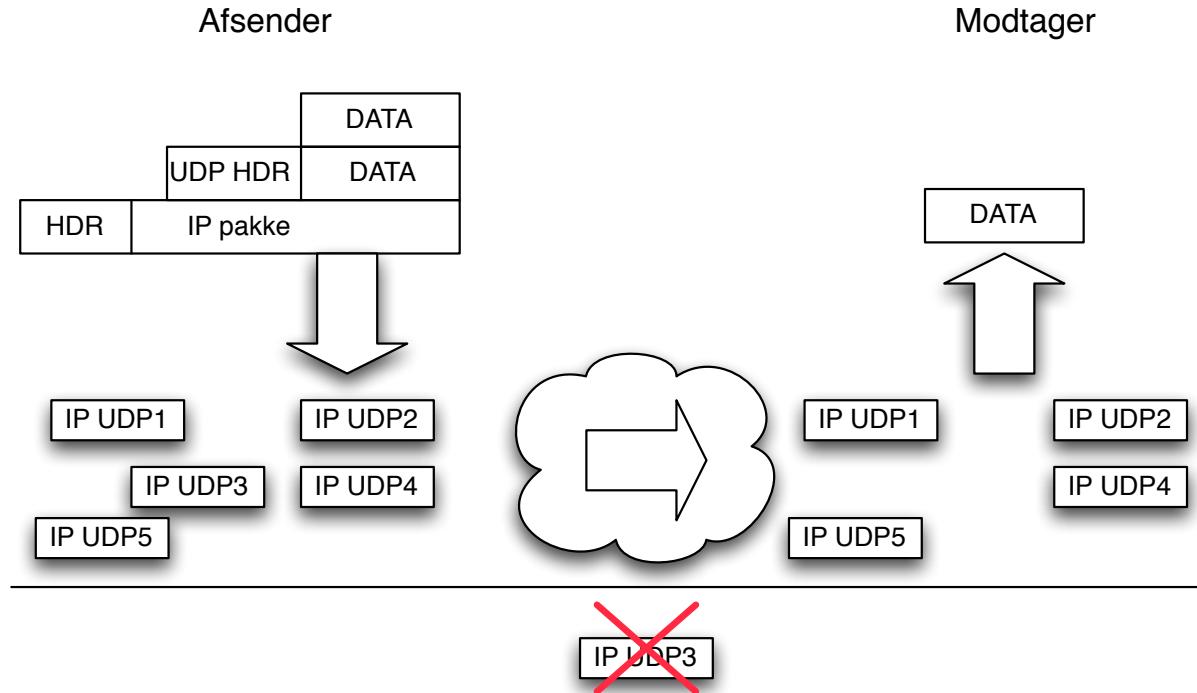


To work with network security the following protocols are the bare minimum to know about.

- ARP Address Resolution Protocol for IPv4
- NDP Neighbor Discovery Protocol for IPv6
- IPv4 & IPv6 – the basic packet fields source, destination,
- ICMPv4 & ICMPv6 Internet Control Message Protocol
- UDP User Datagram Protocol
- TCP Transmission Control Protocol
- DHCP Dynamic Host Configuration Protocol
- DNS Domain Name System

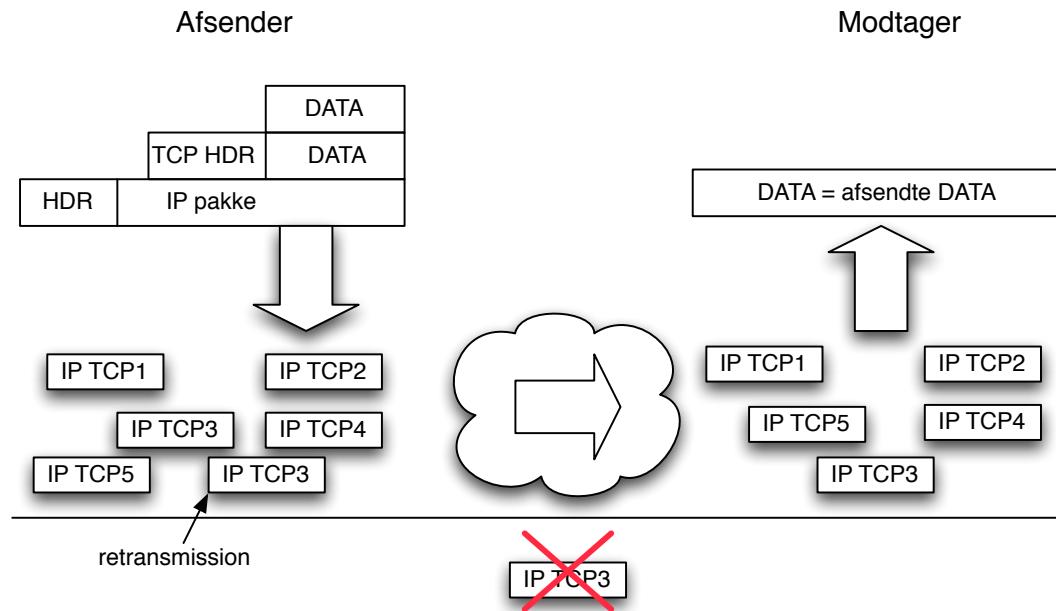
A little Linux knowledge is also **highly recommended**

# UDP User Datagram Protocol



Connectionless [https://en.wikipedia.org/wiki/User\\_Datagram\\_Protocol](https://en.wikipedia.org/wiki/User_Datagram_Protocol)

# TCP Transmission Control Protocol



Connection-oriented [https://en.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](https://en.wikipedia.org/wiki/Transmission_Control_Protocol)

## Well-Known Port Numbers



IANA maintains a list of magical numbers in TCP/IP  
Lists of protocol numbers, port numbers etc.

A few notable examples:

- Port 25/tcp Simple Mail Transfer Protocol (SMTP)
- Port 53/udp and 53/tcp Domain Name System (DNS)
- Port 80/tcp Hyper Text Transfer Protocol (HTTP)
- Port 443/tcp HTTP over TLS/SSL (HTTPS)

Source: <http://www.iana.org>

## Whois – Where do IP addresses come from



All these magical numbers we use on the internet are administered by IANA <https://www.iana.org/>  
They have handed out portions to the Region Internet Registries (RIR)

- RIPE (Réseaux IP Européens) <http://ripe.net>
- ARIN American Registry for Internet Numbers <http://www.arin.net>
- Asia Pacific Network Information Center <http://www.apnic.net>
- LACNIC (Regional Latin-American and Caribbean IP Address Registry) - Latin America and some Caribbean Islands

AFRINIC <https://afrinic.net/>

They are memberbased, and members are called Local Internet Registries (LIRs) og National Internet Registry (NIR)

# Ping



## ICMP - Internet Control Message Protocol

Benyttes til fejlbeskeder og til diagnosticering af forbindelser

ping programmet virker ved hjælp af ICMP ECHO request og forventer ICMP ECHO reply

```
$ ping 192.168.1.1
PING 192.168.1.1 (192.168.1.1): 56 data bytes
64 bytes from 192.168.1.1: icmp_seq=0 ttl=150 time=8.849 ms
64 bytes from 192.168.1.1: icmp_seq=1 ttl=150 time=0.588 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=150 time=0.553 ms
```



## traceroute

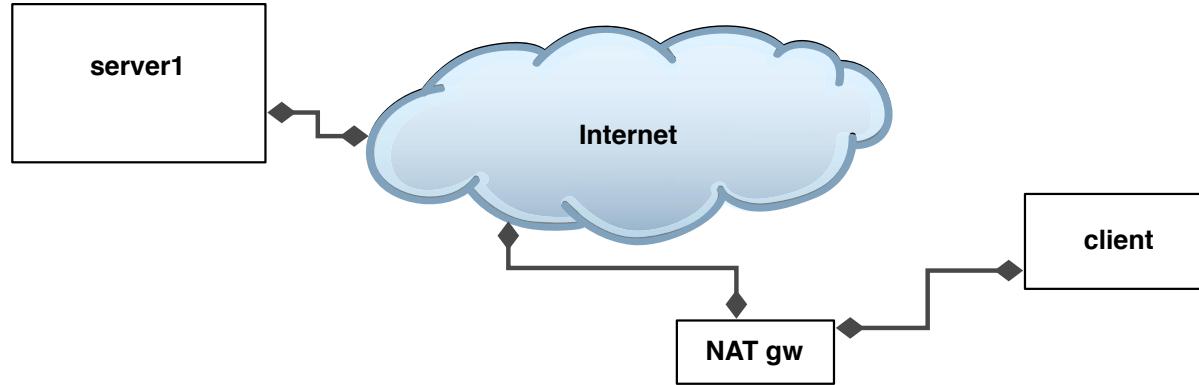
traceroute programmet virker ved hjælp af TTL

levetiden for en pakke tælles ned i hver router på vejen og ved at sætte denne lavt opnår man at pakken *timer ud* - besked fra hver router på vejen

default er UDP pakker, men på UNIX systemer er der ofte mulighed for at bruge ICMP

```
$ traceroute 185.129.60.129
traceroute to 185.129.60.129 (185.129.60.129),
30 hops max, 40 byte packets
 1  safri (10.0.0.11)  3.577 ms  0.565 ms  0.323 ms
 2  router (185.129.60.129)  1.481 ms  1.374 ms  1.261 ms
```

# NAT Network Address Translation



- NAT is used for connecting private networks to the Internet
- NAT gateway replaces source address and forwards packets
- A quick and dirty fix that keeps messing up networks and protocols
- The NAT router/firewall has state tables

# RFC-1918 Private Networks



There is a list of network prefixes anyone can use, for private networks:

- 10.0.0.0 - 10.255.255.255 (10/8 prefix)
- 172.16.0.0 - 172.31.255.255 (172.16/12 prefix)
- 192.168.0.0 - 192.168.255.255 (192.168/16 prefix)

Address Allocation for Private Internets RFC-1918 adresserne!

To use these typically there will be a NAT device in front

The blocks 192.0.2.0/24 (TEST-NET-1) , 198.51.100.0/24 (TEST-NET-2) ,  
and 203.0.113.0/24 (TEST-NET-3) are provided for use in  
documentation.

169.254.0.0/16 has been ear-marked as the IP range to use for end node  
auto-configuration when a DHCP server may not be found

# Course Network

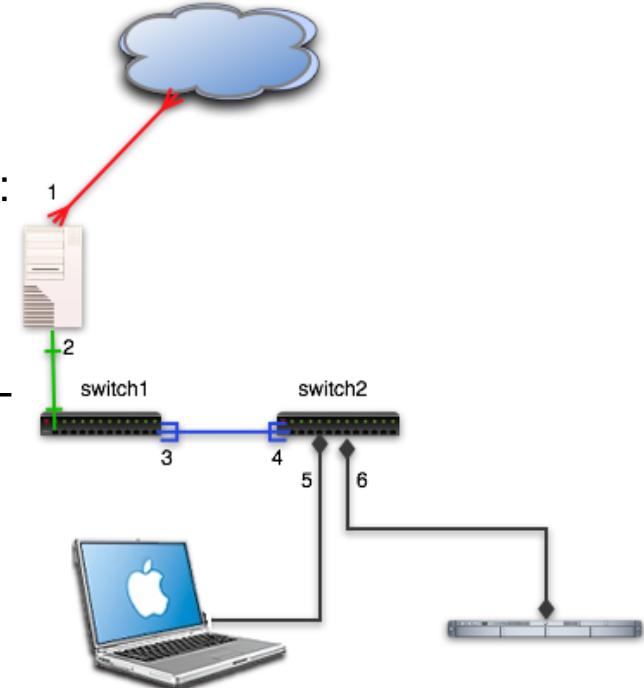


.

I have a course network with me which has the following information:

- Router APU24D – single board computer with OpenBSD
- Switches Juniper and TP-Link
- wireless access-points Unifi AP [https://en.wikipedia.org/wiki/Autonomous\\_system\\_\(Internet\)](https://en.wikipedia.org/wiki/Autonomous_system_(Internet))

You are encouraged to use the network





# Wireshark - graphical network sniffer

http-example.cap

Apply a display filter: <None>

No.	Time	Source	Destination	Protocol	Info
1	0.080000	172.24.65.102	91.182.91.18	TCP	58816 - http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM=0
2	0.080170	172.24.65.102	91.182.91.18	TCP	58817 - http [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=16 TSval=745562412 TSecr=0 SACK_PERM=0
3	0.127053	91.182.91.18	172.24.65.102	TCP	58816 [SYN, ACK] Seq=0 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=1855239975 TSecr=0
4	0.127167	91.182.91.18	172.24.65.102	TCP	58817 [SYN, ACK] Seq=1 Ack=1 Win=16384 Len=0 MSS=1460 SACK_PERM=1 WS=8 TSval=1855239975 TSecr=0
5	0.127226	172.24.65.102	91.182.91.18	TCP	58816 - http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=1855239975
6	0.127241	172.24.65.102	91.182.91.18	TCP	58817 - http [ACK] Seq=1 Ack=1 Win=131760 Len=0 TSval=745562538 TSecr=1855239975
7	0.127365	172.24.65.102	91.182.91.18	HTTP	GET / HTTP/1.1
8	0.141320	91.182.91.18	172.24.65.102	HTTP	HTTP/1.1 304 Not Modified
9	0.141421	172.24.65.102	91.182.91.18	TCP	58816 - http [ACK] Seq=583 Ack=190 Win=131568 Len=0 TSval=745562551 TSecr=1855239975

Frame 7: 568 bytes on wire (4544 bits), 568 bytes captured (4544 bits)  
Ethernet II, Src: Apple\_6c:87:5e (7c:dc:c3:6c:87:5e), Dst: Cisco\_32:09:30 (44:2b:03:32:09:30)  
Internet Protocol Version 4, Src: 172.24.65.102 (172.24.65.102), Dst: 91.182.91.18 (91.182.91.18)  
Transmission Control Protocol, Src Port: 58816 (58816), Dst Port: http (80), Seq: 1, Ack: 1, Len: 562

HyperText Transfer Protocol  
GET / HTTP/1.1\r\nHost: 91.182.91.18\r\nUser-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_9\_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36\r\nAccept-Encoding: gzip,deflate,sdch\r\nAccept-Language: en-US,en;q=0.8,cs;q=0.6,da;q=0.4\r\nIf-None-Match: "7053a63e1516a5b27a295edbd31d07524a6e0a3"\r\nIf-Modified-Since: Tue, 17 Nov 2009 11:22:22 GMT\r\n\r\n

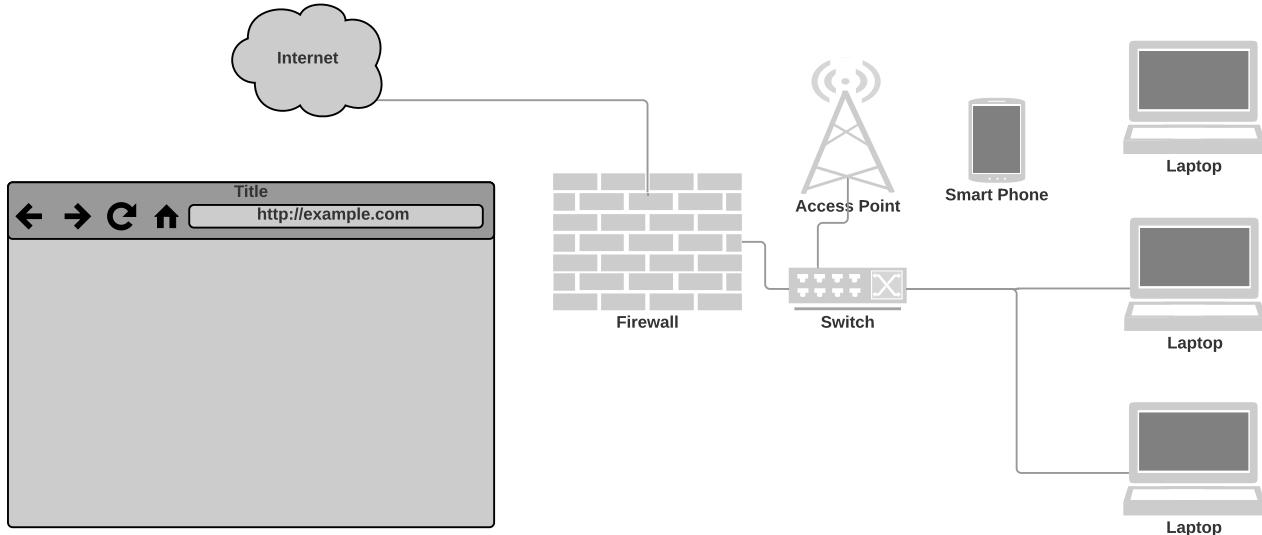
[Full request URI: http://91.182.91.18/] [HTTP request 1/1] [Response in frame: 8]

0000 44 2b 03 32 09 30 7c d1 c3 6c 87 5e 00 00 45 00 D+.2.0|N Ál.^..E.  
0010 02 2a 9e d7 40 00 40 06 ff ff ac 18 41 66 5b 66 .\*.x@.ø. öý~Af[f  
0020 5b 12 e5 c0 00 50 00 00 0e c7 03 14 0c 19 88 18 [.ñÁ.P.é Ç.....  
0030 2b 0f c8 00 00 00 00 00 00 00 00 00 00 00 00 00 +.ñ.....  
0040 07 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 GET / HTTP/1.1  
0050 0d 0e 48 6f 73 74 3a 20 39 31 2e 31 30 32 2e 39 ..Host: 91.102.9  
0060 31 2e 31 38 0d 0a 43 6f 6e 66 65 63 74 69 6f 6e 1.18.Co nnection  
0070 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 43 61 : keep-a live..Ca  
0080 63 68 65 2d 43 6f 6e 74 72 6f 6e 3a 20 6a 61 78 che-Cont rol: max  
0090 2d 61 67 65 3d 30 0d 0a 41 61 63 65 70 74 3a 20 -ages=0.. Accept:  
00a0 63 68 65 2f 6d 6f 6e 74 72 6f 6e 3a 20 6a 61 78 text/html,application  
00b0 61 67 65 2d 6d 6f 6e 74 72 6f 6e 3a 20 6a 61 78 attachment; charset=  
00c0 61 70 78 6c 69 63 61 74 69 6f 6e 2f 78 6d 6c 3b application/xml;

Packets: 9 - Displayed: 9 - Marked: 0 - Load time: 0:0:0 - Profile: Default

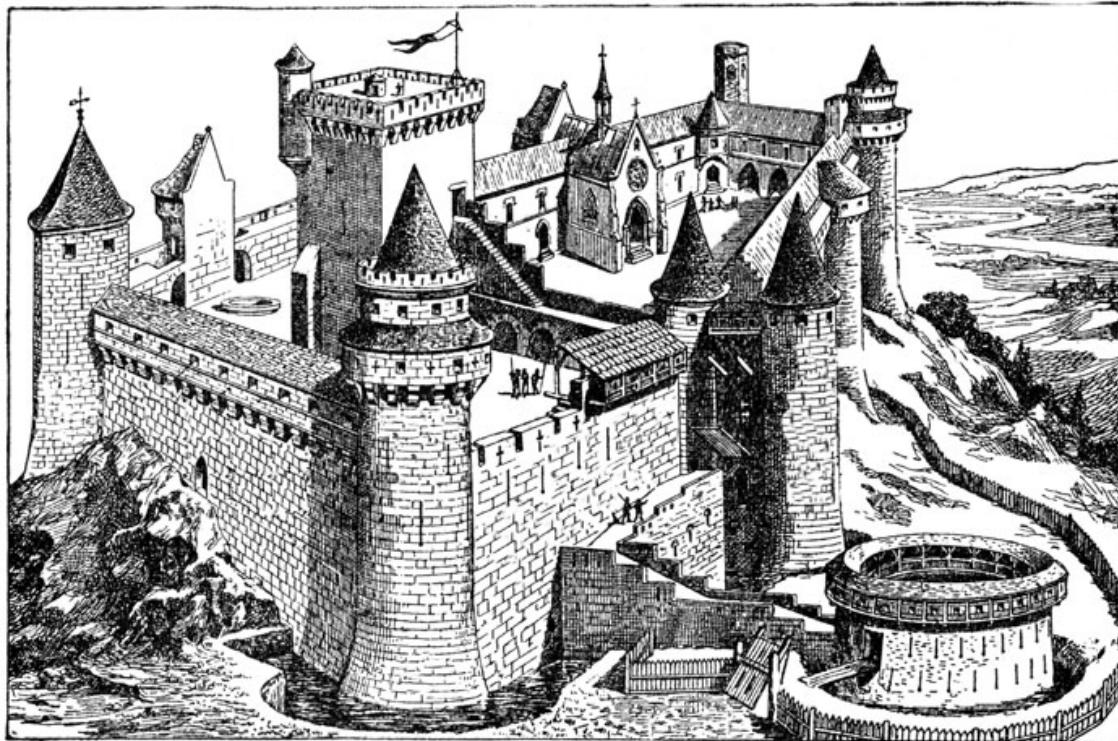
Capture - Options, select a network interface  
<http://www.wireshark.org>

# Your Privacy



- Your data travels far
- Often crossing borders, virtually and literally

# Defense in depth



Picture originally from: <http://karenswhimsy.com/public-domain-images>



# Network Segmentation – Firewalls

\$ firewall

1. (I) **An internetwork gateway that restricts data communication traffic to and from one of the connected networks** (the one said to be "inside" the firewall) and thus protects that network's system resources against threats from the other network (the one that is said to be "outside" the firewall). (See: guard, security gateway.)

2. (O) **A device or system that controls the flow of traffic between networks using differing security postures.**  
Wack, J. et al (NIST), "Guidelines on Firewalls and Firewall Policy", Special Publication 800-41, January 2002.

Tutorial: A firewall typically protects a smaller, secure network (such as a corporate LAN, or even just one host) from a larger network (such as the Internet). The firewall is installed at the point where the networks connect, and the firewall applies policy rules to control traffic that flows in and out of the protected network.

Source: RFC4949 *Internet Security Glossary, Version 2*

<https://datatracker.ietf.org/doc/html/rfc4949> 2007

## Continued



**A firewall is not always a single computer.** For example, a firewall may consist of a pair of filtering routers and one or more proxy servers running on one or more bastion hosts, all connected to a small, dedicated LAN (see: buffer zone) between the two routers. The external router blocks attacks that use IP to break security (IP address spoofing, source routing, packet fragments), while proxy servers block attacks that would exploit a vulnerability in a higher-layer protocol or service. The internal router blocks traffic from leaving the protected network except through the proxy servers. The difficult part is defining criteria by which packets are denied passage through the firewall, because a firewall not only needs to keep unauthorized traffic (i.e., intruders) out, but usually also needs to let authorized traffic pass both in and out.

Source: RFC4949 *Internet Security Glossary, Version 2*

<https://datatracker.ietf.org/doc/html/rfc4949> 2007

- Network layer, packet filters, application level, stateless, stateful
- Firewalls are by design a choke point, natural place to do network security monitoring!
- Older but still interesting Cheswick chapter 2 PDF *A Security Review of Protocols: Lower Layers*  
<http://www.wilyhacker.com/>

# Modern Firewall Infrastructures



A firewall **blocks** traffic on a network

A firewall **allows** traffic on a network

The interesting part is typically what it allows!

A firewall infrastructure must:

- Prevent attackers from entering
- Prevent data exfiltration
- Prevent worms, malware, virus from spreading in networks
- Be part of an overall solution with ISP, routers, other firewalls, switched infrastructures, intrusion detection systems and the rest of the infrastructure
- ...

Difficult – and requires design and secure operations

# Open source based firewalls



- Linux firewalls IP tables, use command line tool ufw Uncomplicated Firewall!
- Firewall GUIs on top of Linux – lots! Some are also available as commercial ones
- OpenBSD PF <http://www.openbsd.org>
- FreeBSD IPFW og IPFW2 <http://www.freebsd.org>
- Mac OS X uses OpenBSD PF
- FreeBSD has an older version of the OpenBSD PF, should really be renamed now



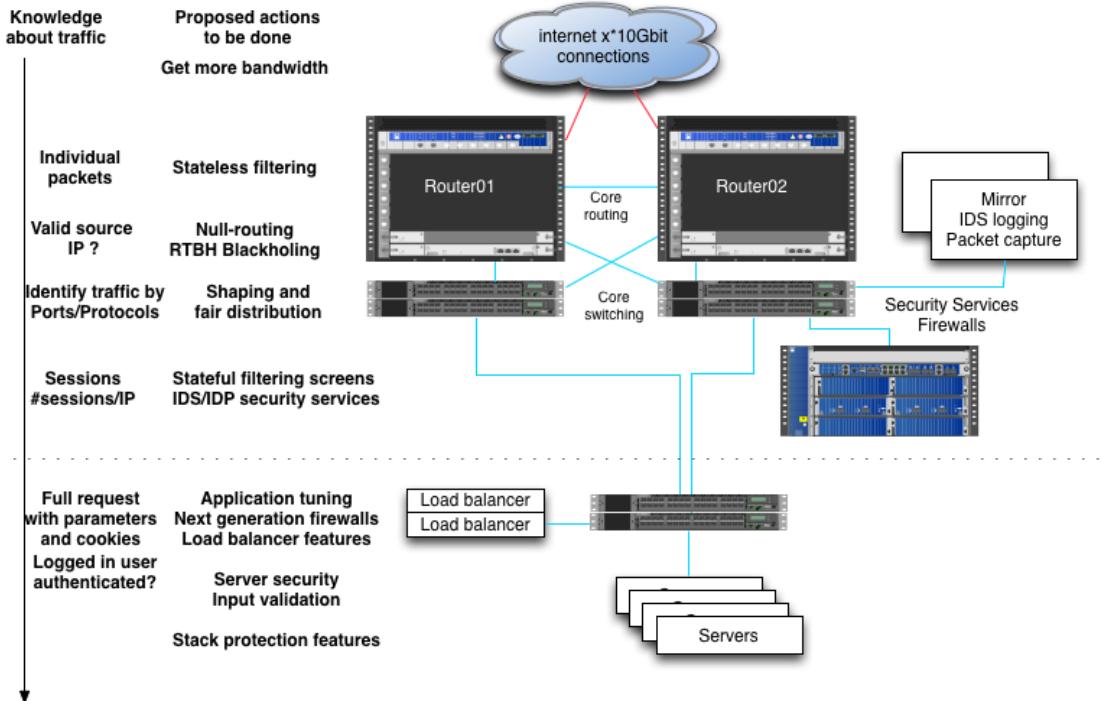
# Uncomplicated Firewall (UFW)

```
root@debian01:~# apt install ufw
...
root@debian01:~# ufw allow 22/tcp
Rules updated
Rules updated (v6)
root@debian01:~# ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
root@debian01:~# ufw status numbered
Status: active
```

To	Action	From
--	-----	----
[ 1] 22/tcp	ALLOW IN	Anywhere
[ 2] 22/tcp (v6)	ALLOW IN	Anywhere (v6)

- Extremely easy to use – I recommend and use the (Uncomplicated Firewall) UFW

# Firewalls are NOT Alone

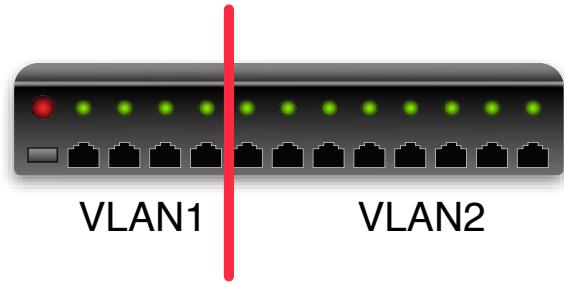


Use Defense in Depth – all layers have features



## Together with Firewalls - Virtual LAN (VLAN)

Portbased VLAN



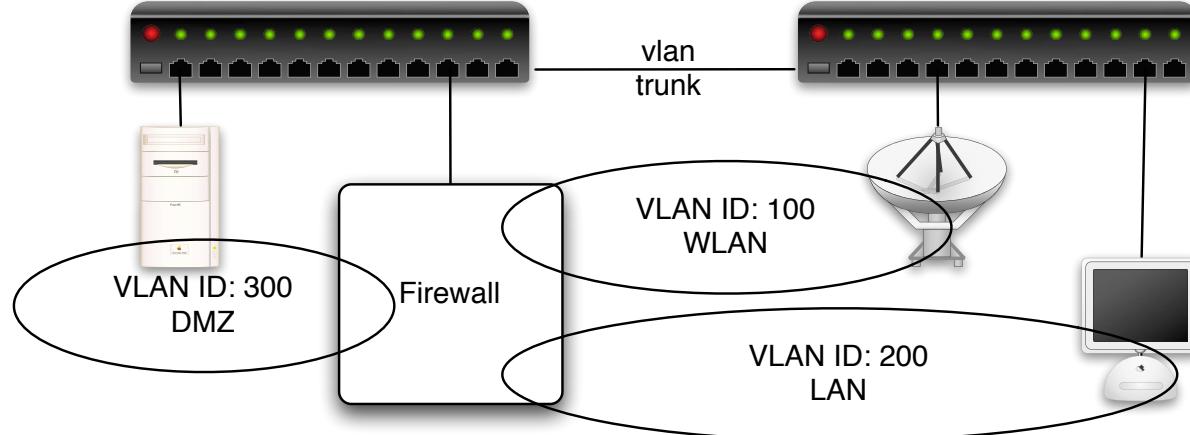
Managed switches often allow splitting into zones called virtual LANs

Most simple version is port based

Like putting ports 1-4 into one LAN and remaining in another LAN

Packets must traverse a router or firewall to cross between VLANs

# Virtual LAN (VLAN) IEEE 802.1q



Using IEEE 802.1q VLAN tagging on Ethernet frames

Virtual LAN, to pass from one to another, must use a router/firewall

Allows separation/segmentation and protects traffic from many security issues

# ARP in IPv4



Server



10.0.0.1

00:30:65:22:94:a1

IP adresser



MAC adresser - Ethernet

Client



10.0.0.2

00:40:70:12:95:1c



## ARP request and reply

**ping 10.0.0.2 from server**

ARP Address Resolution Protocol request/reply:

- ARP request broadcasted on layer 2 - Who has 10.0.0.2 Tell 10.0.0.1
- ARP reply (from 10.0.0.2) 10.0.0.2 is at 00:40:70:12:95:1c

IP ICMP request/reply:

- Echo (ping) request from 10.0.0.1 to 10.0.0.2
- Echo (ping) reply from 10.0.0.2 to 10.0.0.1
- ...

ARP is performed on Ethernet before IP can be transmitted



# IPv6 neighbor discovery protocol (NDP)

OSI	IPv4	IPv6
Network	IP / ICMP	
Link	ARP	IPv6 / ICMPv6
Physical	Physical	Physical

NDP replaces ARP, compare arp -an and ndp -an

RFC4861 Neighbor Discovery for IP version 6 (IPv6)

# Hello neighbors



```
$ ping6 -w -I en1 ff02::1
PING6(72=40+8+24 bytes) fe80::223:6cff:fe9a:f52c%en1 --> ff02::1
30 bytes from fe80::223:6cff:fe9a:f52c%en1: bigfoot
36 bytes from fe80::216:cbff:feac:1d9f%en1: mike.kramse.dk.
38 bytes from fe80::200:aaff:feab:9f06%en1: xrx0000aab9f06
34 bytes from fe80::20d:93ff:fe4d:55fe%en1: harry.local
36 bytes from fe80::200:24ff:fec8:b24c%en1: kris.kramse.dk.
31 bytes from fe80::21b:63ff:fef5:38df%en1: airport5
32 bytes from fe80::216:cbff:fec4:403a%en1: main-base
44 bytes from fe80::217:f2ff:fee4:2156%en1: Base Station Koekken
35 bytes from fe80::21e:c2ff:feac:cd17%en1: arnold.local
```

# Exercise



Now lets do the exercise

## ⚠ Enable firewall - 15min

which is number **5** in the exercise PDF.