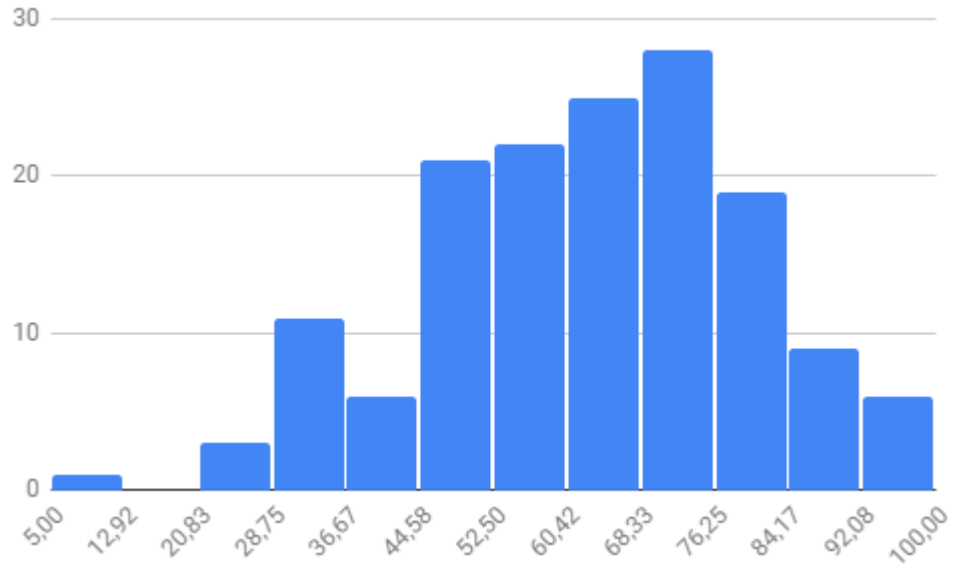
2nd midterm results

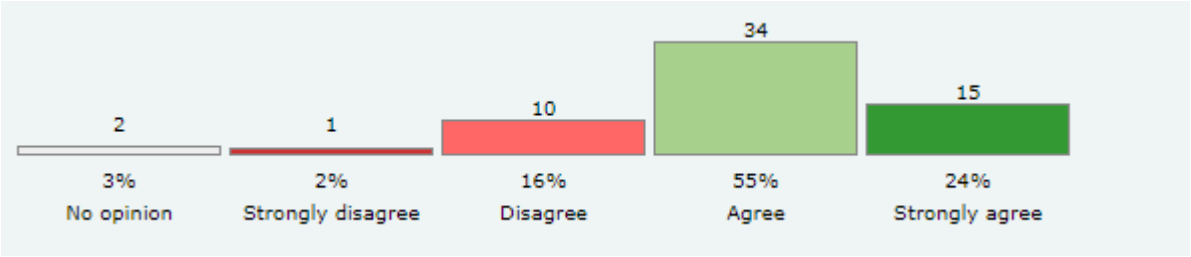


1. This is **NOT** your final grade
   (but less than 50 in both exams is worrisome)

2. **Revision**: Thursday 13th between 14 and 16h in BC329
   If you cannot come, send an email **before** Tuesday 11th

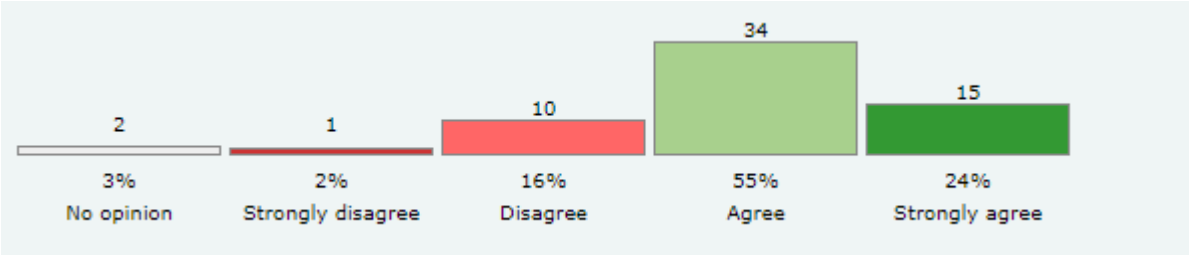
Intermediate assessment

Is the course good? (62 out of 181 inscribed)



Intermediate assessment

Is the course good? (62 out of 181 inscribed)

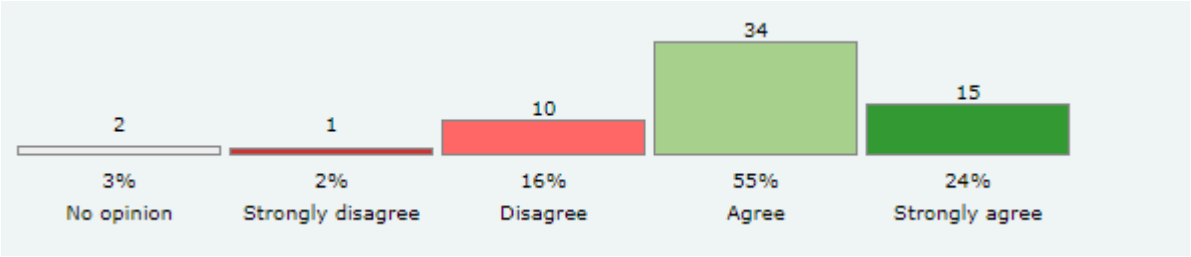


😔 This is far from ideal

☺️ We pass on our first attempt!

Intermediate assessment

Is the course good? (62 out of 181 inscribed)



😔 This is far from ideal

☺️ We pass on our first attempt!

- Lack of structure / lack of a reference book
- Slides lack content, difficult to study

- Too theoretical
- Boring / underexplained exercises

+ Speed adjustment!
+ Teaching team effort appreciated
    + we appreciate the positive comments too!
+ Interesting topics

Intermediate assessment

Best comment

```
----------------------------
Hint: This is EPFL, student DO NOT answer when you ask questions in class
----------------------------
```

Intermediate assessment

Best comment

```
----------------------------
Hint: This is EPFL, student DO NOT answer when you ask questions in class
----------------------------
```

Intermediate assessment

Best comment

Hint: This is EPFL, student DO NOT answer when you ask questions in class

---

Why we principles are important?

A Marauder's Map of
Security and Privacy in Machine Learning:
An overview of current and future research directions for making machine learning secure and private*

Nicolas Papernot
Google Brain
papernot@google.com

Abstract

There is growing recognition that machine learning (ML) exposes new security and privacy vulnerabilities in software systems, yet the technical community's understanding of the nature and extent of these vulnerabilities remains limited but expanding. In this talk, we explore the threat model space of ML algorithms through the lens of Saltzer and Schroeder's principles for the design of secure computer systems. This characterization of the threat space prompts an investigation of current and future research directions. We structure our discussion around three of these

Why we principles are important?

A Marauder's Map of
Security and Privacy in Machine Learning:
An overview of current and future research directions for making machine learning secure and private*

Nicolas Papernot
Google Brain
papernot@google.com

Abstract

There is growing recognition that machine learning (ML) exposes new security and privacy vulnerabilities in software systems, yet the technical community's understanding of the nature and extent of these vulnerabilities remains limited but expanding. In this talk, we explore the threat model space of ML algorithms through the lens of Saltzer and Schroeder's principles for the design of secure computer systems. This characterization of the threat space prompts an investigation of current and future research directions. We structure our discussion around three of these

Fail-safe defaults. Classifier should not give an answer if no confident

Open Design. Classifiers can be attacked even without knowing their inner workings

Separation of Privilege. Federated learning (decentralized learning)

Economy of mechanism. Simple clear interfaces help.

Complete mediation. Ideally we would like to verify every input/output

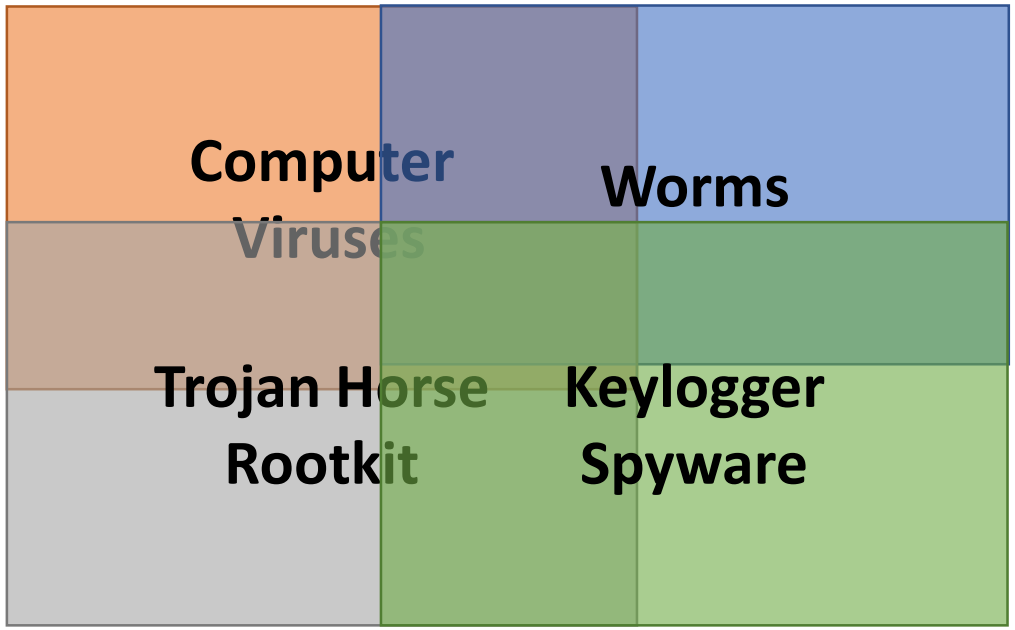
Least privilege. Let the ML learn as little as possible. Then it cannot leak

. . .

Last week – Malware

MALWARE (MALICIOUS SOFTWARE)

Software i**ntentionally** written to cause adverse effects



Modern malware combines "the best" of the categories to achieve its purpose

Last week – Virus

Piece of software

infects programs /macros / OS – inserts code on other executables
it **executes** secretly when the program executes
has the same permissions as the program

Goal: monitor operation / steal sensitive data / destroy systems

Replicates to infect other content or machines
through the network (web, email) or using hardware

Defenses: antivirus (signature / behavioural), sandbox

Last week – Worm

Computer program

self-replicating

email harvesting

scanning (random or targeted)

Goal: transport virus or other malware, Denial of Service

Defenses:

Host-level: Protecting software from remote exploitation (stack protection techniques), achieve diversity, antivirus.

Network-level: Limit the number of outgoing connections, block unknown SMTP connections, Intrusion detection systems

Last week – Intrusion detection systems

Run in the host (monitor files/processes) or in the network (monitor traffic)

Signature based vs. Anomaly-based detection

Signature: identifies known patterns

+ low false alarms

- expensive (need up-to-date signatures), can't find new attacks

Anomaly: attempts to identify behavior different than legitimate

+ adapt to new attacks (legitimate does not change!)

- high false alarms

Last week – Trojan

Piece of software

hidden in an apparently benign software

acts when the program is executed

cannot replicate on its own

Goal: any malicious task (steal, monitor, …)

Defenses:

Train the user to not download / execute

Antivirus can help

Last week – Rootkits and backdoors

Rootkit malware **installed inside the TCB**

Replace system programs with trojaned versions

Modify kernel data structures to hide processes, files, and network activities

Defense: very difficult (integrity checkers at kernel level)

Last week – Rootkits and backdoors

Rootkit malware **installed inside the TCB**

Replace system programs with trojaned versions

Modify kernel data structures to hide processes, files, and network activities

Defense: very difficult (integrity checkers at kernel level)

Backdoor **hidden** functionality that allows the adversary to bypass some security mechanism

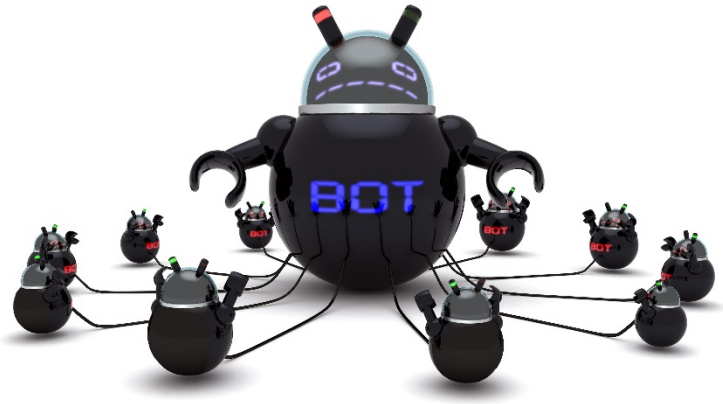
any of the previous malware may create it

Defense: if you can't trust, try to get more votes!

New content!!

Botnets

Attacks at scale!!



Multiple (millions) compromised **hosts** under the control of a single entity

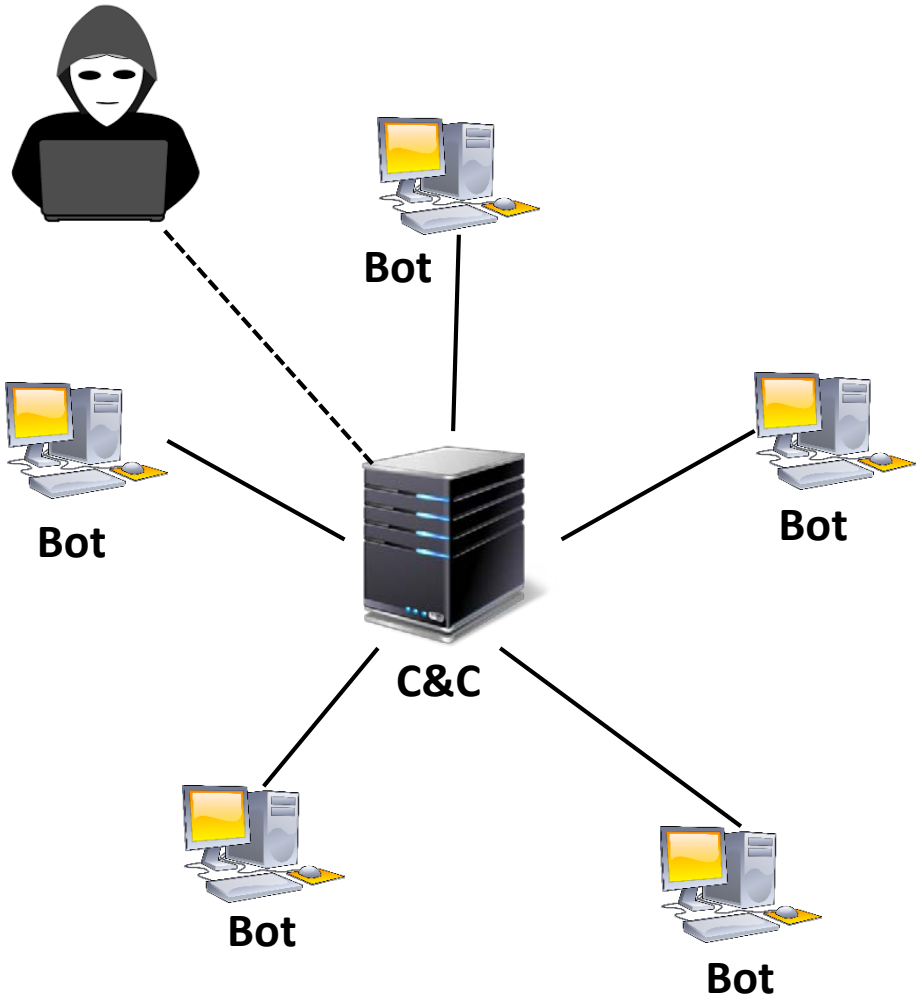
“zombies” or “bots”

uses

Bot-net command & control (C&C)

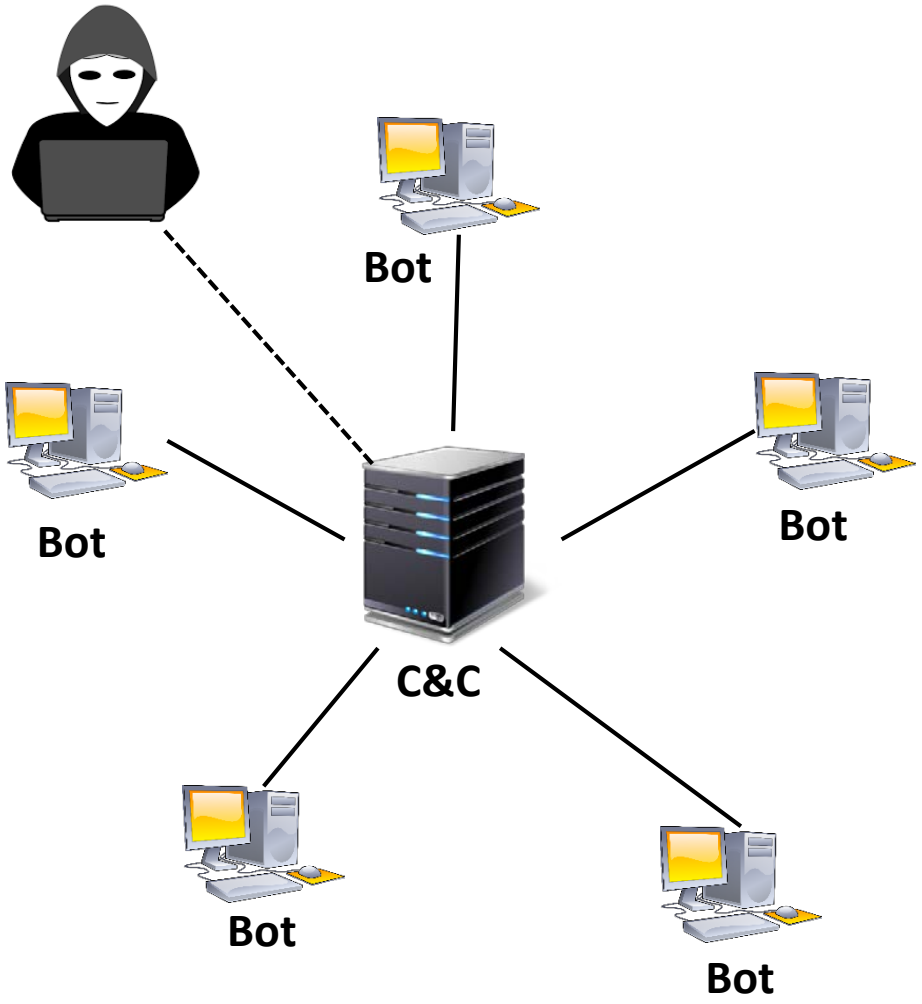
System to keep track of bots and send commands to them

Botnets - Star Topology



What is the problem here?

Botnets - Star Topology

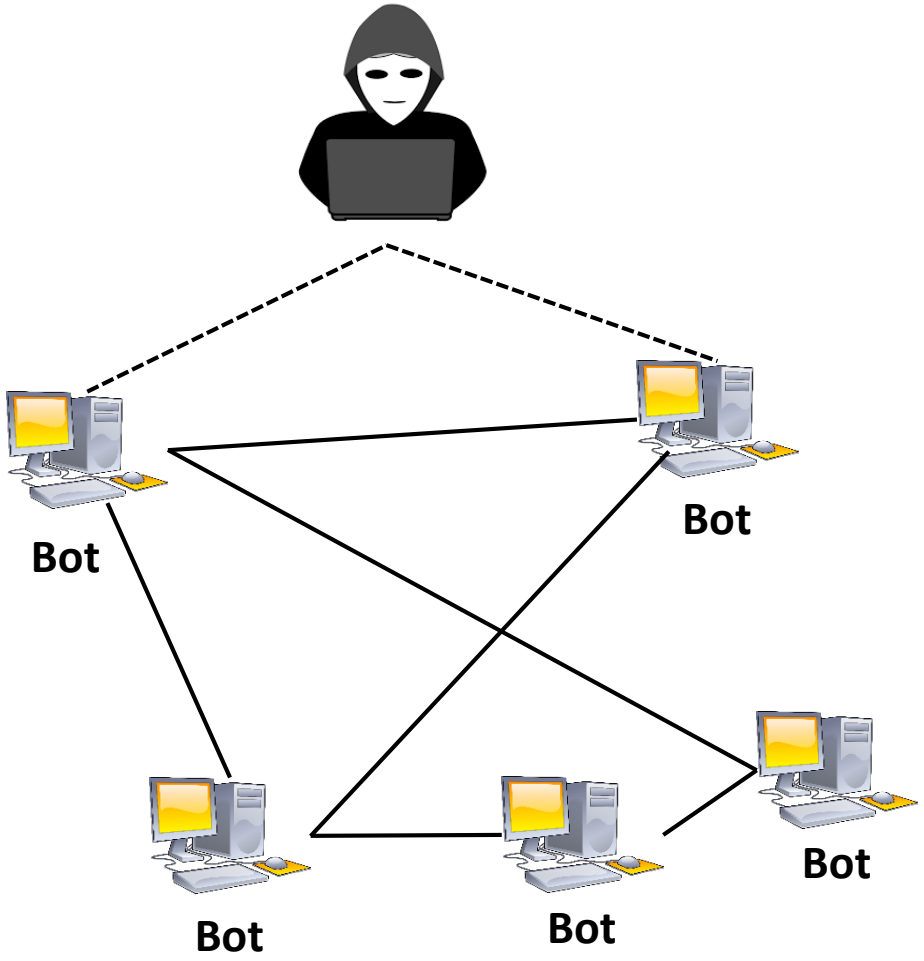


What is the problem here?

C&C single point of failure

the botnet violates the least common mechanism principle!

Botnets – P2P Topology

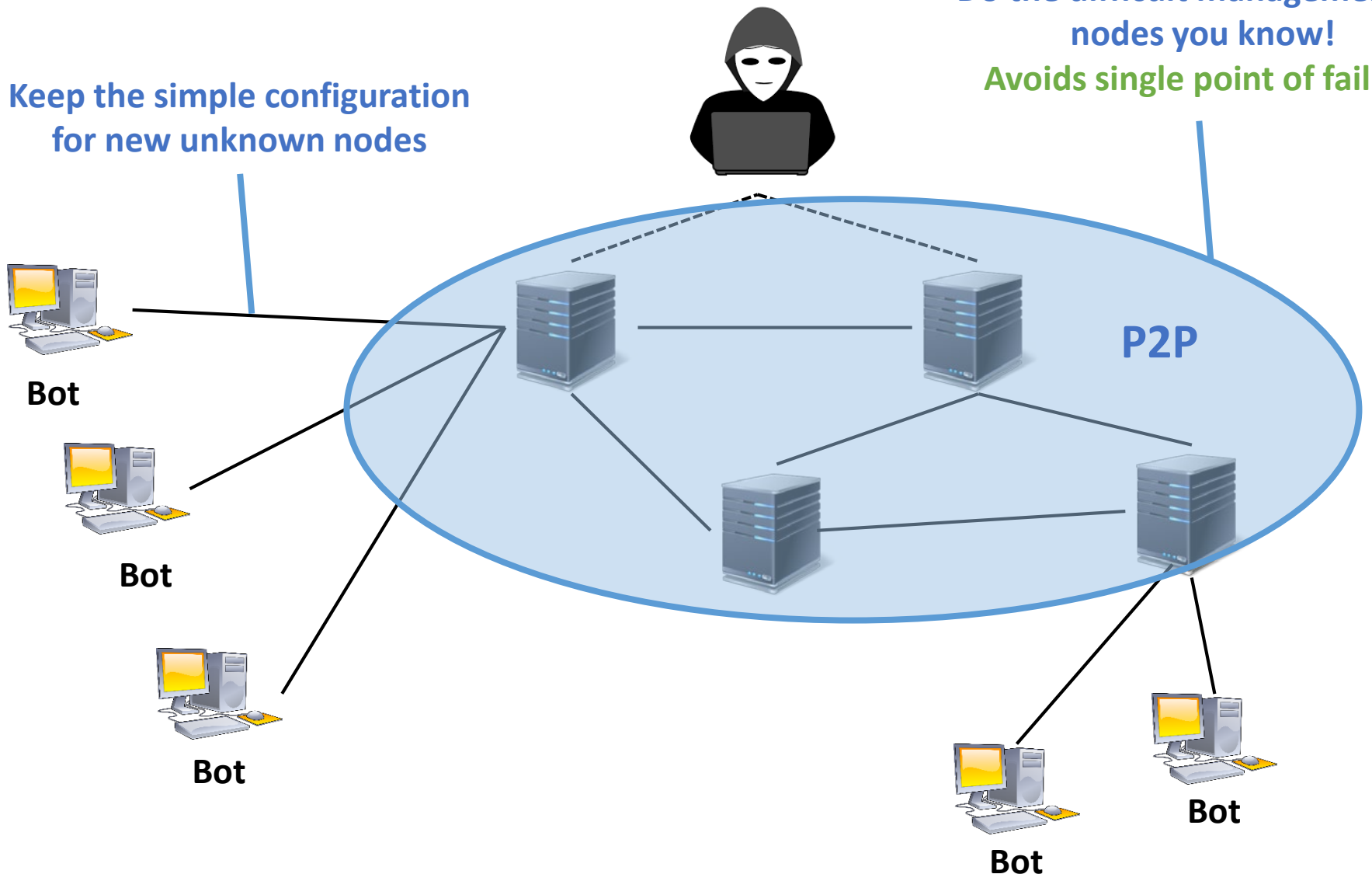


No Command and Control!!

Difficult management (join? leave?)

Vulnerable to attacks in which too many bots are taken over
(these are called Sybil attacks)

Botnets – Hybrid



Monetizing Botnets

Rental – “Pay me money, and I’ll let you use my botnet…”

DDoS extortion – “Pay me or I take down you legitimate business”

Bulk traffic selling – “Pay me to boost visit counts on your website”

Click fraud – “Simulate clicks on advertised links to generate revenue”

Distribute Ransomware – “I’ve encrypted your hard drive, pay!”

Advertise products – “Pay me, I will leave comments all around the web”

Bitcoin mining!!

…

DDoS Botnet Evolution

Trend Highlights:
-Bitcoin has allowed monetization of botnets
-Botnet threat isn't new, but attacker motivations have shifted
-Rapid growth in IoT is fueling the current botnet growth

2000-2005
Objective: Spam, limited DDoS
Examples:
-**Storm** (160k nodes)
-**marina** (6.2M nodes)

2005-2010
Objective: Spam, limited DDoS
Examples:
-**Akbot** (1.3M nodes)
-**BredoLab** (32M nodes)
-**Festi** (250k nodes)

2010-2015
Objective: Spam, DDoS for hire, extortion
Examples:
-**Kelihos** (300k nodes)
-**BASHLITE** (1M nodes)

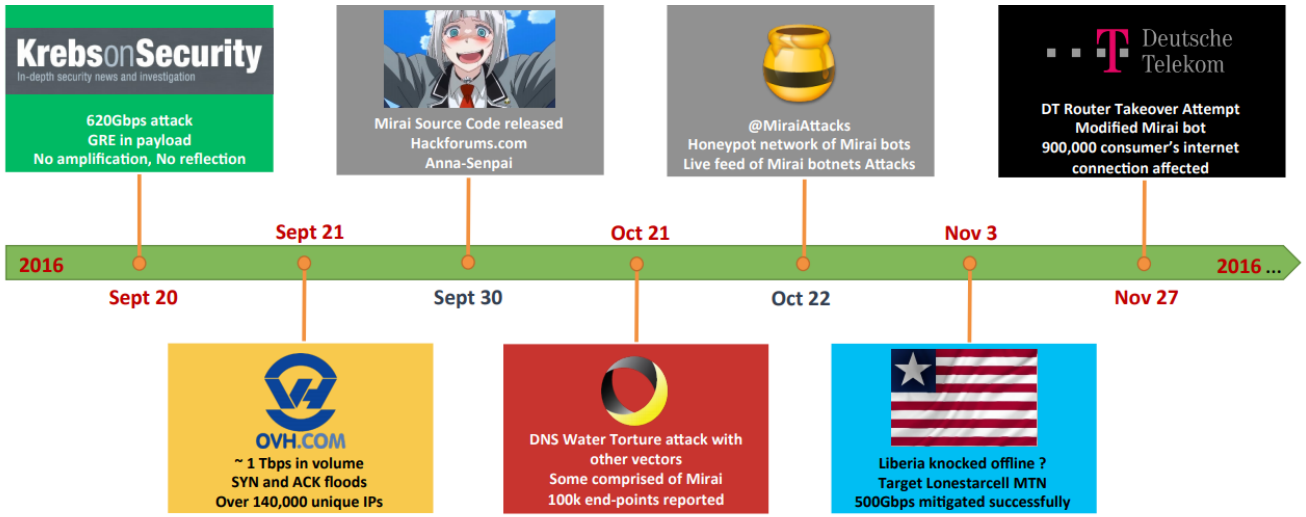
2015-future
Objective: Spam, DDoS for hire, extortion, IoT
Examples:
-**Miral** (380k nodes)

https://www.incapsula.com/blog/how-to-identify-a-mirai-style-ddos-attack.html

Example Botnet – Mirai (2016)

Target: IoT devices

scanning of Telnet ports, attempted to log in using 61 username/password combos



Open source code – variants appear all the time

Wicked (2018): scans ports 8080, 8443, 80, and 81 and attempts to locate vulnerable, unpatched IoT devices running on those ports.

https://www.nanog.org/sites/default/files/1_Winward_Mirai_The_Rise_v1.pdf
https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-antonakakis.pdf

Botnets: defense

Attack C&C infrastructure

Take communication channel off-line

Hijack/poison DNS to route traffic to black hole

Honeypots

Vulnerable computer that serves no purpose other than to attract attackers and study their behavior in controlled environments

Study botnet behavior to find defense (or study ecosystem)

Other malware

Rabbit: code that replicates itself w/o limit to exhaust resources

Logic (time) bomb: code that triggers action when condition (time) occurs

Dropper: code that drops other malicious code

Tool/toolkit: program used to assemble malicious code (not malicious itself)

Scareware: false warning of malicious code attack

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

SPRING
SECURITY AND PRIVACY ENGINEERING LABORATORY

Computer Security (COM-301)

Network security

Carmela Troncoso

SPRING Lab

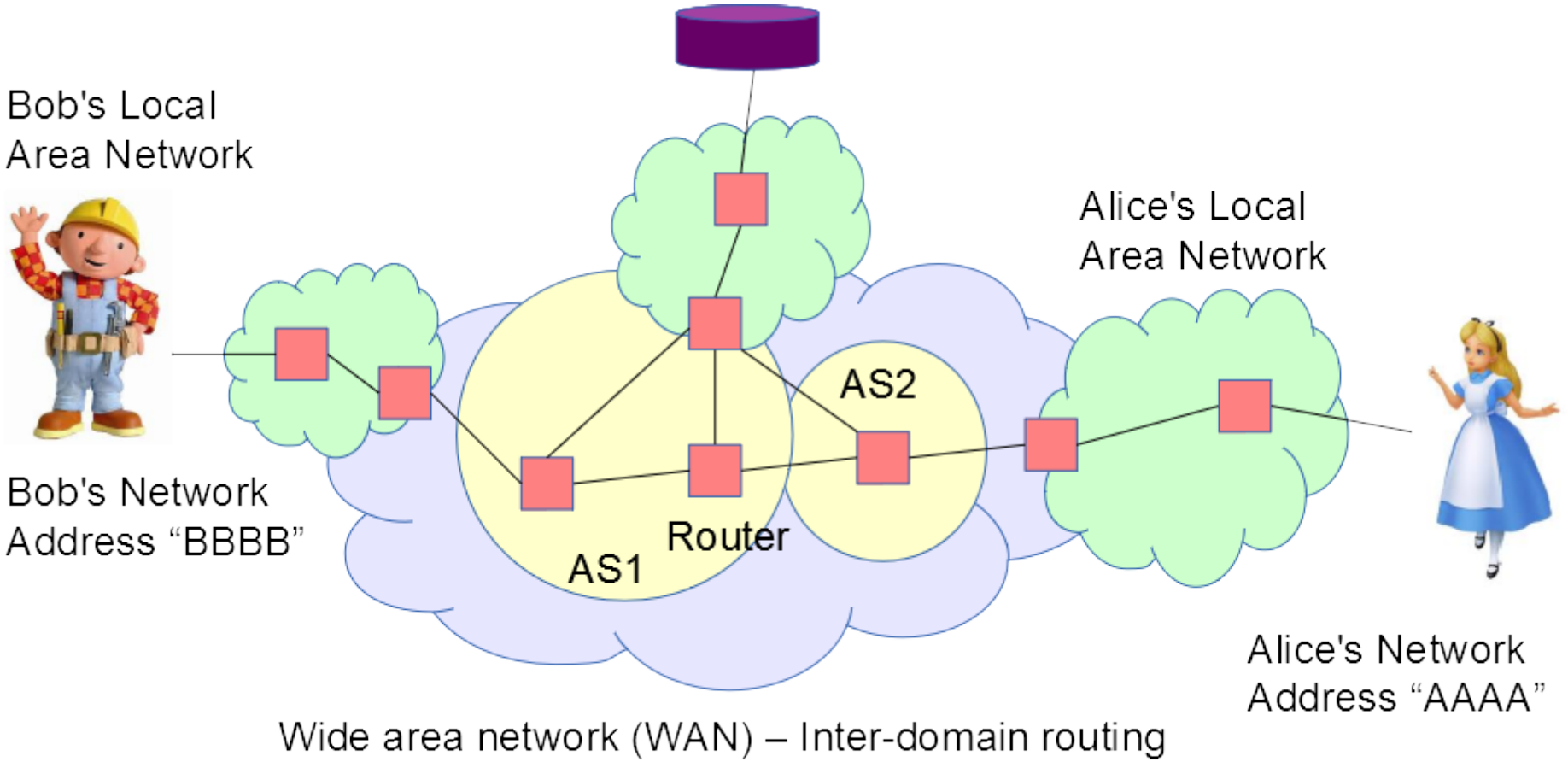
carmela.troncoso@epfl.ch

Some slides/ideas adapted from: George Danezis

Up to here: attacks on hosts
What about the network?

Up to here: attacks on hosts
What about the network?

The network is not a tube!!!



Desired properties

Confidentiality, Integrity, Availability, Authentication, Authorization?

Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary
Session security: Messages within the same session, cannot be modified (keep ordering and no adding/removing messages)
Content security: The content of the messages must not be readable or influenced by adversaries

Desired properties

Confidentiality, Integrity, Availability, Authentication, Authorization?

Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary	**Integrity** **Authentication** **Availability (naming service)**
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	
Session security: Messages within the same session, cannot be modified (keep ordering and no adding/removing messages)	
Content security: The content of the messages must not be readable or influenced by adversaries	

Desired properties

Confidentiality, Integrity, Availability, Authentication, Authorization?

Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary	**Integrity** **Authentication** **Availability (naming service)**
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	**Integrity** **Authentication** **Availability** **Authorization**
Session security: Messages within the same session, cannot be modified (keep ordering and no adding/removing messages)	
Content security: The content of the messages must not be readable or influenced by adversaries	

Desired properties

Confidentiality, Integrity, Availability, Authentication, Authorization?

Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary	**Integrity** **Authentication** **Availability (naming service)**
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	**Integrity** **Authentication** **Availability** **Authorization**
Session security: Messages within the same session, cannot be modified (keep ordering and no adding/removing messages)	**Integrity** **Authentication**
Content security: The content of the messages must not be readable or influenced by adversaries	

Desired properties

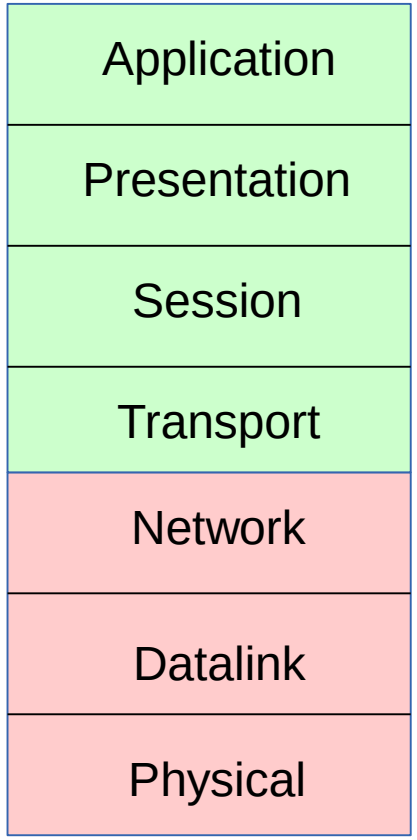
Confidentiality, Integrity, Availability, Authentication, Authorization?

Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary	**Integrity** **Authentication** **Availability (naming service)**
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	**Integrity** **Authentication** **Availability** **Authorization**
Session security: Messages within the same session, cannot be modified (keep ordering and no adding/removing messages)	**Integrity** **Authentication**
Content security: The content of the messages must not be readable or influenced by adversaries	**Confidentiality** **Integrity**

This lecture

- Do deployed network protocols provide the desired properties?
  - Naming security
  - Routing security
  - Session security
  - Content security
- What are the existing solutions to improve network security?

Where are the problems?



Open Systems Initiative (OSI) Model '94

Web, Bittorrent, SMTP/POP/IMAP, XMPP/IRC, VoIP

(SSL, TLS)

Transmission Control Protocol (TCP), UDP

Internet Protocol (IP) (Naming and routing: DNS, BGP)

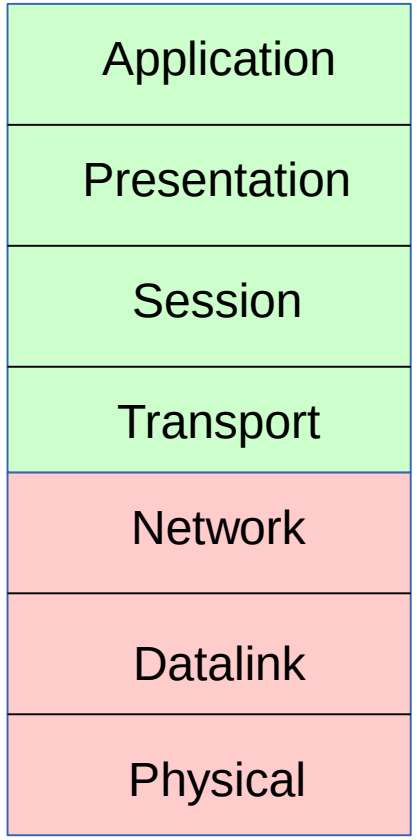
IEEE802.3 (Ethernet) (Naming & routing: ARP)

Modulation & coding

...

Where are the problems?

In this lecture security issues on…



Open Systems Initiative (OSI) Model '94

Web, Bittorrent, SMTP/POP/IMAP, XMPP/IRC, VoIP

(SSL, TLS)

Transmission Control Protocol (TCP) UDP

Internet Protocol (IP) (Naming and routing: DNS, BGP)

IEEE802.3 (Ethernet) (Naming & routing ARP)

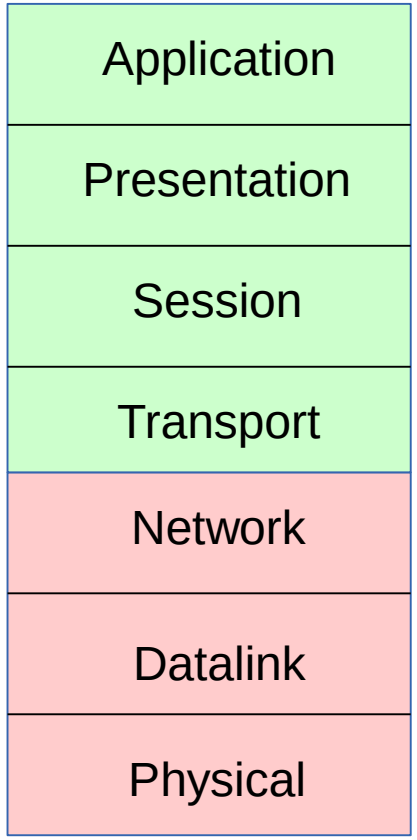
Modulation & coding

...

Naming security
Routing security
Session security
Content security

Where are the problems?

In this lecture security issues on…



Open Systems Initiative (OSI) Model '94

Web, Bittorrent, SMTP/POP/IMAP, XMPP/IRC, VoIP

(SSL, TLS)

Transmission Control Protocol (TCP) UDP

Internet Protocol (IP) (Naming and routing: DNS, BGP)

IEEE802.3 (Ethernet) (Naming & routing ARP)

Modulation & coding

...

We will not see UDP, similar concepts, but differences in the implementation

Naming security
Routing security
Session security
Content security

Routing: routing IP on an Ethernet LAN

Refresher

- **Ethernet:**
  - Local area network (LAN) technology
  - Machines have a “unique” 48 bit MAC address (Medium Access Code)

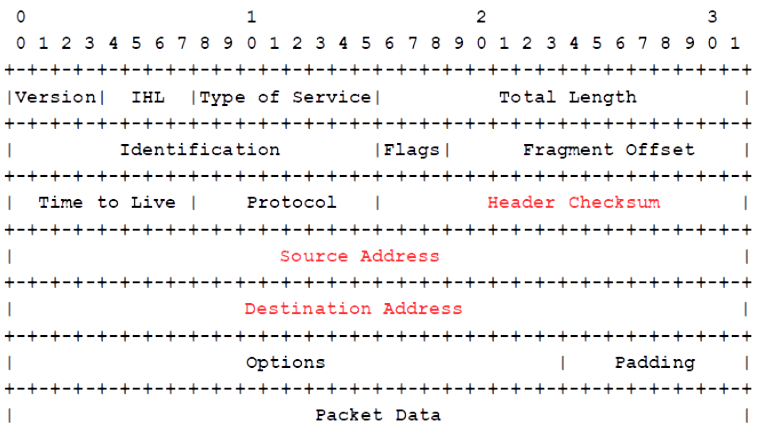
RFC791 “INTERNET PROTOCOL” http://www.ietf.org/rfc/rfc791.txt

Routing: routing IP on an Ethernet LAN

Refresher

- **Ethernet:**
  - Local area network (LAN) technology
  - Machines have a “unique” 48 bit MAC address (Medium Access Code)

- **Internet Protocol (IP)** on the LAN
  - Hosts communicate using the IP protocol
  - Each machine has an IP address (4 bytes in IPv4).
    - Part of the address denotes the network and part the host



Example Internet Datagram Header

RFC791 “INTERNET PROTOCOL” http://www.ietf.org/rfc/rfc791.txt

Routing: routing IP on an Ethernet LAN

Refresher

How does IP routing work?

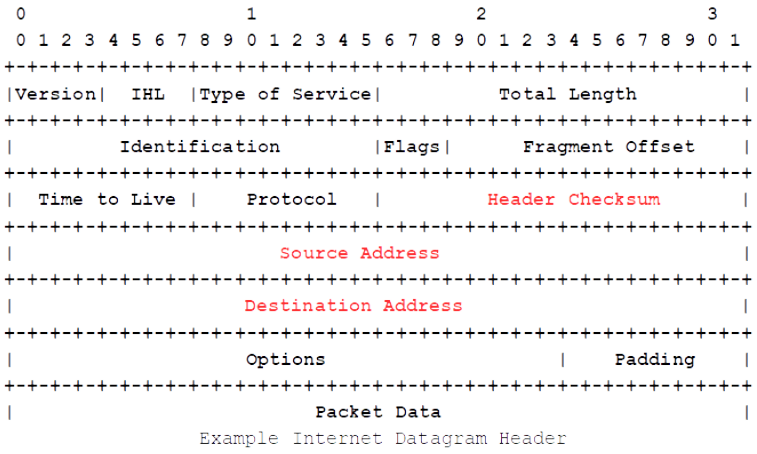
- Alice needs:
  - Her own IP address (eg. 192.128.5.130)
  - Bob's IP address (eg. 192.128.5.125)
  - Her “subnet mask” (eg. 255.255.255.0)
  - Her “gateway” (eg. 192.128.5.1)

- Option 1: Alice and Bob are on the same subnet
  - Address Alice AND mask = Address Bob AND mask
  - Route through the LAN

- Option 2: they are on different subnets
  - Send to gateway
  - Route through the WAN (Wide Area Network)

Alice

Send this packet from address 192.128.5.130 to 192.128.5.125



Example Internet Datagram Header

Routing: routing IP on an Ethernet LAN

Refresher

How does IP routing work?

Send this packet from address 192.128.5.130 to 192.128.5.125

Alice

and inside the LAN?

- Alice does not (want to) know network details
- Alice does not know Bob's MAC address

How can she learn about Bob's MAC?

Routing: routing IP on an Ethernet LAN

Refresher

How does IP routing work?

Alice

Send this packet from address 192.128.5.130 to 192.128.5.125

and inside the LAN?

- Alice does not (want to) know network details
- Alice does not know Bob's MAC address

ARP: "translation" between IP address and MAC address

- Each host maintains a cached table of IP ↔ MAC mappings
- If not available: broadcast an ARP request to query for target IP
- An ARP reply responds with the MAC address for that IP

```
*-----------------------*
| HTYPE (2 bytes)       |
| PTYPE (2 bytes)       |
| HLEN (1)    | PLEN (1)|
| OPERATION (2)         |
| Sender HA (HLEN)      |
| Sender PA (PLEN)      |
| Target HA (HLEN)      |
| Target PA (PLEN)      |
*-----------------------*
```

Routing: routing IP on an Ethernet LAN

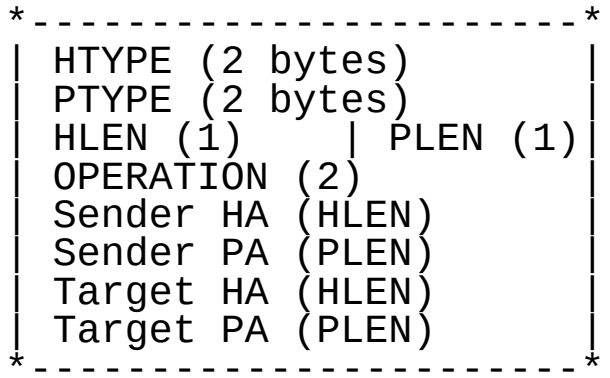
Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary

Integrity
Authentication

Does ARP provide naming security?

ARP: “translation” between IP address and MAC address

- Each host maintains a cached table of IP ↔ MAC mappings
- If not available: broadcast an ARP request to query for target IP
- An ARP reply responds with the MAC address for that IP



Routing: routing IP on an Ethernet LAN

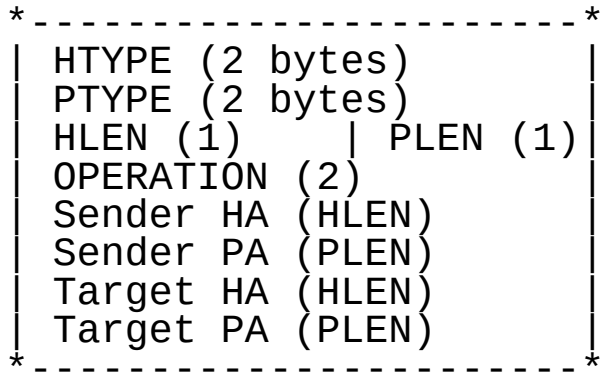
Naming security: The association between lower level names (eg. network addresses) and higher level names (e.g. Alice / Bob) must not be influenced by the adversary

Integrity
Authentication

Does ARP provide naming security?

ARP: “translation” between IP address and MAC address

- Each host maintains a cached table of IP ↔ MAC mappings
- If not available: broadcast an ARP request to query for target IP
- An ARP reply responds with the MAC address for that IP



No Integrity check, nor Authentication

ARP spoofing

If nobody checks…

You can impersonate! (provide the identity of others)

What can you achieve?

ARP spoofing

If nobody checks…

You can impersonate! (provide the identity of others)

What can you achieve?

- Just impersonation is bad

- Man in the middle: provide two hosts (sender/receiver) with your MAC address

- Monitor communication or tamper with it

- **Abuse resource allocation**

- **Denial of Service**: avoid that packets arrive to one host

ARP spoofing

If nobody checks…

You can impersonate! (provide the identity of others)

What can you achieve?

- Just impersonation is bad

- Man in the middle: provide two hosts (sender/receiver) with your MAC address

- Monitor communication or tamper with it

- Abuse resource allocation

Also bad for

Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	**Integrity** **Authentication** **Availability** **Authorization**

ARP spoofing

If nobody checks…

You can imperson

The same happens in DNS, IP, Ethernet,…
No network protocol was (initially) designed with security in mind!

BECAUSE OF A VERY NAÏVE THREAT MODEL:
outsiders are bad, insiders behave, trust them!

What can you achieve?

- Just impersonation is bad

- Man in the middle: provide two hosts (sender/receiver) with your MAC address

- Monitor communication or tamper with it

- Abuse resource allocation

Also bad for

Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary	**Integrity** **Authentication** **Availability** **Authorization**

ARP spoofing - Defenses

- Use of static, read-only entries for critical services in the ARP cache of a host

- Use ARP spoofing detection and prevention software
    - check if one IP has more than one MAC or one MAC reported by multiple IPs
    - certifies requests by cross-checking
    - sends email if IP-MAC association change

ARP spoofing - Defenses

- Use of static, read-only entries for critical services in the ARP cache of a host

- Use ARP spoofing detection and prevention software
    - check if one IP has more than one MAC or one MAC reported by multiple IPs
    - certifies requests by cross-checking
    - sends email if IP-MAC association change

ARP spoofing - Defenses

- Use of static, read-only entries for critical services in the ARP cache of a host

- Use ARP spoofing detection and prevention software
  - check if one IP has more than one MAC or one MAC reported by multiple IPs
  - certifies requests by cross-checking
  - sends email if IP-MAC association change

Separation of privilege: force the adversary to gain control of more entities

DNS spoofing – Attacks

DNS Spoofing

- **Cache poisoning**: corrupt the DNS resolver with fake pairs (IP,domain)
- **DNS Hijacking**: corrupt the DNS responses with fake pairs → censorship

What can you achieve?

DNS spoofing – Attacks

DNS Spoofing

Cache poisoning: corrupt the DNS resolver with fake pairs (IP,domain)

DNS Hijacking: corrupt the DNS responses with fake pairs → censorship

What can you achieve?

- **Denial of Service**: avoid that packets arrive to one host
- **Redirection**: reroute clients to malicious host
  - Malicious host attacks client (e.g., serving malware…)
  - Malicious host act as man in the middle (e.g., monitoring)

DNS spoofing – Defenses

Domain Name System Security Extensions (DNSSEC)

- Extensions to DNS that provide **origin authentication**
    - DNS responses are **digitally signed by authoritative resolvers** – prevents poisoning!
    - DNSSEC responses are not encrypted – **does not provide confidentiality**!

- 1st attempt (RFC 2535) 99-01: impractical, non-scalable, complex key management
- Nowadays (*DNSSEC-bis* RFC 4033): simplified messages and key management

DNS spoofing – Defenses

Domain Name System Security Extensions (DNSSEC)

- Extensions to DNS that provide **origin authentication**
    - DNS responses are **digitally signed by authoritative resolvers** – prevents poisoning!
    - DNSSEC responses are not encrypted – **does not provide confidentiality**!

- 1st attempt (RFC 2535) 99-01: impractical, non-scalable, complex key management
- Nowadays (*DNSSEC-bis* RFC 4033): simplified messages and key management

DNS-over-HTTPS (DoH) (RFC8484)

- Recent development – DNS queries over HTTPS connection (confidentiality & integrity)

- Deployed by Cloudflare (integrated in Firefox), Google, others

Others: DNS-over-TLS, DNSCrypt, DNSCurve

If we fix DNS, do we solve the routing problem?

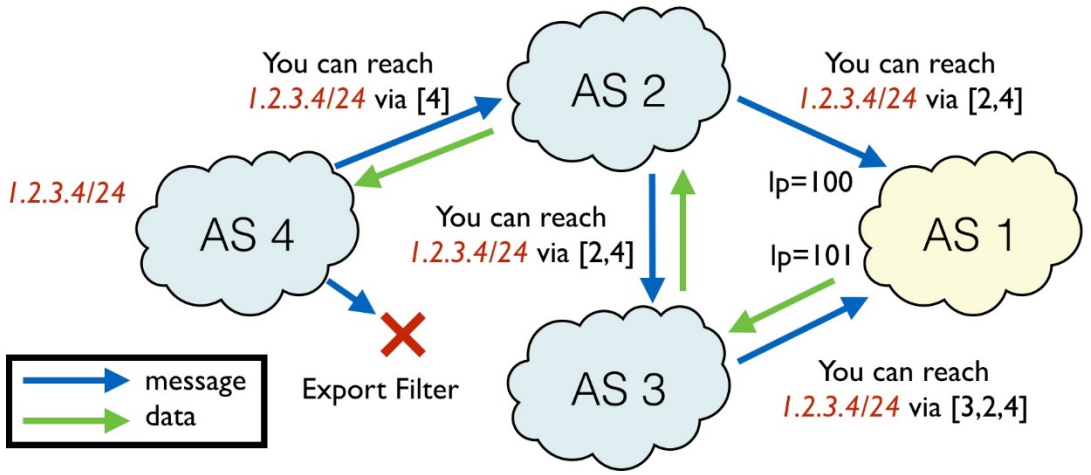
Routing security: The route over the network and the eventual delivery of messages must not be influenced by the adversary

Integrity
Authentication
Availability
Authorization

If we fix DNS, do we solve the routing problem? BGP (Border Gateway Protocol) (RFC 4271)

Refresher

- BGP **constructs the routing tables** between AS - Autonomous Systems with independent routing domains
  - Routers maintain tables of (IP subnet → Router IP, cost)
  - Routes change (faults, new contracts, new cables) - BGP updates constantly
  - Cost is **crucial**: the routes with lowest cost are chosen to route (real money!)



BGP Security

Weak authentication mechanism between routers (RFC 2385):

- Aimed at preventing DoS
- Short shared secret (up to 80 bytes of ASCII)
- Ad-hoc message authentication code based on the weak algorithm MD5

Does this guarantee the integrity of the advertised routes?

BGP Security

Weak authentication mechanism between routers (RFC 2385):

- Aimed at preventing DoS
- Short shared secret (up to 80 bytes of ASCII)
- Ad-hoc message authentication code based on the weak algorithm MD5

Does this guarantee the integrity of the advertised routes? NO!!

BGP Security

Weak authentication mechanism between routers (RFC 2385):

- Aimed at preventing DoS
- Short shared secret (up to 80 bytes of ASCII)
- Ad-hoc message authentication code based on the weak algorithm MD5

Does this guarantee the integrity of the advertised routes? NO!! BGP Hijacking!

- An adversary controls or compromises a router *somewhere* on the Internet
- Injects false low-cost routes to redirect portions of traffic to themselves
- The routing information propagates to routing tables until it expires

What can you achieve?

BGP Security

Weak authentication mechanism between routers (RFC 2385):

- Aimed at preventing DoS
- Short shared secret (up to 80 bytes of ASCII)
- Ad-hoc message authentication code based on the weak algorithm MD5

Does this guarantee the integrity of the advertised routes? NO!! BGP Hijacking!

- An adversary controls or compromises a router *somewhere* on the Internet
- Injects false low-cost routes to redirect portions of traffic to themselves
- The routing information propagates to routing tables until it expires

What can you achieve?

- **Redirection**: surveillance, injection, modification, or censorship.

Example 1: Belarus hijacks internet (2013)

- Global traffic redirected to Belarusian ISP GlobalOneBel.
  - Daily basis throughout February,
  - Changing set of victims: major financial institutions, governments, and network service providers.
  - Affected countries included the US, South Korea, Germany, the Czech Republic, Lithuania, Libya, and Iran

https://dyn.com/blog/mitm-internet-hijacking/

Example 2: Turkish Government hijacks Google and Level 3 DNS services (2014)

- Turkish government attempts to censor Twitter via DNS poisoning
  - Turkish citizens connect to other DNS services
    - Google DNS IP - 8.8.8.8 / 8.8.4.4
    - Level 3 – 4.2.2.1 / 4.2.2.2

- TurkTelekom (Turkish's national telecom provider) hijacked the DNS servers of using the Border Gateway Protocol (BGP)
  - Redirect citizens to their own DNS → serving their own content

https://dyn.com/blog/turkish-internet-censorship/
https://security.googleblog.com/2014/03/googles-public-dns-intercepted-in-turkey.html

BGP Spoofing - Defenses

Filtering could alleviate
(some routes should really not come from some routers).

But… there is no authority to guarantee the correctness of routes (all contractual).

Fundamental flaw (again): Design did not consider insiders as adversaries!

BGPsec

Each AS is given a certificate that links its verification key to its IP blocks.
Updates are only accepted if they are signed by the authority for the AS/IP Block.
Delegation is possible
Effort started in 2003 (RFC8205) -- weakly deployed

Spoofing: lesson to be learned

1. The network is hostile!

Routing security attacks, facilitated through **poor association of high level and low level names & addresses** (IP to Ethernet MAC / Router to router).

- **Threat model**: assumes network “insiders” are trusted to provide authoritative information.
- Also **no** integrity or confidentiality.

Spoofing: lesson to be learned

1. The network is hostile!

Routing security attacks, facilitated through **poor association of high level and low level names & addresses** (IP to Ethernet MAC / Router to router).

- **Threat model**: assumes network "insiders" are trusted to provide authoritative information.
- Also **no** integrity or confidentiality.

2. The solution is intimately linked to cryptography

Why? There is **no centralized authority** to act as either (a) originator of policy or (b) provide a trusted computing base

- Cryptography allows mutually distrustful actors to achieve some collective security properties
- Asymmetric cryptography (certificates and signatures) particularly useful for all to verify name and route associations!

Spoofing: lesson to be learned

1. The network is hostile!

Routing security attacks, facilitated through **poor association of high level and low level names & addresses** (IP to Ethernet MAC / Router to router).

- **Threat model**: assumes network "insiders" are trusted to provide authoritative information.
- Also **no** integrity or confidentiality.

2. The solution is intimately linked to cryptography

Why? There is **no centralized authority** to act as either (a) originator of policy or (b) provide a trusted computing base

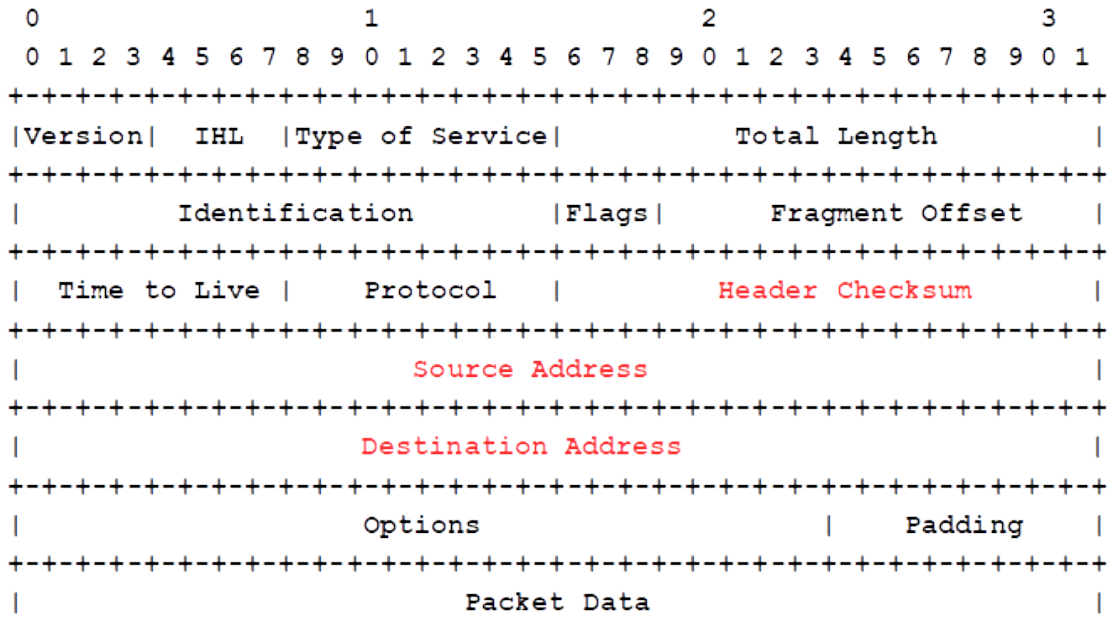
- Cryptography allows mutually distrustful actors to achieve some collective security properties
- Asymmetric cryptography (certificates and signatures) particularly useful for all to verify name and route associations!

But also… Who has authority?

Not a cryptographic question! related to name resolution & security policy

So what about IP?

Refresher



Example Internet Datagram Header

So what about IP?
IPSec - Internet Protocol Security

- Cryptographic security properties at the IP level
  - Key exchange based on public key cryptography or shared symmetric keys
  - **Authentication Header (AH)**: authentication & integrity (HMAC), protection from replay attacks (sequence number)
  - **Encapsulating Security Payload (ESP)**: confidentiality

So what about IP?
IPSec - Internet Protocol Security

- Cryptographic security properties at the IP level
  - Key exchange based on public key cryptography or shared symmetric keys
  - **Authentication Header (AH)**: authentication & integrity (HMAC), protection from replay attacks (sequence number)
  - **Encapsulating Security Payload (ESP)**: confidentiality

- Two modes:
  - **Transport**:

    protects IP packet payload using AH/ESP

    sent with the **original IP headers**
  - **Tunnel**:

    protects the whole packet (Headers + Payload) is protected and placed inside another packet

So what about IP?
IPSec - Internet Protocol Security

- Cryptographic security properties at the IP level
  - Key exchange based on public key cryptography or shared symmetric keys
  - **Authentication Header (AH)**: authentication & integrity (HMAC), protection from replay attacks (sequence number)
  - **Encapsulating Security Payload (ESP)**: confidentiality

- Two modes:
  - **Transport**:

    protects IP packet payload using AH/ESP

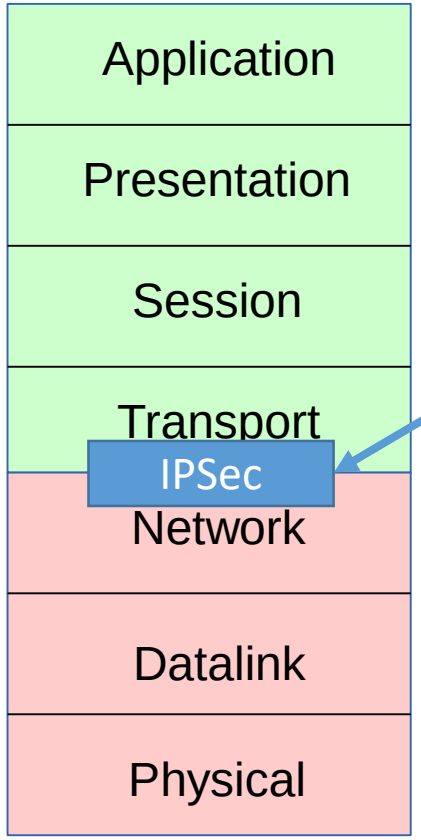
    sent with the original IP headers
  - **Tunnel**:

    protects the whole packet (Headers + Payload) is protected and placed inside another packet

Weak deployment

..but mandatory in IPv6

Where does IPSec happen?



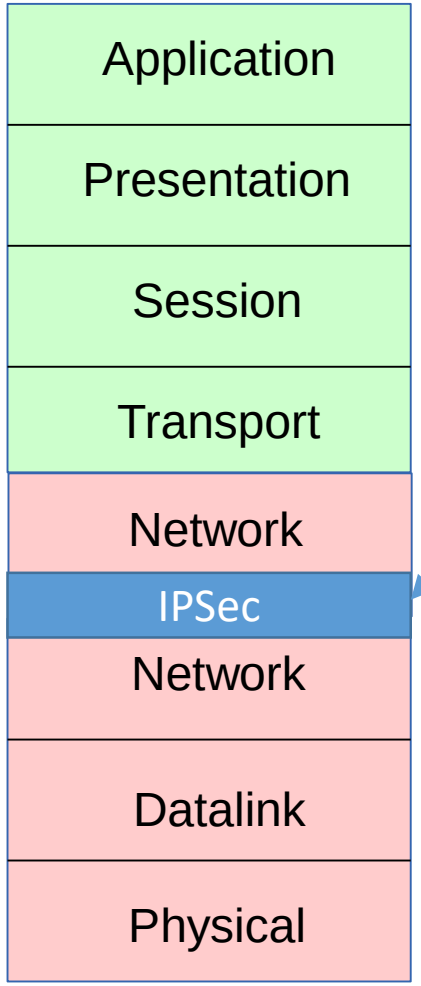
Open Systems Initiative (OSI) Model '94

IPSec in **TRANSPORT MODE**, **encrypts payload** but keeps the headers.

Transmission Control Protocol (TCP), UDP

Internet Protocol (IP)

Where does IPSec happen?



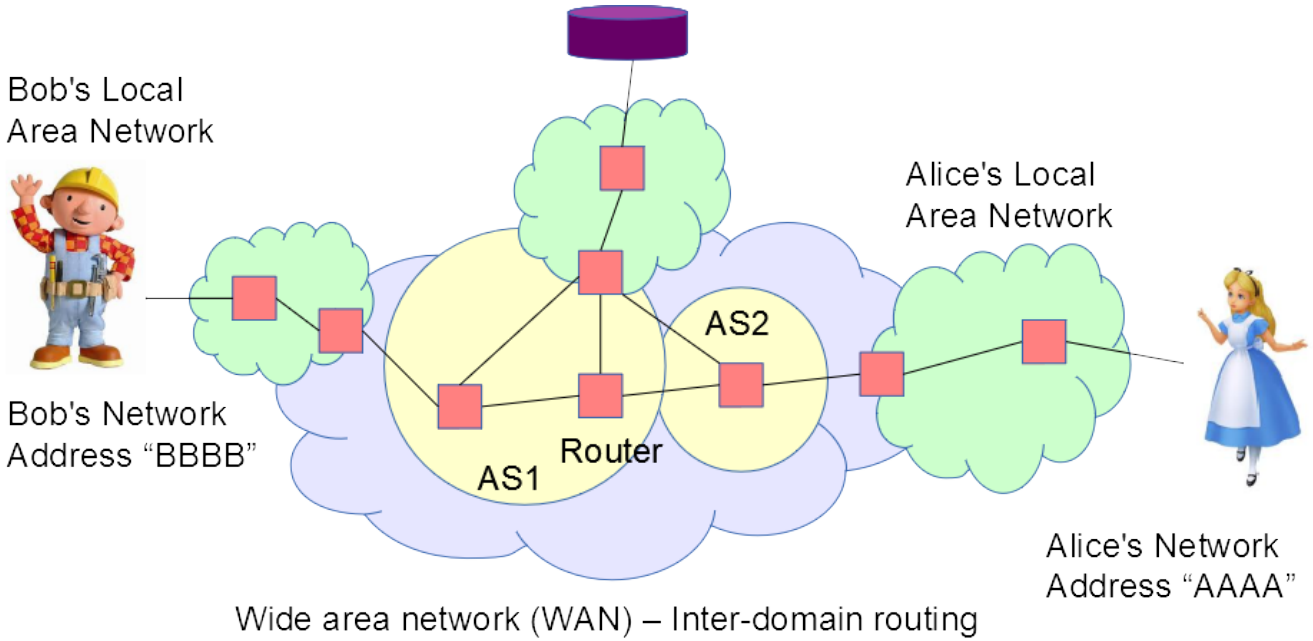
Open Systems Initiative (OSI) Model '94

IPSec in **TUNNEL MODE**, **<u>encrypts payload</u>** and **<u>the headers</u>**.

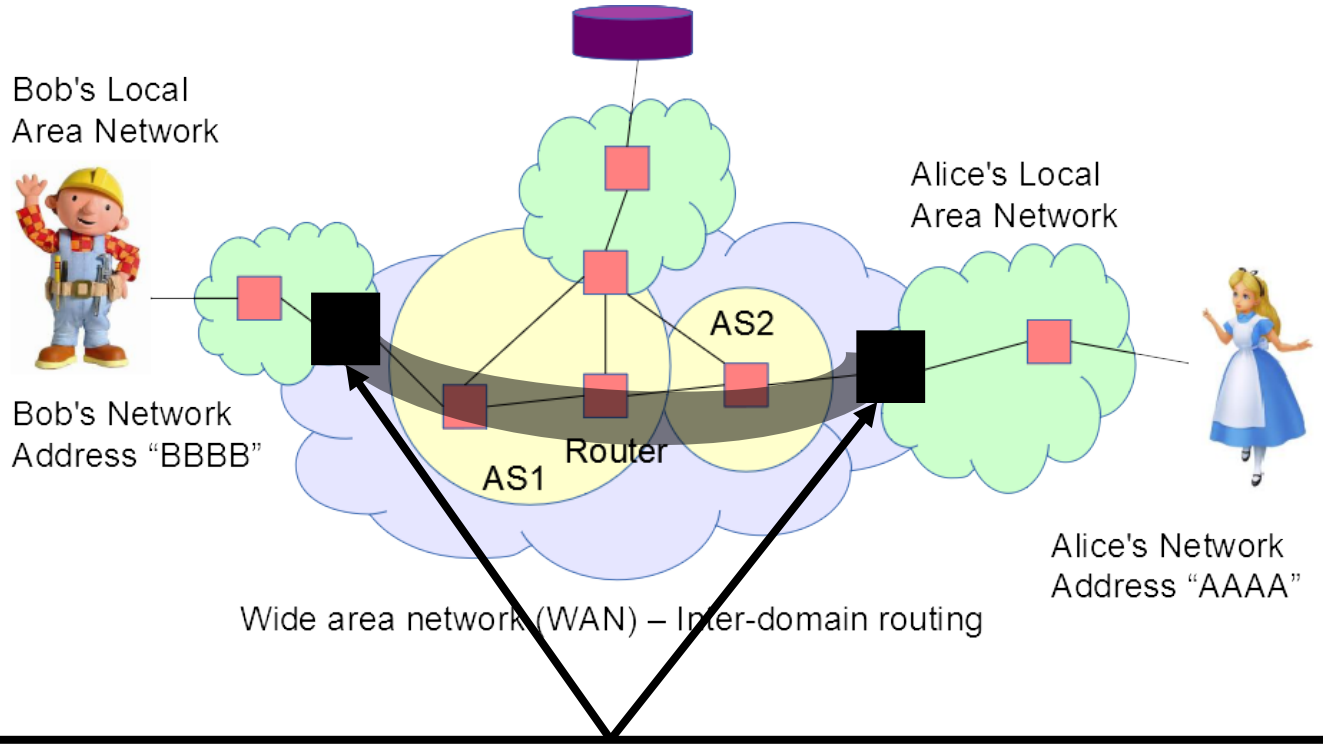
Transmission Control Protocol (TCP), UDP

Internet Protocol (IP)

Virtual Private Network



Virtual Private Network

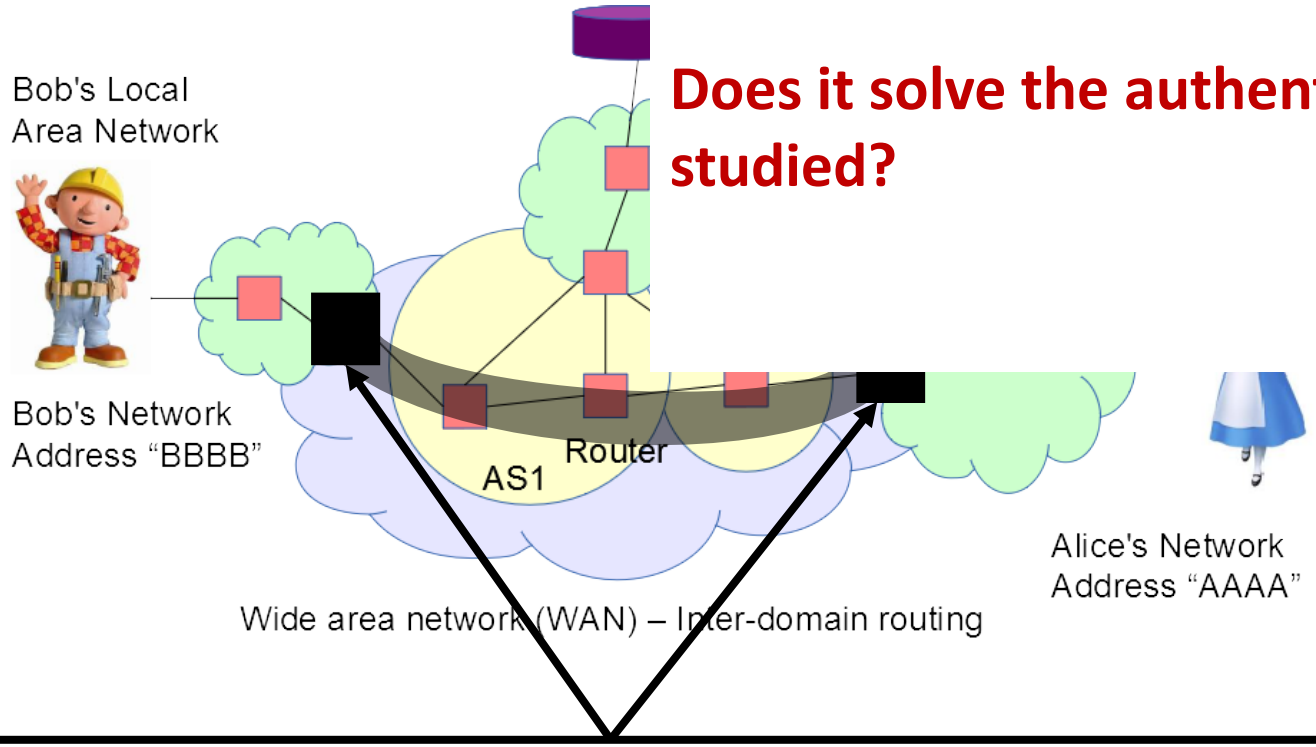


- IPSec in tunnel mode. The VPN
  - Looks like one single network
  - Routing internally
  - Inside VPN "tunnel" fully protected packets: confidentiality, authentication, integrity, reply

Virtual Private Network

Does it protect against Denial of Service?

Does it solve the authentication problem we have studied?

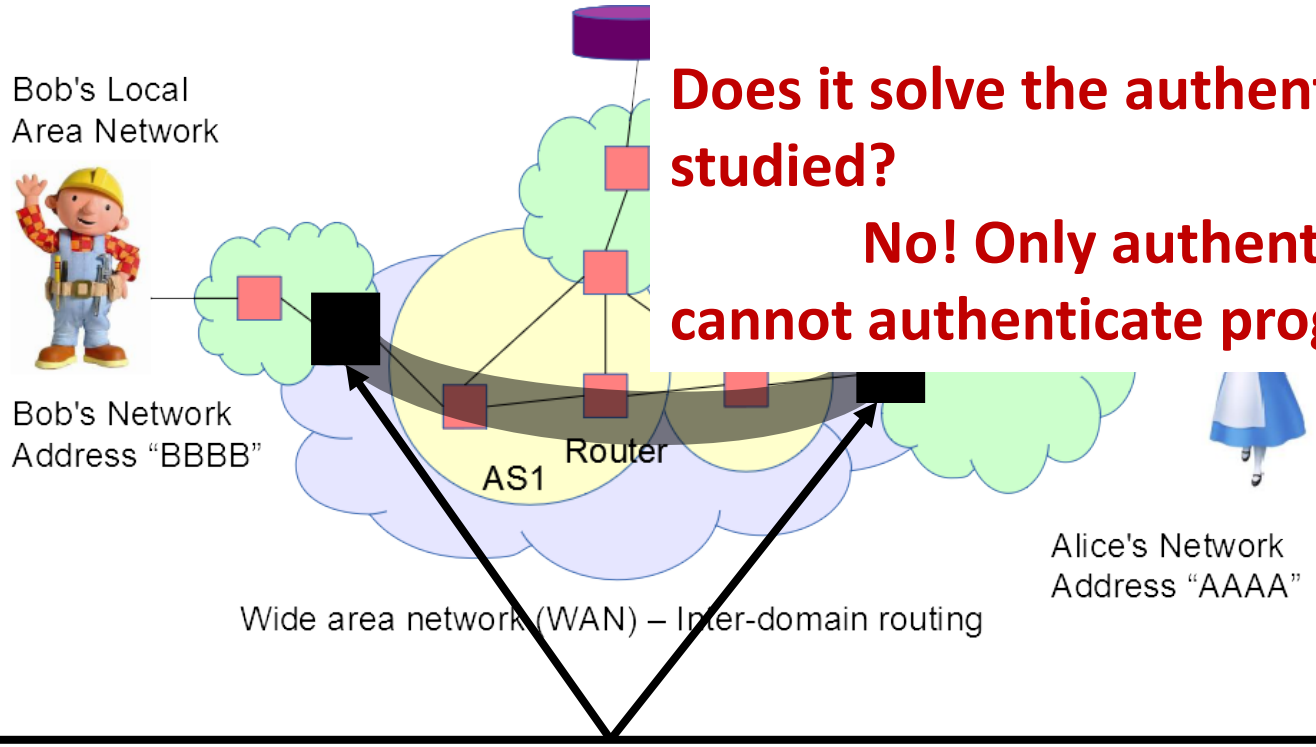


- IPSec in tunnel mode. The VPN
  - Looks like one single network
  - Routing internally
  - Inside VPN "tunnel" fully protected packets: confidentiality, authentication, integrity, reply

Virtual Private Network

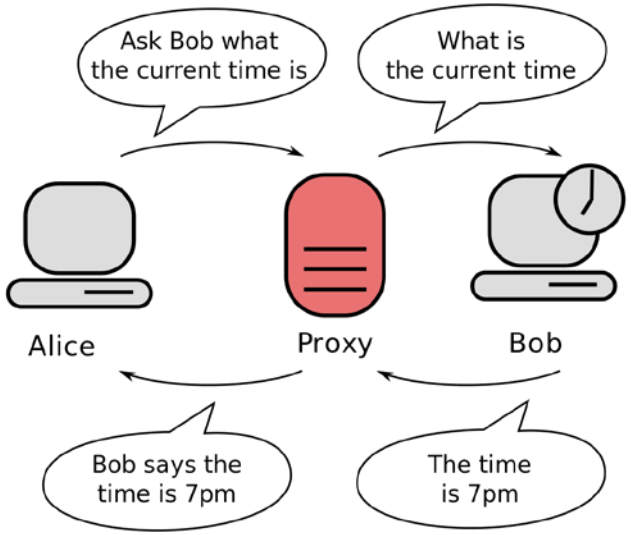
Does it protect against Denial of Service?
No! Your IP still exists

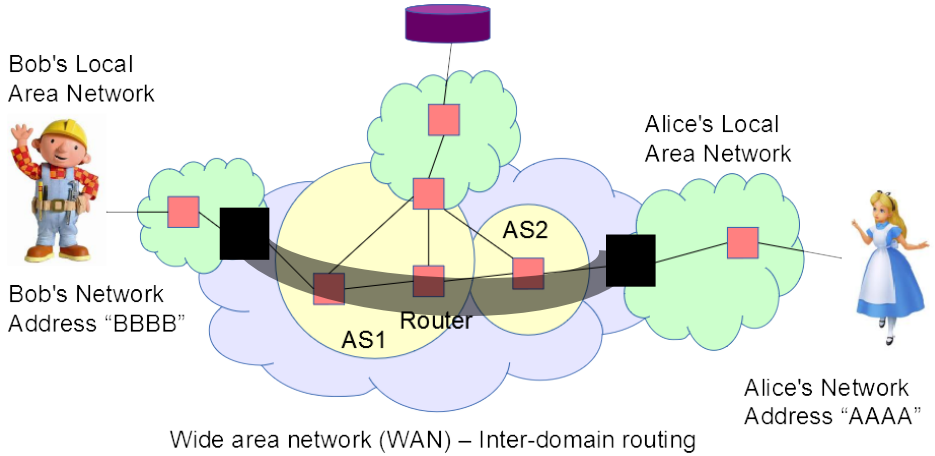
Does it solve the authentication problem we have studied?
No! Only authentication at network level. It cannot authenticate programs or applications



- IPSec in tunnel mode. The VPN
  - Looks like one single network
  - Routing internally
  - Inside VPN "tunnel" fully protected packets: confidentiality, authentication, integrity, reply

Is a VPN the same as a proxy?





IP limitations

Refresher

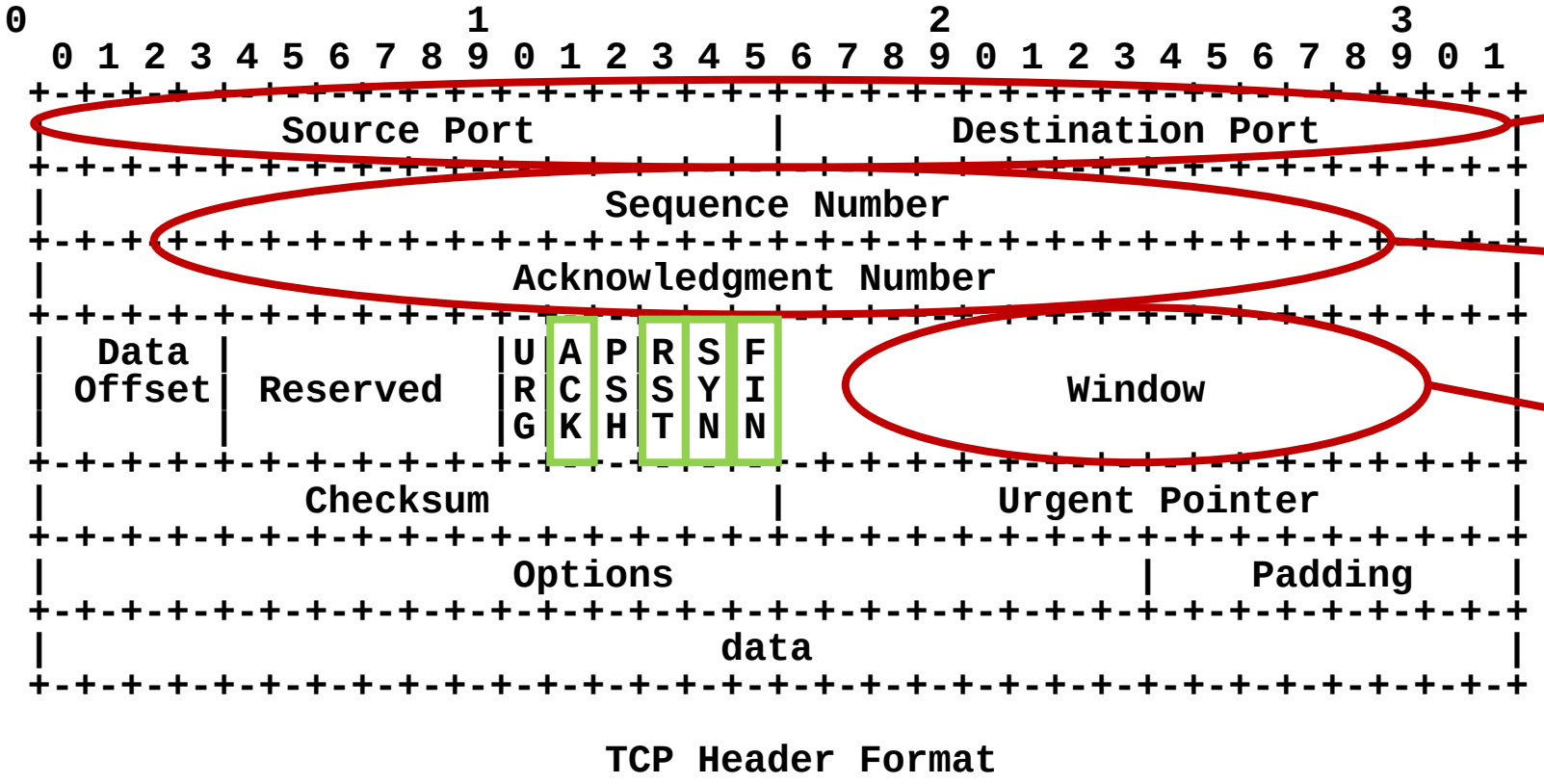
- **No reliability**: messages can get dropped, there is no mechanism to ensure a message was received
- **No congestion/flow control**: no mechanism to avoid congestion either in the network or the end hosts
- **No sessions**: no way to associate messages together (and in both directions) into one logical “session”
- **No multiplexing**: no way to associate messages to a network address to specific applications / users on host.

The Transmission Control Protocol (TCP)

- Protocol run “inside/above” the IP protocol
- Addresses the issues above

TCP header

Refresher



TCP Header Format

Multiplexing

Reliability
Congestion control

Flow Control

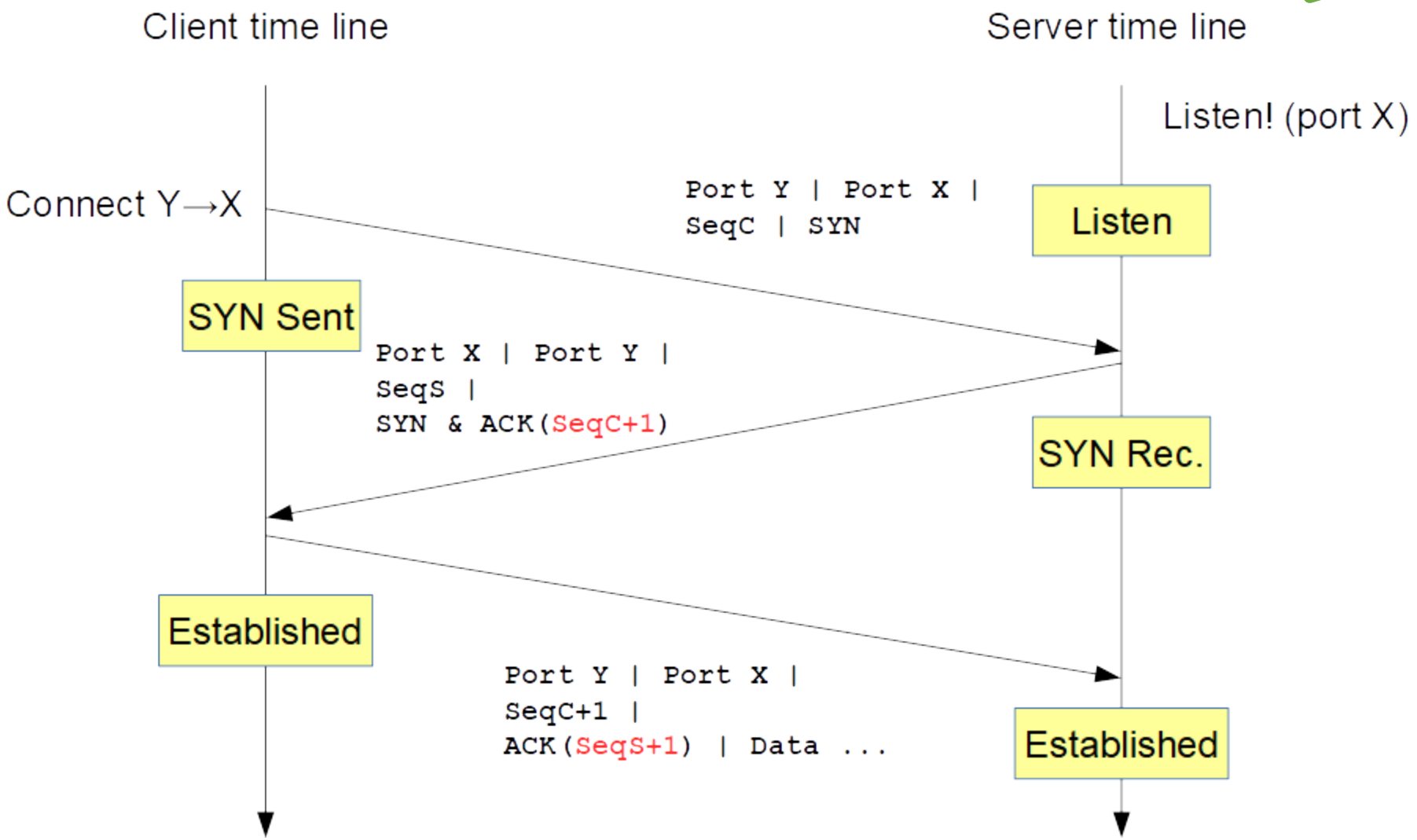
Well known Ports:

20-21 – FTP
22 – SSH
25 – SMTP
53 – DNS
80 – HTTP
110 – POP3
143 – IMAP
443 – HTTPS

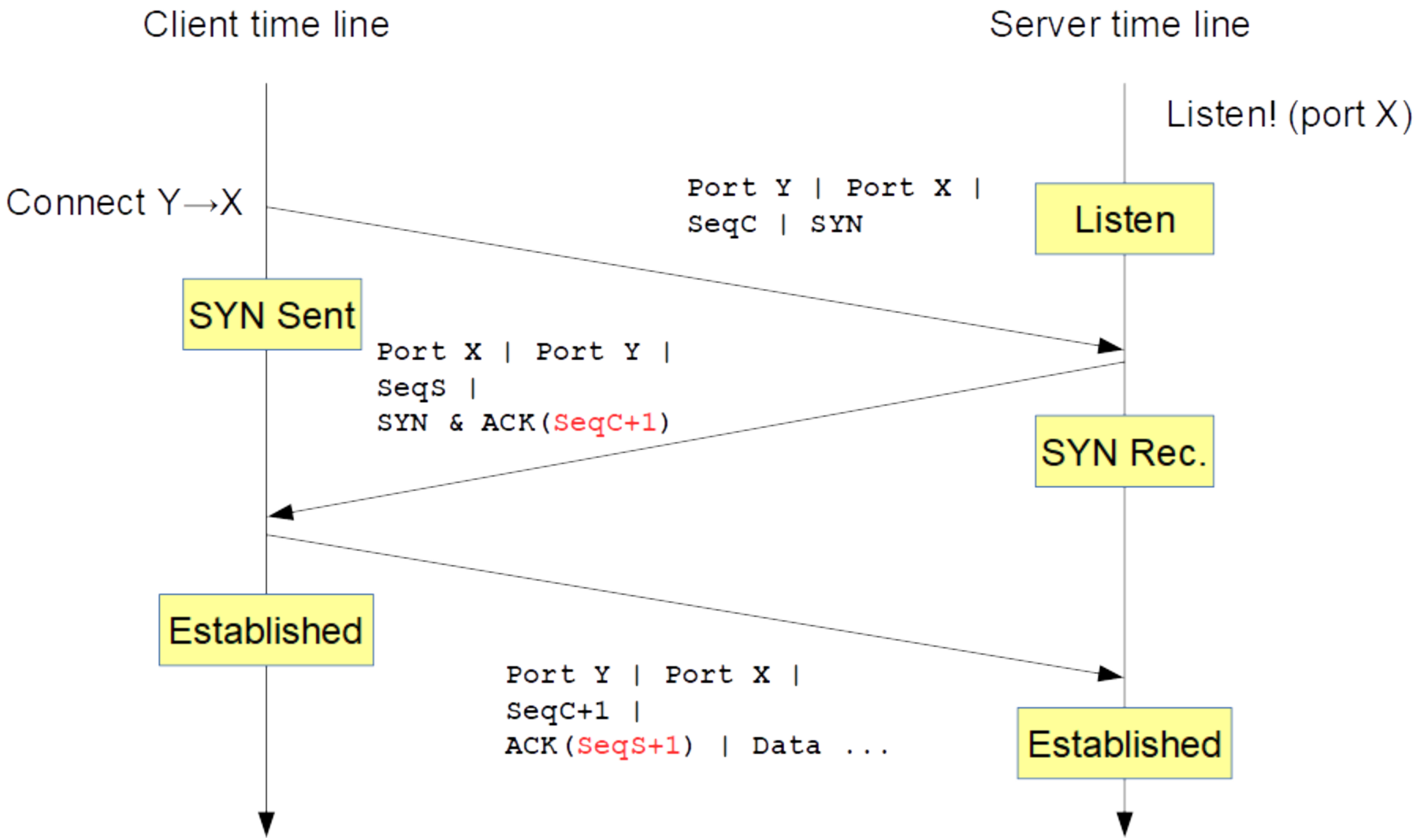
RFC793 (1981) http://www.ietf.org/rfc/rfc793.txt

TCP 3-way handshake

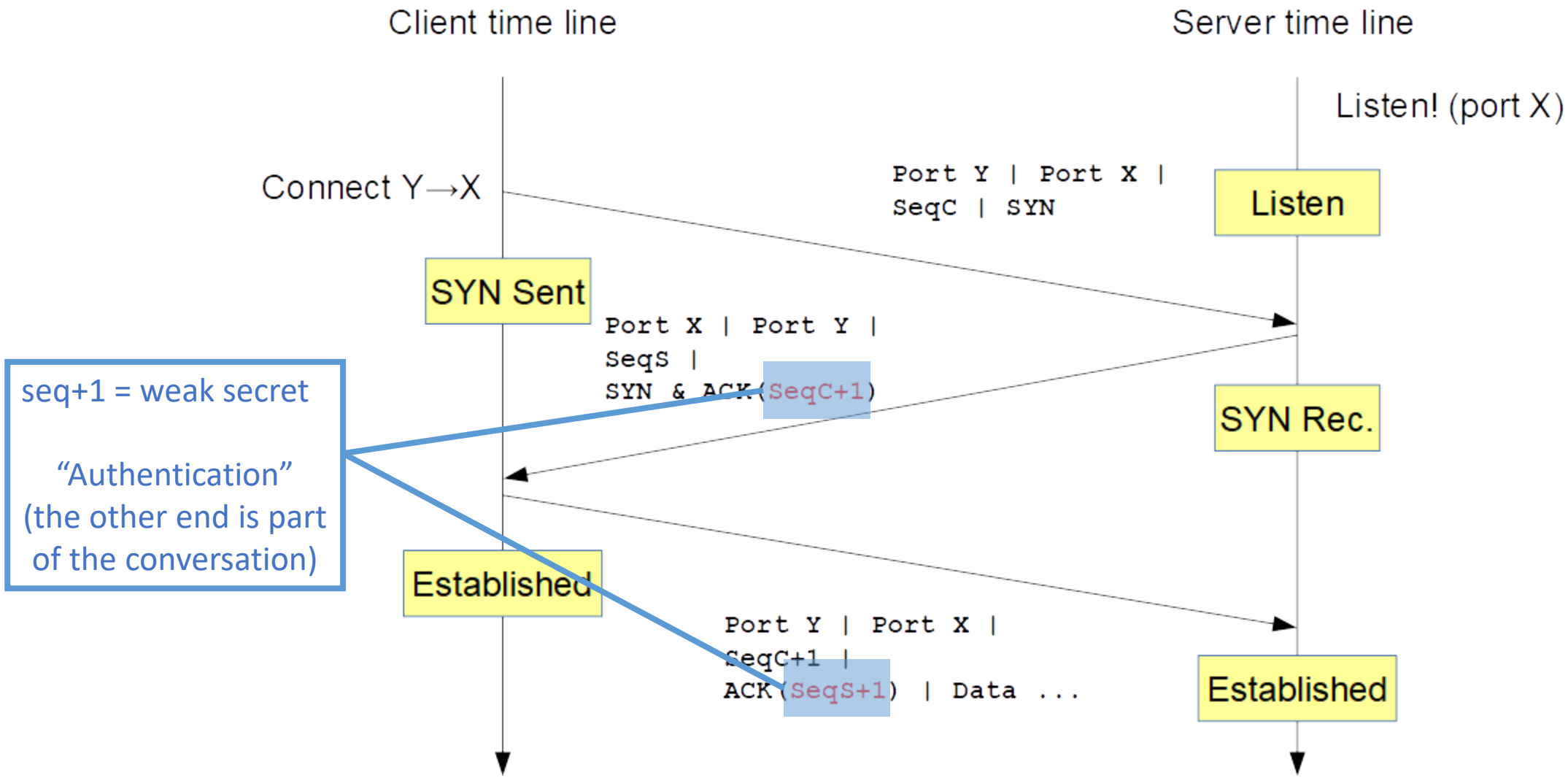
Refresher



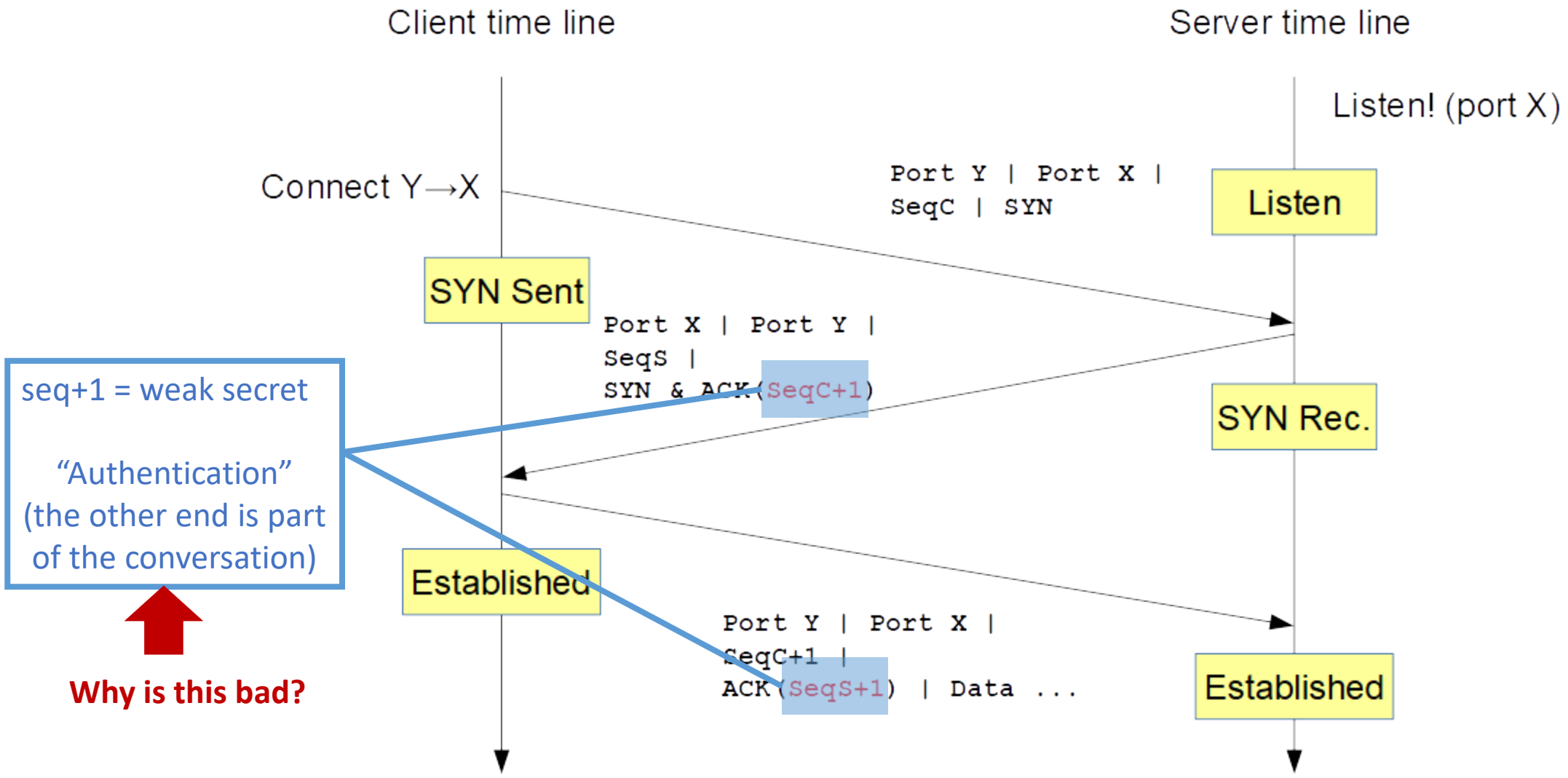
TCP 3-way handshake – Security considerations



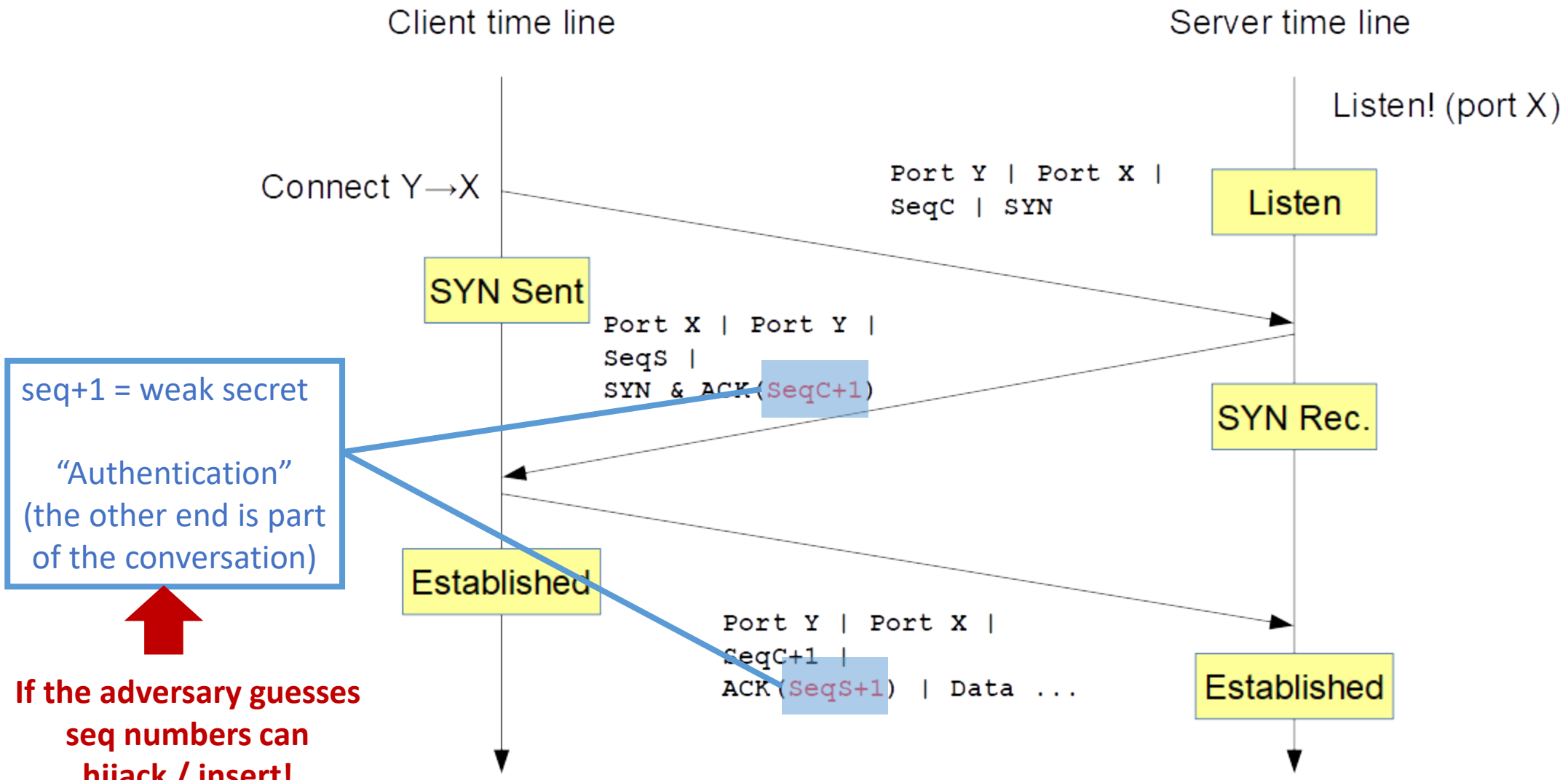
TCP 3-way handshake – Security considerations



TCP 3-way handshake – Security considerations



TCP 3-way handshake – Security considerations



https://www.techrepublic.com/article/tcp-hijacking/

TCP 3-way handshake – Security considerations

Can the adversary guess???

- Weak random numbers generation
- Observation (if connection in the clear)

Example attack

- The (historical) “rsh” UNIX utility that provides a remote shell implemented authentication and authorization on the basis of remote IP address only! (**Bad idea**)

https://www.techrepublic.com/article/tcp-hijacking/

TCP 3-way handshake – Security considerations

Can the adversary guess???

- Weak random numbers generation
- Observation (if connection in the clear)

Guess bc adversary is spoofing, not observing

Example attack

- The (historical) “rsh” UNIX utility that provides a remote shell implemented authentication and authorization on the basis of remote IP address only! (**Bad idea**)
- The Robert Morris Attack:
  1) Send a SYN packet **spoofed** as if it was from authorized host.
  2) Guess server SeqS and send an ACK with SeqS+1 and some data.
  3) The data is interpreted as a shell command and executed!

https://www.techrepublic.com/article/tcp-hijacking/

Basic steps of TCP hijacking

Who: a man in the middle adversary (MITM)

- can observe communication
- can intercept and inject packets

What:

1- Wait for TCP session to be established between client and server

2- Wait for authentication phase to be over

3- Only then use knowledge of sequence numbers to take over the session and inject malicious traffic.

4- Use malicious traffic to execute commands, …

5- The genuine connection gets cancelled (desynchronized or reset).

https://www.techrepublic.com/article/tcp-hijacking/

Basic steps of TCP hijacking

Who: a man in the middle adversary (MITM)

- can observe communication
- can intercept and inject packets

How can we solve this?

Cryptographically authenticate all exchanges! Not only at the start

But TCP cannot do that…

What:

1- Wait for TCP session to be established between client and server

2- Wait for authentication phase to be over

3- Only then use knowledge of sequence numbers to take over the session and inject malicious traffic.

4- Use malicious traffic to execute commands, …

5- The genuine connection gets cancelled (desynchronized or reset).

https://www.techrepublic.com/article/tcp-hijacking/

The Transport Layer Security (TLS)

- Cryptographic protocols above TCP/IP -- “middlelayer”

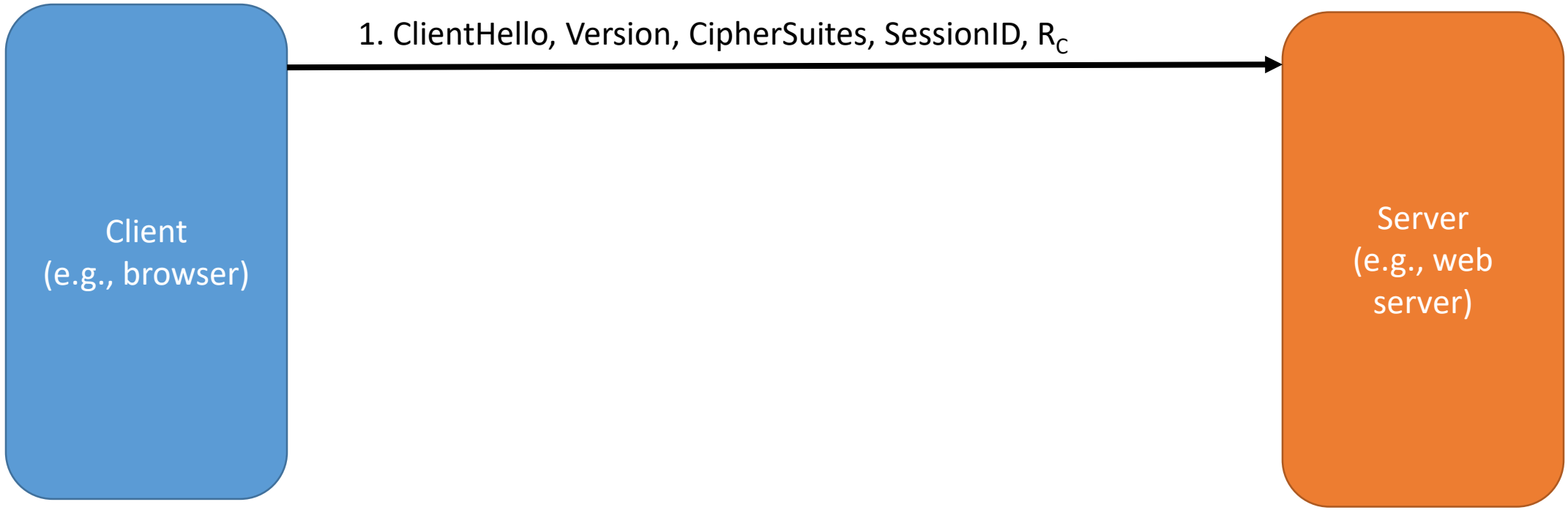
- **Goal**: providing communications security:
    - Confidentiality: symmetric encryption
    - Authentication (One or two-side): public key cryptography
    - Integrity: MAC and signatures

- Provides **forward secrecy**
    - Learning a secret at one point in time does not reveal anything about the past

- State of the art: TLS v3
    - Reality: a zoo in the Internet (it is difficult to upgrade a huge number of computers)
    - SSL, same principles but many vulnerabilities -- deprecated!

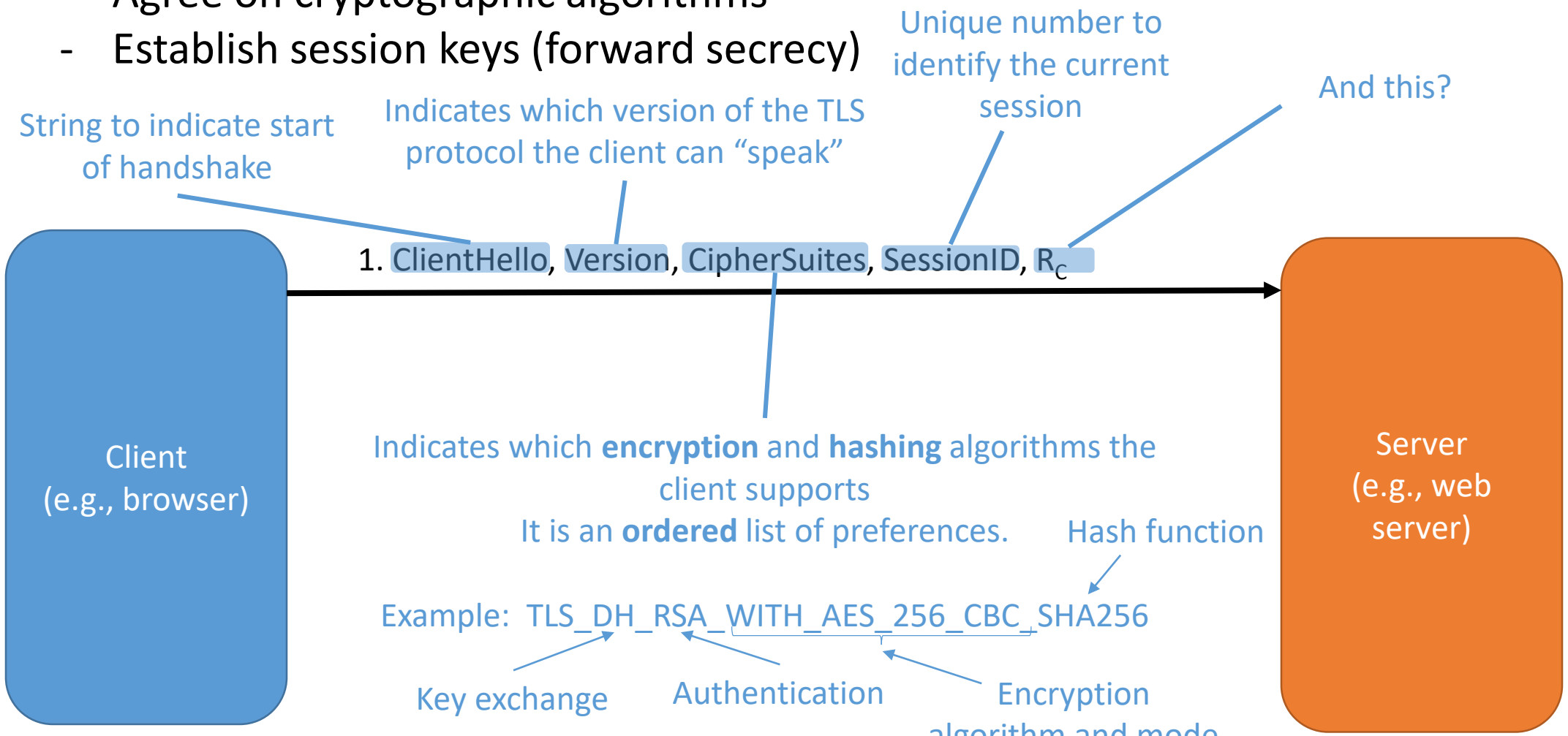
The TLS handshake

- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)



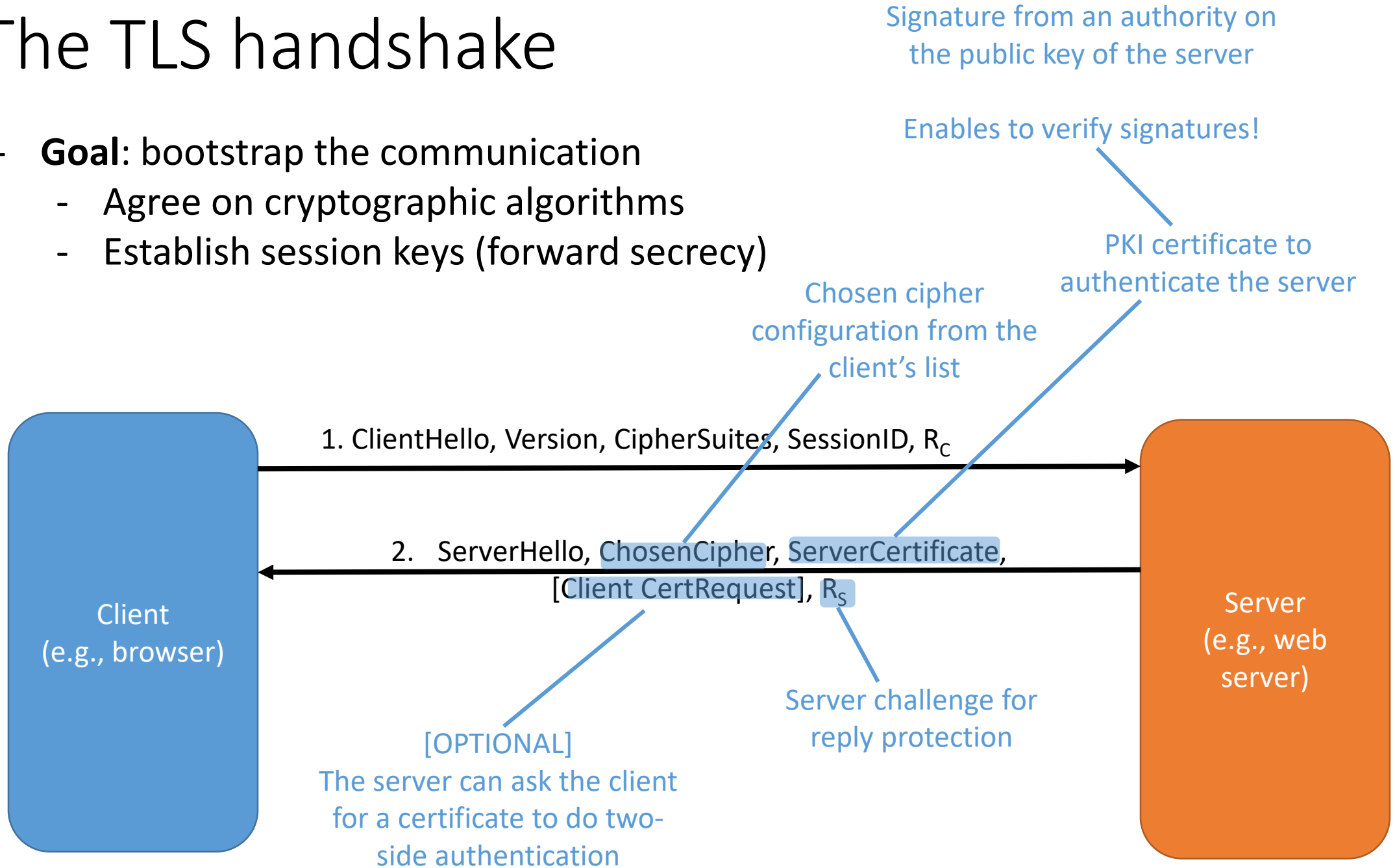
The TLS handshake

- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)



The TLS handshake

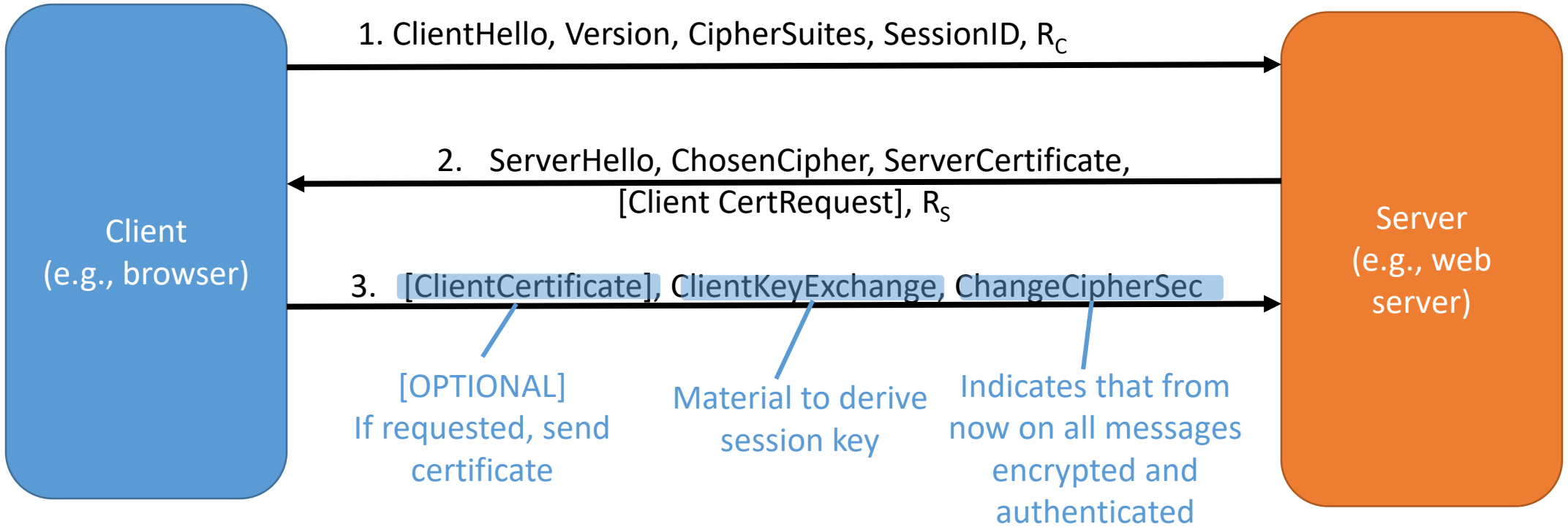
- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)



The TLS handshake

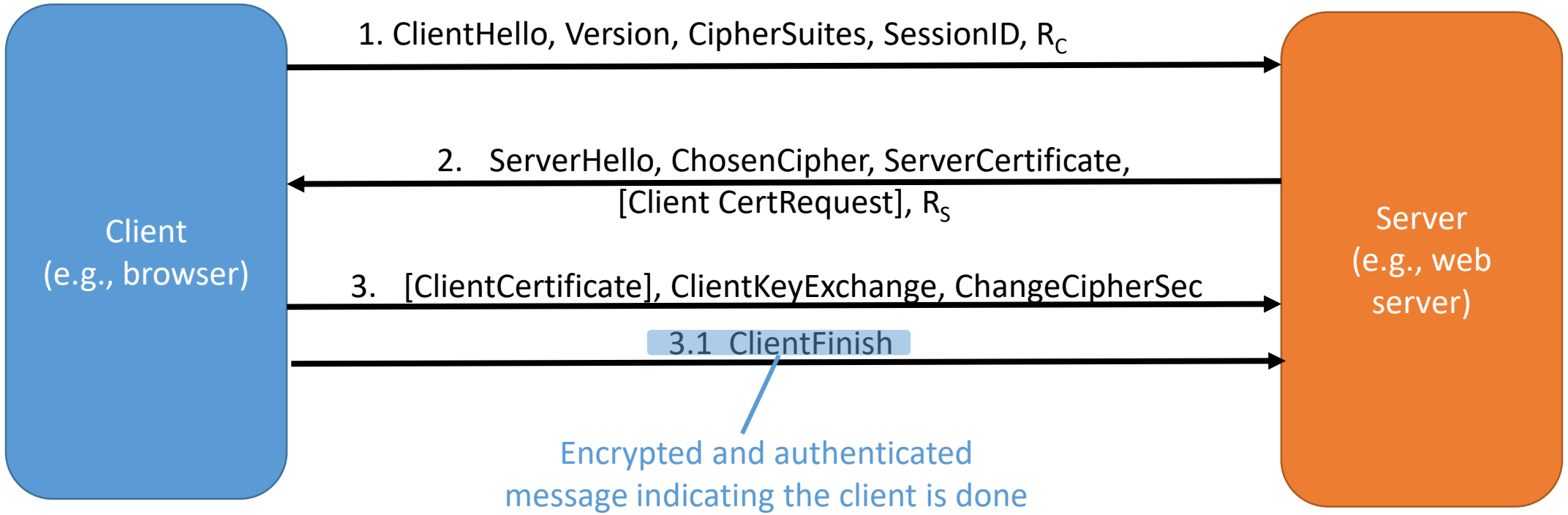
- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)

After step 3 Client and server have a shared session key!!!



The TLS handshake

- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)



The TLS handshake

- **Goal**: bootstrap the communication
  - Agree on cryptographic algorithms
  - Establish session keys (forward secrecy)

Server does the same: indicates that from now on everything will be encrypted and authenticated

And sends an authenticated encrypted Finished message