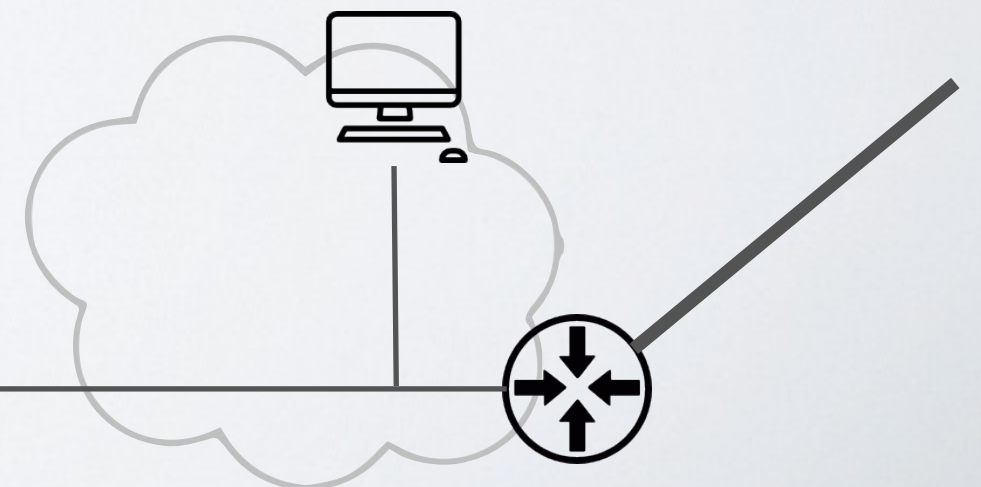
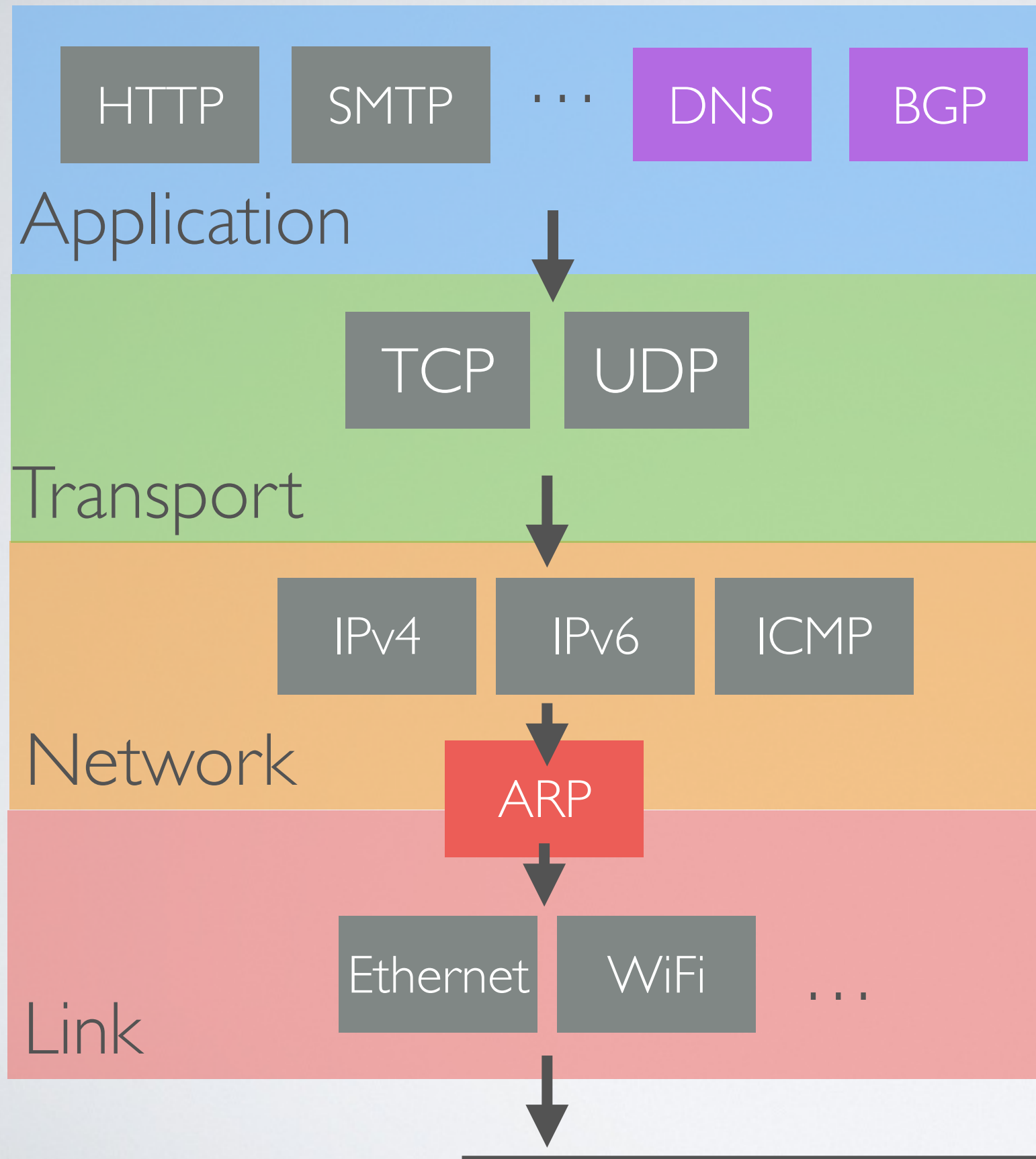


# Network Security

Thierry Sans

# The Protocol Stack



confidentiality  
integrity  
availability



The attacker is capable of ...

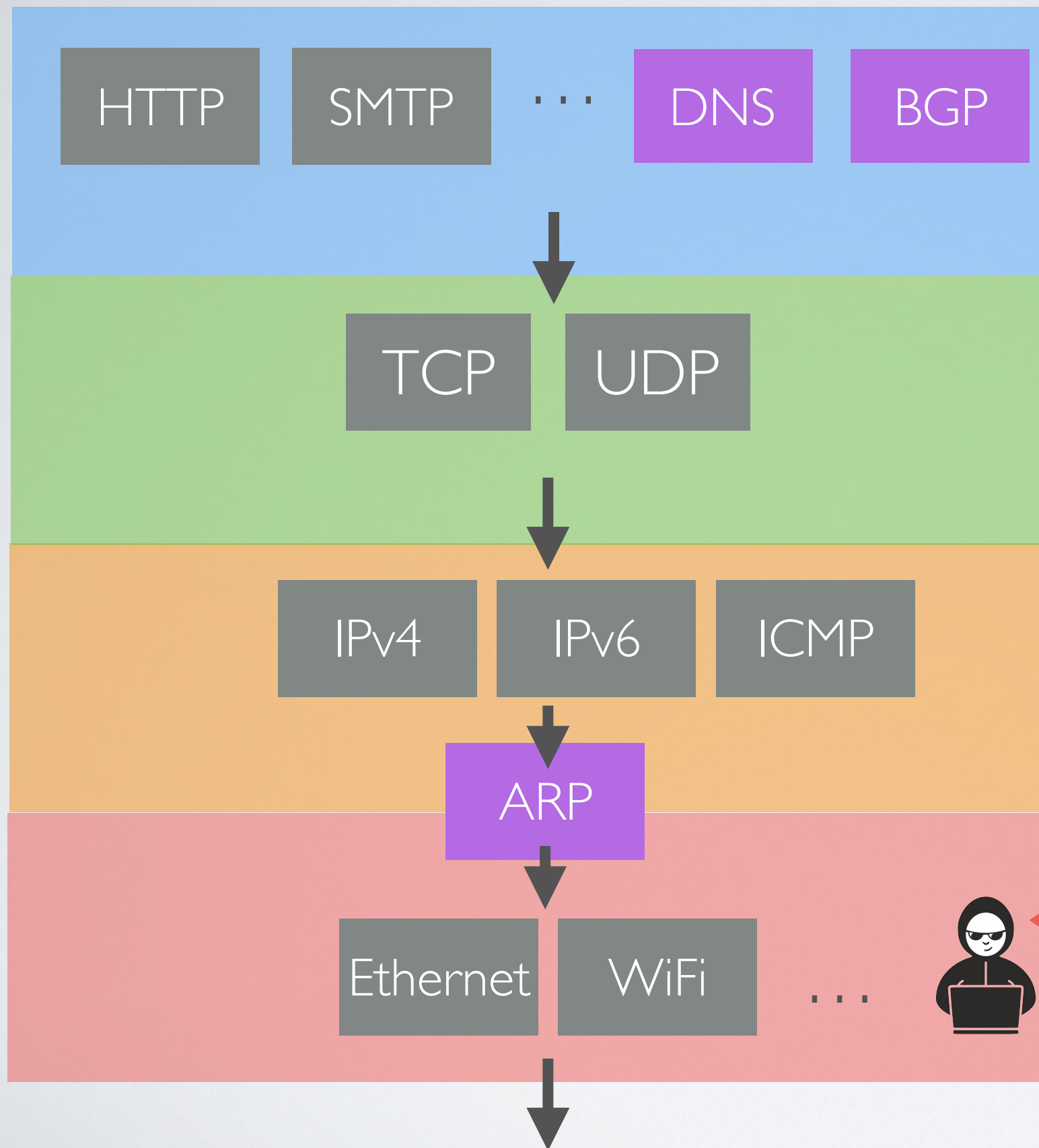
**Scanning** - survey the network and its hosts

**Eavesdropping** - read messages

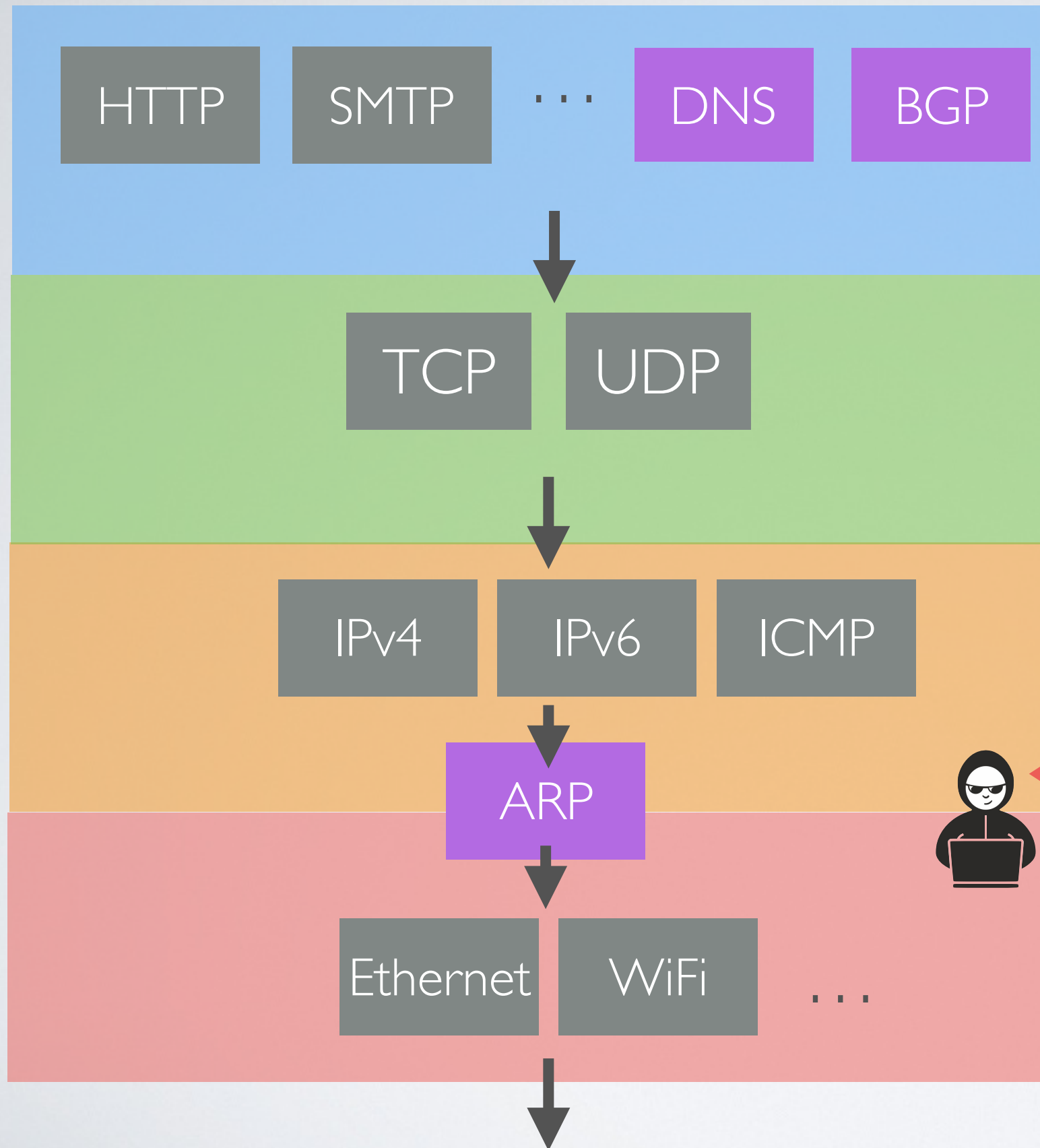
**Spoofing** - forge illegitimate messages

**DOS** (Denial of Service) - disrupt the communications

➡ The attacker can target any layer in the network stack

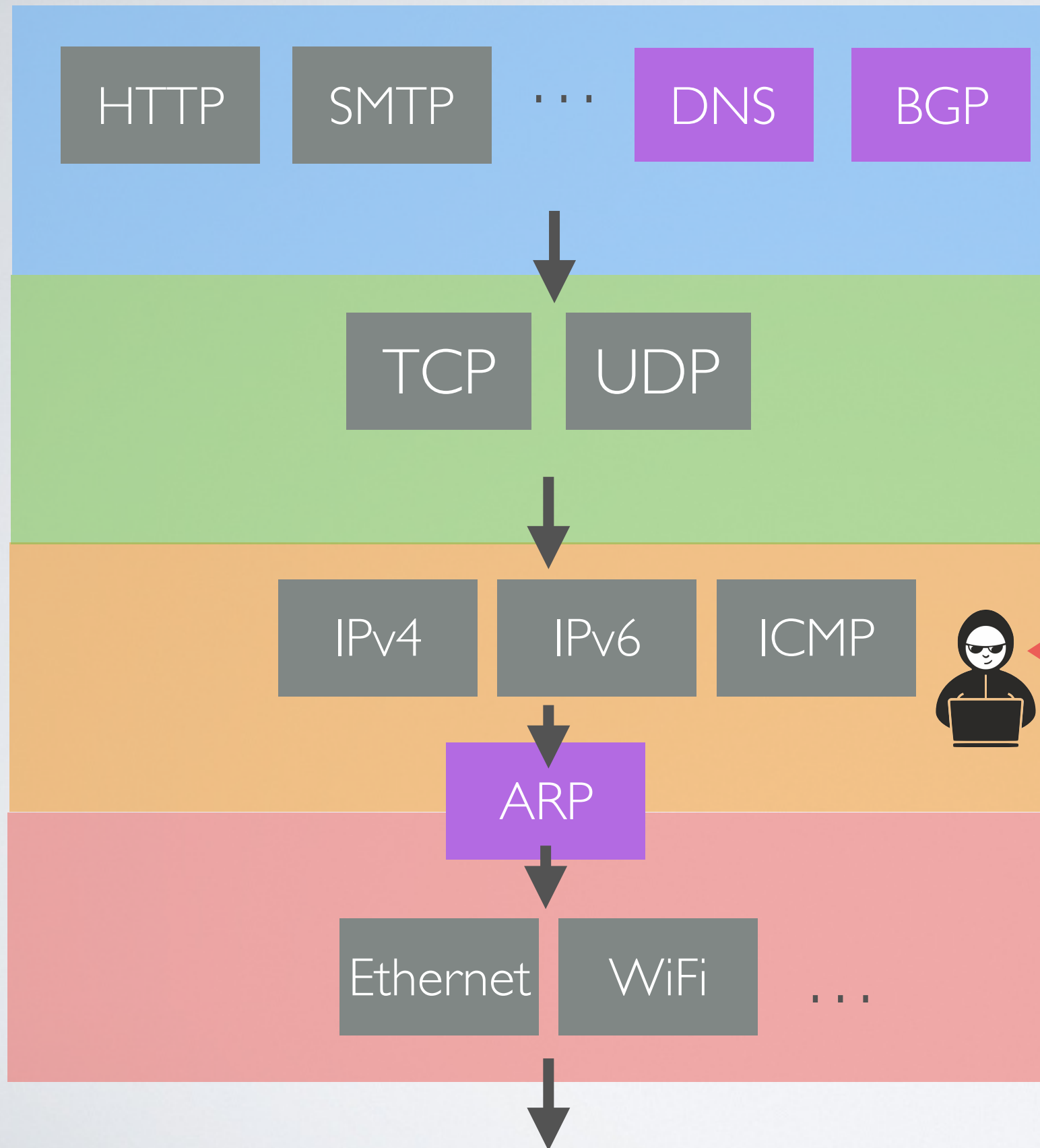


Packet Sniffing (eavesdropping)

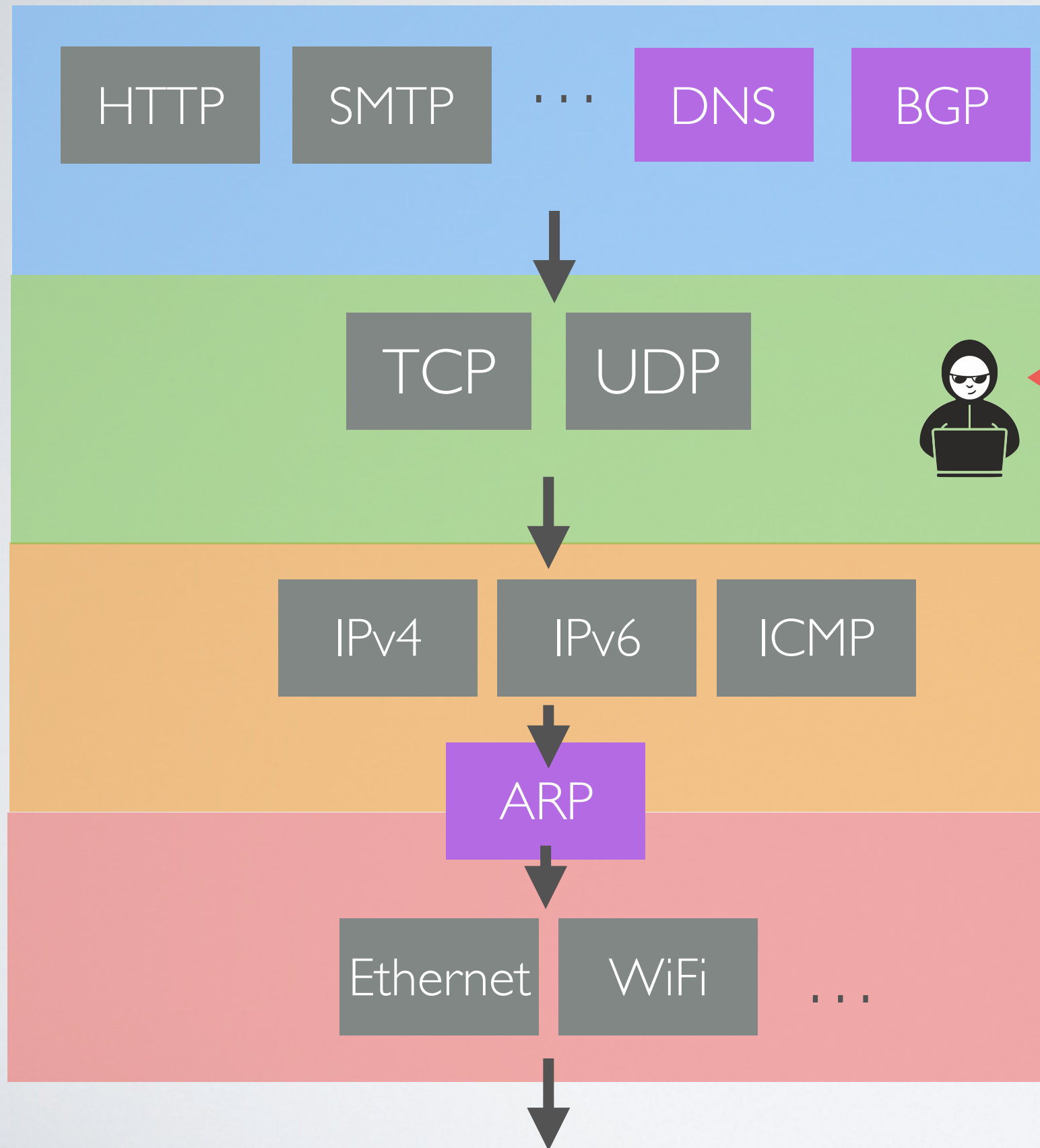


ARP-cache poisoning (spoofing)

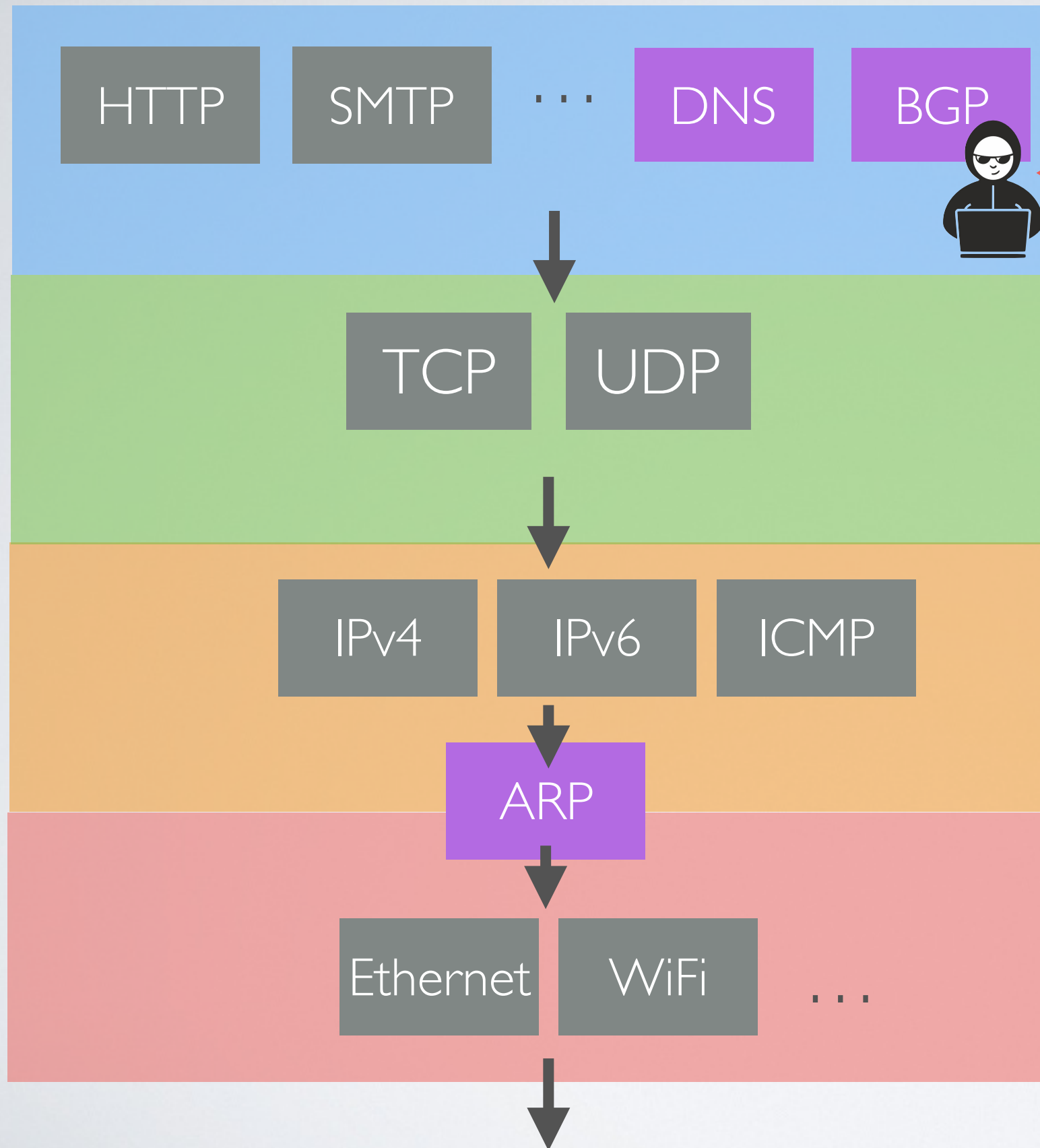




- Host discovery (scanning)
- IP forgery (spoofing)
- ICMP Ping flooding (DOS)



- Port scanning (scanning)
- TCP forgery (spoofing, DOS)
- TCP-syn flooding (DOS)
- UDP flooding (DOS)



- Route Hijacking (spoofing, DOS)
- DNS-cache poisoning (spoofing, DOS)

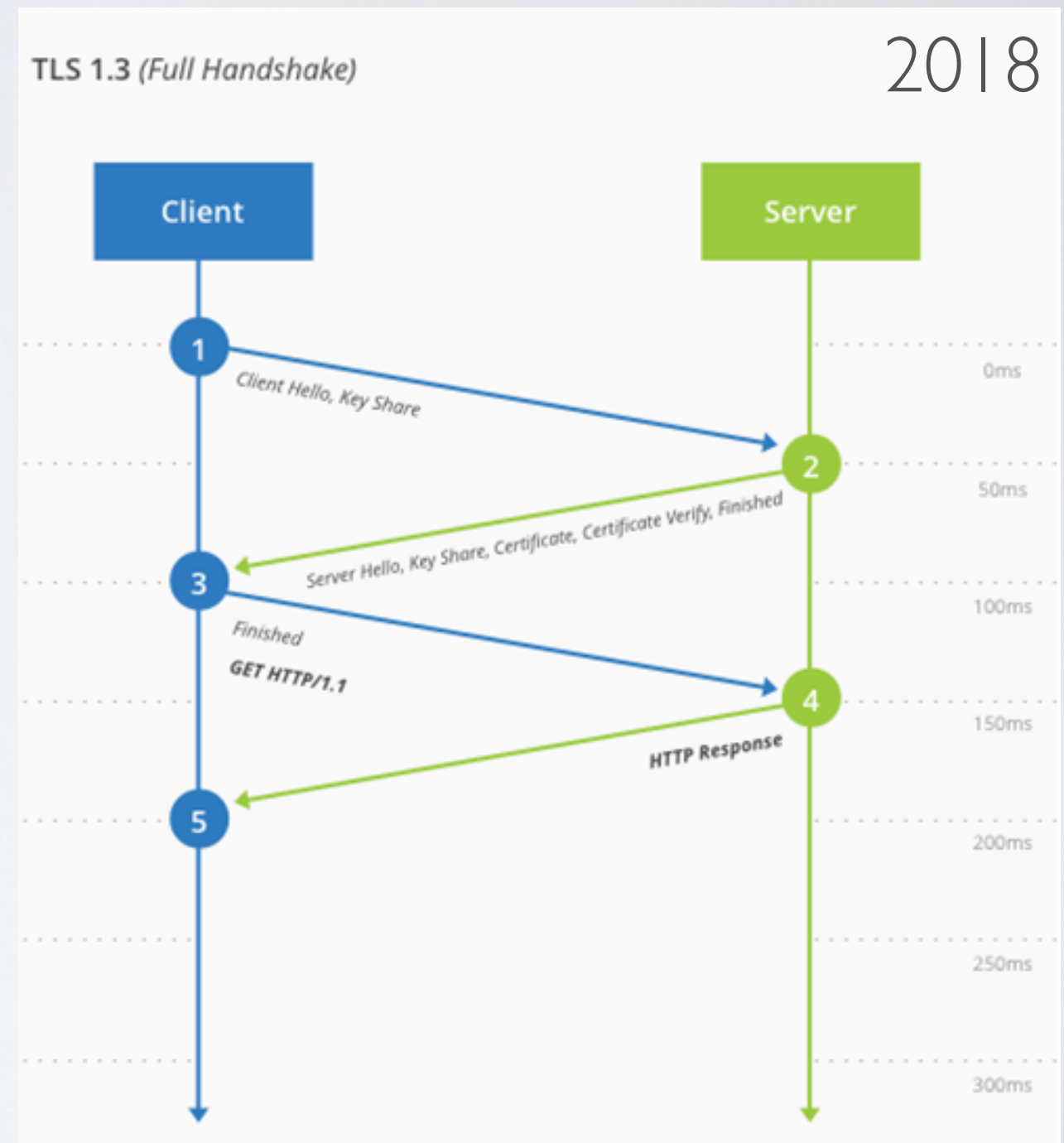
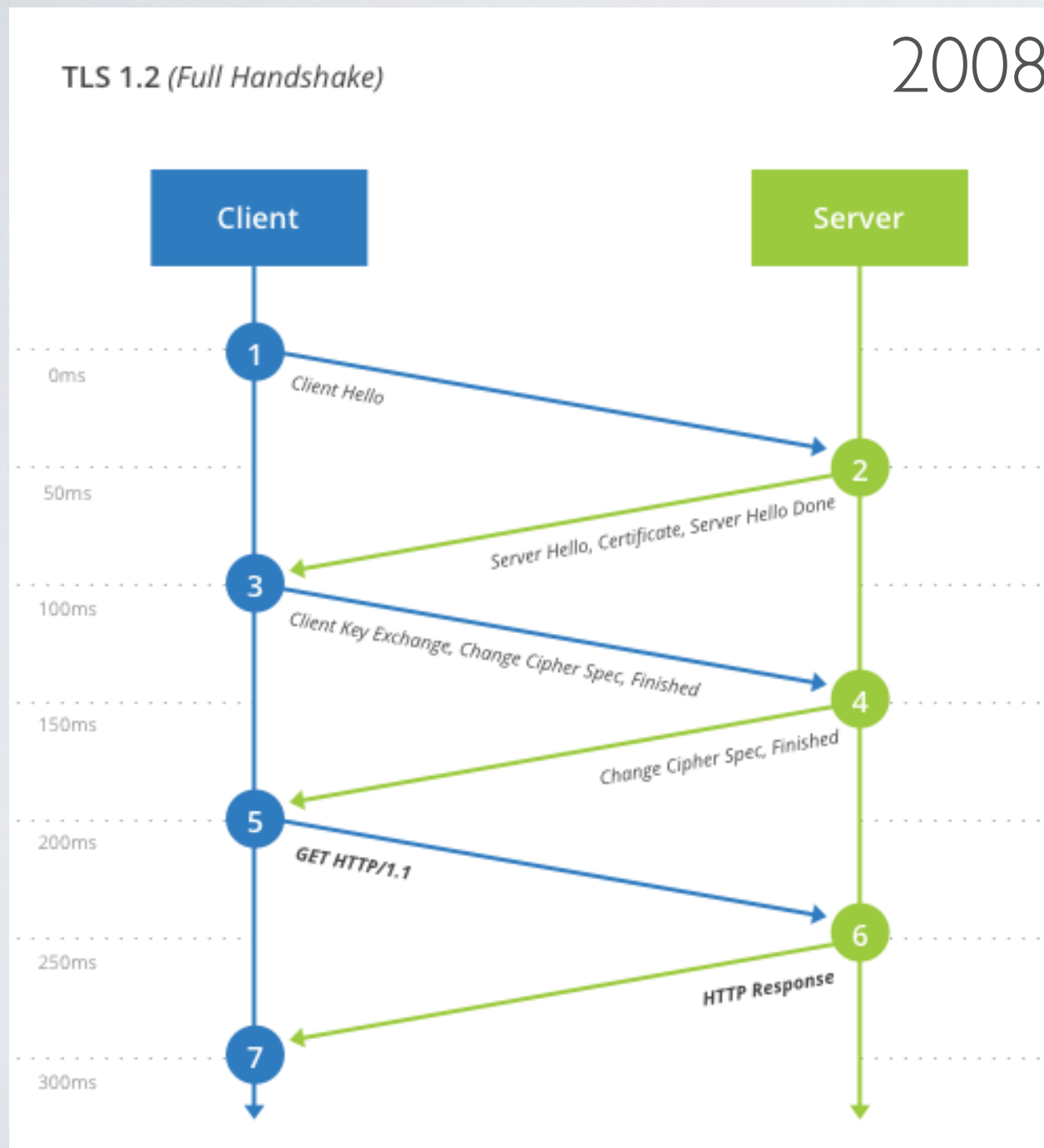


# TLS - Transport Layer Security

# TLS - Transport Layer Protection

- ➔ Transport Layer Security (formerly SSL) provides
  - **integrity:** authentication handshake
  - **confidentiality:** end-to-end secure channel
- ✓ Prevents all kinds of eavesdropping and spoofing for application protocols e.g HTTP + TLS = HTTPS
- ⦿ 2-10 times slower than an insecure TCP connection
- ⦿ Not used in practice to secure DNS and BGP

# TLS Authentication Handshake 1.2 vs 1.3



source <https://www.cloudflare.com/learning-resources/tls-1-3/>

# Specific attacks of HTTPS

Webpages can be delivered either with HTTPS or HTTP

➡ The browser can automatically switch between HTTP and HTTPS

Sometime within the same webpage (mixed-content)

e.g the main page loads over HTTPS

but images, scripts or css load with HTTP

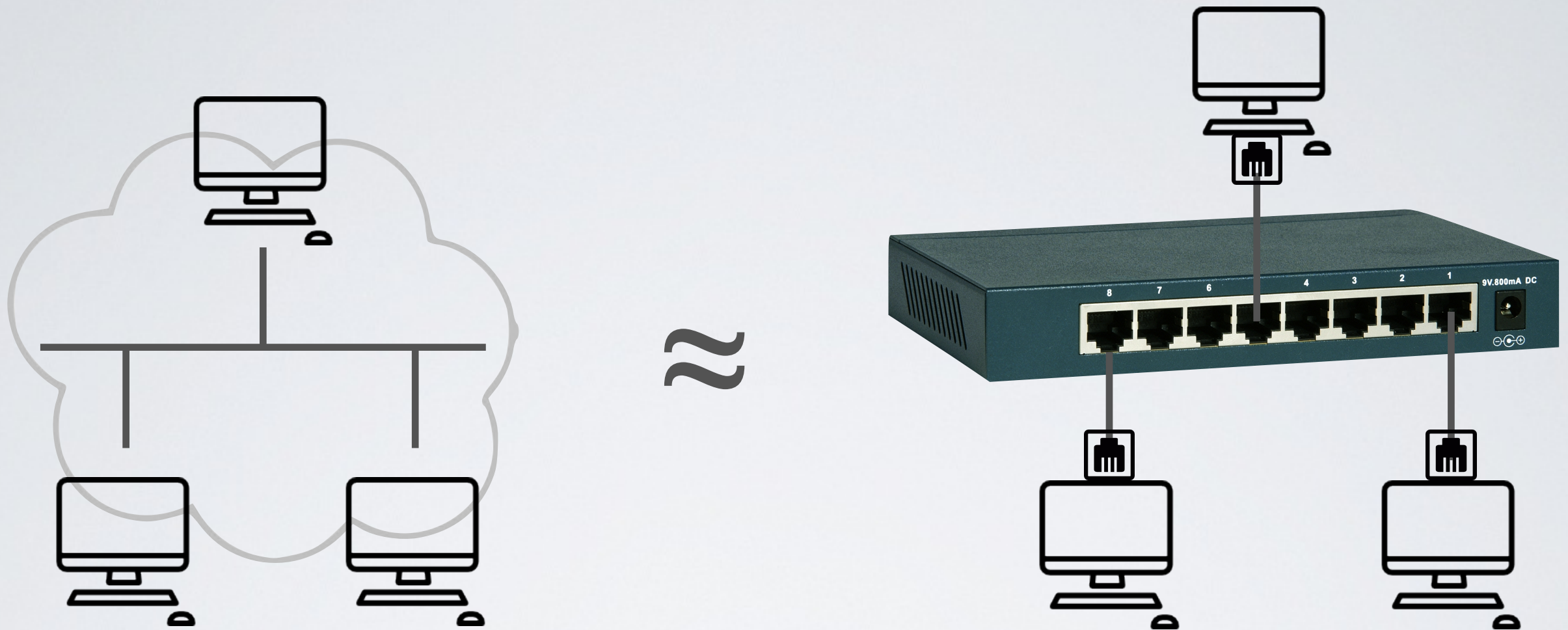
An attacker can do a MitM attack and remove the SSL protection

➡ **SSLStripping** attack (challenge coming next)

# Preventing eavesdropping attacks



# Preventing packet sniffing over Ethernet



**Hub :** broadcast all messages on all ports

**Switch :** (smart HUB) forward messages on specific port based on their MAC addresses

➡ isolate Ethernet traffics (no straightforward packet sniffing)



# Packet sniffing over a wireless network

➡ Encrypt message before sending them over the air

Wireless Security	WEP	WPA	WPA2	
			Personal	Enterprise
Authentication	Shared Key	Shared Key	Shared Key	RADIUS Server
Cryptography	RC4	TKIP and RC4	CCMP and AES	
Security	Broken	Broken	External attackers only	Good

# Preventing spoofing attacks

# Preventing ARP-cache poisoning

- **Authenticating ARP messages** has been proposed (research) but never implemented
- **Static ARP** tables (not practical in dynamic environment)
- **Detection and correction** tools

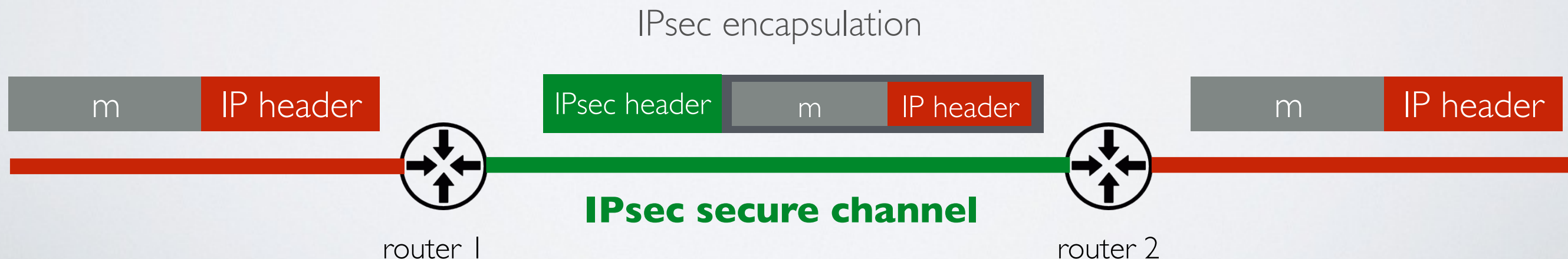
# Preventing IP forgery

**IPsec - Internet Protocol Security** provides authentication (and optionally encryption) of IP traffic

➔ Uses SHA2 and AES (previously SHA1 and 3DES)

✓ Used usually between routers (link and network layers only)

⦿ However IPsec is rarely deployed in practice



# Preventing DNS spoofing

**DNSSEC** - Domain Name System Security Extensions provides authentication (but not encryption) between DNS servers

- Not widely deployed

**DNS over HTTPS** (since 2018) provides authentication and encryption between client/server and server/server

- Pushed by Google and Mozilla



# Preventing route hijacking (BGP)

## **Bogon Filtering**

*Best Current Practice* to limit fake route advertisement

Deny route advertised by hosts with spoofed addresses

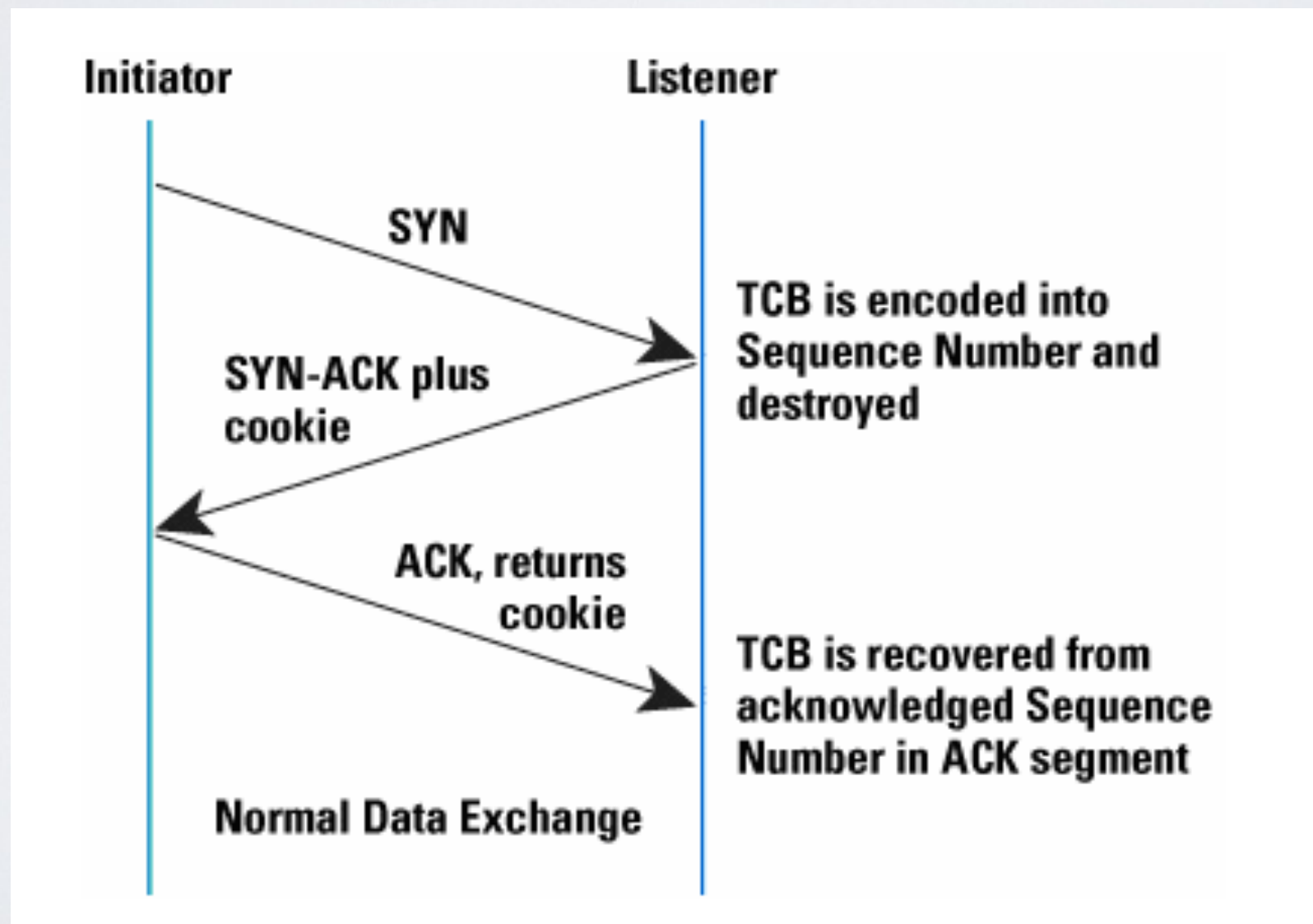
➡ Implemented by ISPs (Internet Service Providers)



# Preventing DOS attacks

# Preventing TCP-syn flooding

**TCP-syn cookie** prevents from maintaining a queue of half-opened TCP connections



# Preventing DOS and DDOS attacks in general

## **Network Ingress Filtering** (a.k.a BCP 38)

*Best Current Practice* to limit the impact of DOS and DDOS

1. Deny access to network traffic with spoofed addresses
  2. Ensure that traffic is traceable to its correct source network
- ➡ Implemented by ISPs (Internet Service Providers)

# Preventing scanning attacks (and beyond)

# Preventing host discovery and port-scanning

**Host discovery** uses ICMP ping echo message

- ➡ ICMP can be disabled or reserved to hosts on the same network

**Port Scanning** uses TCP-syn messages

- ➡ TCP connections can be rejected if a source attempts to initiate multiple connections on multiple ports simultaneously
- ➡ **Packet filtering** can prevent these two scanning techniques



# Limitation of a host-by-host packet filtering solution

How to enable packet filtering on every host on the network?

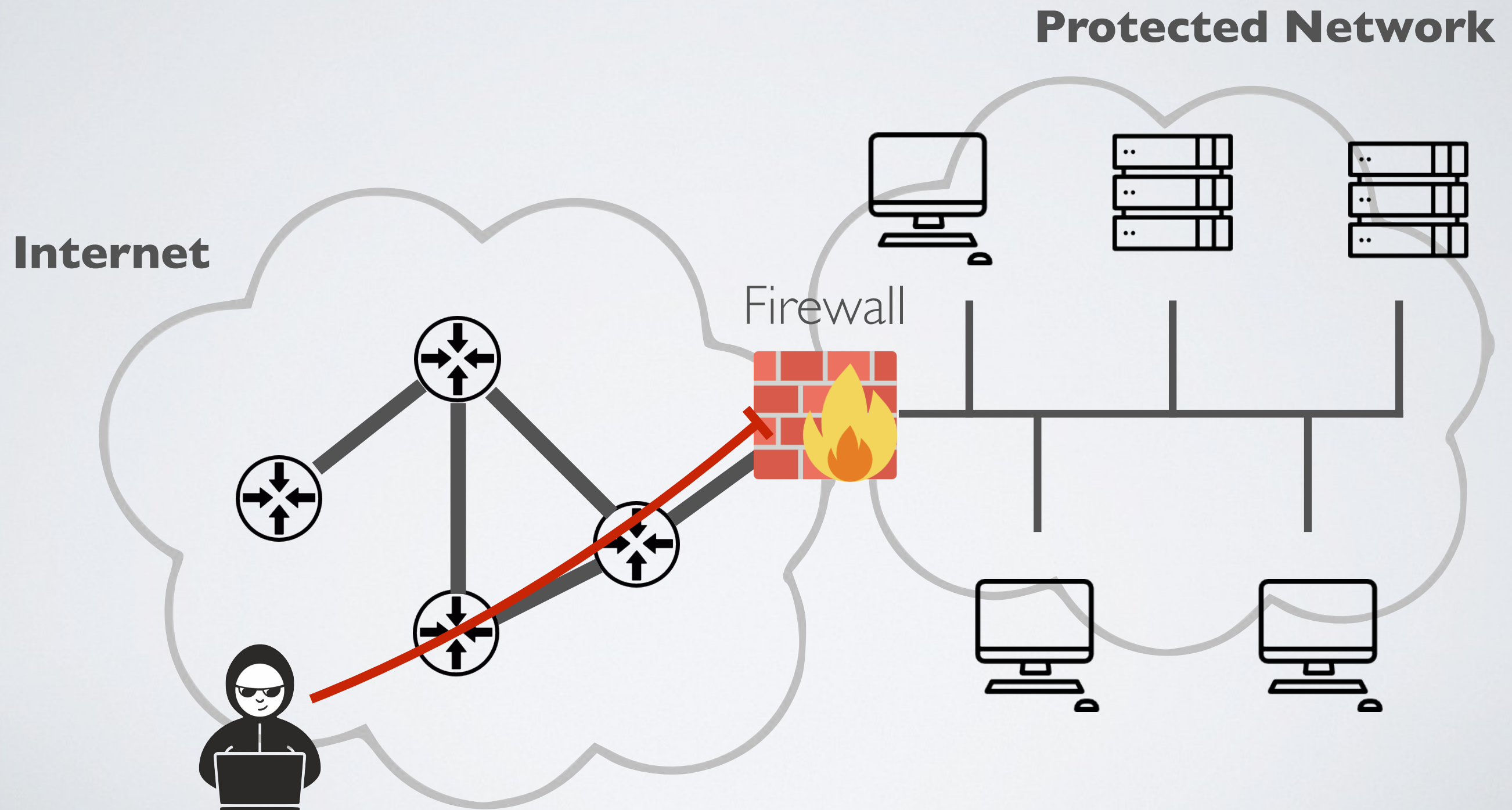
1. Each host needs to have **packet filtering capability** across different hardware, OS and versions
2. The admin needs to have **administrative privilege on every host** to push the packet filtering policy

➡ Impossible in practice



Firewall

# Network Firewall



# Network Firewall

**A firewall** defines a logical defense parameter and acts as an access control between two networks

- ➔ Packet filtering based on IP addresses (TCP filtering)
- inbound traffic from the Internet trying to get into the protected network
- outbound traffic going the other way
- ✓ For the most part, we trust the outbound but not the inbound

# Widely used in practice

Assuming the attacks comes from outside, a firewall can prevent

- Most scanning attacks
  - Some spoofing attacks
  - Some flooding attacks (as long as it can handle the load)
  - Anomalous messages e.g smurf attack
  - and others
- ➡ But more generally, it can restrict access to protected hosts



# Two type of firewalls

## **Stateless packet filtering**

is purely based on the IP address and the port

## **Stateful packet filtering**

tracks the status of every connection (TCP 3 way handshake)

# Example of a stateful firewall policy

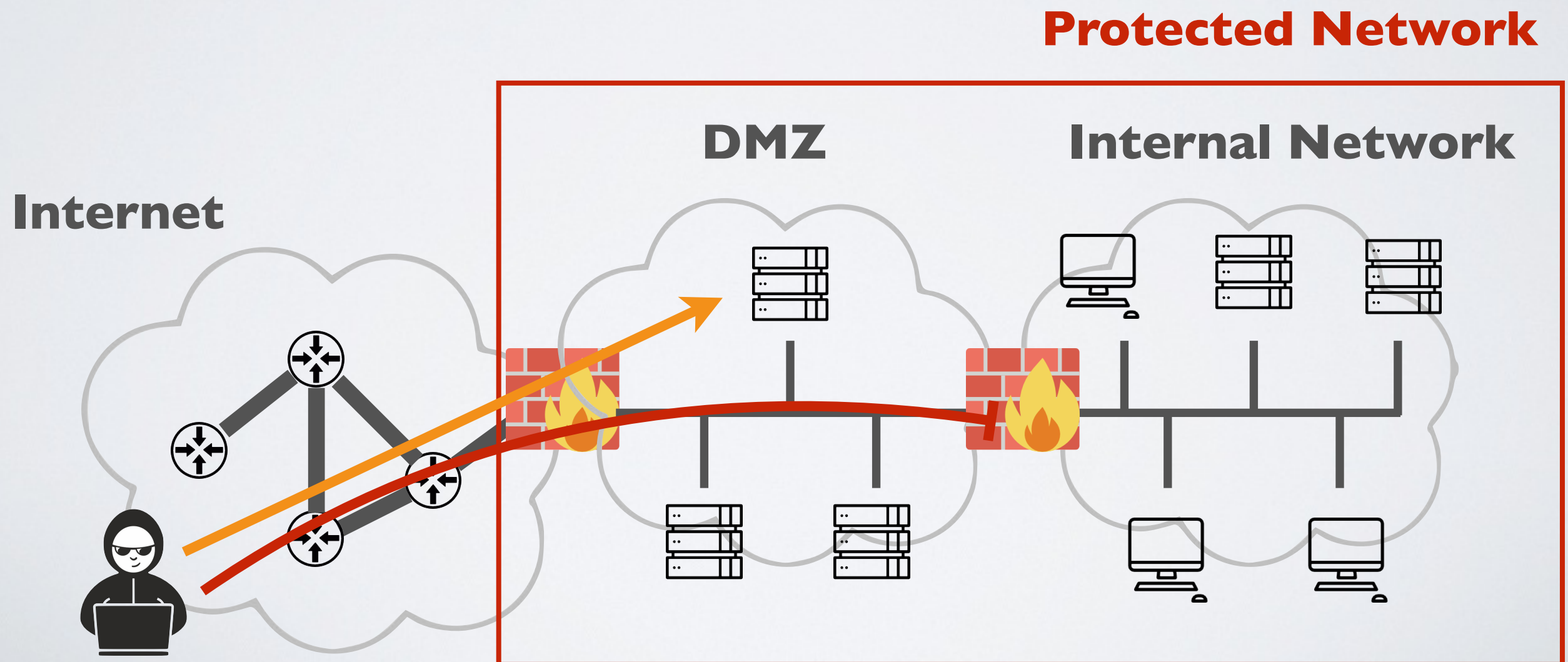
## **ACL** - Access Control Lists

action	protocol	IP src	port src	IP dst	port dst	state
allow	TCP	222.22/16	>1023	! 222.22/16	80	any
allow	TCP	! 222.22/16	80	222.22/16	>1023	ack
allow	UDP	222.22/16	>1023	! 222.22/16	53	-
allow	UDP	! 222.22/16	53	222.22/16	>1023	-
deny	all	all	all	all	all	all



# Concept of DMZ

**DMZ** - DeMilitarized Zone isolates exposed public servers e.g web, mail, database and so on



# Intrusion Detection

# Two approaches to build an IDS

## **Signature-based IDS**

Have pre-defined malicious message pattern

➡ Relies on a signature database

## **Heuristic-based**

Builds a model of acceptable message exchange patterns

➡ Relies on machine learning

# (Network) Intrusion Detection Systems

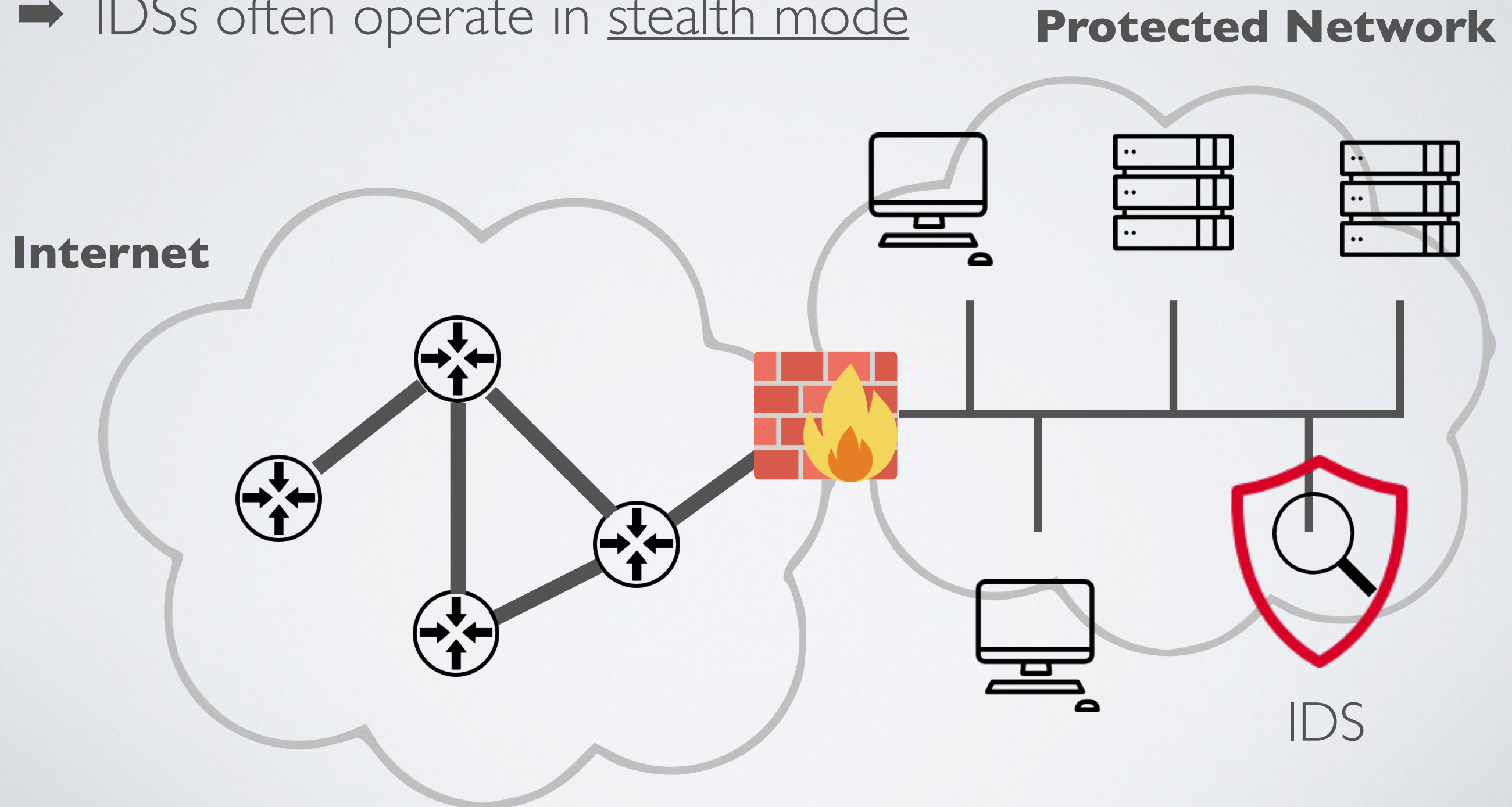
**IDS** - Intrusion detection systems performs deep packet inspection

- Looks at the headers
- Look at packet contents (payload)
- Looks at the packet fragmentation



# IDS in the protected network

➔ IDSs often operate in stealth mode



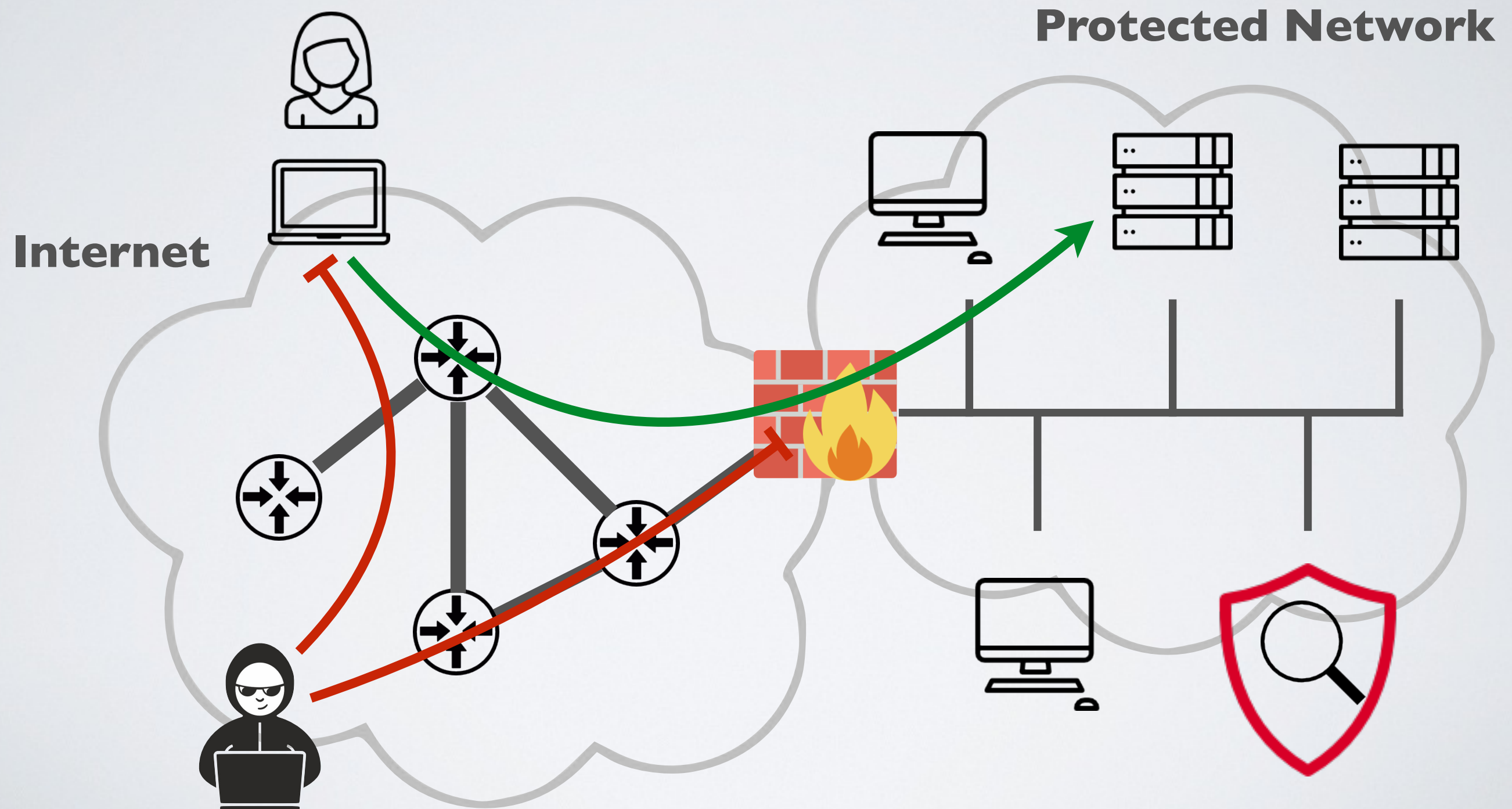


# IPS - Intrusion Prevention system

**IPS** = IDS + Firewall

➡ IP addresses sending malicious packets can be filtered

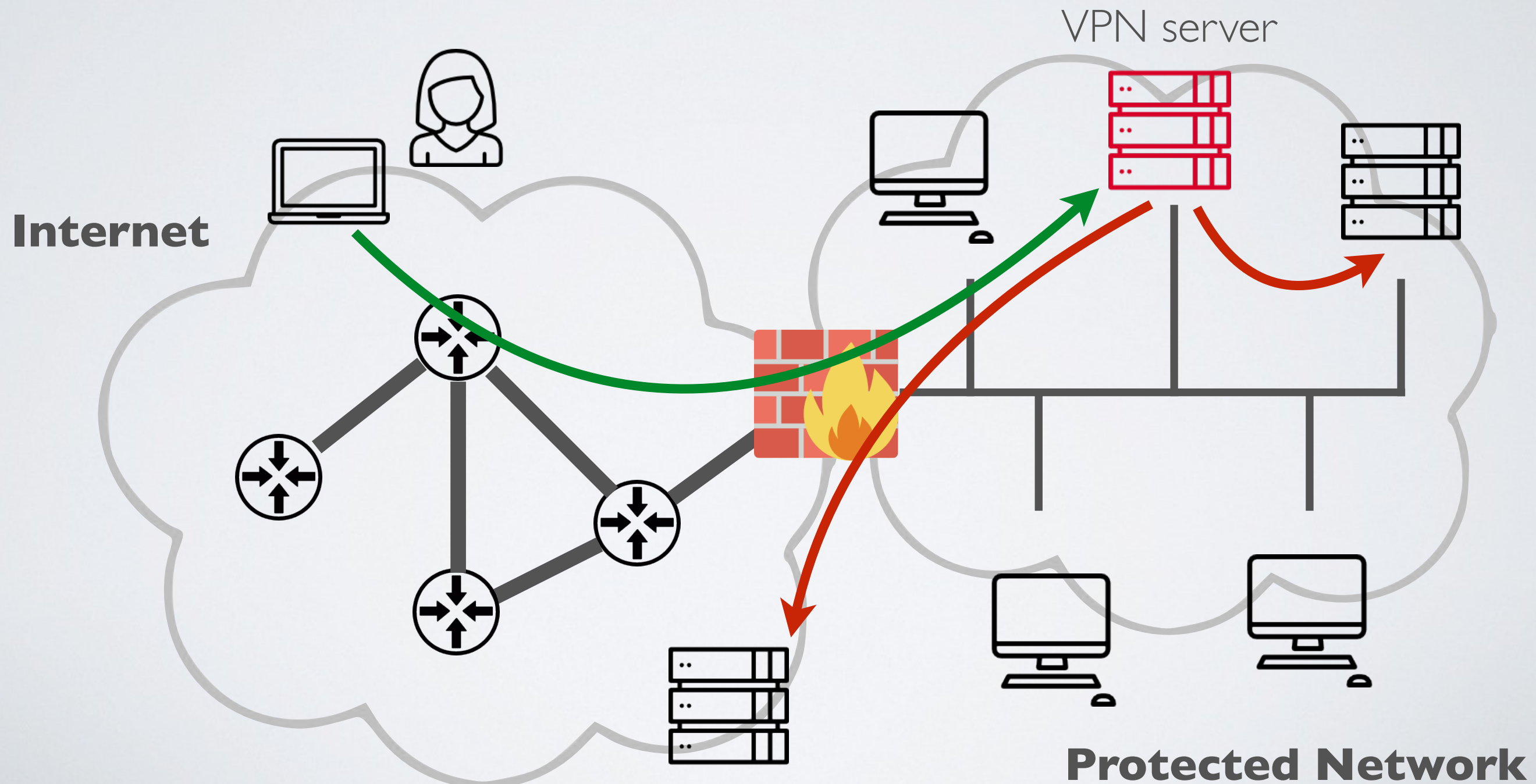
# Problem with nomad hosts



VPN - Virtual Private Network

# VPN - Virtual Private Network

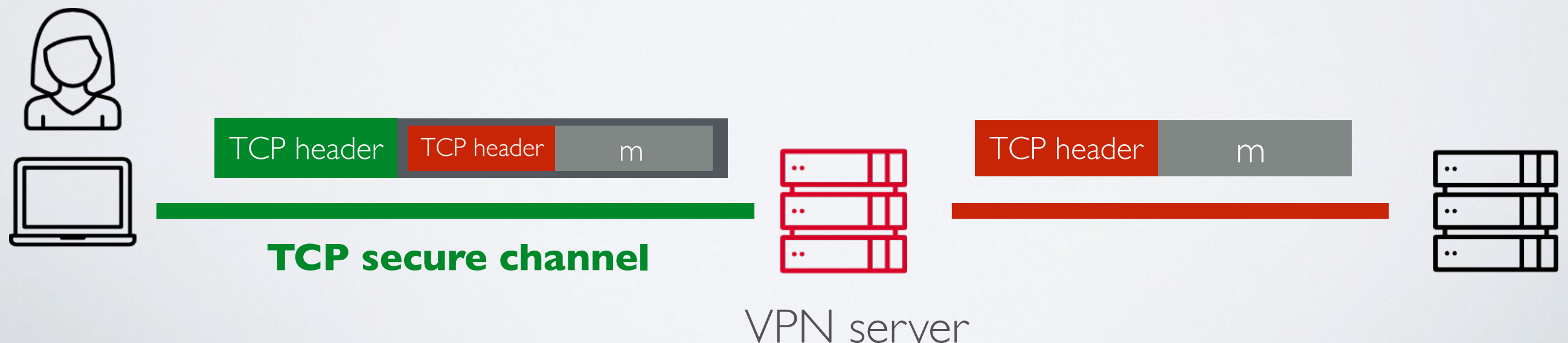
**VPN** protected nomad hosts outside the protected network





# Tunneling protocol

1. Alice's message is encapsulated and sent to the VPN server
  2. The VPN extract this traffic and send it to the destination
  3. Same thing on the way back
- ➡ Provides anonymity (from the IP perspective at least)





# Different type of VPNs

VPN can be built using different technology e.g.

- IPsec
- TLS (e.g openVPN)
- SSH

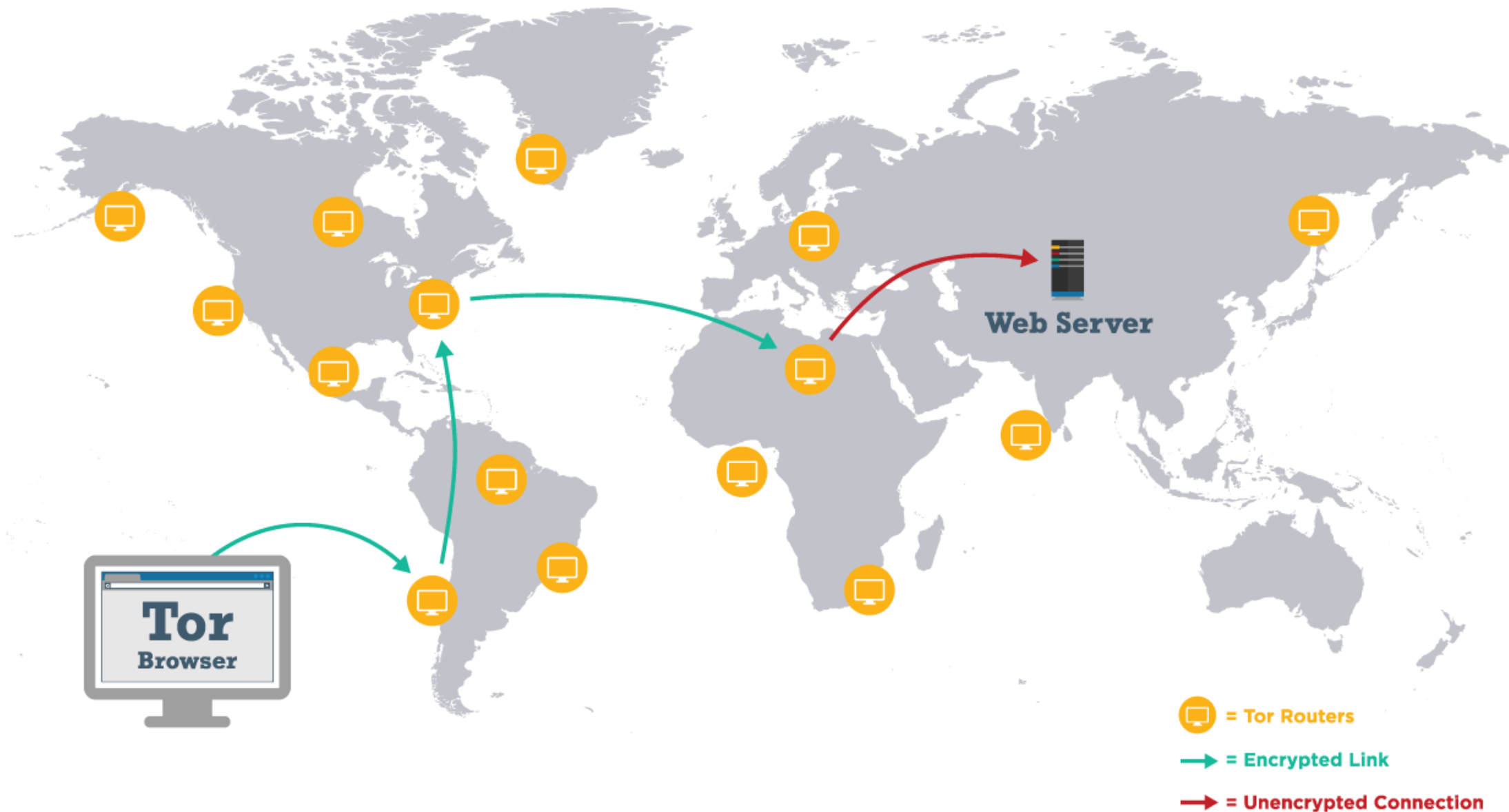
# VPN to enforce security ... or evade it :)

- ➡ Evade censorship and geo-restrictions  
by masking the real IP address

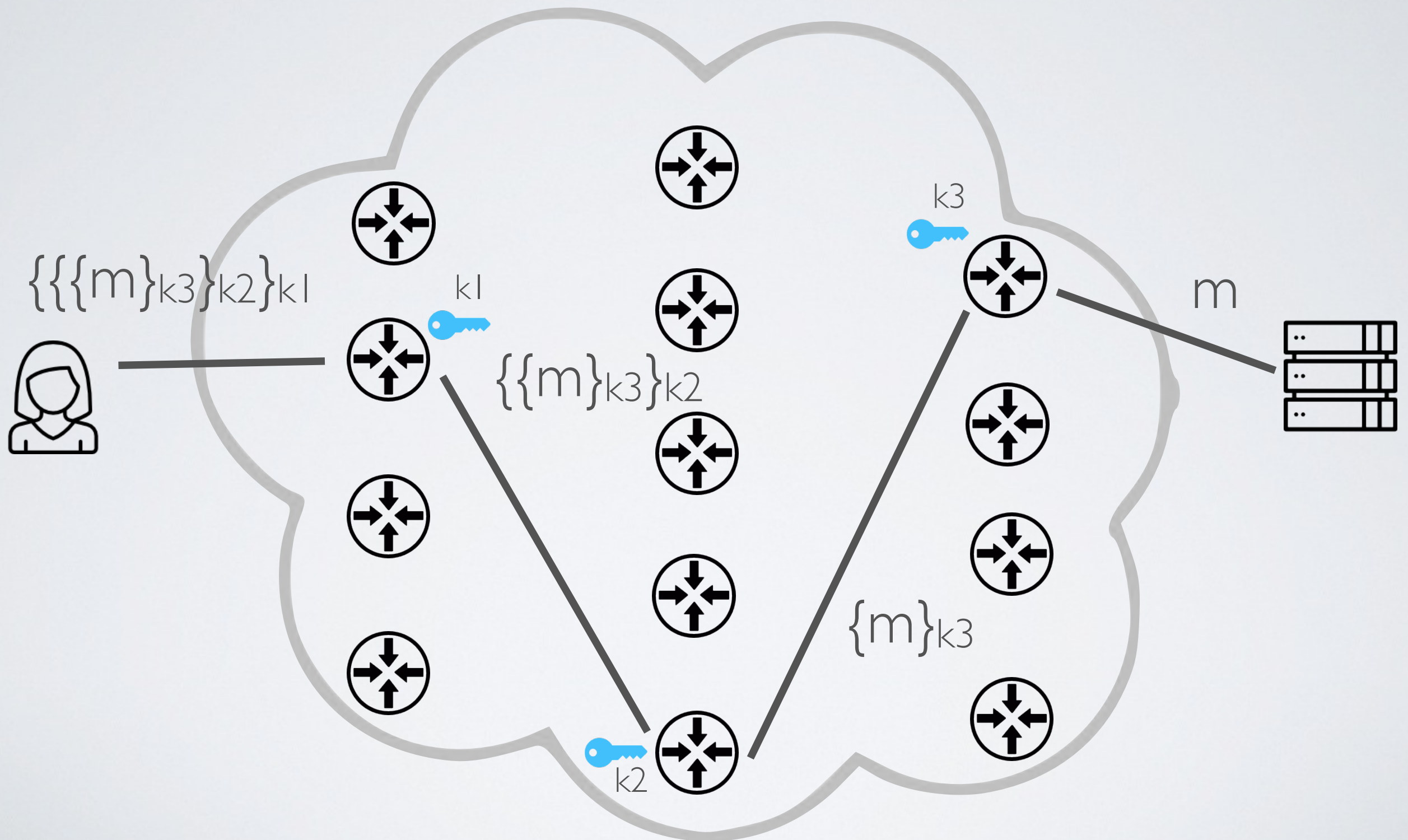
# TOR - The Onion Router

# The TOR network a.k.a Onion Routing

## How The Tor Network Works



# Hiding Alice behind TOR





	knows about
TOR #1 (guard node)	Alice's and TOR #2 IP addresses
TOR #2 Middle Node)	TOR #1 and TOR #2 IP addresses
TOR #3 (Exit node)	TOR #2 and Bob's IP addresses and Alice's content (but not Alice's IP)
Bob	TOR #3 IP address and Alice's content (but not Alice's IP)

➡ Nobody knows about Alice's IP and Alice's content at once

✓ The more TOR nodes are available in the TOR network  
The more secure it is

# The exit node

- ◎ Whatever Alice does illegally on the Internet the exit node might be blame for it
- ➡ Tips for running an exit node (from "TOR blog")  
<https://blog.torproject.org/tips-running-exit-node>

# Limitation of TOR

- ✓ TOR prevents people from identifying you based on your IP address
- TOR does not prevent you from be identified based on application identify information (e.g web tracking)
- ➡ TOR should be used with the TOR browser that deactivates scripts and other tracking mechanisms

# Hiding Bob behind TOR (a.k.a .onion server)

For <http://8t3D01PwqN5fap4n.onion>  
meet me at that RP node (Rendez-vous Point)

