IAIK TU Graz

Data Link & Network Security

Information Security 2019

Johannes Feichtner
johannes.feichtner@iaik.tugraz.at

Computer Networks – How?

Using four layers…

- Application
  - Everything else (HTTP, user applications, etc.)
- Transport
  - Ensure that sent data arrives (TCP)
- Internet
  - Addressing other nodes, routing of packets (IP)
- Link
  - Type of Network: Wireless, Cables, Protocols, Networks

Internet Protocol Suite
Application Layer
BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP · (more)
Transport Layer
TCP · UDP · DCCP · SCTP · RSVP · ECN · (more)
Internet Layer
IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec · (more)
Link Layer
ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) · (more)
v · d · e

IAIK TU Graz

Attack Scenarios

Blogging in Tunesia

- Assumption
  - You are a blogger in Tunesia
  - The government does not like your critical comments on Facebook

- True story happening during the „Arab Spring“ Source: http://goo.gl/6frkcF

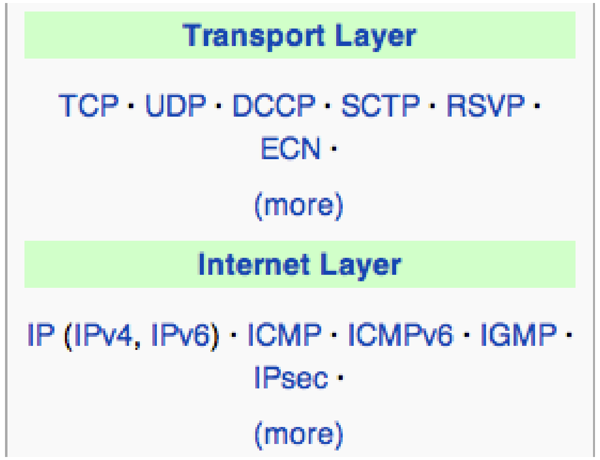
Link: Where are you?

- At home? In a café with WiFi? Using a smartphone?
- Technologies:
  - Wired: LAN
  - WiFi: 802.11a, 802.11b, ..., 802.11ax
  - Mobile Networks: GPRS, ..., HSPA(+), LTE

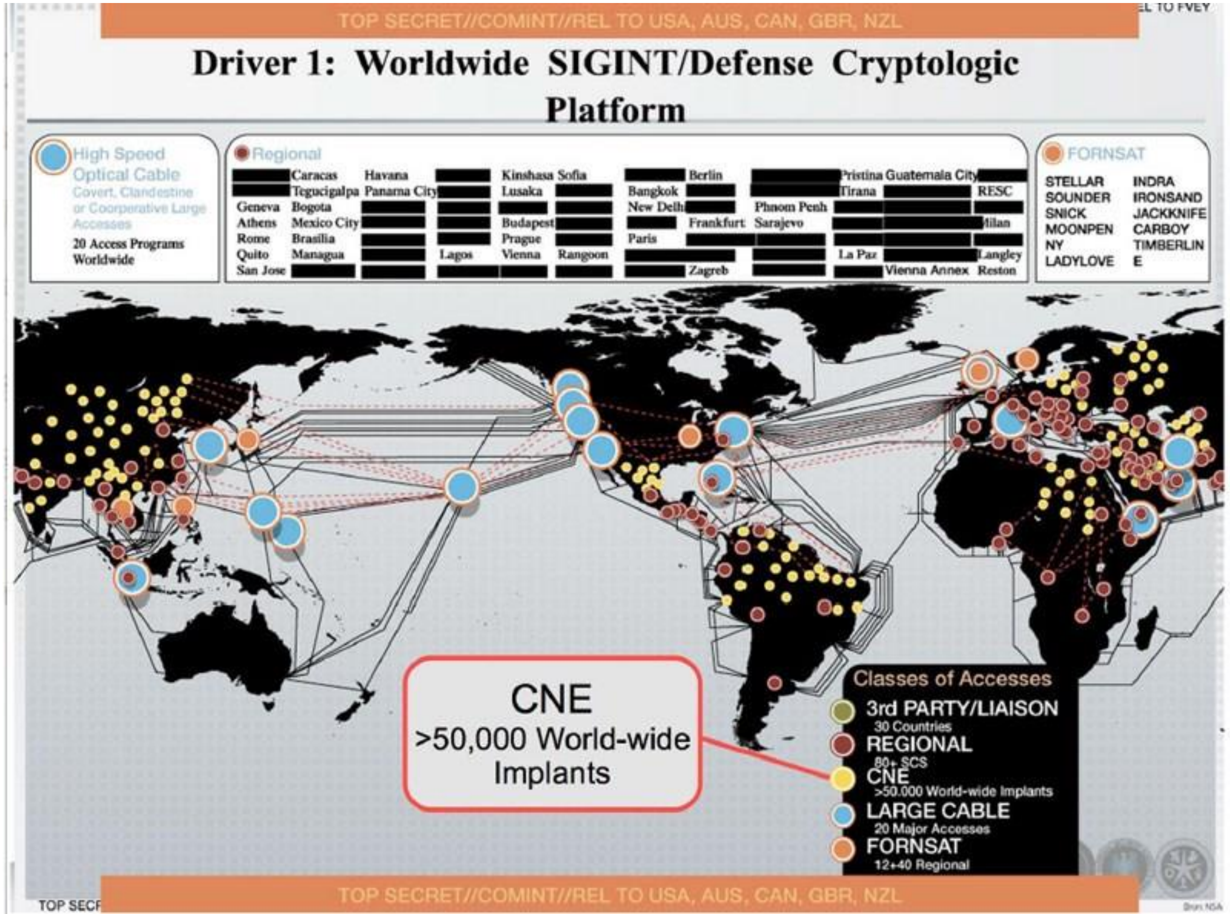
Link Layer

ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) · (more)

Network Layer – Packets



- From your computer to Facebook servers
  - Over the Internet: TCP, UDP, IP
  - Via your local ISP
    through the Internet to Facebook
    - In times of cloud: Which server?
  - Which links do packets take?
  - Leaking meta information?
    - User Profiling?



Application Layer

Application Layer

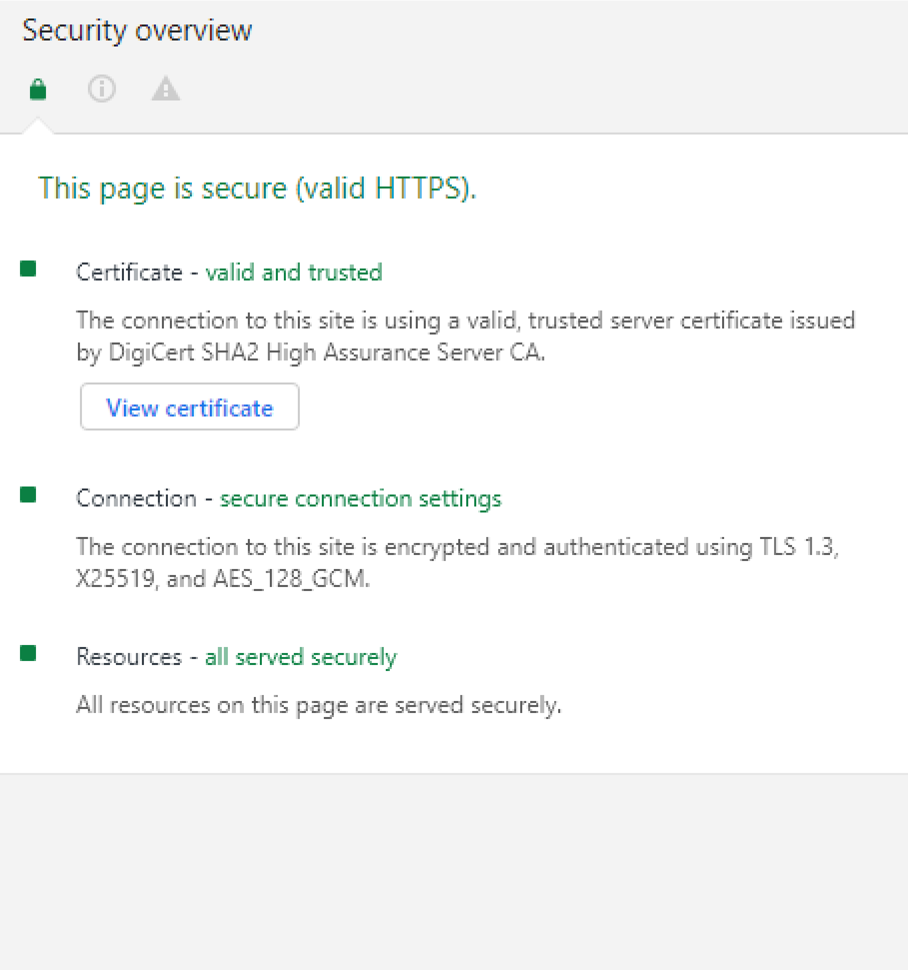
BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·

(more)

Facebook

- Web application: JavaScript, HTML, …
- Communication: AJAX, HTTP
- HTTPS via your browser
- Communication: DNS

```
GET /hprofile-ak-snc4/187700_43202447_1193426_q.jpg HTTP/1.1
Host: profile.ak.fbcdn.net
User-Agent: Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.6; en-US; rv:1.9.2.13) Gecko/20101203 Firefo
Accept: image/png,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Referer: http://www.facebook.com/
```



Scenario 1 – Blogging in Tunesia

- **Abstract**
  - Join WiFi network
  - Fire up your browser and go to Facebook
  - Post something, read feeds…
- **Tools**
  - Browser
  - Wireshark
    - Network Sniffer
    - Captures data on every layer

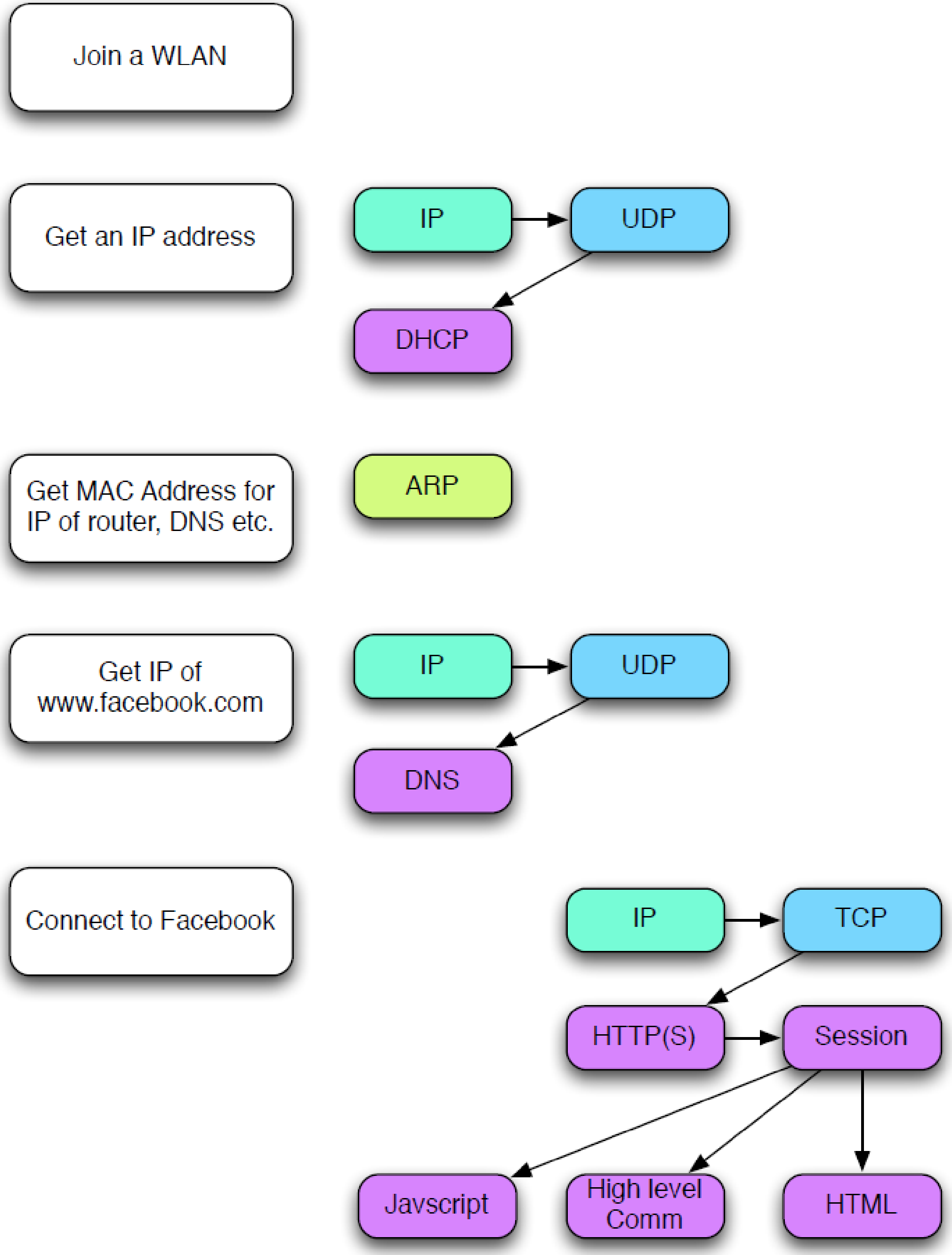
Big Picture...

Application Layer

Transport Layer

Internet Layer

Link Layer



Internet Protocol Suite

Application Layer

BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·

(more)

Transport Layer

TCP · UDP · DCCP · SCTP · RSVP · ECN ·

(more)

Internet Layer

IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec ·

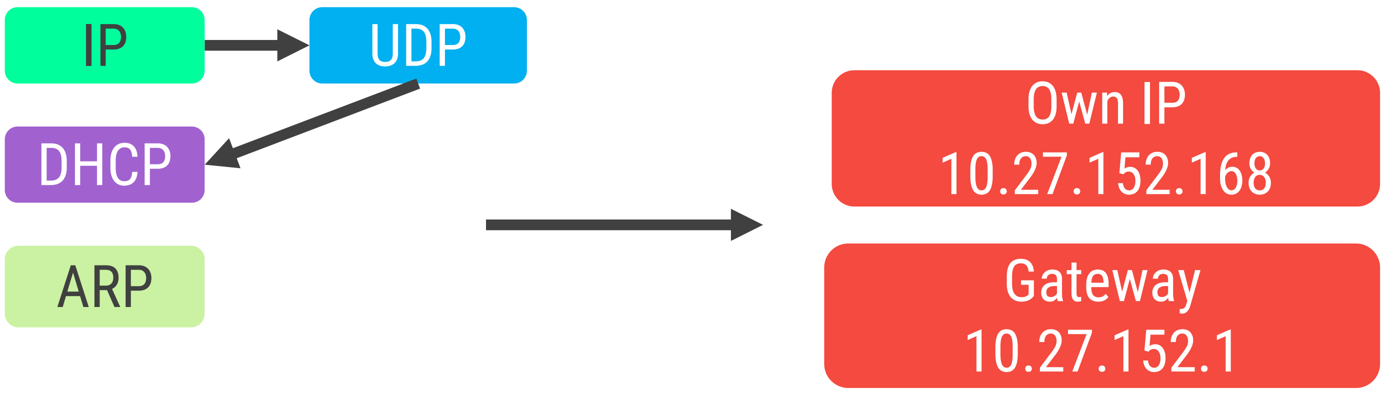
(more)

Link Layer

ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) ·
(more)

v · d · e

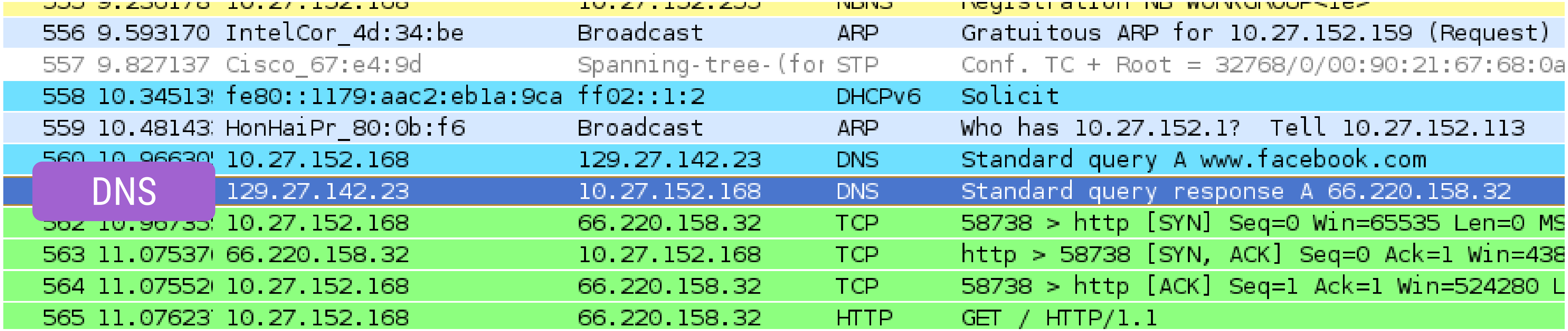
Join a WiFi



1	0.000000	Cisco_67:e4:9d	Spanning-tree-(for	STP	Conf. TC + Root = 32768/0/00:90:21:67:68:0a Cost = 4 Port
2	0.000007	174.36.30.8	10.27.152.168	HTTP	HTTP/1.1 200 OK (text/html)
3	0.000936	fe80::c62c:3ff:fe15:ae5a	ff02::1:ff15:ae5a	ICMPv6	Multicast listener report
		:2c:03:15:ae:5a	3Com_f6:ab:8e	ARP	Who has 10.27.152.1? Tell 10.27.152.168
		.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0x34c277d5
		om_f6:ab:8e	c4:2c:03:15:ae:5a	ARP	10.27.152.1 is at 00:04:75:f6:ab:8e
		:2c:03:15:ae:5a	Broadcast	ARP	Gratuitous ARP for 10.27.152.168 (Request)
		.27.152.5	10.27.152.168	DHCP	DHCP ACK - Transaction ID 0x34c277d5
		:2c:03:15:ae:5a	Broadcast	ARP	Who has 169.254.255.255? Tell 10.27.152.168
10	0.069249	10.27.152.168	224.0.0.2	IGMP	V2 Leave Group 224.0.0.251
11	0.069500	10.27.152.168	224.0.0.251	IGMP	V2 Membership Report / Join group 224.0.0.251
		4:2c:03:15:ae:5a	Broadcast	ARP	Who has 10.27.152.1? Tell 10.27.152.168
		Com_f6:ab:8e	c4:2c:03:15:ae:5a	ARP	10.27.152.1 is at 00:04:75:f6:ab:8e
14	0.072125	10.27.152.168	129.27.142.23	DNS	Standard query PTR lb._dns-sd._udp.0.152.27.10.in-addr.arpa
15	0.072170	10.27.152.168	129.27.142.23	DNS	Standard query TXT cf._dns-sd._udp.0.152.27.10.in-addr.arpa
16	0.072217	10.27.152.168	129.27.142.23	DNS	Standard query PTR b._dns-sd._udp.0.8.16.172.in-addr.arpa
17	0.072262	10.27.152.168	129.27.142.23	DNS	Standard query PTR db._dns-sd._udp.0.8.16.172.in-addr.arpa
18	0.072308	10.27.152.168	129.27.142.23	DNS	Standard query PTR r._dns-sd._udp.0.8.16.172.in-addr.arpa
19	0.072353	10.27.152.168	129.27.142.23	DNS	Standard query PTR dr._dns-sd._udp.0.8.16.172.in-addr.arpa
20	0.072399	10.27.152.168	129.27.142.23	DNS	Standard query PTR lb._dns-sd._udp.0.8.16.172.in-addr.arpa
21	0.072444	10.27.152.168	129.27.142.23	DNS	Standard query TXT cf._dns-sd._udp.0.8.16.172.in-addr.arpa

DHCP
ARP
DHCP
ARP

Where is Facebook?

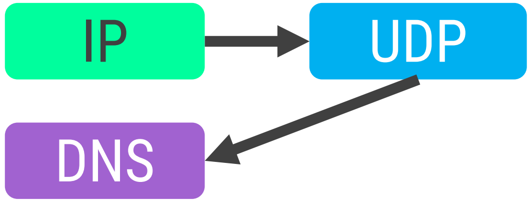


555	9.230178	10.27.152.168	10.27.152.255	NBNS	Registration NB WORKGROUP<1e>
556	9.593170	IntelCor_4d:34:be	Broadcast	ARP	Gratuitous ARP for 10.27.152.159 (Request)
557	9.827137	Cisco_67:e4:9d	Spanning-tree-(for	STP	Conf. TC + Root = 32768/0/00:90:21:67:68:0a
558	10.34513!	fe80::1179:aac2:eb1a:9ca	ff02::1:2	DHCPv6	Solicit
559	10.48143:	HonHaiPr_80:0b:f6	Broadcast	ARP	Who has 10.27.152.1? Tell 10.27.152.113
560	10.96630!	10.27.152.168	129.27.142.23	DNS	Standard query A www.facebook.com
		129.27.142.23	10.27.152.168	DNS	Standard query response A 66.220.158.32
562	10.96733:	10.27.152.168	66.220.158.32	TCP	58738 > http [SYN] Seq=0 Win=65535 Len=0 MS
563	11.07537!	66.220.158.32	10.27.152.168	TCP	http > 58738 [SYN, ACK] Seq=0 Ack=1 Win=438
564	11.07552!	10.27.152.168	66.220.158.32	TCP	58738 > http [ACK] Seq=1 Ack=1 Win=524280 L
565	11.07623'	10.27.152.168	66.220.158.32	HTTP	GET / HTTP/1.1

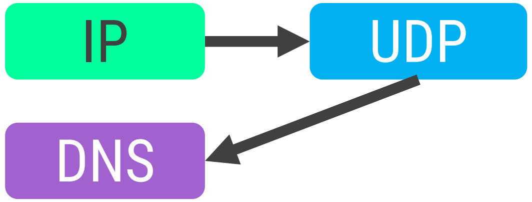
```
▷ Frame 560 (76 bytes on wire, 76 bytes captured)
▷ Ethernet II, Src: c4:2c:03:15:ae:5a (c4:2c:03:15:ae:5a), Dst: 3Com_f6:ab:8e (00:04:75:f6:ab:8e)
▽ Internet Protocol, Src: 10.27.152.168 (10.27.152.168), Dst: 129.27.142.23 (129.27.142.23)
    Version: 4
    Header length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 62
    Identification: 0xfdb7 (64951)
  ▷ Flags: 0x00
    Fragment offset: 0
    Time to live: 255
    Protocol: UDP (0x11)
  ▷ Header checksum: 0x0c01 [correct]
    Source: 10.27.152.168 (10.27.152.168)
    Destination: 129.27.142.23 (129.27.142.23)
▽ User Datagram Protocol, Src Port: 49766 (49766), Dst Port: domain (53)
    Source port: 49766 (49766)
    Destination port: domain (53)
    Length: 42
  ▷ Checksum: 0xb28f [validation disabled]
▽ Domain Name System (query)
    [Response In: 561]
    Transaction ID: 0xe278
  ▷ Flags: 0x0100 (Standard query)
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▽ Queries
    ▽ www.facebook.com: type A, class IN
        Name: www.facebook.com
        Type: A (Host address)
        Class: IN (0x0001)
```

Which IP does Facebook have?

Let’s ask the DNS server 129.27.142.23

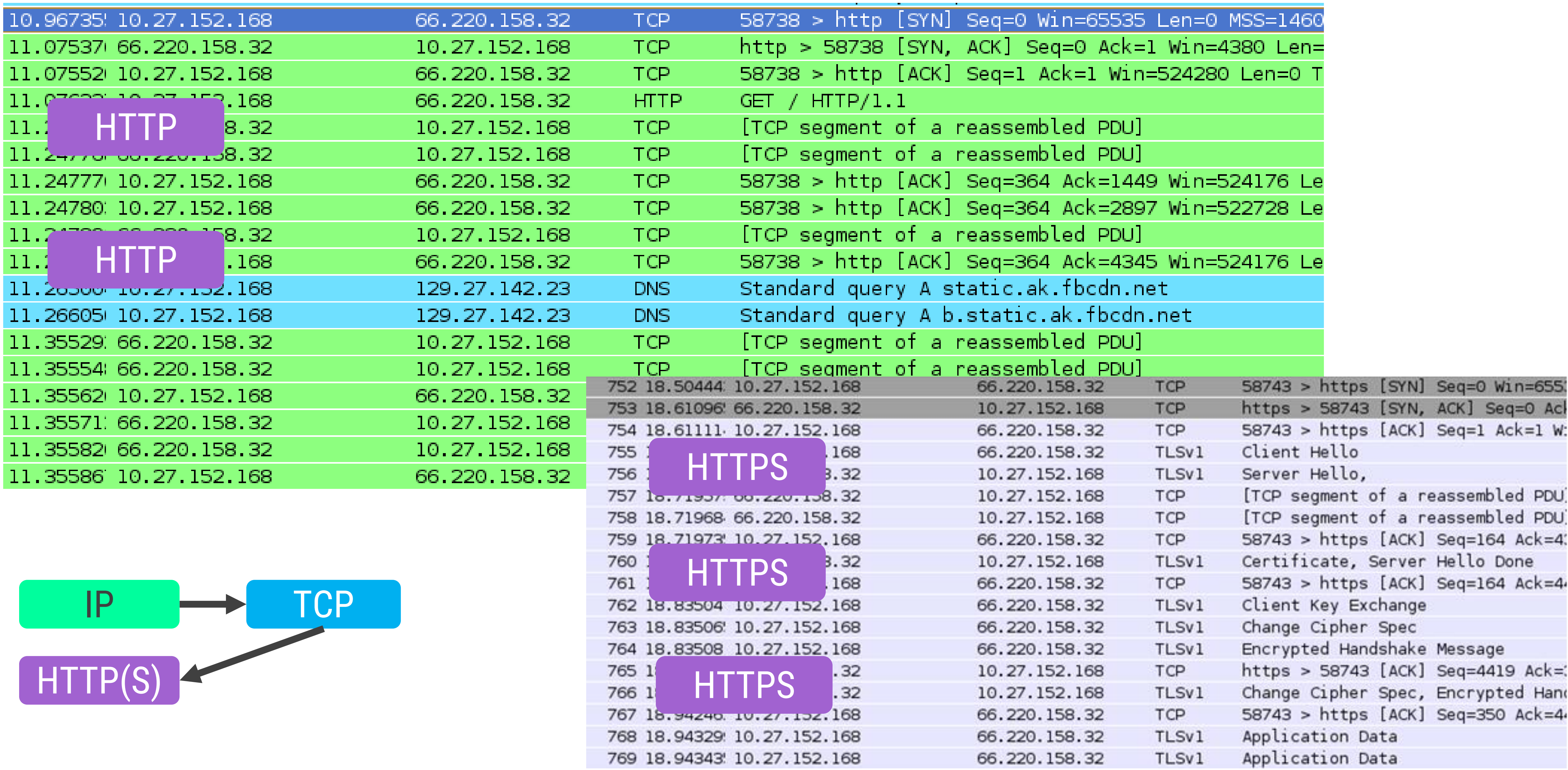


```
▷ Frame 561 (92 bytes on wire, 92 bytes captured)
▷ Ethernet II, Src: 3Com_f6:ab:8e (00:04:75:f6:ab:8e), Dst: c4:2c:03:15:ae:5a (c4:2c:03:15:ae:5a)
▷ Internet Protocol, Src: 129.27.142.23 (129.27.142.23), Dst: 10.27.152.168 (10.27.152.168)
▷ User Datagram Protocol, Src Port: domain (53), Dst Port: 49766 (49766)
▽ Domain Name System (response)
    [Request In: 560]
    [Time: 0.000758000 seconds]
    Transaction ID: 0xe278
  ▷ Flags: 0x8180 (Standard query response, No error)
    Questions: 1
    Answer RRs: 1
    Authority RRs: 0
    Additional RRs: 0
  ▽ Queries
    ▽ www.facebook.com: type A, class IN
        Name: www.facebook.com
        Type: A (Host address)
        Class: IN (0x0001)
  ▽ Answers
    ▽ www.facebook.com: type A, class IN, addr 66.220.158.32
        Name: www.facebook.com
        Type: A (Host address)
        Class: IN (0x0001)
        Time to live: 42 seconds
        Data length: 4
        Addr: 66.220.158.32
```



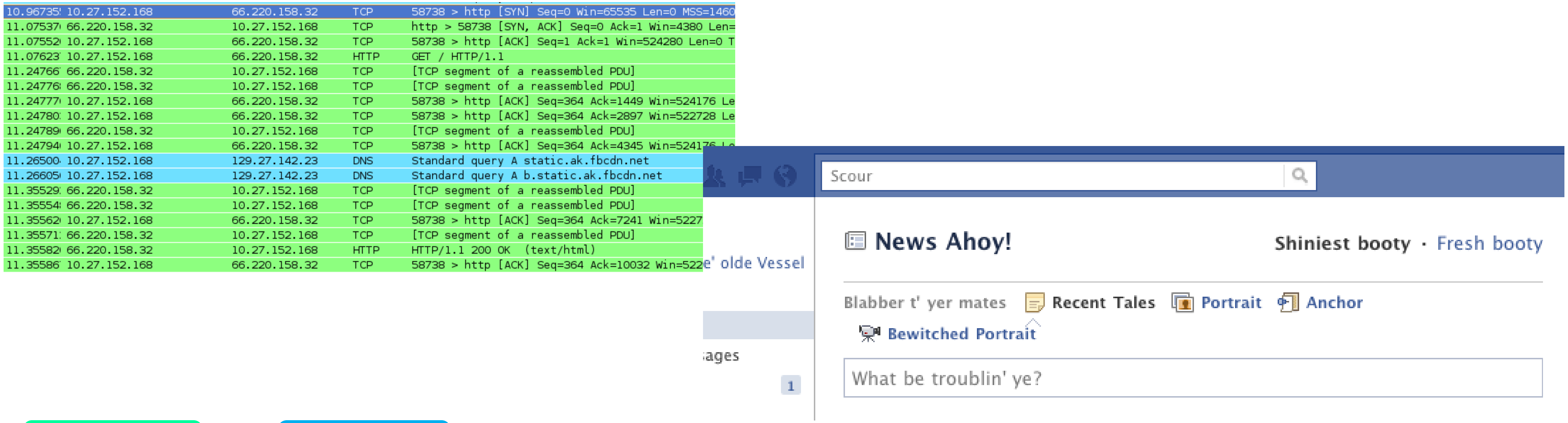
The answer is 66.220.158.32

Let‘s Login

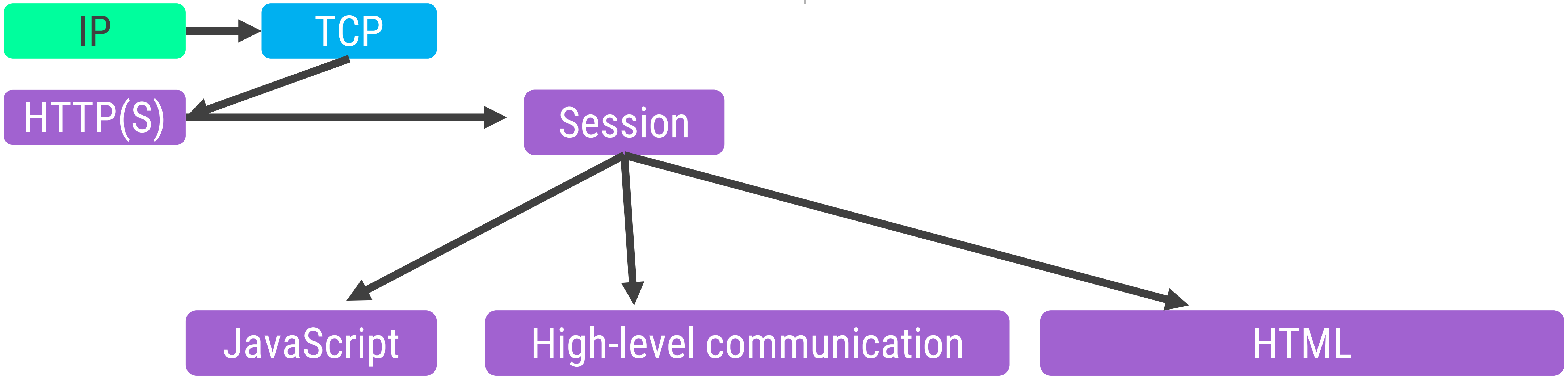


10.96735 10.27.152.168 66.220.158.32 TCP 58738 > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460
11.07537 66.220.158.32 10.27.152.168 TCP http > 58738 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=
11.07552 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=1 Ack=1 Win=524280 Len=0 T
66.220.158.32 HTTP GET / HTTP/1.1
10.27.152.168 TCP [TCP segment of a reassembled PDU]
10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.24777 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=1449 Win=524176 Le
11.24780 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=2897 Win=522728 Le
10.27.152.168 TCP [TCP segment of a reassembled PDU]
66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=4345 Win=524176 Le
129.27.142.23 DNS Standard query A static.ak.fbcdn.net
11.26605 10.27.152.168 129.27.142.23 DNS Standard query A b.static.ak.fbcdn.net
11.35529 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.35554 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.35562 10.27.152.168 66.220.158.32
11.35571 66.220.158.32 10.27.152.168
11.35582 66.220.158.32 10.27.152.168
11.35586 10.27.152.168 66.220.158.32
HTTP
HTTP
752 18.50444 10.27.152.168 66.220.158.32 TCP 58743 > https [SYN] Seq=0 Win=655
753 18.61096 66.220.158.32 10.27.152.168 TCP https > 58743 [SYN, ACK] Seq=0 Ack
754 18.61111 10.27.152.168 66.220.158.32 TCP 58743 > https [ACK] Seq=1 Ack=1 W
755 66.220.158.32 TLSv1 Client Hello
756 10.27.152.168 TLSv1 Server Hello,
757 10.27.152.168 TCP [TCP segment of a reassembled PDU]
758 18.71968 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
759 66.220.158.32 TCP 58743 > https [ACK] Seq=164 Ack=4
760 10.27.152.168 TLSv1 Certificate, Server Hello Done
761 66.220.158.32 TCP 58743 > https [ACK] Seq=164 Ack=4
762 66.220.158.32 TLSv1 Client Key Exchange
763 18.83506 10.27.152.168 66.220.158.32 TLSv1 Change Cipher Spec
764 18.83508 10.27.152.168 66.220.158.32 TLSv1 Encrypted Handshake Message
765 10.27.152.168 TCP https > 58743 [ACK] Seq=4419 Ack=
766 10.27.152.168 TLSv1 Change Cipher Spec, Encrypted Hand
767 66.220.158.32 TCP 58743 > https [ACK] Seq=350 Ack=4
768 18.94329 10.27.152.168 66.220.158.32 TLSv1 Application Data
769 18.94343 10.27.152.168 66.220.158.32 TLSv1 Application Data
HTTPS
HTTPS
HTTPS
IP
TCP
HTTP(S)

Let's Post



10.96735 10.27.152.168 66.220.158.32 TCP 58738 > http [SYN] Seq=0 Win=65535 Len=0 MSS=1460
11.07537 66.220.158.32 10.27.152.168 TCP http > 58738 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=
11.07552 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=1 Ack=1 Win=524280 Len=0 T
11.07623 10.27.152.168 66.220.158.32 HTTP GET / HTTP/1.1
11.24766 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.24776 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.24777 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=1449 Win=524176 Le
11.24780 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=2897 Win=522728 Le
11.24789 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.24794 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=4345 Win=524176
11.26500 10.27.152.168 129.27.142.23 DNS Standard query A static.ak.fbcdn.net
11.26605 10.27.152.168 129.27.142.23 DNS Standard query A b.static.ak.fbcdn.net
11.35529 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.35554 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.35562 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=7241 Win=5227
11.35571 66.220.158.32 10.27.152.168 TCP [TCP segment of a reassembled PDU]
11.35582 66.220.158.32 10.27.152.168 HTTP HTTP/1.1 200 OK (text/html)
11.35586 10.27.152.168 66.220.158.32 TCP 58738 > http [ACK] Seq=364 Ack=10032 Win=522
Scour
News Ahoy!
Shiniest booty · Fresh booty
e' olde Vessel
Blabber t' yer mates
Recent Tales
Portrait
Anchor
Bewitched Portrait
ages
1
What be troublin' ye?



IP
TCP
HTTP(S)
Session
JavaScript
High-level communication
HTML

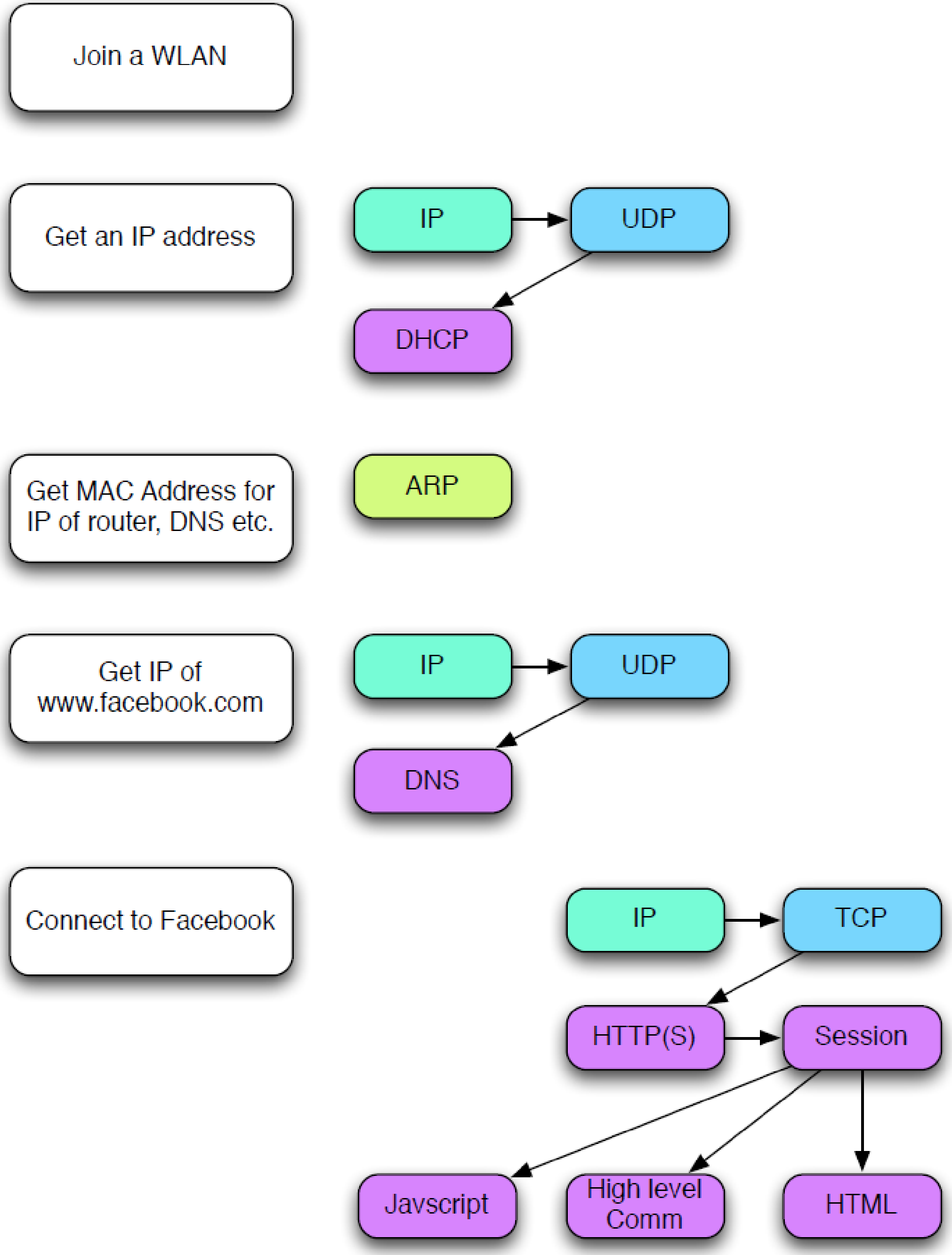
Big Picture...

Application Layer

Transport Layer

Internet Layer

Link Layer



Internet Protocol Suite

Application Layer

BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·

(more)

Transport Layer

TCP · UDP · DCCP · SCTP · RSVP · ECN ·

(more)

Internet Layer

IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec ·

(more)

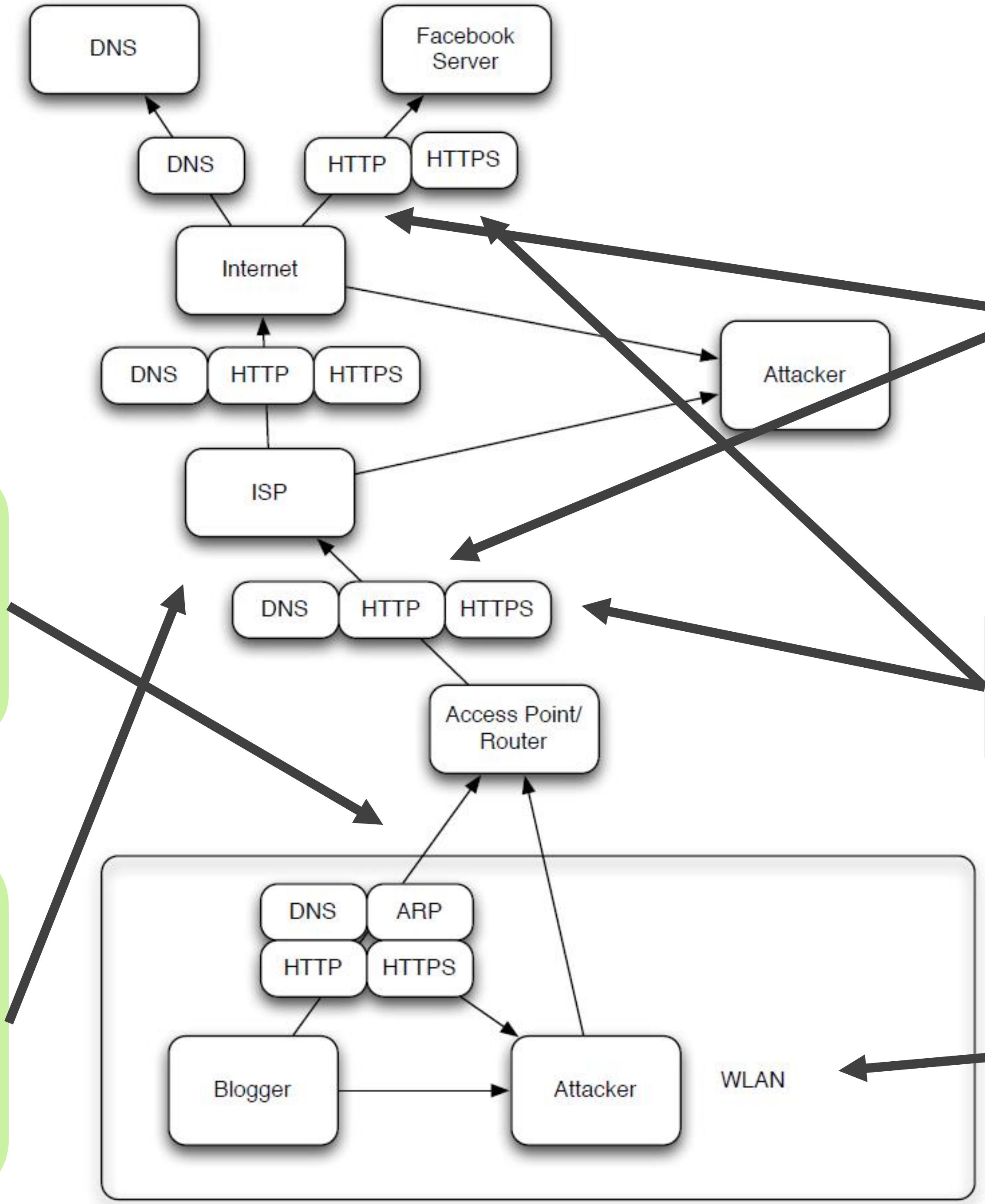
Link Layer

ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) ·

(more)

v · d · e

Attacks



Capture HTTP Session

Poison ARP cache, make traffic flow through attacker

Fake HTTPS certificate, make user accept it

Fake DNS Reply, bring user to another domain for account data phishing

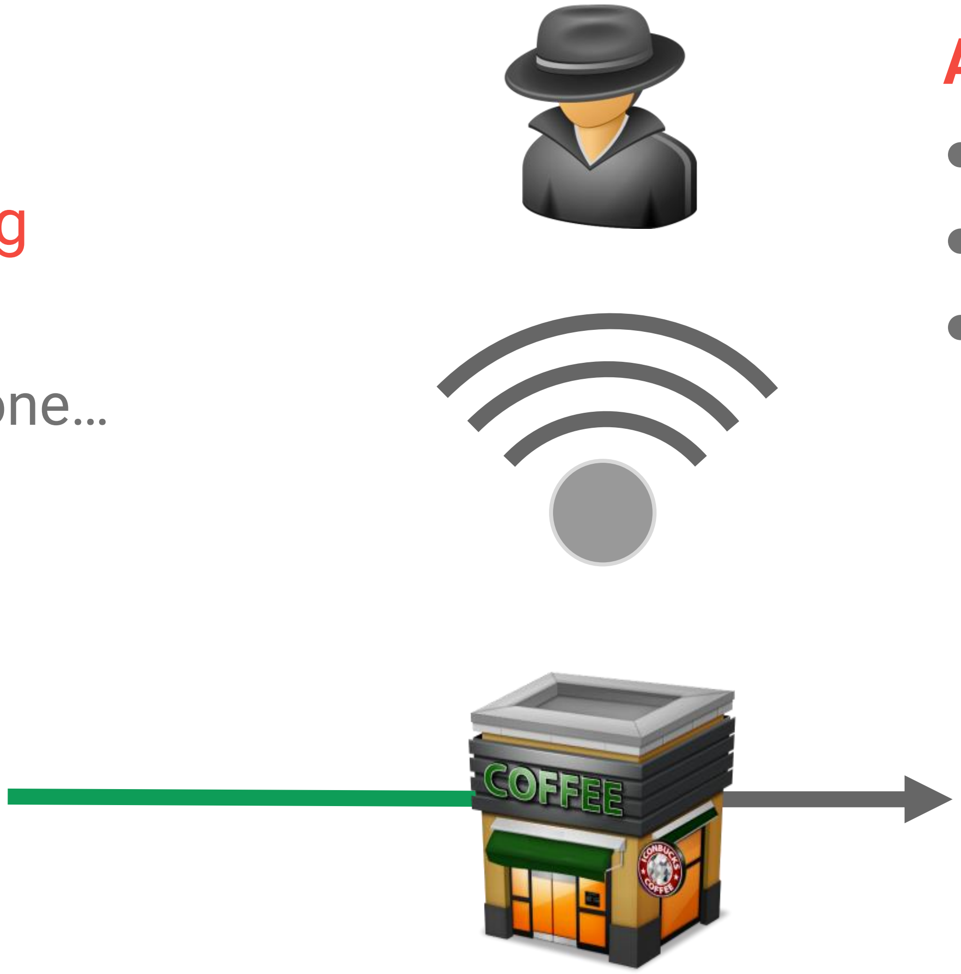
Fake Access Point

Attack – Open WLAN

How?

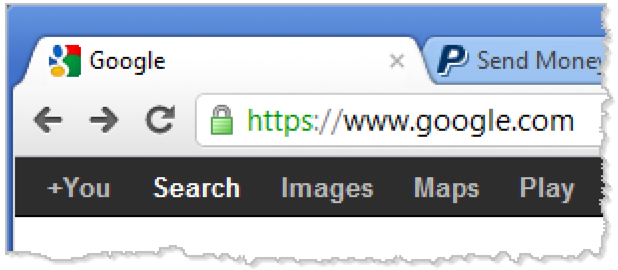
a) Join WLAN,
   start ARP Poisoning

b) Create own AP
   – E.g. with smartphone…



Attacker

- Sniff data
- Manipulate data
- Attack HTTPS connections



Client

http://www.apple.com
http://www.microsoft.com
https://www.google.com

IAIK TU Graz

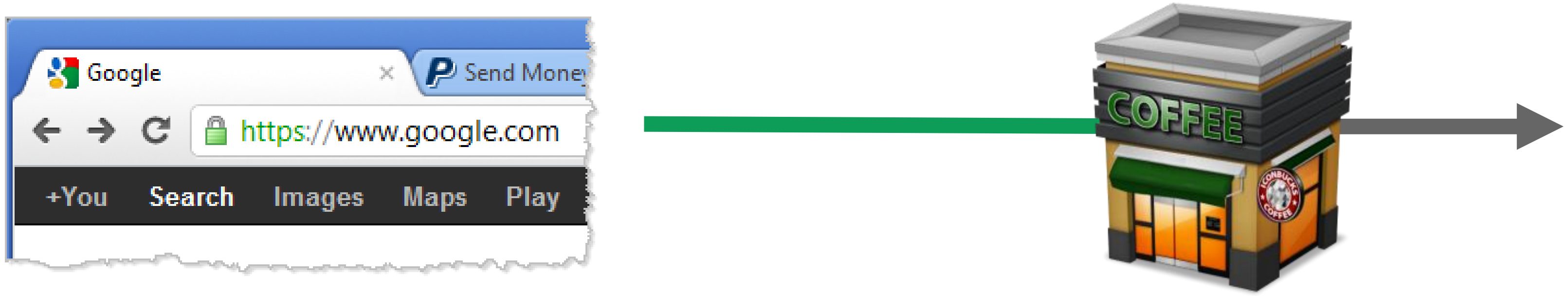
Attack – Sniff data

Unencrypted (HTTP)

- Credentials, cookies, content
- Derive usage patterns
  - Which hosts visited?
  - Called URLs?

Encrypted (HTTPS)

- Find out communication partners (IP addresses)

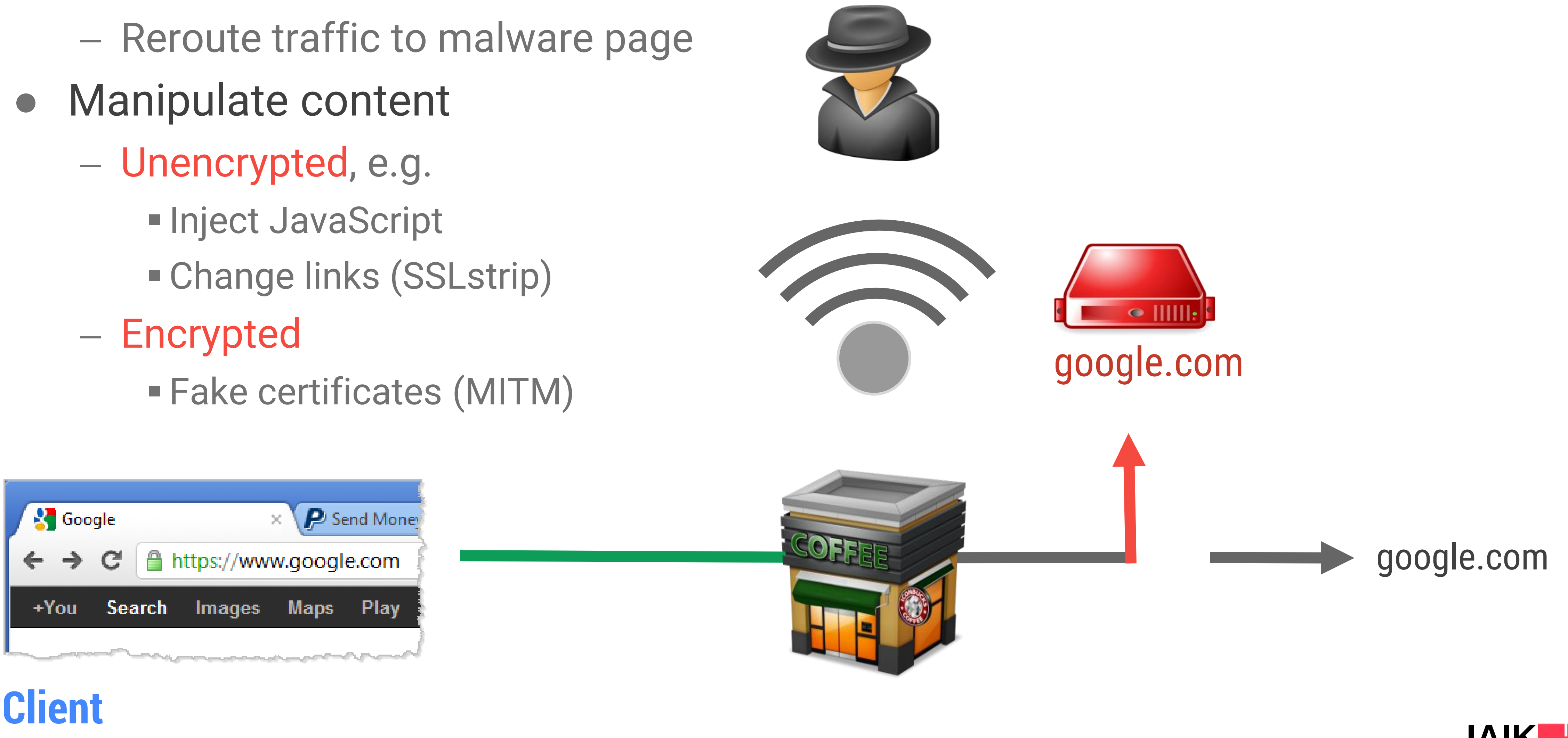


http://www.apple.com
http://www.microsoft.com
https://www.google.com

Client

Attack – Manipulate data

- Fake DNS replies
  - Reroute traffic to malware page
- Manipulate content
  - Unencrypted, e.g.
    - Inject JavaScript
    - Change links (SSLstrip)
  - Encrypted
    - Fake certificates (MITM)



IAIK TU Graz

Attack – HTTPS Traffic

...by faking server certificates

The problem

- Users often accept invalid certificates anyway

- We have ~130 certificate authorities (CA) in our browsers‘ trust stores
  - They are not equally rigid when issuing certificates

  →Certificate could be obtained and misused

- Especially with mobile apps
  - Can overwrite certificate validation routines
  - Many apps silently (without warning) accept invalid certificates

Back in Tunesia…

Attacks in 2011/2012

- Facebook was largely HTTP
- HTTPS only
  - For login (password protection)
  - If explicitly requested (https://www.facebook.com)
- Tunesia had a national (stated-owned) ISP

Scenario

- ISP has access to Facebook HTTP transmissions
- Injects JavaScript code into Facebook logon page
- JS in your browser reads entered user and password → posts it to non-existing URI, e.g. http://www.facebook.com/fake/user/password
- ISP catches HTTP URLs and reads out user/password

Scenario 1 - Turkey (2014)

Situation: Government decides to block Twitter

How?

- Send orders to every national ISP
- Task: Any DNS request to `*.twitter.com` should not be resolved

Remedy: Change your ISP‘s DNS servers to use alternatives, e.g. from Google or OpenDNS

Source: http://goo.gl/aqPFFu

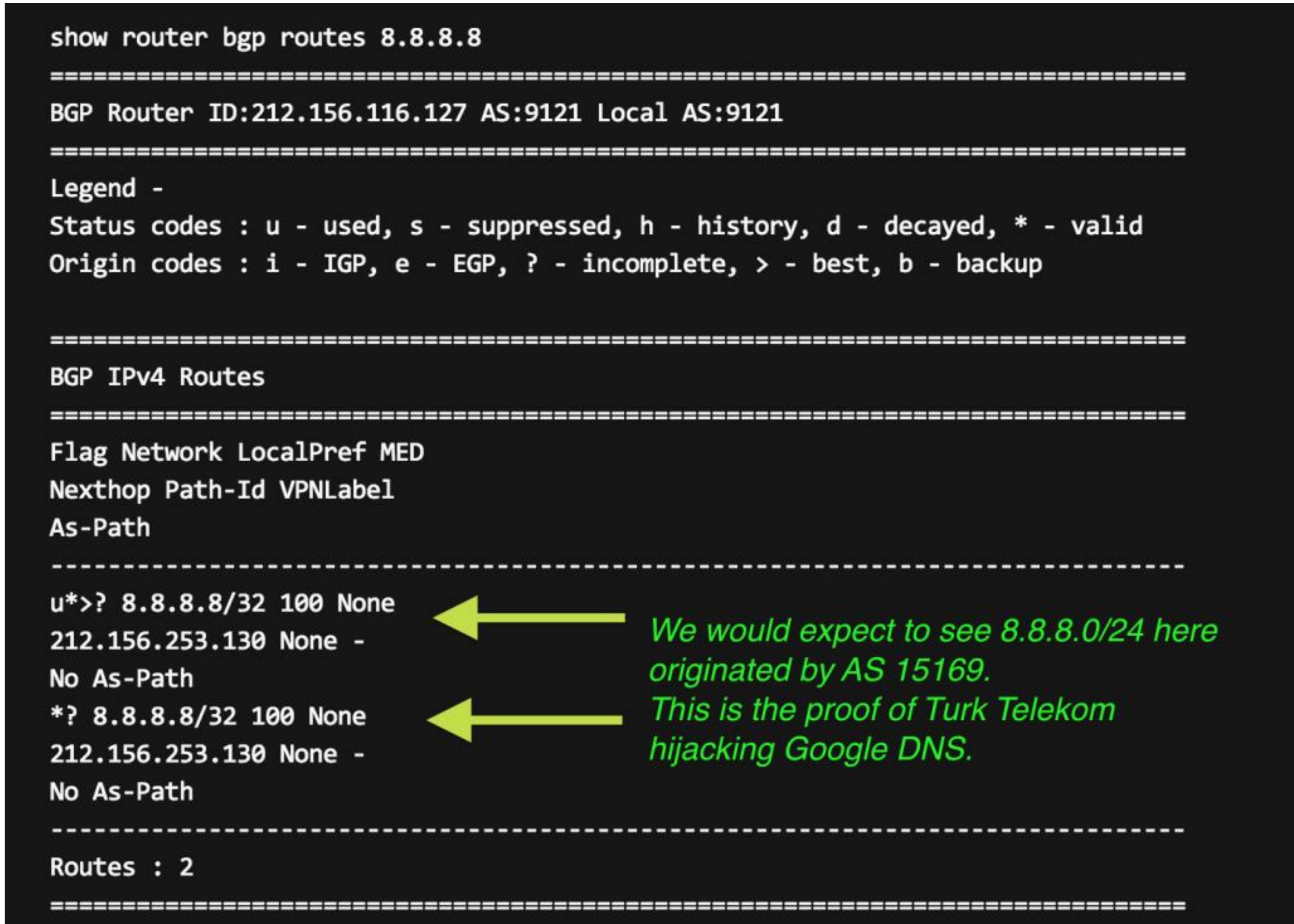
Turkey cont. (2014)

New situation: Government decides to block Google‘s DNS servers

How?

On AS 9121(TurkTelecom)
re-route all traffic to 8.8.8.8
to 212.156.253.130 instead.
→ BGP Hijacking

Null routing would break
connectivity for all users of 8.8.8.8

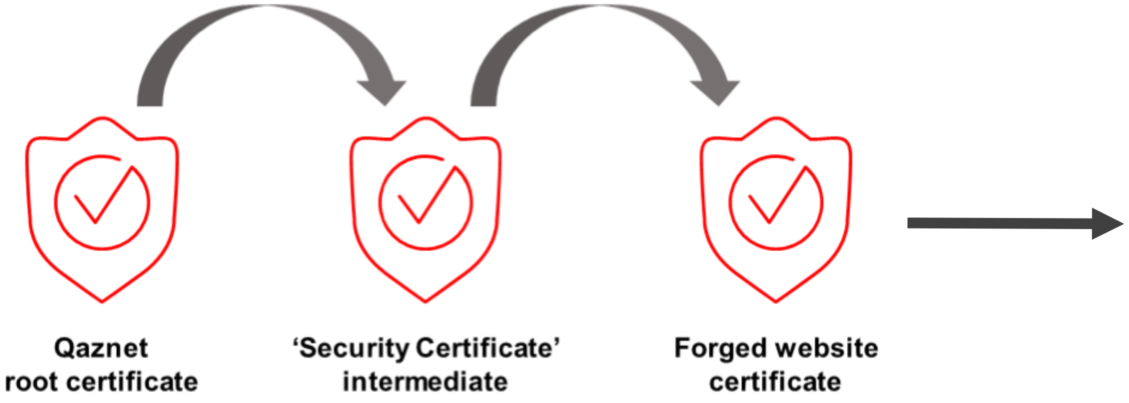


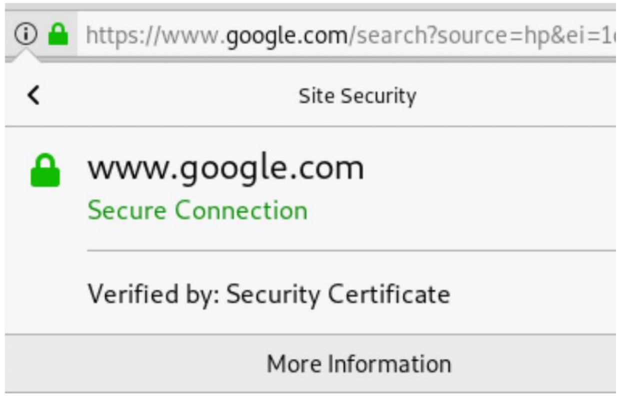
Source: https://goo.gl/x31tCE

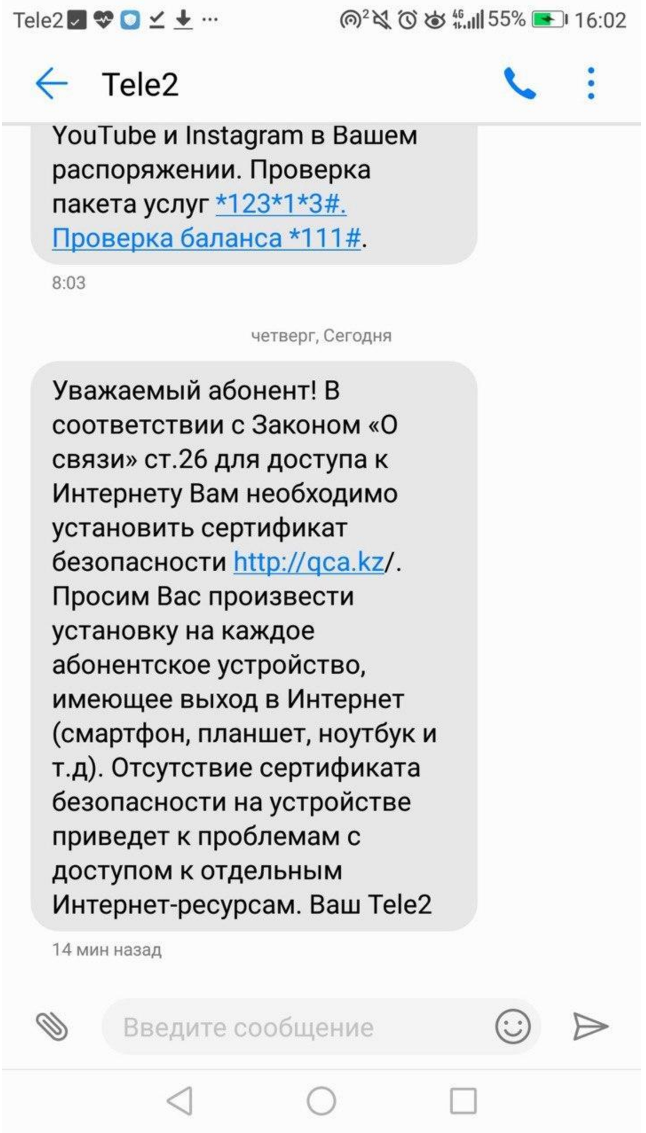
Scenario 2 – Kazakhstan (2019)

- Government requires citizens to install trust anchor
  - Law to „improve nation‘s security“
- In 07/2019 users receive SMS with request to visit qca.kz and install a certificate from „Qaznet Trust Network“

→ How to prevent/bypass this attack? :-)

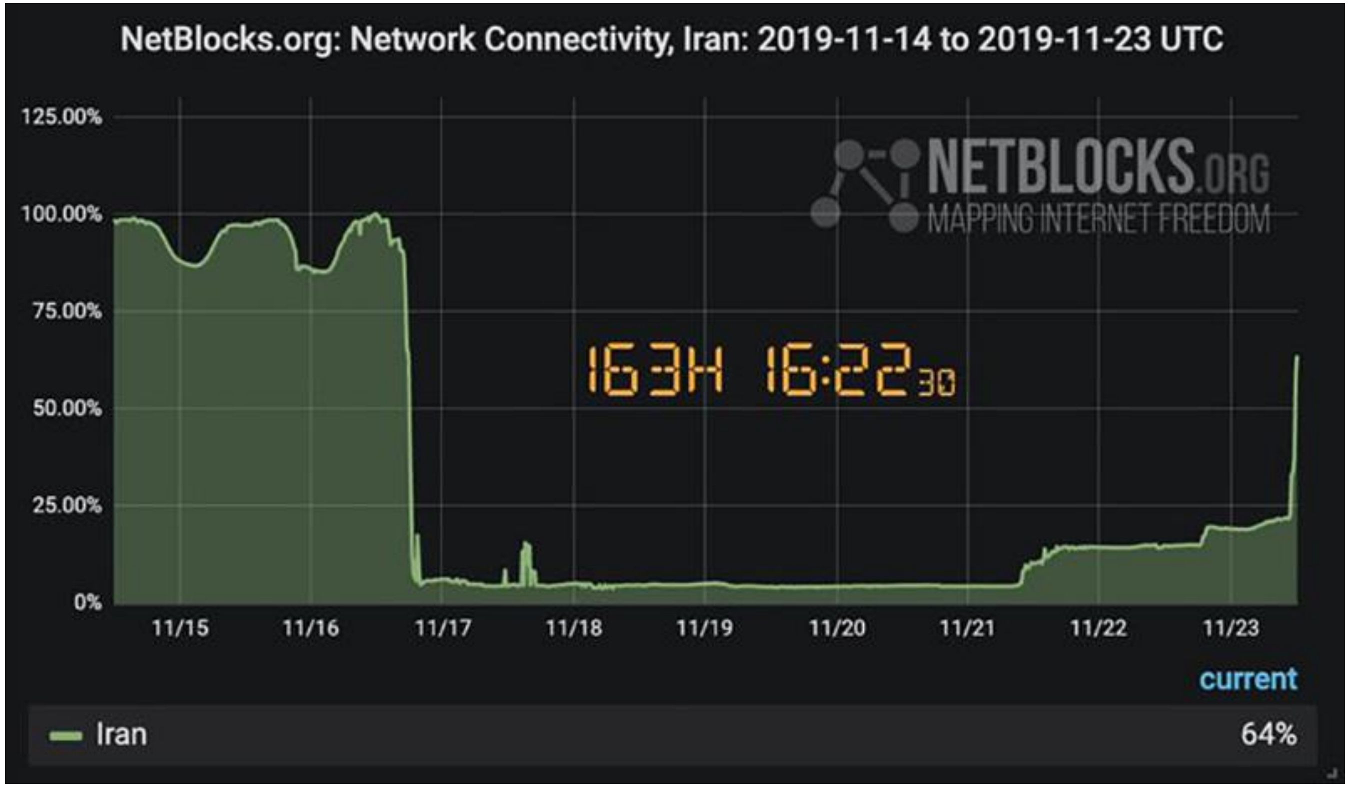






Scenario 4 – Iran (2019)

Within 24h → government sends shutdown requests to all ISPs



Source: https://goo.gl/x31tCE

1. Only downlinks blocked, uplinks still worked
2. Everything blocked

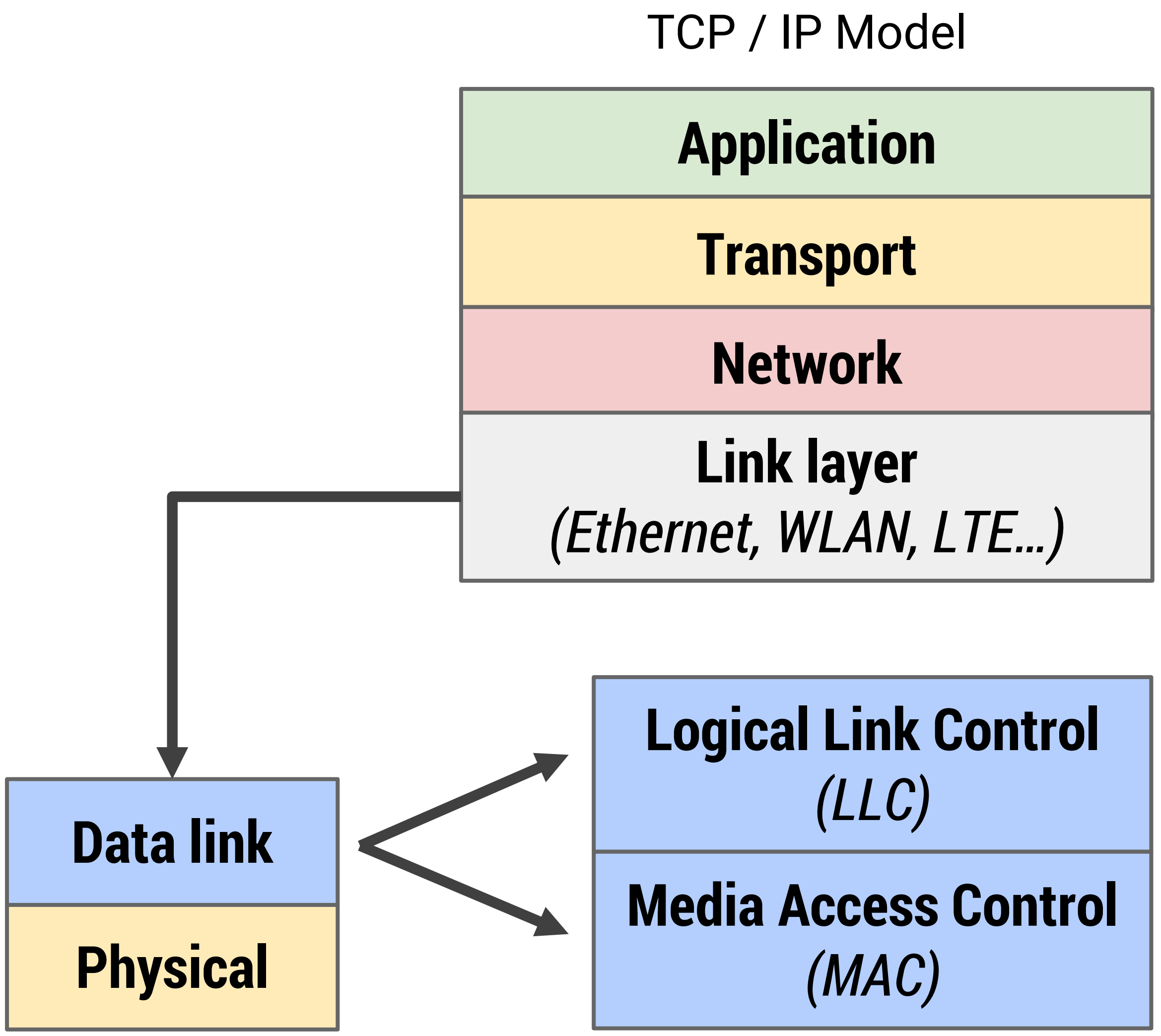
Techniques

Review: Link Layer

- IEEE 802
  - Logical Link Control (LLC)
  - Media Access Control (MAC)
  - Ethernet (LAN)
    - Frame Collisions
  - VLANs

- Cables, Hubs, Switches

- Wireless Networks
  - Basics: CSMA/CA, Channels
  - Attacks



IAIK TU Graz

Switch / Hub – Security

Hub attack

- Every node sees whole traffic
- Sniffing is easy by setting network card into „promiscuous mode“

Switch attacks

- MAC Flooding
  - Flood switch with fake MAC addresses until memory exhausted
  - Switch then changes mode and behaves like hub

- MAC Spoofing
  - Fake foreign MAC address. Switch then redirects traffic to port of attacker

- ARP Poisoning

Attacks on Switched Networks

- MAC Flooding
  - Flood switch with fake MAC addresses until memory exhausted
  - Switch then changes mode and behaves like hub
- MAC Spoofing
  - Emulate foreign MAC address
  - Targets the SAT of the switch
- ARP Poisoning / Spoofing
  - Targets other clients

MAC Spoofing

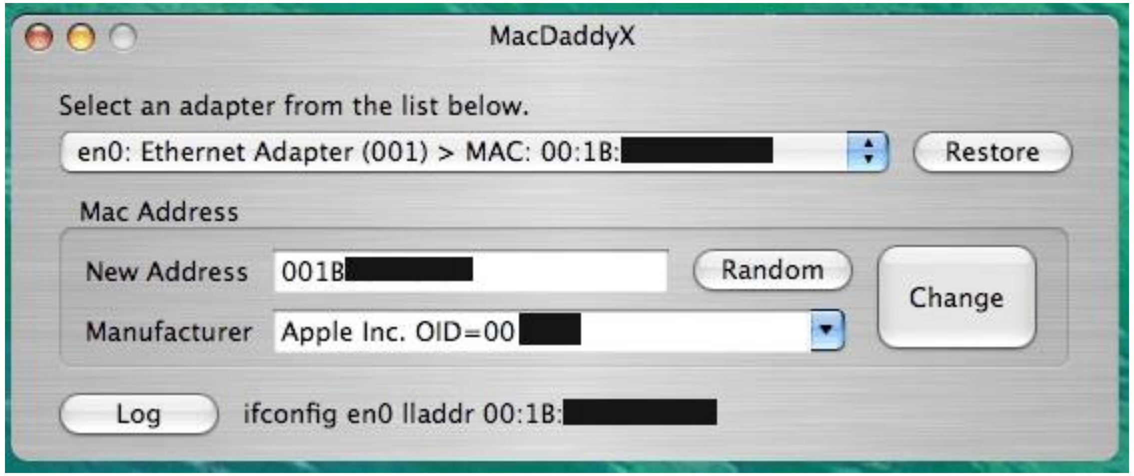
Workflow

1. Attacker forges MAC address of victim host
   - Switch refreshes SAT mapping: MAC address <-> switch port
2. Switch redirects incoming frames to attacker
3. Works until victim sends new frame

Purpose

- Can suffice to capture credentials
- Denial-of-Service (DoS) attack
  - Prevent IP connectivity with spoofed victims
- Fake identity in WiFis that require username / password

See: https://goo.gl/I8AOKL



IAIK TU Graz

Review: TCP / UDP

- Service provisioned to higher layers through ports
  - Port 80 for HTTP, 443 for HTTPS / TLS, 21 for FTP, …
- Session: Communication client / server via socket pair
  - TCP: Established after fulfilling a handshake
    - Connection-oriented
    - Reliable → error detection, flow & congestion control
  - UDP: Identified on higher layer, e.g. using session cookies
    - Connection-less
    - Unreliable → sender does not know if destination reached
    - No congestion control

HTTP!

Internet Protocol Suite
Application Layer
BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·
(more)
Transport Layer
TCP · UDP · DCCP · SCTP · RSVP · ECN ·
(more)
Internet Layer
IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec ·
(more)
Link Layer
ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) · (more)
v · d · e

TCP Scanning

Portscan → Check whether specific ports are open on host

How?

- TCP SYN Scan - „Half-open“ scanning
  - Attacker sends SYN packet
    - If server answers with SYN/ACK packet → port open
    - If server answers with RST packet → port closed
  - Attacker sends RST packet instead of ACK

- TCP FIN Scan
  - Attacker sends FIN packet
    - If port is open → server ignores FIN packet
    - If port is closed → server answers with RST packet

Nmap

Leading tool for portscanning: https://nmap.org/

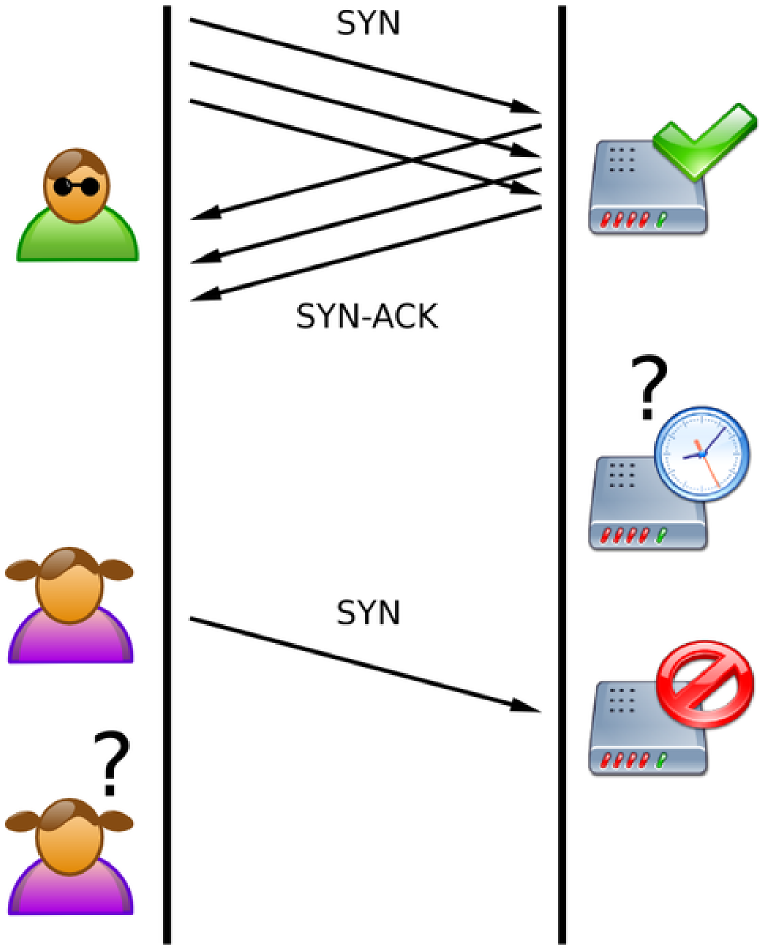
Features include

- IP & Port Scans – UDP / TCP (SYN, FIN Scanning)
- OS Fingerprinting
  - Determine running OS by checking reaction to uncommon packets
    - Use of reserved flags in TCP header
    - Use of weird flag combination
    - Selection of initial SEQ numbers
    - Analysis to ICMP responses
  - Each TCP/IP implementation is different in handling corner cases

Source: http://goo.gl/XPe0Ok

Attack: SYN Flooding

Very common DoS attack!



Source: https://goo.gl/6lUQ7a

Idea

1. Attacker starts handshake with SYN segment
2. Victim replies with SYN-ACK
   → Allocates data structures (reassembly buffer, etc.)
3. Attacker host stays silent

Problem:

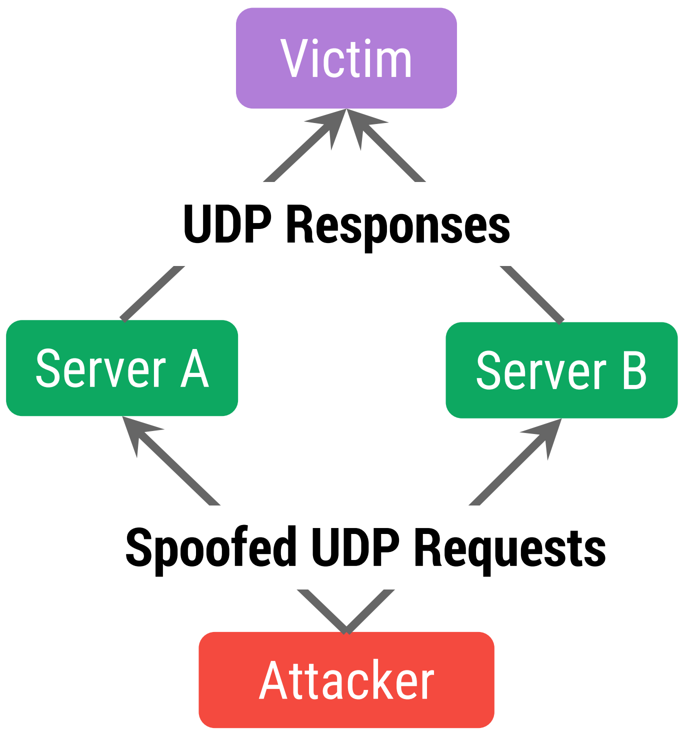
Hosts can only keep limited number of TCP connections in *half-open* state to limit memory usage → after that limit, no more connections accepted!

Solution (not always): Drop half-open connections (FIFO), SYN cookies

Attack: UDP Reflection Attack

Distributed Denial of Service attack

→ Send packets with forged source IP address and let server answer large replies to victim



Basic Workflow

1. Attacker sends UDP requests spoofing the victim‘s IP address, e.g. NTP or DNS request
2. Servers sends response to victim
   → x-times larger than request

Consequences

- Victim overwhelmed with traffic
- Attacker barely traceable

ARP Spoofing

Scenario: Blogging in Tunesia

Internet Protocol Suite

Application Layer

BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·

(more)

Transport Layer

TCP · UDP · DCCP · SCTP · RSVP · ECN ·

(more)

Internet Layer

IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec ·

(more)

Link Layer

ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) ·

(more)

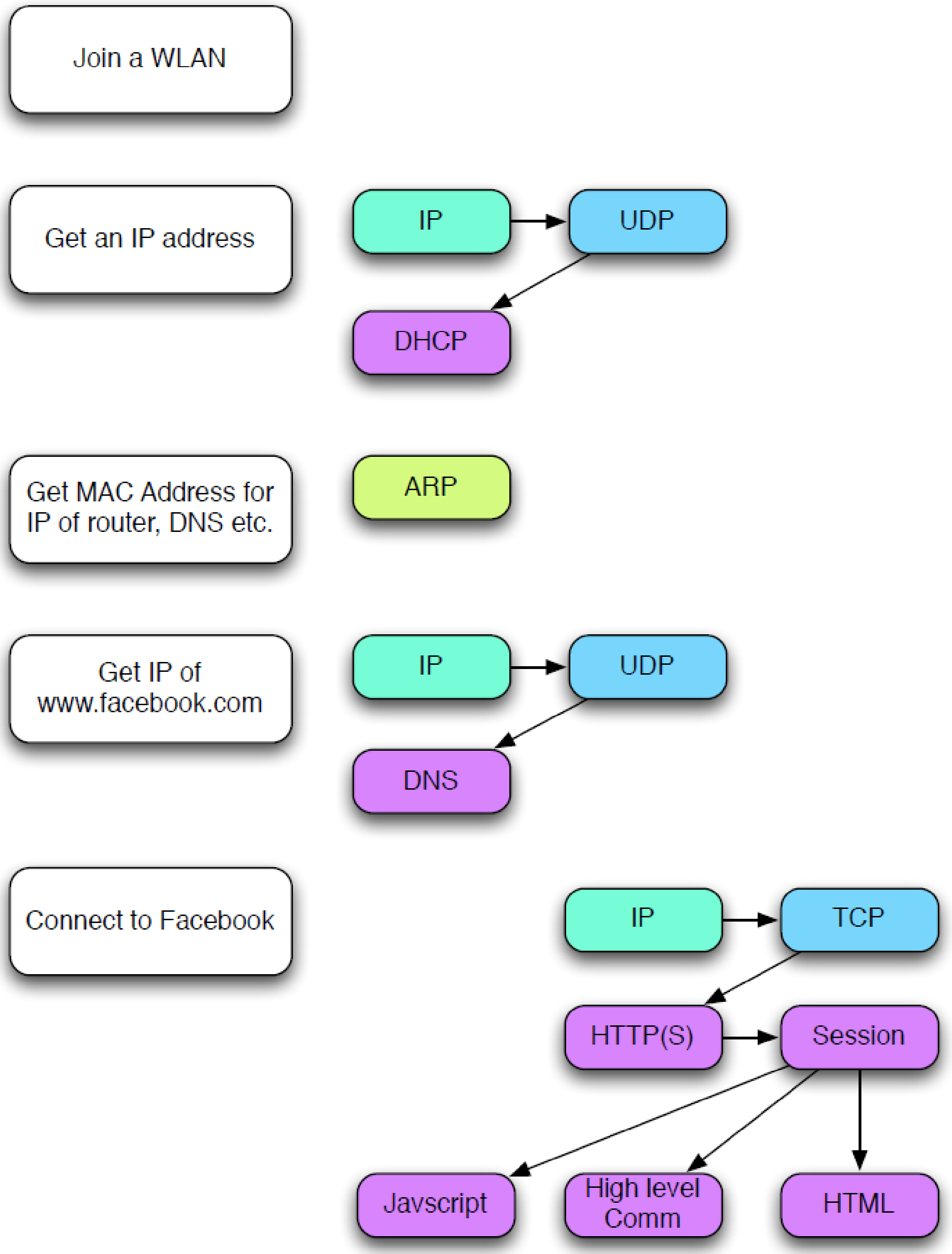
v · d · e

Application Layer

Transport Layer

Internet Layer

Link Layer



Scenario: Blogging in Tunesia

Internet Protocol Suite

Application Layer

BGP · DHCP · DNS · FTP · HTTP · IMAP · IRC · LDAP · MGCP · NNTP · NTP · POP · RIP · RPC · RTP · SIP · SMTP · SNMP · SSH · Telnet · TLS/SSL · XMPP ·

(more)

Transport Layer

TCP · UDP · DCCP · SCTP · RSVP · ECN ·

(more)

Internet Layer

IP (IPv4, IPv6) · ICMP · ICMPv6 · IGMP · IPsec ·

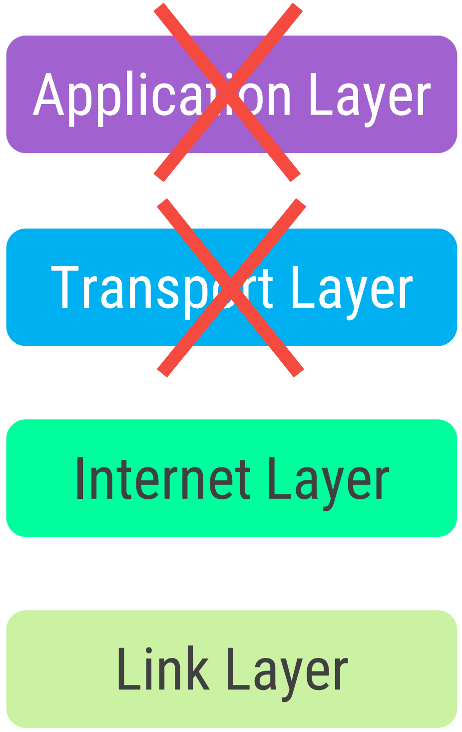
(more)

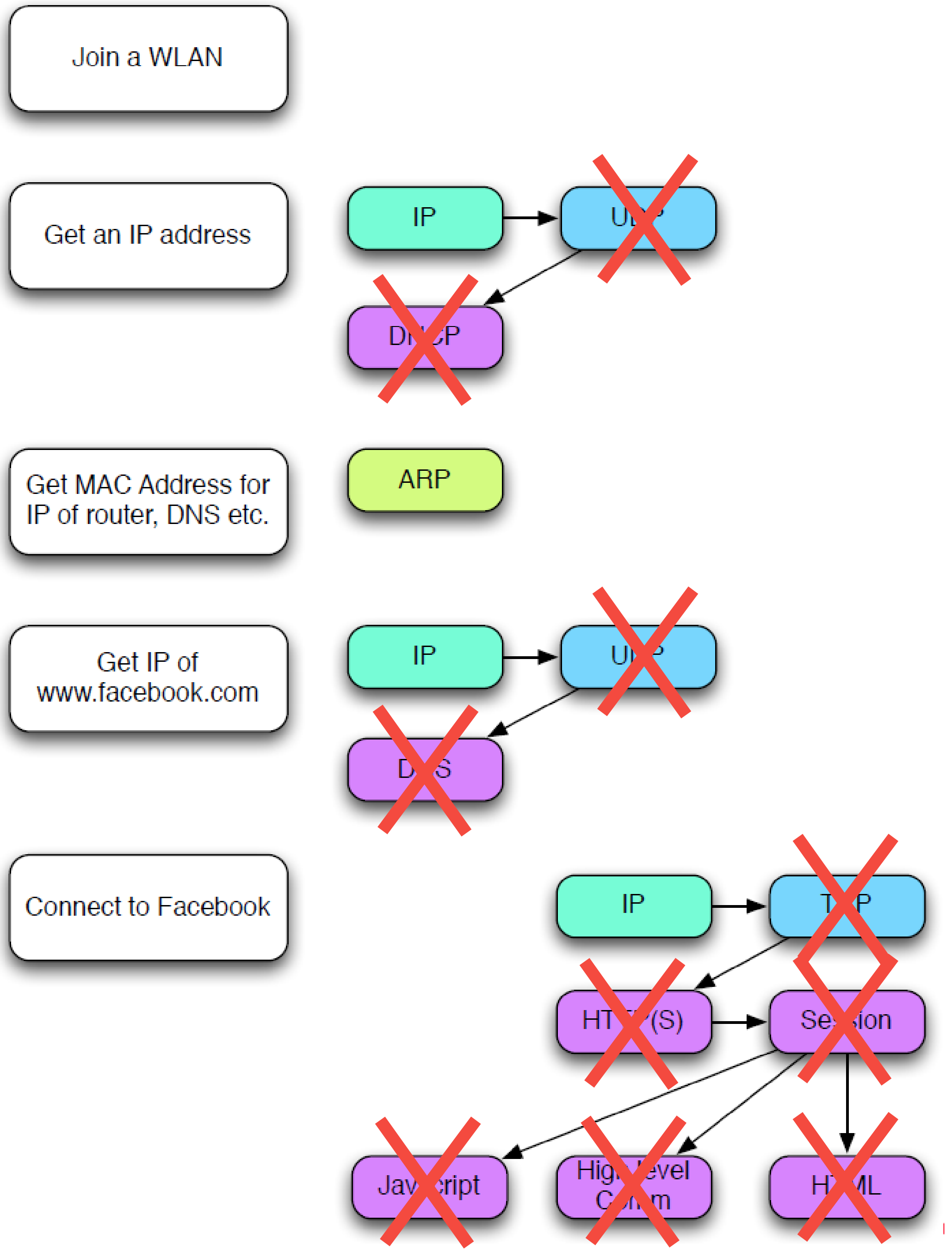
Link Layer

ARP/InARP · NDP · OSPF · Tunnels (L2TP) · PPP · Media Access Control (Ethernet, DSL, ISDN, FDDI) ·

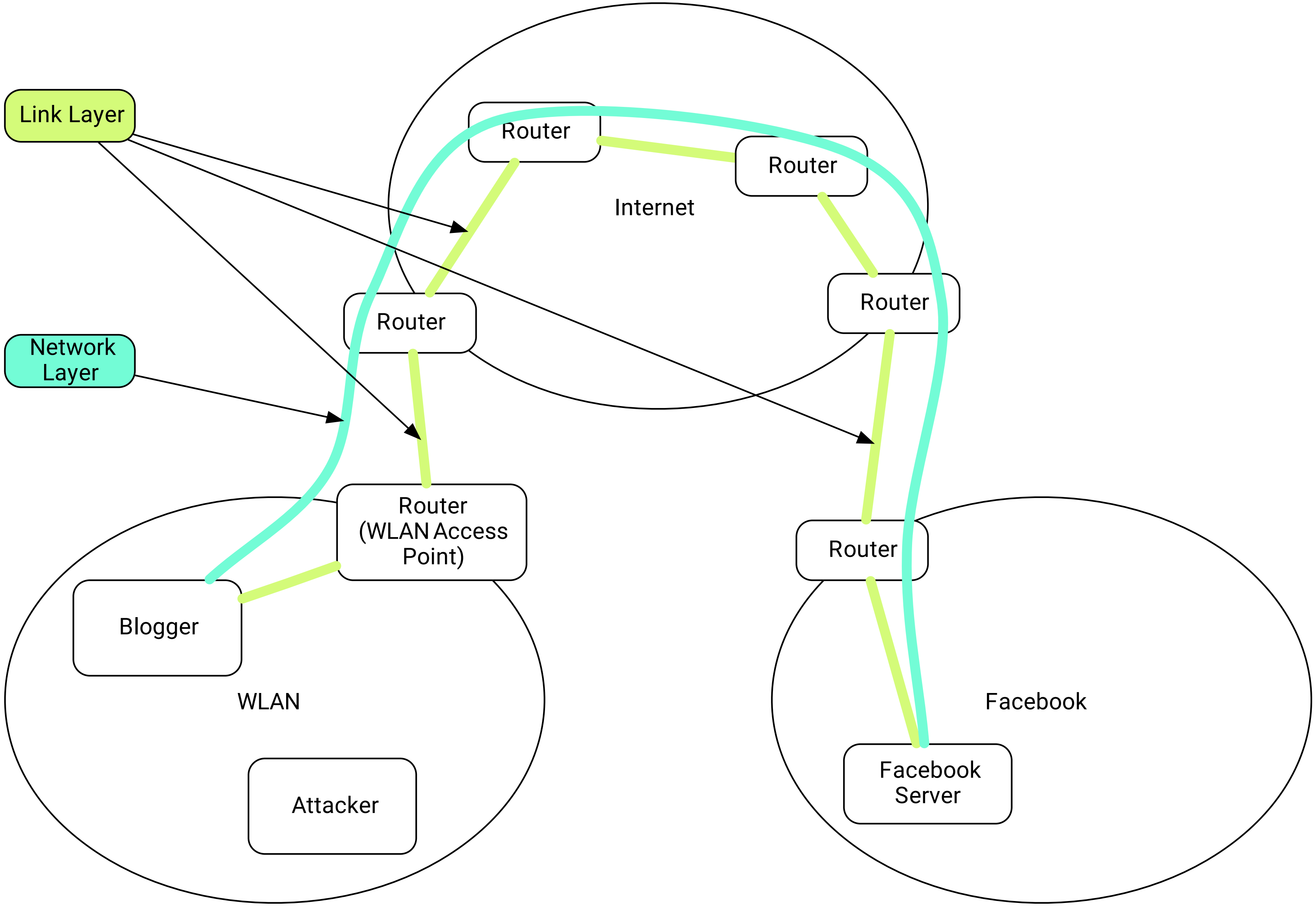
(more)

v · d · e

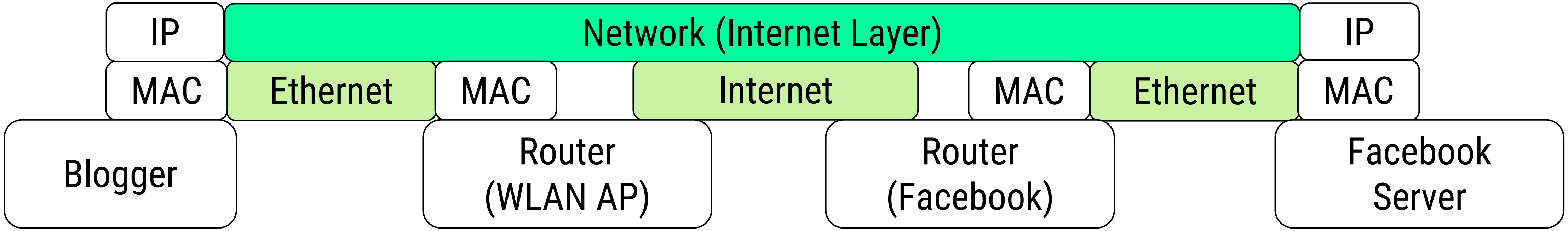




Routing the Blogger

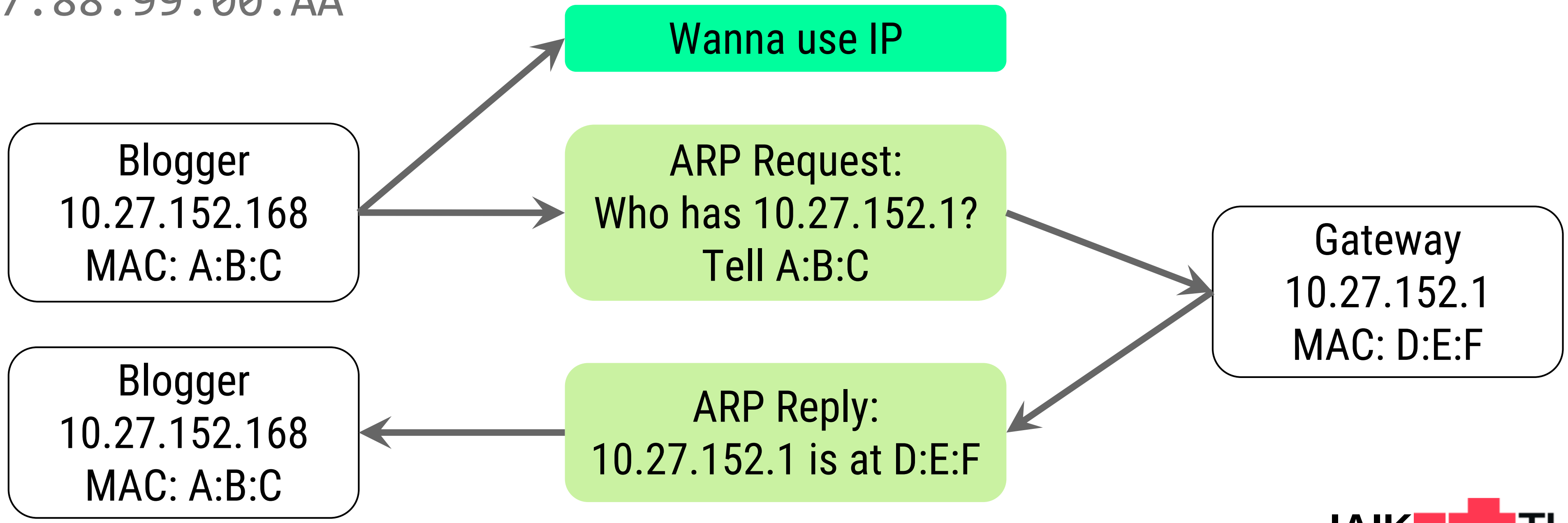


Routing the Blogger



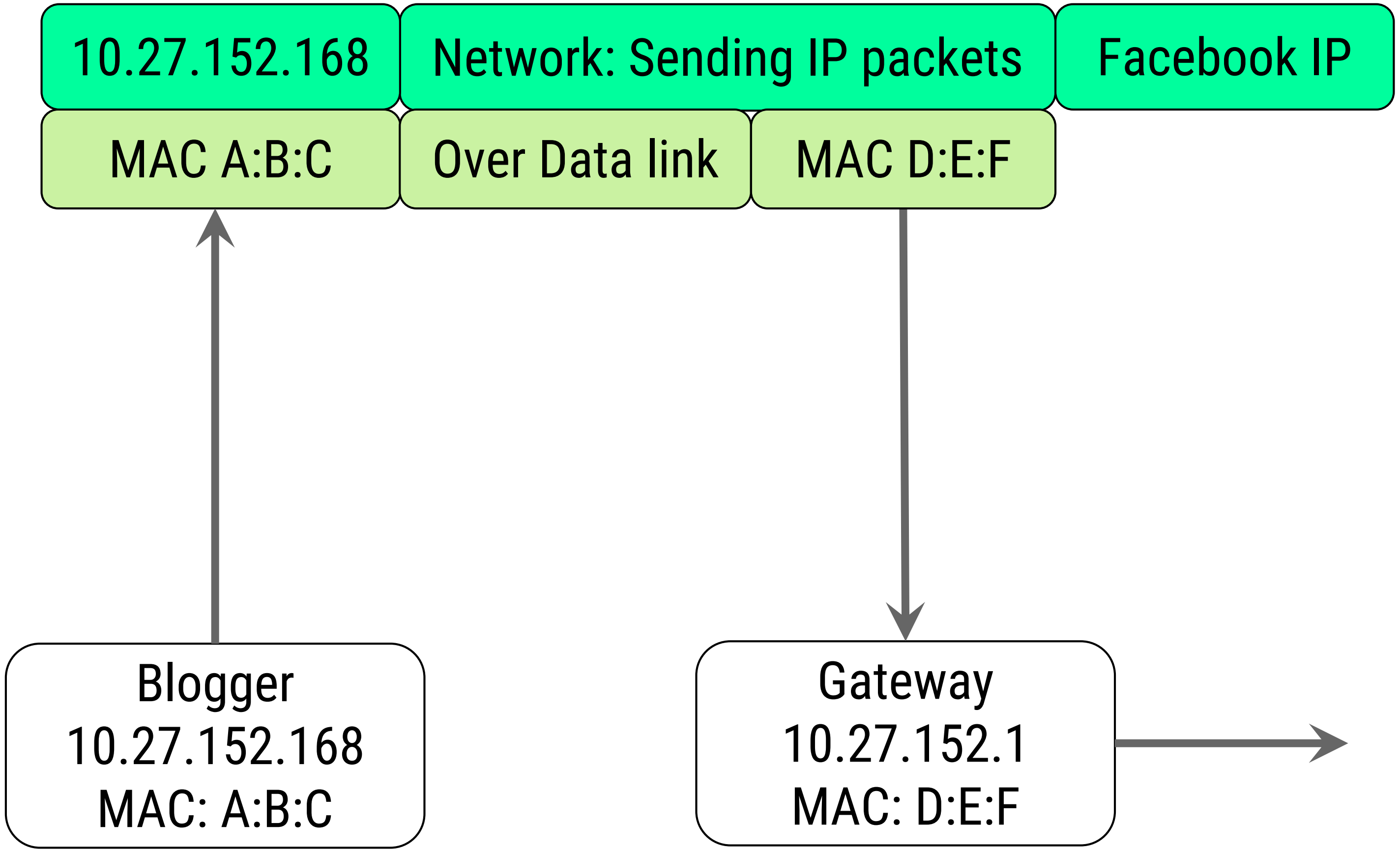
ARP Request on Gateway

1. ARP Request: Who has a.b.c.d
   → Tell to MAC `00:11:22:33:44:55`
2. ARP Reply: That‘s me. My MAC `66:77:88:99:00:AA`
3. ARP Cache: Stores IP to MAC mapping
   – „IP a.b.c.d is linked to MAC `66:77:88:99:00:AA`“



IAIK TU Graz

Contacting Facebook via 10.27.152.1



Wireshark

1	0.000000	Cisco_67:e4:9d	Spanning-tree-(for	STP	Conf. TC + Root = 32768/0/00:90:21:67:68:0a Cost = 4 Port
2	0.000007	174.36.30.8	10.27.152.168	HTTP	HTTP/1.1 200 OK (text/html)
3	0.000936	fe80::c62c:3ff:fe15:ae5a	ff02::1:ff15:ae5a	ICMPv6	Multicast listener report
4	0.004892	c4:2c:03:15:ae:5a	3Com_f6:ab:8e	ARP	Who has 10.27.152.1? Tell 10.27.152.168
5	0.004965	0.0.0.0	255.255.255.255	DHCP	DHCP Request - Transaction ID 0x34c277d5
ARP		Com_f6:ab:8e	c4:2c:03:15:ae:5a	ARP	10.27.152.1 is at 00:04:75:f6:ab:8e
		:2c:03:15:ae:5a	Broadcast	ARP	Gratuitous ARP for 10.27.152.168 (Request)
8	0.029129	10.27.152.5	10.27.152.168	DHCP	DHCP ACK - Transaction ID 0x34c277d5
9	0.040115	c4:2c:03:15:ae:5a	Broadcast	ARP	Who has 169.254.255.255? Tell 10.27.152.168
10	0.069249	10.27.152.168	224.0.0.2	IGMP	V2 Leave Group 224.0.0.251
11	0.069500	10.27.152.168	224.0.0.251	IGMP	V2 Membership Report / Join group 224.0.0.251
ARP		4:2c:03:15:ae:5a	Broadcast	ARP	Who has 10.27.152.1? Tell 10.27.152.168
		Com_f6:ab:8e	c4:2c:03:15:ae:5a	ARP	10.27.152.1 is at 00:04:75:f6:ab:8e
14	0.072125	10.27.152.168	129.27.142.23	DNS	Standard query PTR lb._dns-sd._udp.0.152.27.10.in-addr.arpa
15	0.072170	10.27.152.168	129.27.142.23	DNS	Standard query TXT cf._dns-sd._udp.0.152.27.10.in-addr.arpa
16	0.072217	10.27.152.168	129.27.142.23	DNS	Standard query PTR b._dns-sd._udp.0.8.16.172.in-addr.arpa
17	0.072262	10.27.152.168	129.27.142.23	DNS	Standard query PTR db._dns-sd._udp.0.8.16.172.in-addr.arpa
18	0.072308	10.27.152.168	129.27.142.23	DNS	Standard query PTR r._dns-sd._udp.0.8.16.172.in-addr.arpa
19	0.072353	10.27.152.168	129.27.142.23	DNS	Standard query PTR dr._dns-sd._udp.0.8.16.172.in-addr.arpa
20	0.072399	10.27.152.168	129.27.142.23	DNS	Standard query PTR lb._dns-sd._udp.0.8.16.172.in-addr.arpa
21	0.072444	10.27.152.168	129.27.142.23	DNS	Standard query TXT cf._dns-sd._udp.0.8.16.172.in-addr.arpa

Security?

By sending gratuitous ARP replies we can update ARP caches of other hosts…

But…

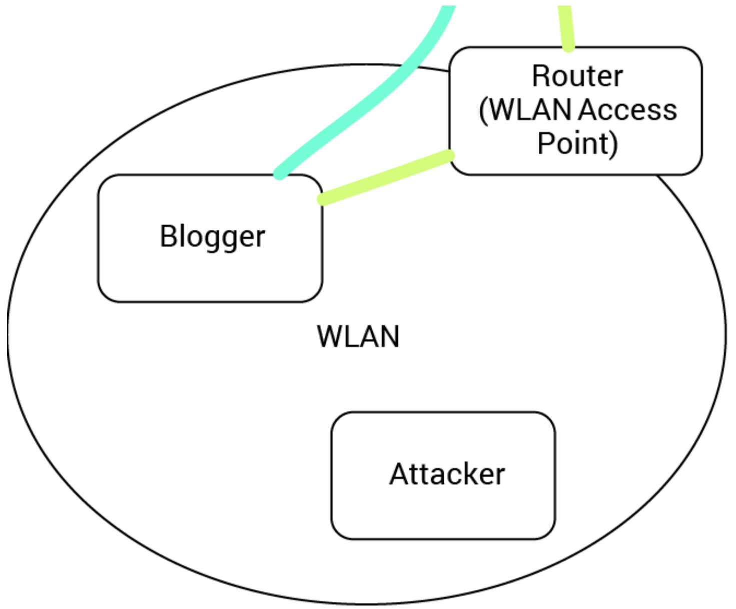
- ARP Request / Replies are not authenticated!
  - Anybody can send them for arbitrary IPs / MAC addresses
- Gratuitous ARP replies can update the ARP cache

→ ***Consequences?***

ARP Poisoning

Assumption

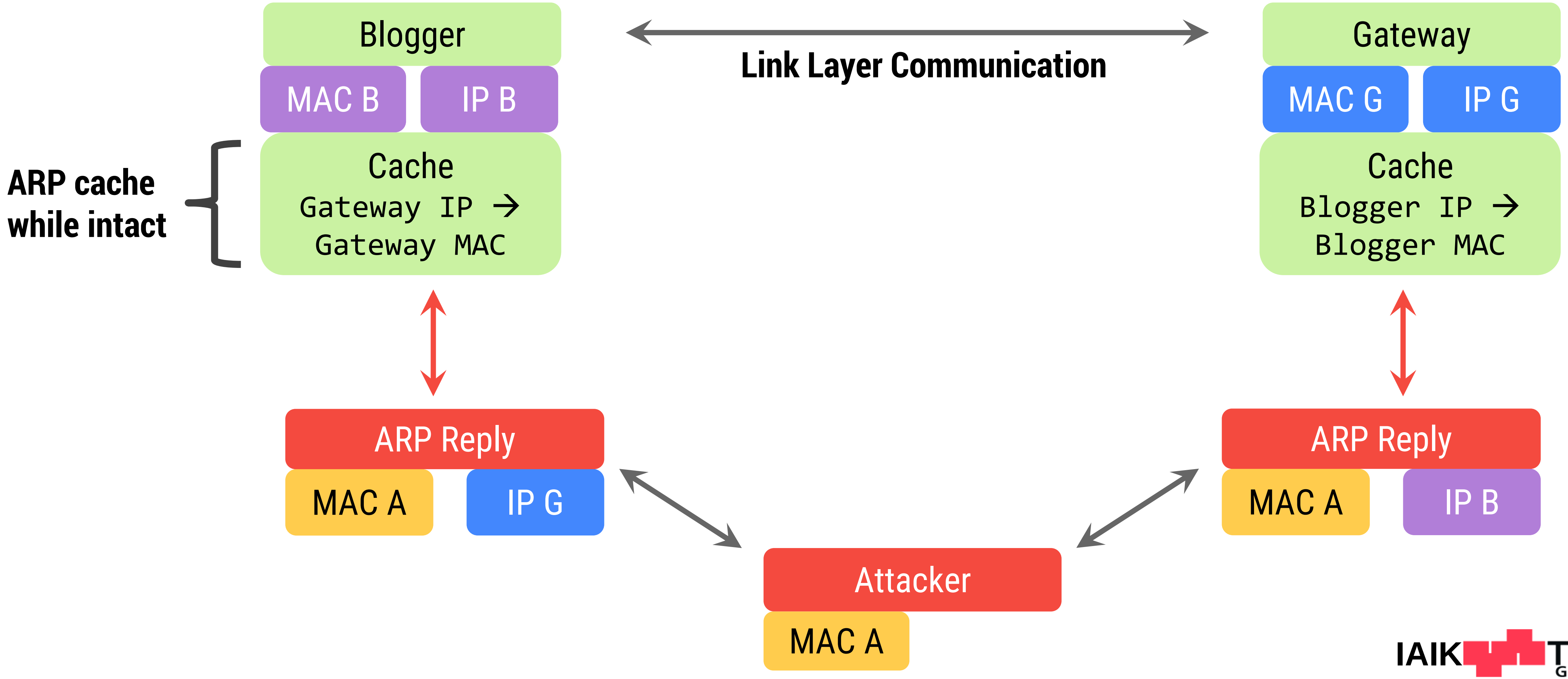
- Attacker is in same (W)LAN
- Blogger has MAC address of gateway (WLAN AP) in cache
- Attacker wants to modify traffic of blogger



→ *Attacker poisons ARP cache of blogger and gateway*

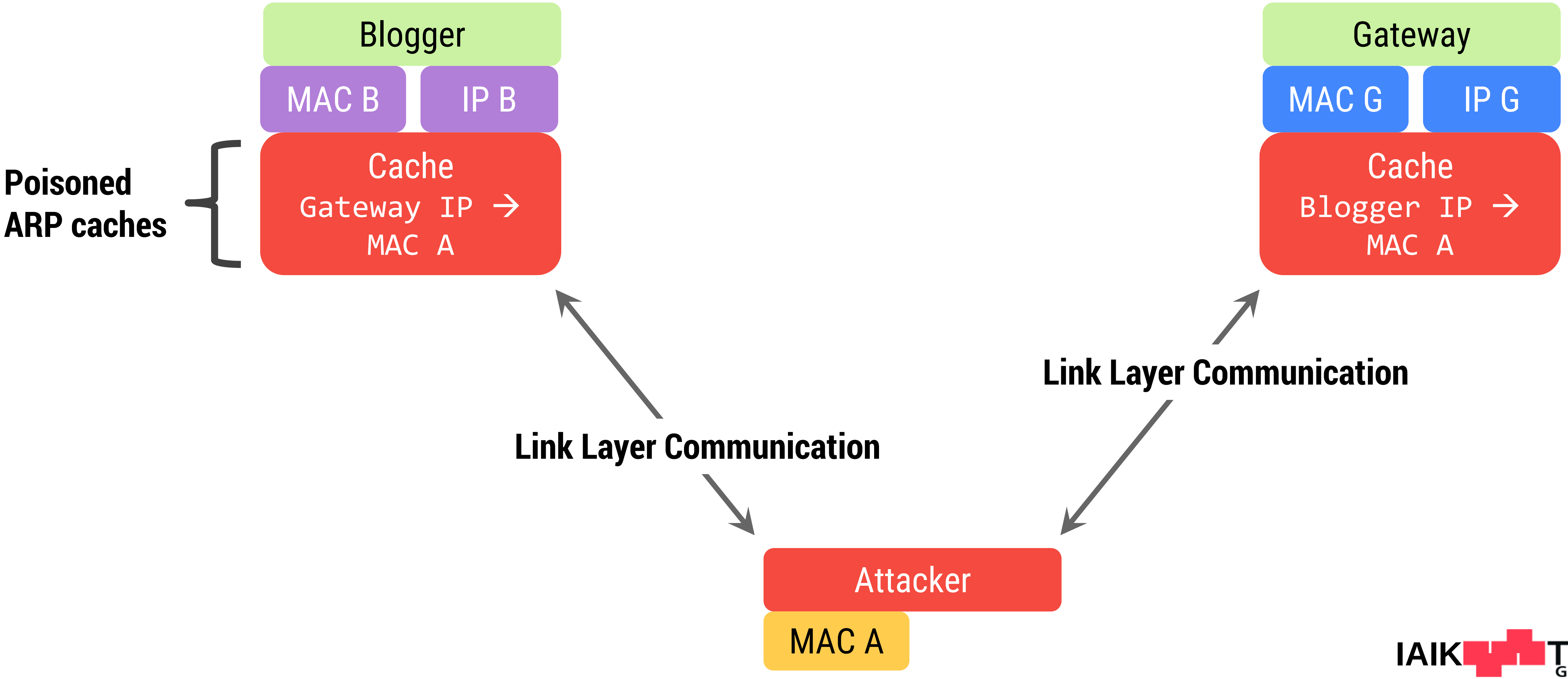
ARP Poisoning

Attacker poisons ARP cache with gratuitous ARP replies



ARP Poisoning

Communication with attacker due to poisoned ARP cache!



ARP Poisoning

Applicable in…

- Old LANs with hubs
  - One big collision domain due to the shared medium

- Switched networks
  - Frames only sent to destination host, others cannot see / modify traffic

  → with ARP Poisoning this becomes possible!

- WLANs
  - Especially in (untrusted) open WLANs

Protection Tips

- Always ensure you are using HTTPS
  - Raises complexity for attacks as ARP Spoofing alone is not enough
- Use VPN
  - Establishes encrypted tunnel
- Use trusted and protected WLANs
- Use firewall or intrusion detection system
  - Some of them remember their own IP <-> MAC mappings
  - Detect cache poisoning attack and identify attacker

Wireless Networks

IAIK TU Graz

Overview

Types

- Ad-hoc networks
- Infrastructure-based networks (access points)

Technologies

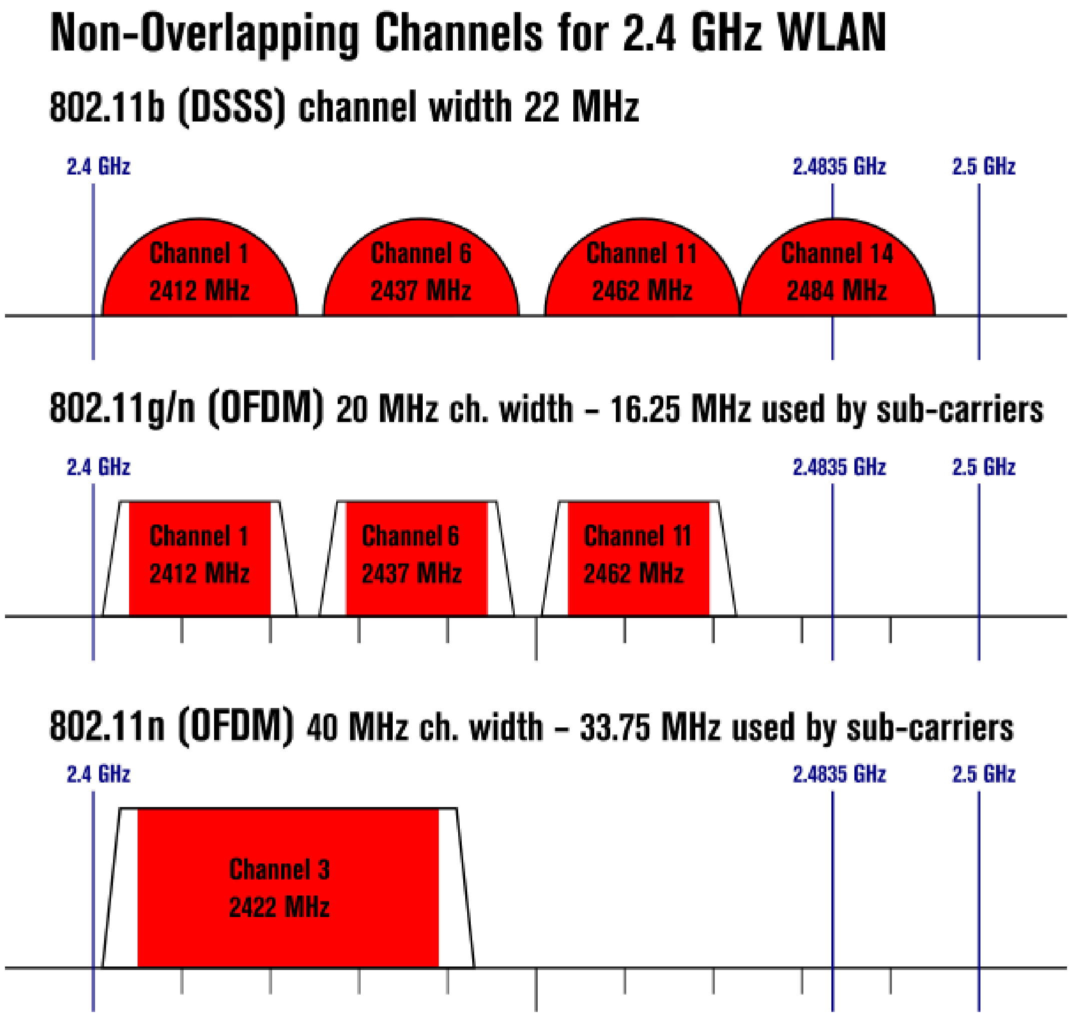
- **Wireless LANs (IEEE 802.11)**
- GSM/UMTS/LTE, ...
- Bluetooth
- ADS-B (Airplanes), AIS (Ships), Satellite Internet

See: http://goo.gl/cQfUY6 See: https://goo.gl/3tsgyO See: https://goo.gl/zolC0t

IAIK TU Graz

Wireless LAN – Evolution

- 1999: **802.11a:** 5 GHz, max. 54 Mbit/s
- 1999: **802.11b:** 2.4 GHz, max. 11 Mbit/s
- 2003: **802.11g:** 2.4 GHz, max. 54 Mbit/s
- 2009: **802.11n:** 2.4 GHz/5 GHz,
  54 Mbit/s to 600 Mbit/s
- 2013: **802.11ac:** 5 GHz, max. 1300 Mbit/s
  via multiple 80 MHz channels



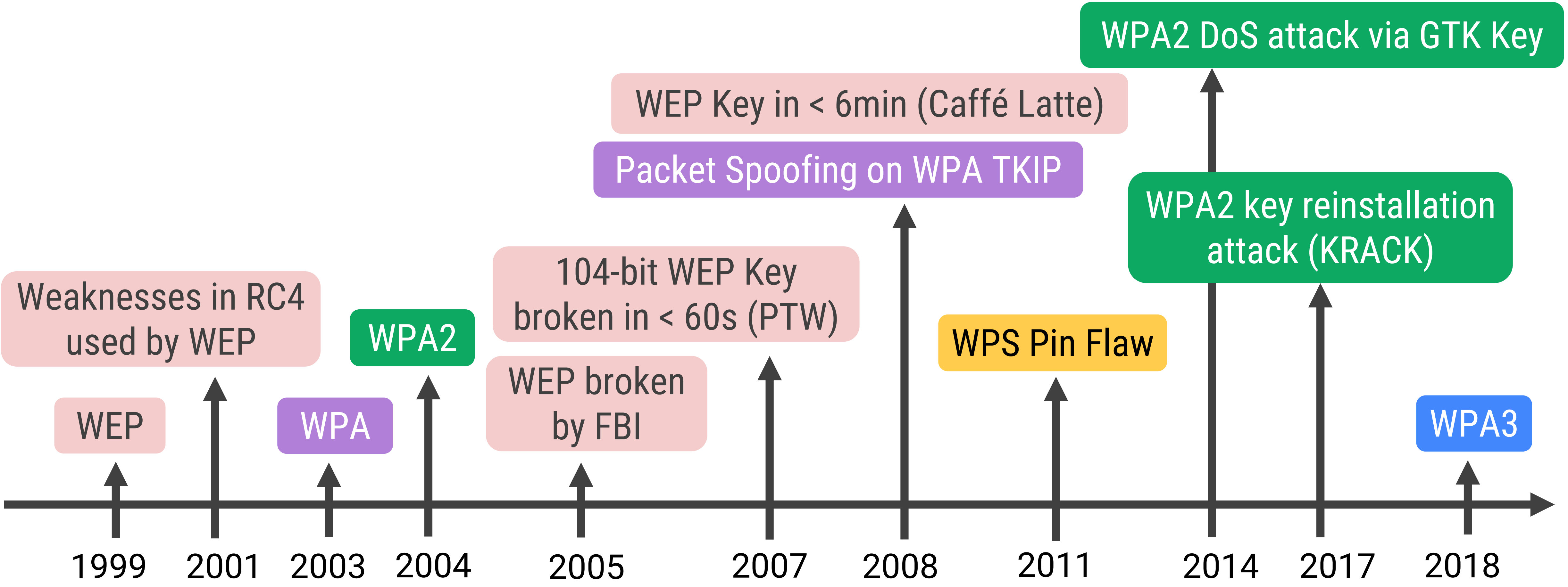
Source: https://goo.gl/Zt1G59

- 2016: **802.11ah:** 900 MHz, „WiFi HaLow“ → Internet of Things
- 2018: **802.11ax:** 1-5 GHz, improved multi-user MIMO, OFDMA for spectrum segregation

Wireless LAN – Security

Many security issues ever since!

Man-in-the-middle (MITM), Denial of Service (DoS), Injection, Spoofing, …



IAIK TU Graz

Attacking WLANs in Practice

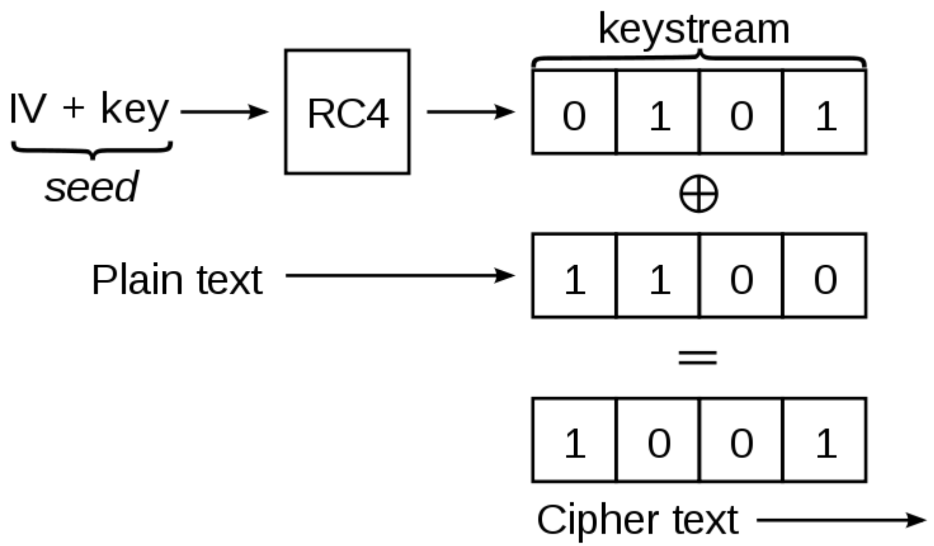
Which of these networks could be vulnerable?

SSID	MAC Address	Security	WPS	Manufacturer
3HuiGate_74F5	C8:51:95:74:F6:14	WPA2-PSK	Yes	Huawei Ltd.
3WebCube207C	D4:40:F0:1B:20:80	WPA2-PSK	No	Huawei Ltd.
A1-6842DG	A4:B1:E9:68:43:DF	WPA-PSK + WPA2-PSK	Yes	Technicolor
NETGEAR_11g	E4:F4:C6:F8:9B:70	WPA2-PSK	No	Netgear
PBS-536755	38:22:9D:53:67:5A	WPA-PSK	No	ADB Broadband Italia
TMOBILE-53141	C8:51:95:9D:A6:76	WPA-PSK + WPA2-PSK	No	Huawei Ltd.
UPC1381027	8C:04:FF:E4:C4:10	WPA-PSK + WPA2-PSK	No	Technicolor USA Inc.
UPC Wi-Free	FE:94:E3:25:4E:38	WPA2-EAP	No	
WLAN_E9	00:1A:2B:01:AC:A0	WEP	No	Ayecom Technology Co.

See: https://goo.gl/kGYaYv

WEP

„Wired Equivalent Privacy“



Source: https://goo.gl/hhQEdm

2 Variants

1. 40-bit key + 24-bit IV → 64-bit RC4 Key
   - 10 hex chars (0-9, A-F) or 5 ASCII chars (0-9, a-z, A-Z)
2. 104-bit key + 24-bit IV → 128-bit RC4 key
   - 26 hex chars or 13 ASCII chars

Attack Idea

- Look for many packets with „weak IVs“ that reveal information about WEP key
- Enough weak IVs found? → Crack WEP key
- Weak IV is key dependent → Takes different amount of time per key

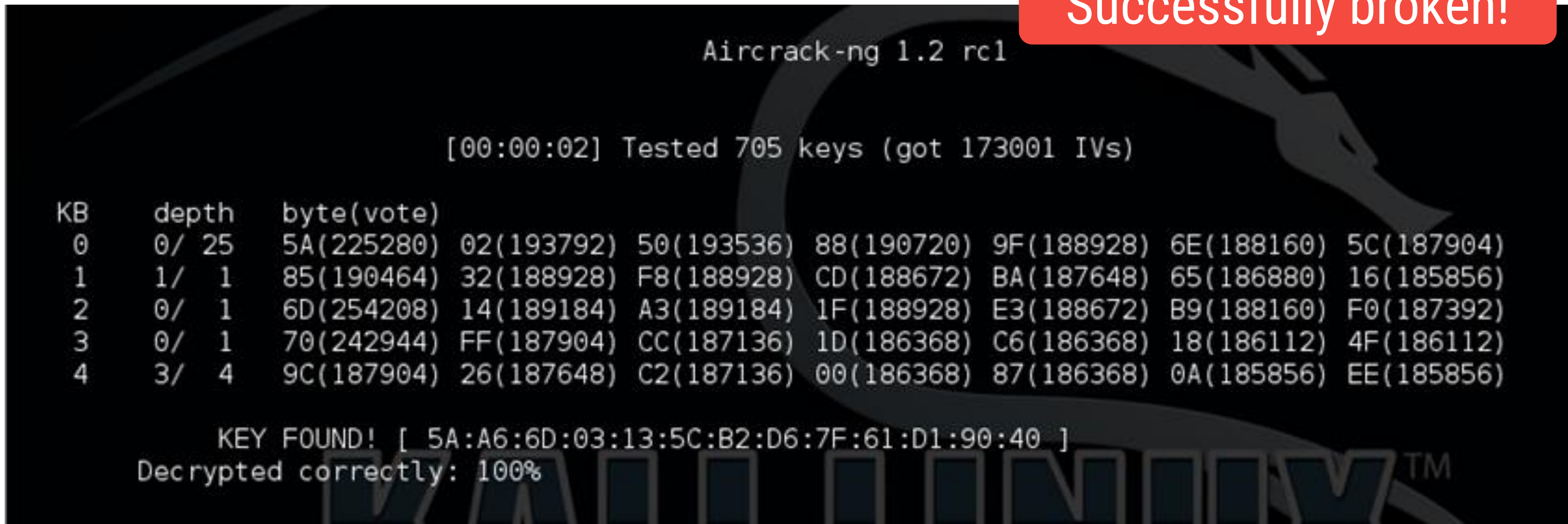
Attacking WEP

Ways to get the key…

- Active attack (traffic generation)
  - Replay attack
    - Stimulate network to send encrypted data packets
  - ARP Replay
    - Send ARP requests, listen for ARP responses

- Passive attack
  - Wait, wait, wait and just capture traffic…
  - Advantage: Undetectable!

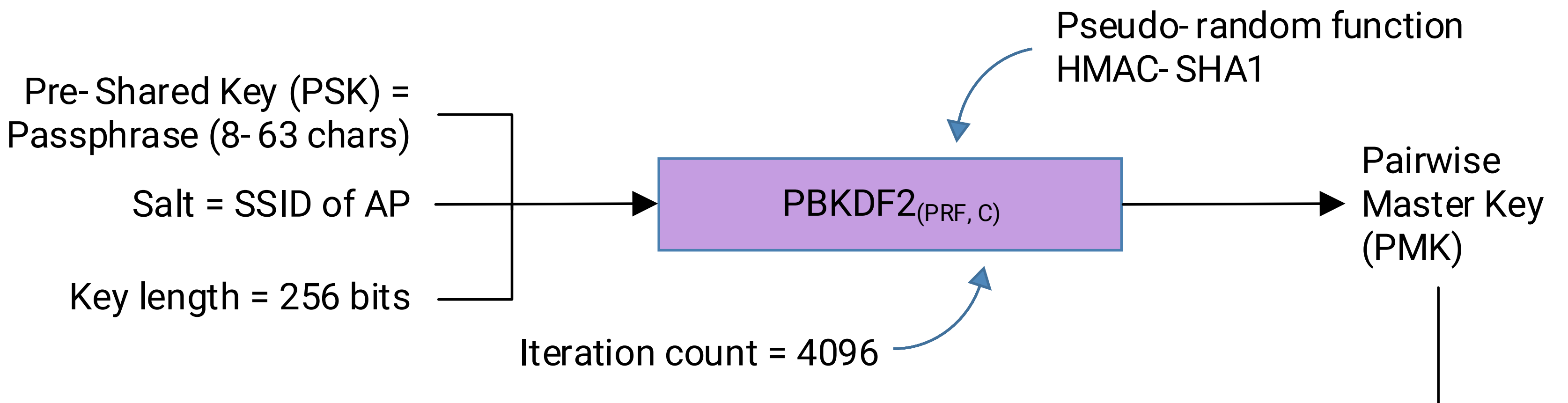
Successfully broken!



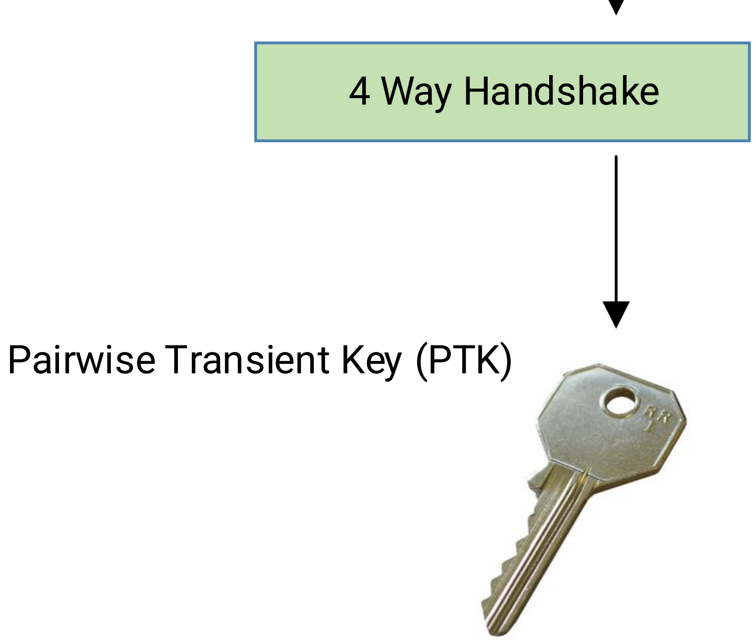
Source: https://goo.gl/0SmCZz

WPA

„Wi-Fi Protected Access“



- Same architecture as WEP but dynamic keys
  - Temporary Key Integrity Protocol (TKIP)
  - Generates new 128-bit key for each packet



- Message Integrity Check (MIC)
  - Prevent attacker from modifying and resending packets
- Two operation modes
  - Personal: Pre-Shared Key (PSK)
  - Enterprise: 802.1x + RADIUS Authentication Server

WPA 2

„Wi-Fi Protected Access II“

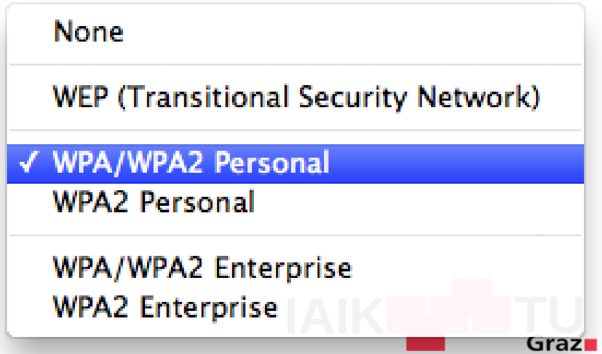
- Replacement of WPA
  - „Long Term Solution“ (802.11i)

- Based on AES-256 (block cipher) instead of RC4 (stream cipher)
  - Counter Mode CBC-MAC (CCMP) protocol instead of TKIP for encryption

Attack Idea

Security depends on quality of used passphrase (PSK)

→ Crack PSK since attacking strong encryption is pointless…



Attacking WPA / WPA 2

Ways to get the key…

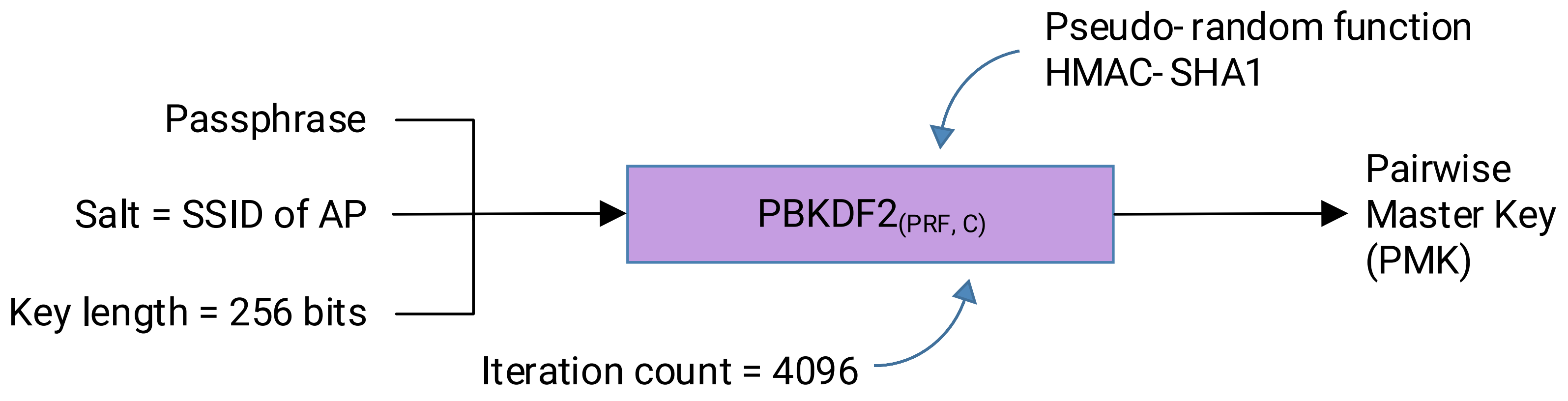
Only known way so far: Brute Force!

Requirements

- Captured handshake
- Passphase to test: Can be provided from a dictionary or generated on-the-fly
- SSID of AP: Serves as IV for PBKDF2

→ *Is this practically feasible?*

Well… it requires high computational resources!



IAIK TU Graz

Attacking WPA / WPA 2

Wikipedia, list of common passwords, …

Fasten up the brute force attack…

- Use pre-calculated rainbow tables
  - List of PMK for one specific SSID and a dictionary of passphrases
  - Most popular SSIDs

linksys	MSHOME	orange
Default	home	USR8054
NETGEAR	hpsetup	101
Wireless	smc	tmobile
WLAN	tsunami	SpeedStream
Belkin54g	ACTIONTEC	…

See: https://goo.gl/P7zSoJ

- Use GPU acceleration
  - Nvidia CUDA or OpenCL → E.g. Pyrit or Elcomsoft EWSA
  - In the cloud, e.g. Amazon EC2

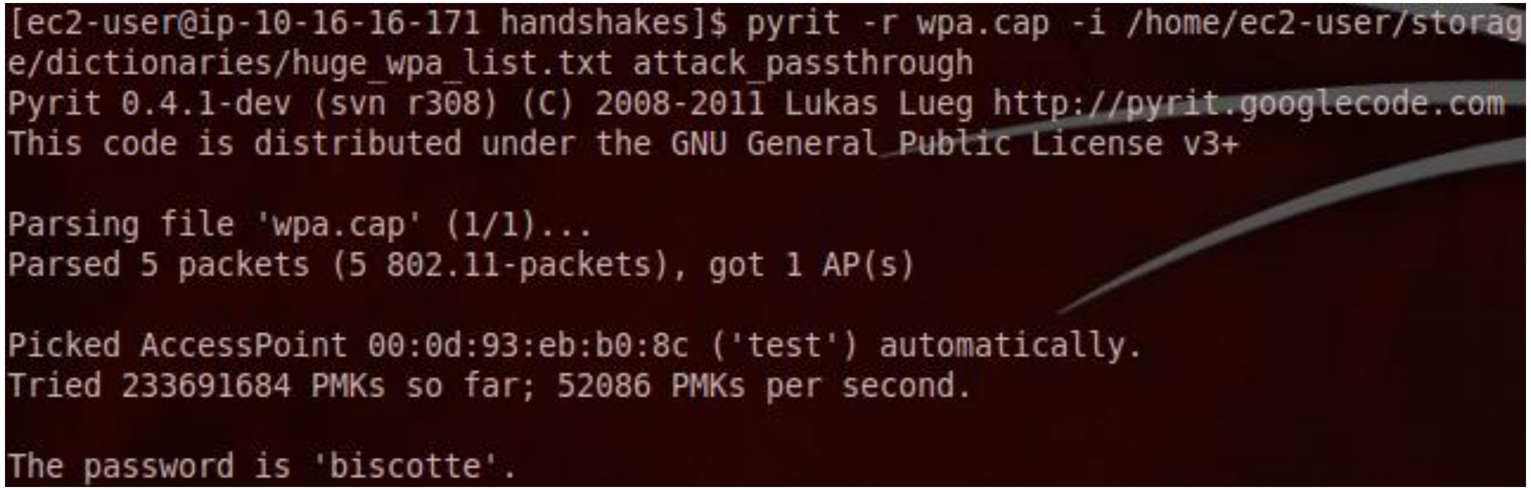
See: https://goo.gl/WEr7uI

See: http://goo.gl/mBzBw4

Attacking WPA / WPA 2

Result?

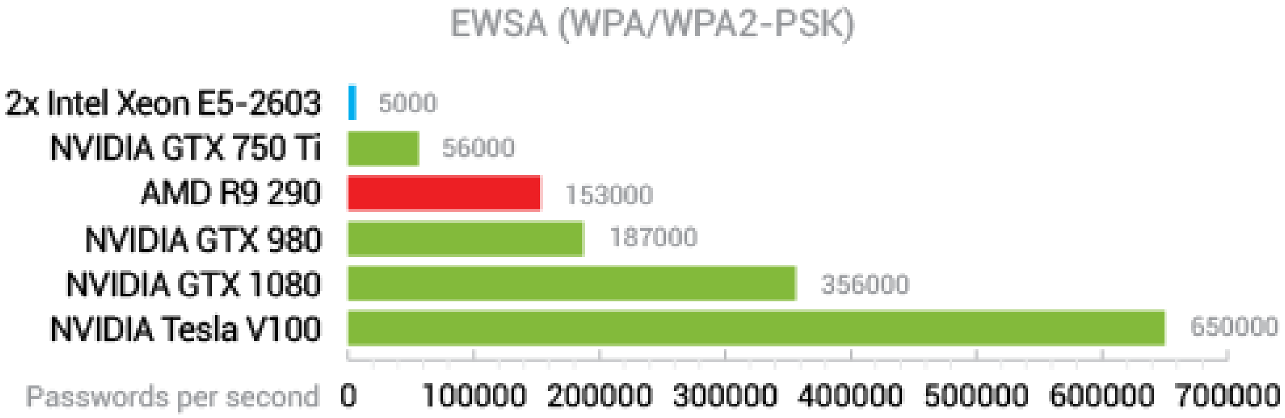
Brute-force always works :-)



Source: https://goo.gl/DV4sBc

but...

- What if passphrase is not in used dictionary?
- What if passphrase is simply too complex or long?



Source: http://goo.gl/ZIl8nm

IAIK TU Graz

WPS

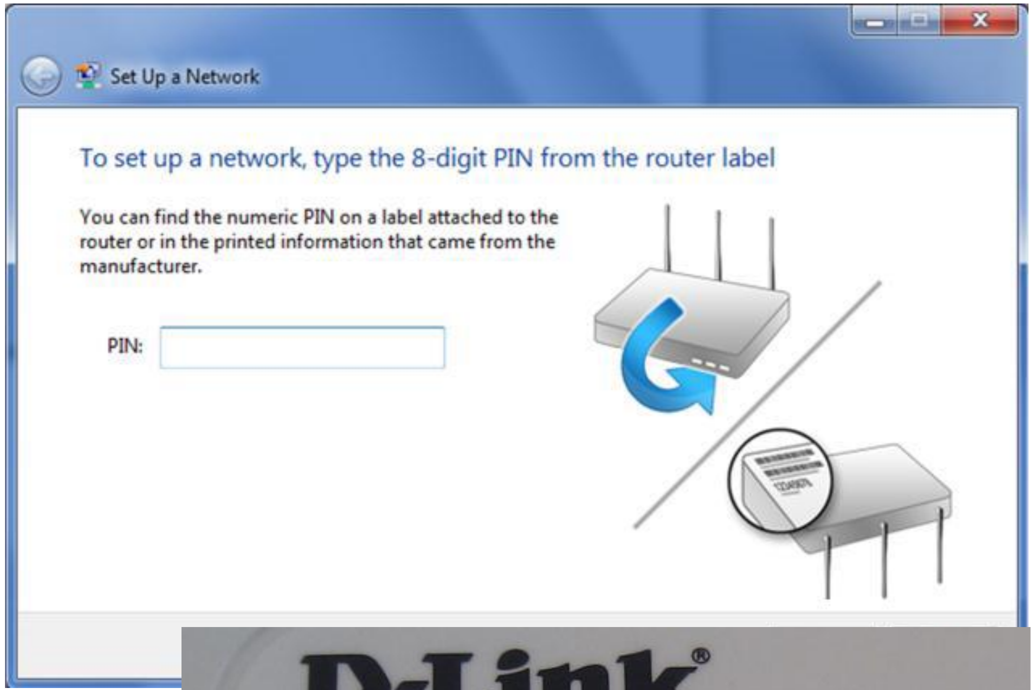
„Wi-Fi Protected Setup“

- 8 digits PIN as alternative to passphrases
- Emphasis on usability
  - Fixed on sticker or dynamically generated

Problem

- Last digit is checksum → only
  10^7 = 10.000.000 possible PINs
- PIN is validated in two separate halves:
  1st half 10.000, 2nd 1.000 possibilities

→ Need max. 11.000 guesses until PIN recovered
(< 4 hours)



Source: https://goo.gl/zPrIEf

Attacking WPS

Result?

See: https://goo.gl/oJrwaQ

```
reaver -i mon0 -b A4:B1:E9:68:43:DF -c 6
[+] Associated with A4:B1:E9:68:43:DF (ESSID: A1-6842DG)
[+] Trying pin 12345670
[+] Trying pin 95946344
[+] Trying pin …
[+] 92.25% complete @ 2016-04-09 17:31:55 (3 seconds/attempt)
[+] Trying pin 19196343
[+] Key cracked in 4942 seconds
[+] WPS PIN: ‘19196343’
[+] WPA PSK: ‘rkndemo’
[+] AP SSID: ‘A1-6842DG’
```

Note: Only works if WPS enabled on AP and amount of tries not limited!

Attacking WLANs

Q: What if AP has WPS disabled, uses WPA 2 and strong passphrase?

A: One option left to consider: *WPA (2) Password-generating algorithms*

Problem

- Many vendors use (known) algorithms to generate a default password which is then attached to sticker

- If password not changed by users → attackers may simply calculate it!

For case studies, see https://goo.gl/SIh07u and https://goo.gl/9uSvne

Source: https://goo.gl/gthik6

Attacking WLANs

Q: How to identify the vendor?

A: MAC Address!

WIRESHARK

OUI Lookup Tool

The Wireshark OUI lookup tool provides an easy way to look up OUIs Wireshark manufacturer database, which is a list of OUIs and MAC ad

Examples:

0000.0c

08:00:20

01-00-0C-CC-CC-CC

missouri

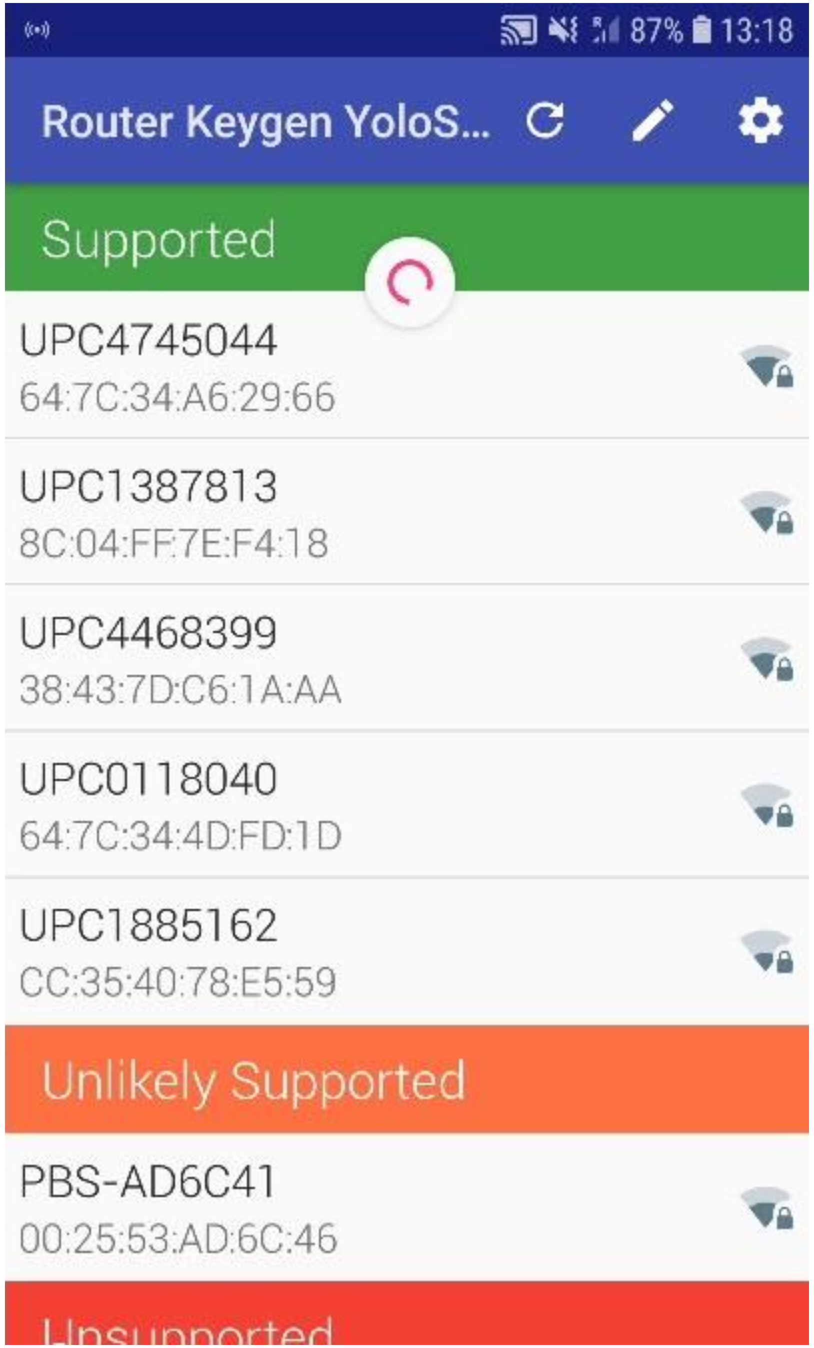
OUI search

8C-04-FF-7E-F4-18

Find

Results

8C:04:FF Technico Technicolor CH USA Inc.

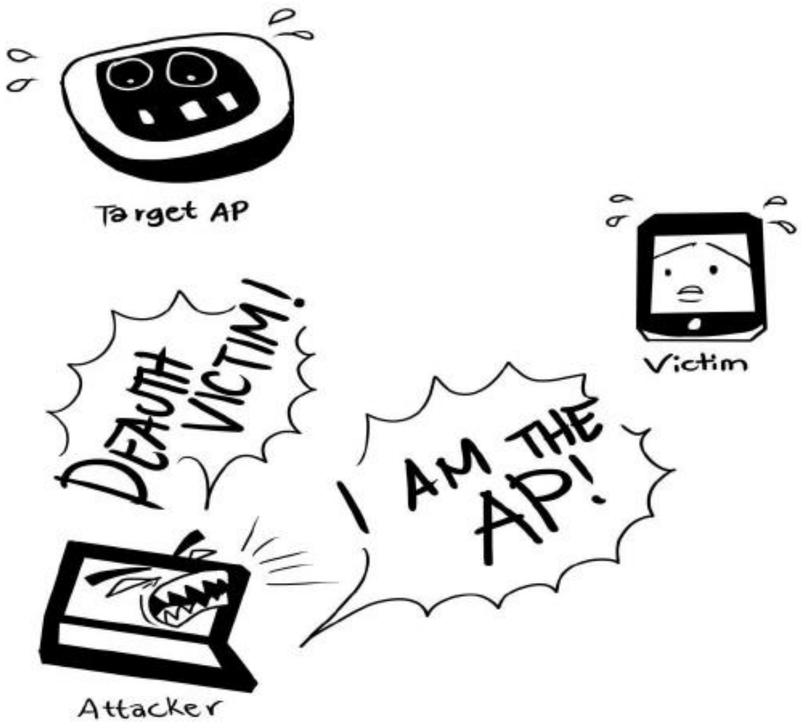


IAIK TU Graz

Evil Twin Attack

How does it work?

- Perform DoS attack to deauthenticate client from WiFi
- Create fake AP with same ESSID and encryption
  - Easy if AP is open: `Freewave`, `Airport-Frankfurt`, `TUGRAZguest`, `freeGRAZwifi`, …
  - Otherwise, you need to know the password
- Client connects:
  - MITM attack
  - Phishing (show fake captive portal, OAuth login, …)



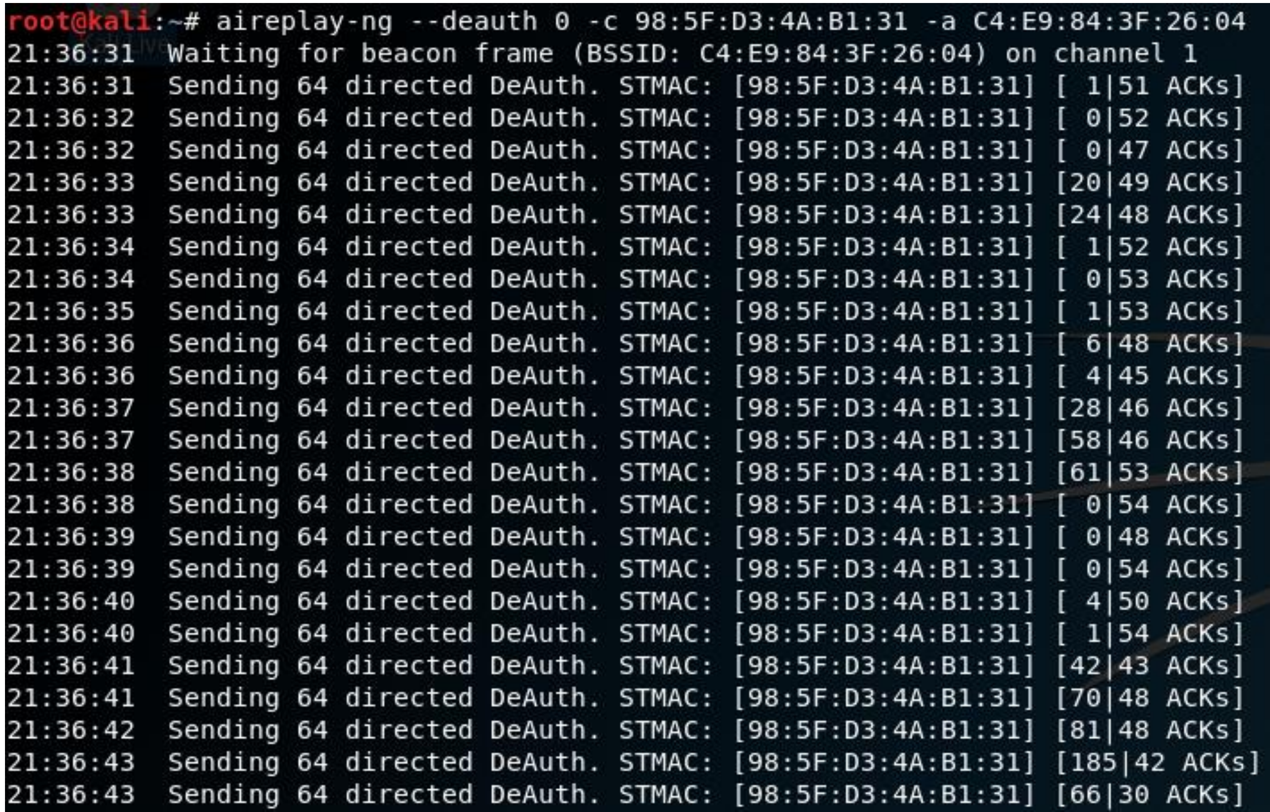
Why does it work?

- Clients choose APs with best signal
- Auto-Connect feature → Usability vs security
  - Enabled by default in Windows, Ubuntu, macOS, …

Evil Twin Attack

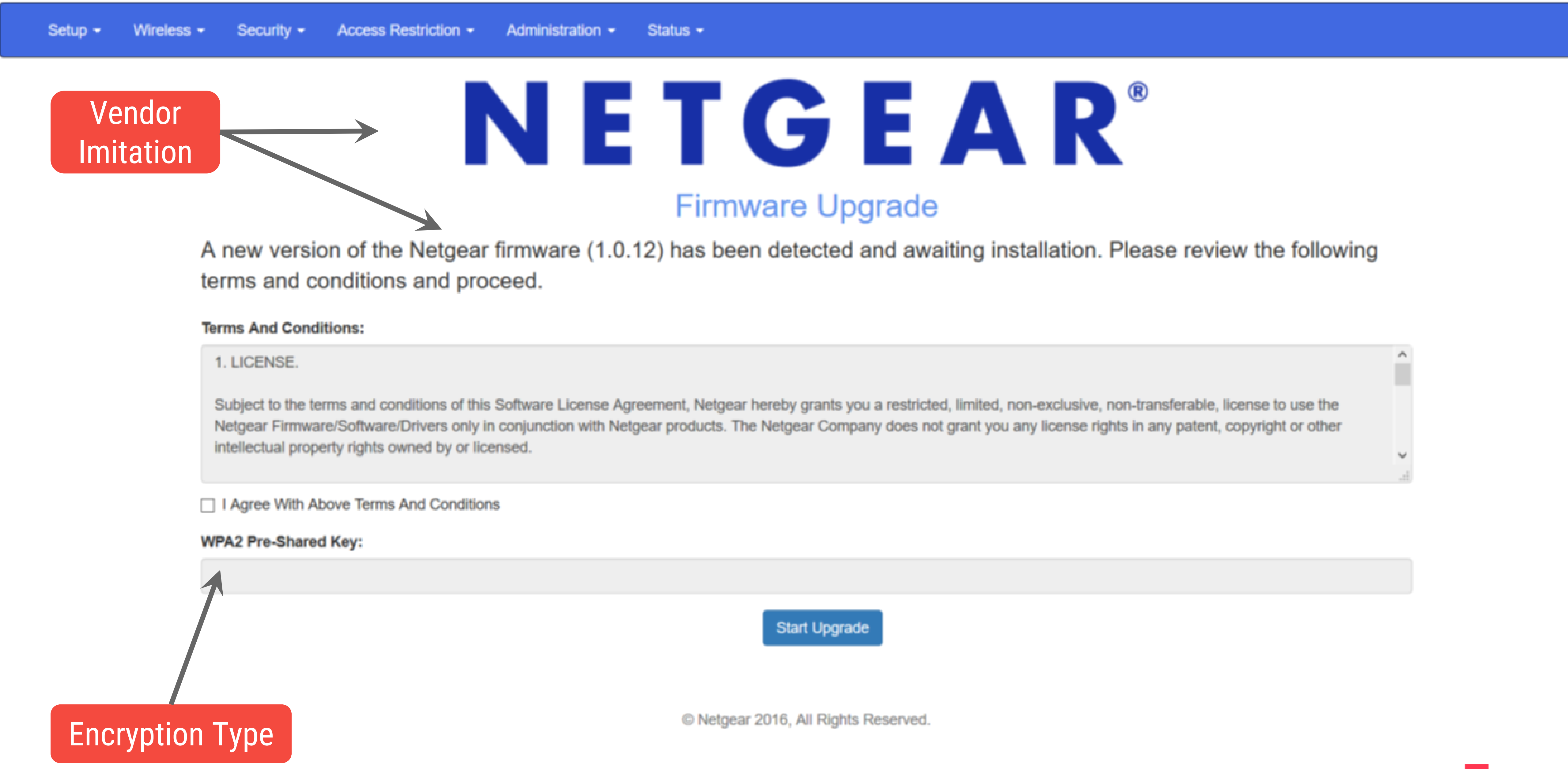
Deauthentication Attack

- Leverage DEAUTH frame (transmitted unencrypted)
  - Sent when all communication is terminated
  - Kick out client from WiFi by forging DEAUTH frames
    - 1 from AP to client
    - 1 from client to AP
    - 1 from AP to broadcast address



Probe Response Flooding

- Keep answering to probe request frames
  - „Sorry, your PSK is incorrect“
- Victim will stay disconected
  - Client will mostly connect manually
  - Also useful for downgrade attacks


Setup
Wireless
Security
Access Restriction
Administration
Status
Vendor Imitation
NETGEAR®
Firmware Upgrade
A new version of the Netgear firmware (1.0.12) has been detected and awaiting installation. Please review the following terms and conditions and proceed.
Terms And Conditions:
1. LICENSE.
Subject to the terms and conditions of this Software License Agreement, Netgear hereby grants you a restricted, limited, non-exclusive, non-transferable, license to use the Netgear Firmware/Software/Drivers only in conjunction with Netgear products. The Netgear Company does not grant you any license rights in any patent, copyright or other intellectual property rights owned by or licensed.
I Agree With Above Terms And Conditions
WPA2 Pre-Shared Key:
Start Upgrade
Encryption Type
© Netgear 2016, All Rights Reserved.

Protection Tips

- Use a manually set, long and really complex passphrase
  - Raises complexity for successful brute force attack
- Change SSID to your own choice
  - Hinders attacks with pre-calculated rainbow tables
- Use WPA 2 CCMP only
  - Drops some deficiencies of TKIP and RC4 in WPA
- Disable WPS in your AP settings

Outlook

- 10.01.2020
  - HTTP Sessions
  - Same Origin Policy, CSP, Client-side Attacks
- 17.01.2020
  - TLS Handshake
  - TLS Security Features


HAPPY HOLIDAYS
&
A SECURE 2020!