

# Connected Factories lecture 5/6: Security

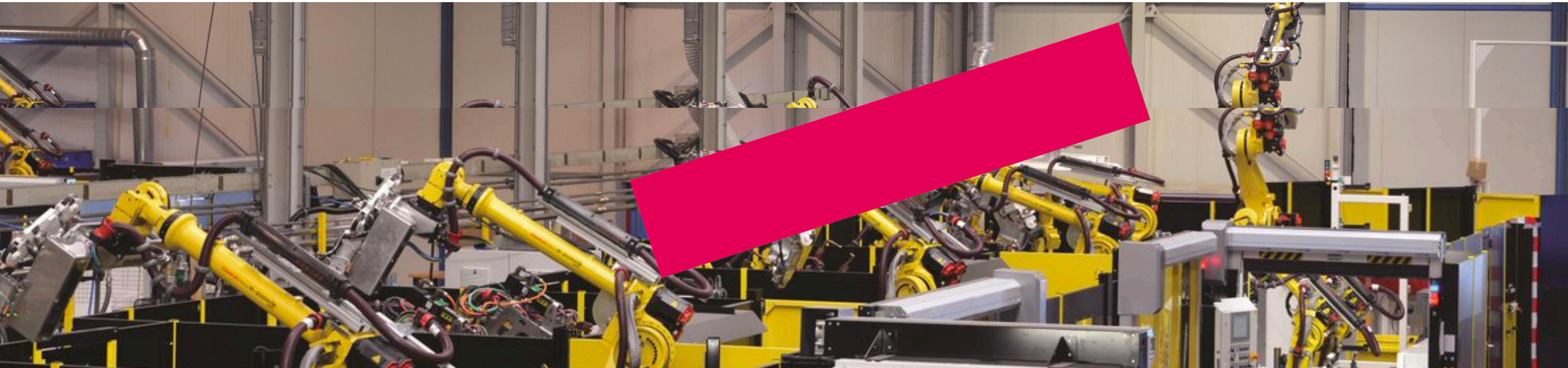


Image: AWL-Techniek

Associate Degree Smart Industry  
Faculty of Engineering and Automotive

johan.korten@han.nl

V1.0 April 8, 2020

# Schedule

	<b>Theme</b>	
Lecture 1	Introduction	
Lecture 2	Network connections	
Lecture 3	Network protocols	
Lecture 4	Interconnections	
Lecture 5	<b>Today: Security</b>	
Lecture 6	Safety	
Assessment		

# Security by design

Security (and safety) should be considered at design time already.

Patching leaks as we go (like the Microsoft approach) has proven to be a road to disaster.

How to enter a ‘secured’ area and wreak havoc:



# Common IoT Security Issues: TEDx by Ken Munro

<https://youtu.be/pGtnC1jKpMg>



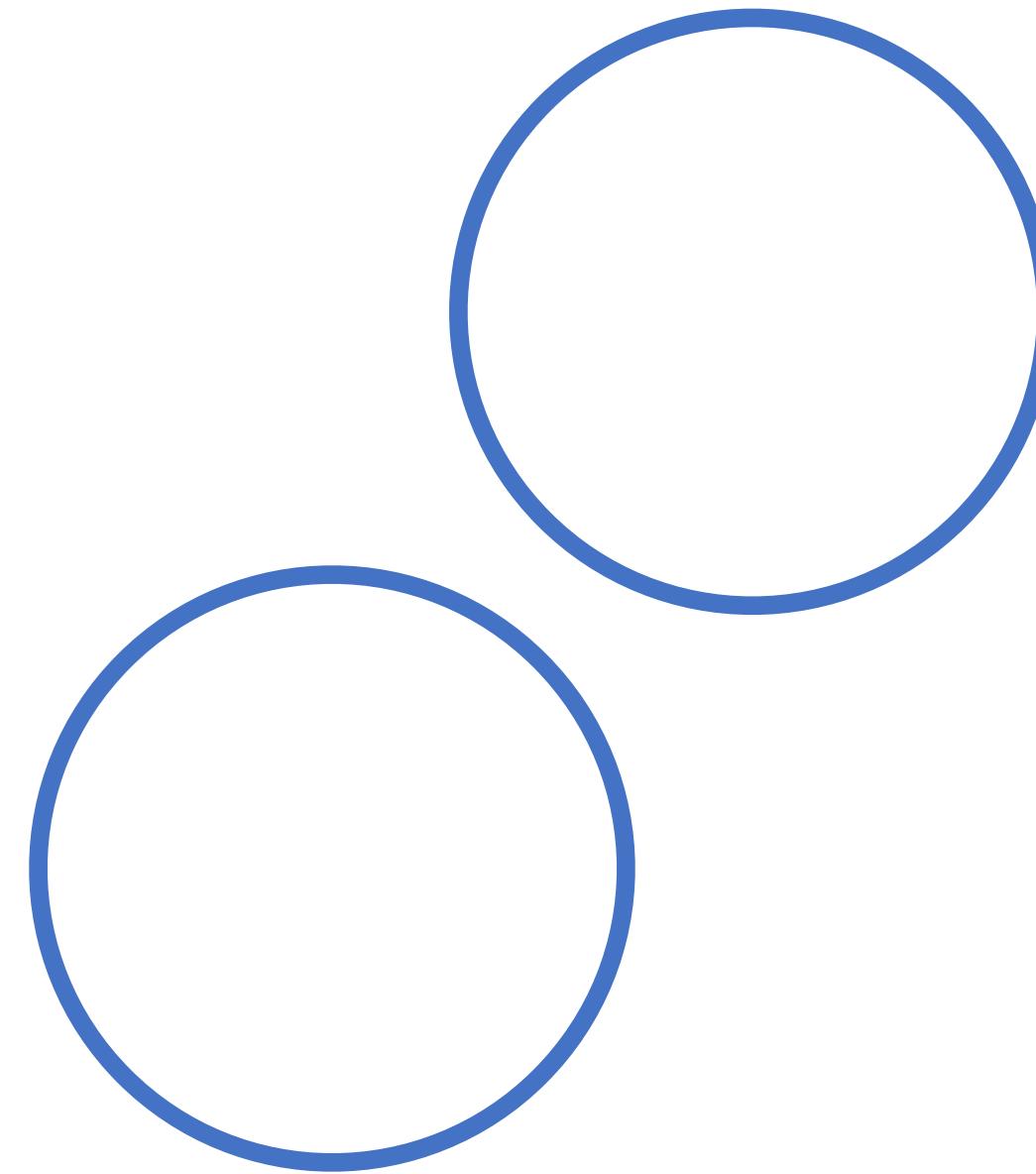
# Common Networking Tools 1/2

Tool name	Usage	Common Flag(s) / usage
netstat	Shows incoming and outgoing TCP connections	-r (ip route), -i (interfaces)
lsof	list open files	-i -n -P   grep programname (e.g. “lsof -i -n -P   grep mosquitto”)
tcpdump	(TCP) packet analyser	
ping	Sends echo request to specified host	e.g. “ping www.han.nl”
traceroute	Shows route of a packet	-i (e.g. “traceroute -i google.com”)
nmap	Network exploration and security scanner tool	

# Common Networking Tools 2/2

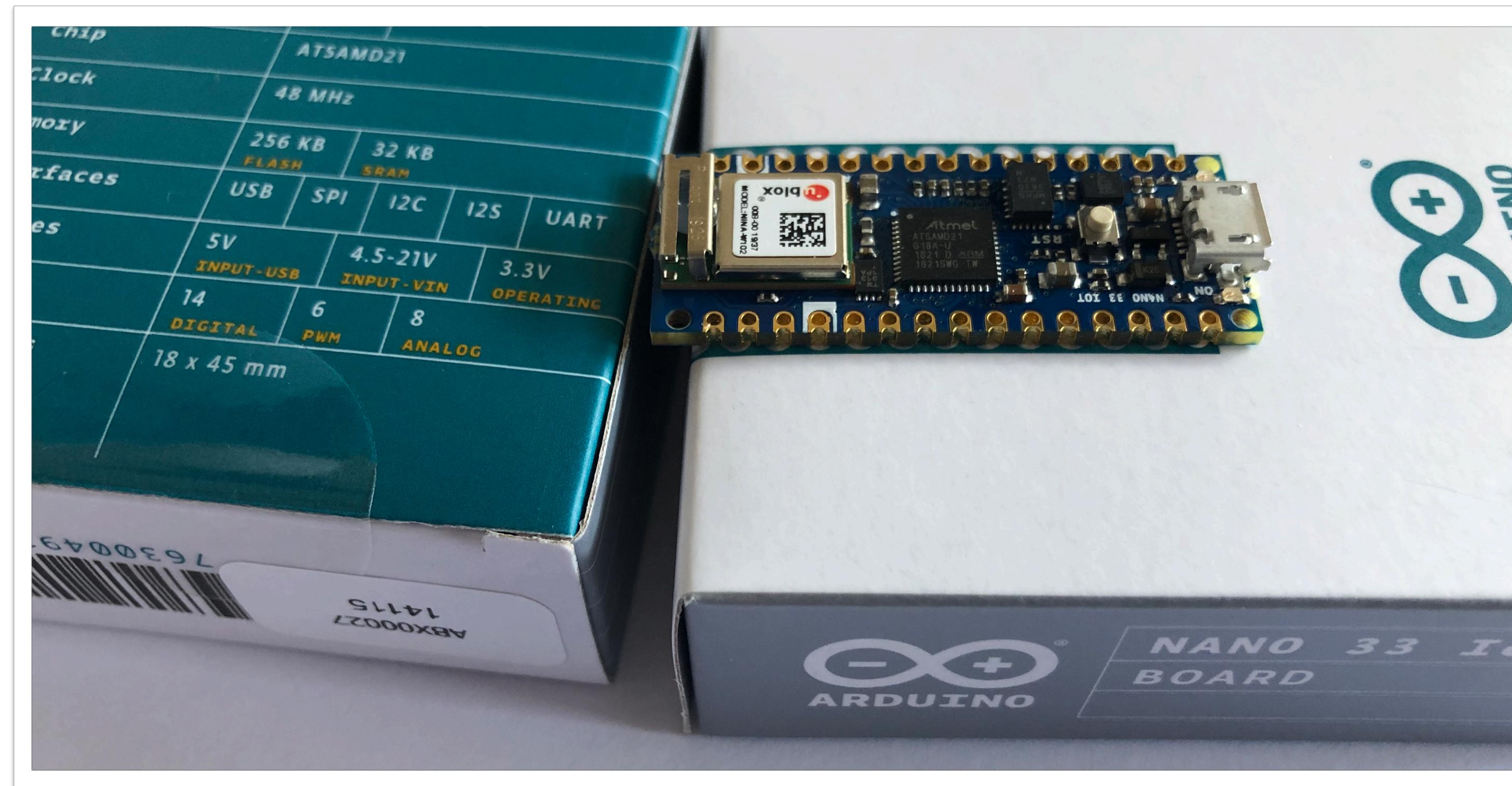
Tool name	Usage	Common Flag(s) / usage
<a href="#">dig</a> / <a href="#">nslookup</a> / <a href="#">host</a>	Show DNS record information	<a href="#">dig</a> / <a href="#">nslookup</a> / <a href="#">host</a>
<a href="#">hostname</a>	Returns hostname of your system	<a href="#">hostname</a>
<a href="#">ifconfig</a>	Network interface configuration overview tool	
<a href="#">whois</a>	Retrieving domain name information	
<a href="#">arp</a>	Display the ARP table	-an
<a href="#">ssh</a>	SSH client to connect to some host	ssh -l pi other_hostname

# Security by design



# Your secure IoT / MQTT device.

Practical: using your Nano 33 IoT Board Crypto chip (ATECC608A)



# Your secure IoT / MQTT device.

Microchip ATECC608A

## Features:

- Cryptographic co-processor with secure hardware-based key storage
- Protected storage for up to 16 Keys, certificates or data
- ECDH: FIPS SP800-56A Elliptic Curve Diffie-Hellman
- NIST standard P256 elliptic curve support
- SHA-256 & HMAC hash including off-chip context save/restore
- AES-128: encrypt/decrypt, galois field multiply for GCM



<https://github.com/MicrochipTech/gcp-iot-core-examples/wiki>

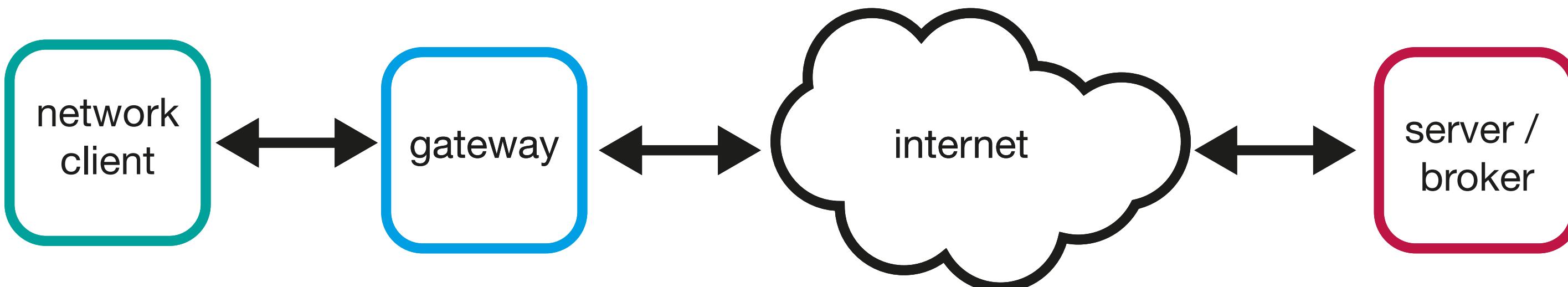
# ARP Poisoning / Spoofing

Why:

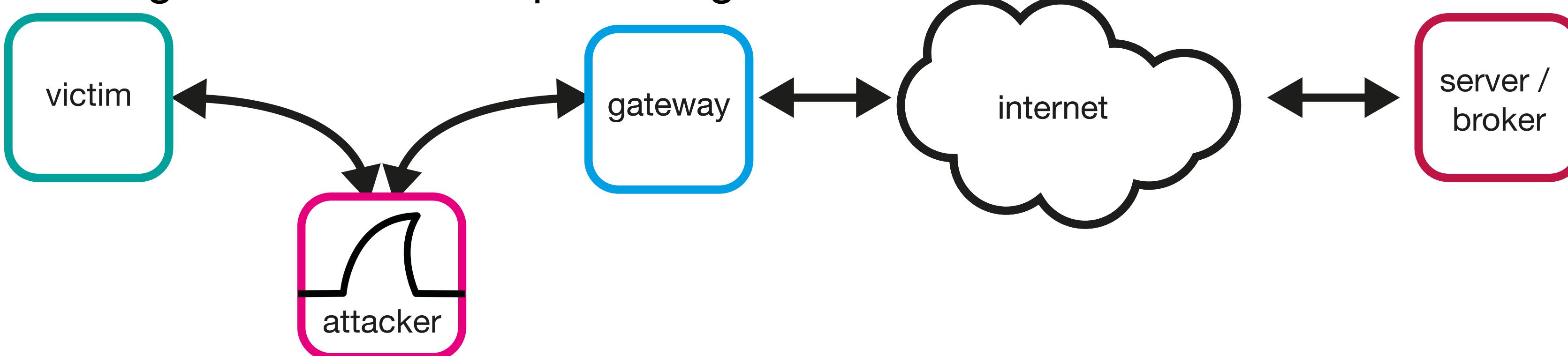
- get more insight into networking
- see why encryption matters
- get some intuition on how security can be tested

# How does ARP Poisoning / Spoofing work?

Normal routing operation:



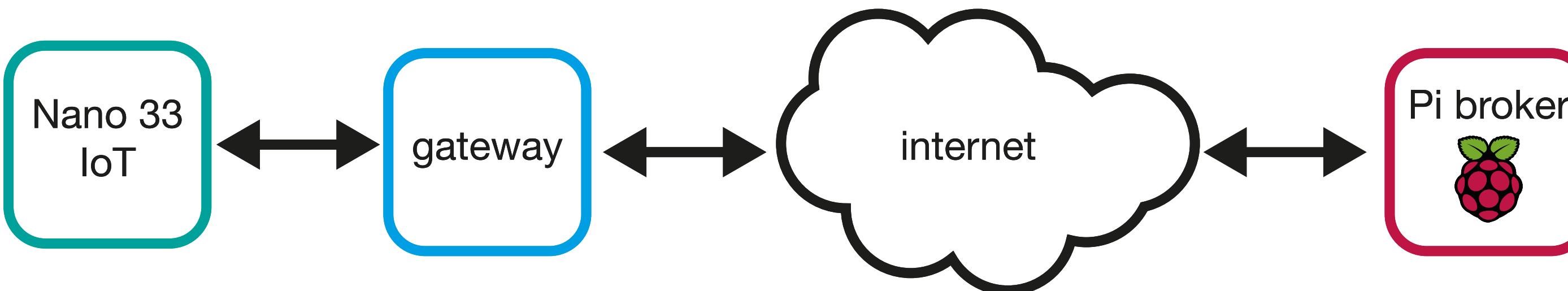
Routing with ARP cache poisoning:



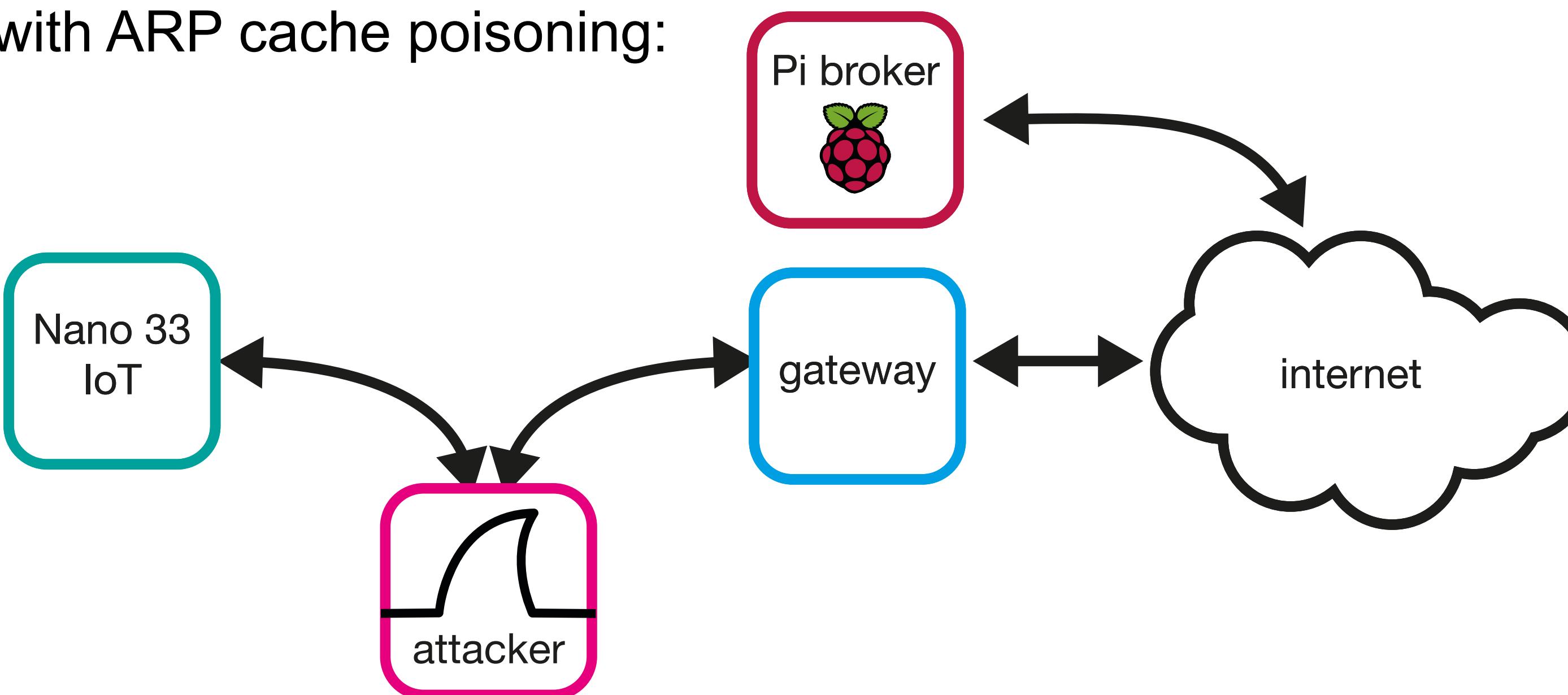
Note: victim needs to be in same subnetwork (wired, wireless, bridged)

# Ok, time for our very evil plans:

Normal routing operation:



Routing with ARP cache poisoning:



# Preparation: moving the broker outside of your local network

Either:

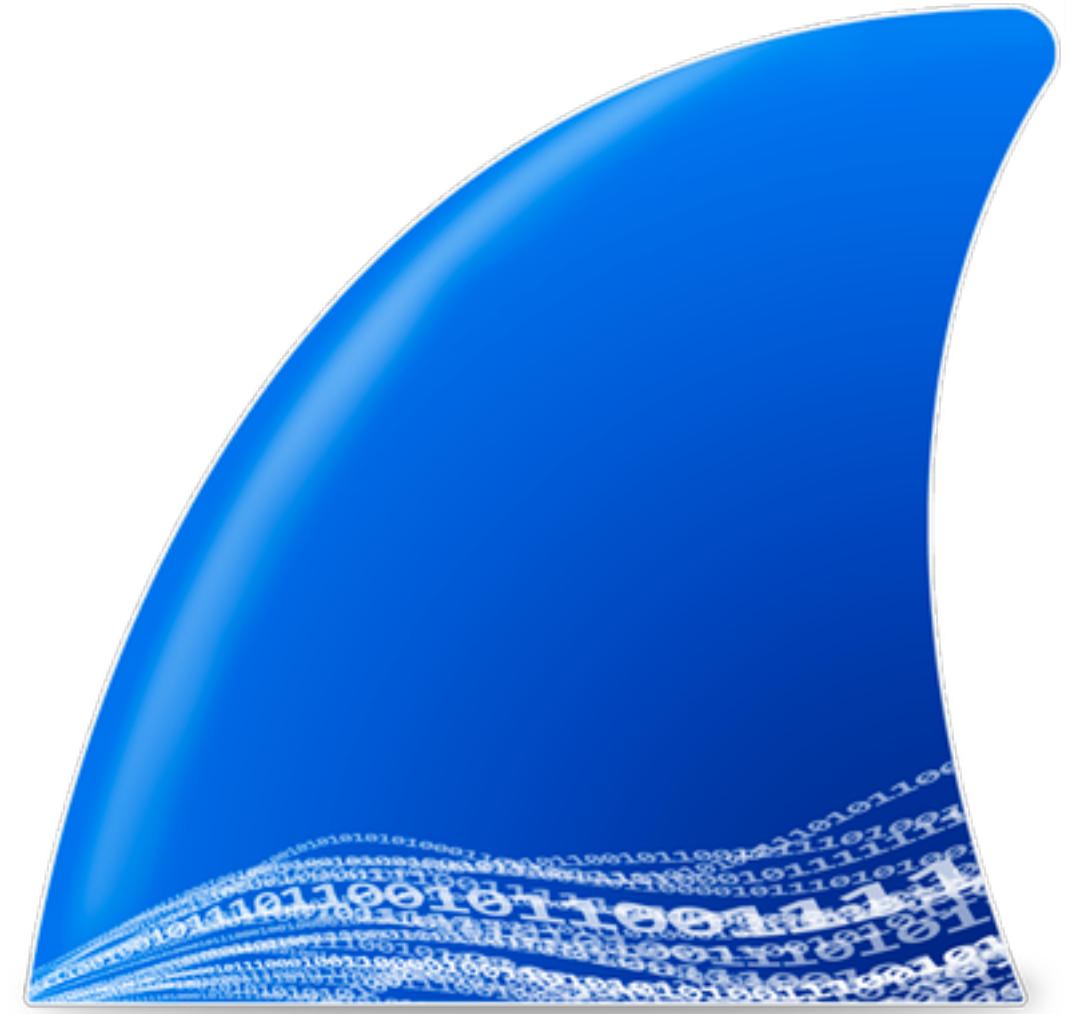
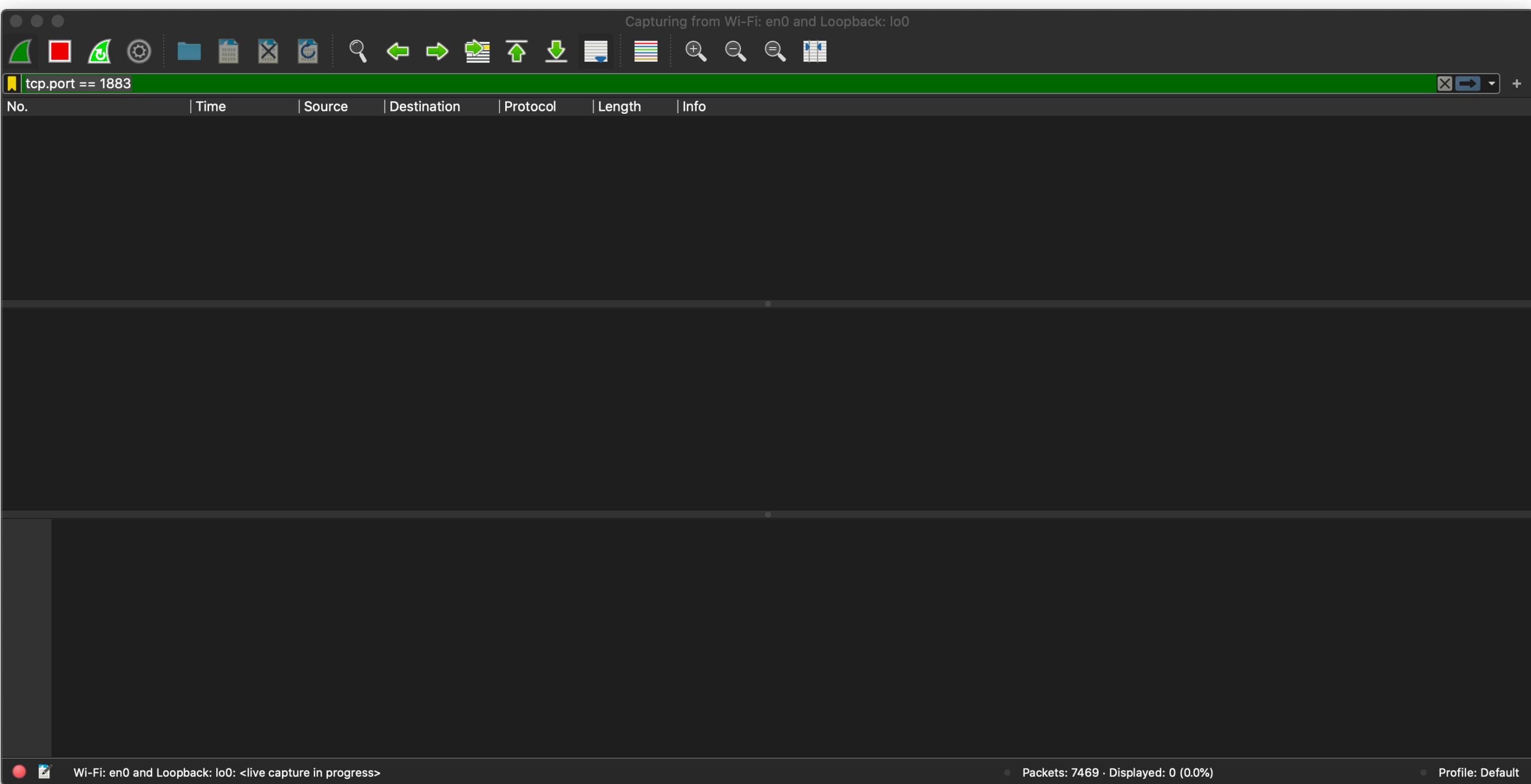
- Put your Pi in the DMZ of your Router (*possibly puts your device at risk!*)

Or:

- Port Forward 1883 in your router to the local IP address of your Pi
- Change MqttNano33 credentials to (external) IP address of your ISP router  
(Find it in your router settings or use <https://www.whatismyip.com>)

# Prepare Wireshark

1. Go to <https://www.wireshark.org>
2. Download and install Wireshark
3. Open Wireshark
4. Monitor your WiFi and local loopback



# ARP Poisoning:

Step 1: get available local network *interfaces*

Step 2: get local network interfaces *gateways*

Step 3: get *gateway information* found in step 2

Step 4: show the *connected devices* to the particular gateway

Step 5: select the *intended victim*

Step 6: poison ARP table (*pretend you are the gateway by broadcasting this*)

Step 7: do your analysis

Step 8: restore ARP tables

# Results

Filter: "tcp.port == 1883"

Wi-Fi: en0 and Loopback: lo0

tcp.port == 1883

No.	Time	Source	Destination	Protocol	Length	Info
3955	55.937407	arduino...	ip4dab43a0...	MQTT	86	Publish Message [factory/indicatorLightState]
4009	56.984987	arduino...	ip4dab43a0...	TCP	86	[TCP Retransmission] 64043 → mqtt(1883) [PSH, ACK] Seq=1 Ack=1 Win=5691 Len=32
4138	60.129684	arduino...	ip4dab43a0...	TCP	86	[TCP Retransmission] 64043 → mqtt(1883) [PSH, ACK] Seq=1 Ack=1 Win=5691 Len=32
4530	65.936120	arduino...	ip4dab43a0...	TCP	86	[TCP Retransmission] 64043 → mqtt(1883) [PSH, ACK] Seq=1 Ack=1 Win=5691 Len=32
5365	75.858333	arduino...	ip4dab43a0...	MQTT	634	Publish Message [factory/touchSensorState], Publish Message [factory/indicatorLightState], Publish Mes.
5366	75.858336	arduino...	ip4dab43a0...	TCP	58	54976 → mqtt(1883) [SYN] Seq=0 Win=5744 Len=0 MSS=1436
5423	76.352188	arduino...	ip4dab43a0...	TCP	54	64043 → mqtt(1883) [ACK] Seq=614 Ack=2 Win=5690 Len=0
5437	76.505751	ip4dab43...	arduino-2f6...	TCP	54	mqtt(1883) → 64043 [FIN, ACK] Seq=1 Ack=1 Win=64075 Len=0
5460	77.016208	ip4dab43...	arduino-2f6...	TCP	54	[TCP Retransmission] mqtt(1883) → 64043 [FIN, ACK] Seq=1 Ack=1 Win=64075 Len=0
5632	77.955662	arduino...	ip4dab43a0...	TCP	86	[TCP Retransmission] 64043 → mqtt(1883) [PSH, ACK] Seq=1 Ack=2 Win=5690 Len=32

► Frame 5365: 634 bytes on wire (5072 bits), 634 bytes captured (5072 bits) on interface en0, id 0  
► Ethernet II, Src: arduino-2f60.home (24:62:ab:b2:2f:60), Dst: Linksys02523.home (f4:0f:24:24:58:11)  
► Internet Protocol Version 4, Src: arduino-2f60.home (10.189.1.67), Dst: ip4dab43a0.direct-adsl.nl (77.171.67.160)  
► Transmission Control Protocol, Src Port: 64043 (64043), Dst Port: mqtt (1883), Seq: 33, Ack: 1, Len: 580

MQ Telemetry Transport Protocol, Publish Message

► Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once delivery (Fire and Forget)  
Msg Len: 27  
Topic Length: 24  
Topic: factory/touchSensorState  
Message: 0

MQ Telemetry Transport Protocol, Publish Message  
MQ Telemetry Transport Protocol, Ping Request  
MQ Telemetry Transport Protocol, Publish Message  
MQ Telemetry Transport Protocol, Publish Message

0000 f4 0f 24 24 58 11 24 62 ab b2 2f 60 08 00 45 00 ···\$X·\$b ···/`·E·  
0010 02 6c 02 1f 00 00 ff 06 1a 22 0a bd 01 43 4d ab ·l··· ·"··CM·  
0020 43 a0 fa 2b 07 5b 00 00 47 0e d0 5f d2 f2 50 19 C··+ [·· G··\_P·  
0030 16 3b 80 81 00 00 30 1b 00 18 66 61 63 74 6f 72 ;···0··factor

• Packets: 7589 · Displayed: 24 (0.3%) • Profile: Default

# Other types of network 'misuse'

DNS Poisoning

TCP/IP Hijacking

etc. etc.

See e.g. [https://www.tutorialspoint.com/ethical\\_hacking/ethical\\_hacking\\_dns\\_poisoning.htm](https://www.tutorialspoint.com/ethical_hacking/ethical_hacking_dns_poisoning.htm)

See also: <https://github.com/ammarx/ARP-spoofing>

# SSL and Let's Encrypt

<https://medium.com/@flynam/securing-your-iot-device-using-ssl-4643110ab901>

See also the PDF “Study of the Development of an IoT-based sensor platform for E-Agriculture”

# Links

Ethical Hacking:

[https://www.tutorialspoint.com/ethical\\_hacking/index.htm](https://www.tutorialspoint.com/ethical_hacking/index.htm)

ARP Poisoning:

[https://www.tutorialspoint.com/ethical\\_hacking/ethical\\_hacking\\_arp\\_poisoning.htm](https://www.tutorialspoint.com/ethical_hacking/ethical_hacking_arp_poisoning.htm)

