# IT Security 2024/2025
# Exercise Sheet 9
# – Device Identification –

Daniel Vogel

Publication: 05.12.2024
Deadline: 11.12.2024 10:00

Lightning Survey ⚡

**Exercise 1** (Device Identification per Layer, 2 points)**.** Device identification can be done on different layers, depending on use case and used protocols.

Describe a scenario where application layer device identification must be used, as device identification on lower layers wouldn't work. Explain why. Write your solutions into `task1.txt`.

**Exercise 2** (Passive and Active Device Identification, 2 points)**.** Which of these requirements must be met for passive device identification to succeed? Mark with True or False. Extend `task2.yml` with your solution.

- ☐ The fingerprinter must have a sensitive receiver set up, such that signals sent by the target device can be received sufficiently by that receiver.
- ☐ The target device must not move out of the fingerprinter's range.
- ☐ There must be DID observable.
- ☐ Target devices can only be identified as a specific device, if a correctly trained classifier for that device is used.

Decide whether the following statements are True or False. Active device identification ...

- ☐ only works in environments, where the target device is willing to connect to the fingerprinter's network.
- ☐ only works in environments, where the target device is willing to connect to another existing network.
- ☐ is only used to identify target devices that use MAC address randomization.
- ☐ requires the target device to respond to messages.

**Exercise 3** (Wi-Fi-based Device Identification, 6 points)**.** For your perimeter, you have set up a Wi-Fi based intrusion detection system. You received an alert by your system that an intruding device has been detected. The system provided you with its Wi-Fi trace around the incident, where the packet with the number 3083 of the Target Device (TD) has been marked as suspicious.

Analyze the given pcap (see `wisec-24.pcap`) to learn more about the intruding device. Use `script.py` to implement a Wi-Fi based IE fingerprinting method akin to the one proposed by VanHoef[1]. Calculate the fingerprint of that suspicious packet. Use that fingerprint to isolate

[1] see Section 3 of Vanhoef, Mathy, et al. "Why MAC address randomization is not enough: An analysis of Wi-Fi network discovery mechanisms." Proceedings of the 11th ACM on Asia conference on computer and communications security. 2016. `https://dl.acm.org/doi/pdf/10.1145/2897845.2897883`

all packets sent by the TD from all other packets not sent by it and solve the following tasks. Write your solution into `task3.txt`.

a) Calculate the fingerprint of the TD. State that fingerprint.

b) How many probe requests have been observed from the TD using that fingerprint?

c) State the first and last timestamp of these observed probe requests of the TD (YYYY-MM-DD hh:mm:ss).

d) State your observation towards the MAC address of the TD.

e) Which SSIDs did the TD probe for?

f) Describe the behavior of the observed RSSI measured for all packets received from the TD.

g) What is the average rate with which the TD sent its probe requests during the observed time period? Towards what change in its transmission state does the probe request rate hint for the TD?

*Hints: You do not need to use sequence numbers for this task. The fingerprint of the first packet in `wisec-24.pcap` should be looking something like this:* `0,1,50,3,45,127,107,supra:02040b16,extrates:0c1218243048606c,htcap:402d,htmcs:000000000000000000000000ffffff,htext:0000,httx:00000000,htasel:00,htampdu:1b,interworking:0f`

**Note:** You are not limited to use the given `script.py`, if you prefer not to use Python. If not using the given `script.py`, if your solution does not match the expected solution, no further points will be awarded. The given `script.py` handles access to the .pcap file. You will need to use the `scapy` library.