Advanced Internet and IoT Technologies

Assignment 02

HAW Hamburg WS 2025

Prof. Dr. Thomas Schmidt, Michel Rottleuthner, M.Sc. Deadline: 11. November 2025

1. IPv6 for the IoT with 6LoWPAN

IoT link layer technologies usually have a low maximum transmission unit (MTU). At the same time, each byte over-the-air prolongs radio transmissions and increases energy expenditures. 6LoWPAN is a convergence protocol that adapts IPv6 to resource-constrained IoT environments.

The IPv6 address of an IoT device and necessary routing entries will be provided during the lab class.

- (a) Find out the MTU for the IEEE 802.15.4 link (lowpan0).
- (b) 6LoWPAN employs two methods to allow larger packets on a small-MTU link. What are these methods?
- (c) Ping an IoT device and gradually increase the payload. What do you observe in tshark on lowpan0 and wpan0? Do the success rates for ping change?
- (d) What is the largest packet size that 6LoWPAN can fragment? Check RFC 4944 for more information on the 6LoWPAN internals.

2. 6LoWPAN header compression

Header compression techniques optimize for minimal packet sizes in the constrained IoT. 6LoWPAN provides a 1) stateless compression with rules defined in the corresponding RFCs, and 2) a stateful compression, which relies on globally deployed context.

Tools: netcat (nc)

- (a) Run tshark on wlan0 and send a UDP message with payload *hello* to ff02::1%wlan0 using source and destination ports 8888. Record the packet size.
- (b) Run tshark on wpan0 and send the same UDP message to ff02::1%lowpan0. Did the size change? Look at the captured 6LoWPAN header for compression details.
- (c) Now change both UDP ports to 61616 and resend the packet on lowpan0. Why did the packet size change again?

3. RESTful applications for the IoT

The Constrained Application Protocol (CoAP) is considered to be the HTTP for the IoT and aims at providing the REST architecture for the most constrained devices and networks.

The IPv6 address of an IoT device and necessary routing entries will be provided during the lab class.

Tools: aiocoap (from python3-pip)

- (a) Use aiocoap-client to discover available CoAP resources of the IoT device.
- (b) After discovering the resources, perform a GET request to request the board name and the status counter.
- (c) Modify the status counter using a PUT request with any random integer number and use a GET request to read the newly assigned value.