Lehrstuhl für Technische Informatik
Prof. Dr. Stefan Katzenbeisser
Organizer: Martin Schmid
Supervisors: Emiliia Geloczi

Passau, 01.10.2025

Security Engineering Lab, Advanced Security Engineering Lab

Intrusion Detection in Smart Home.

Topic Introduction

Every day, the number of IoT devices worldwide increases. While these devices simplify many aspects of daily life, they also introduce risks, as IoT systems continuously generate, transmit, and process large volumes of information that may be critical in certain contexts. This makes smart homes an attractive target for adversaries. To protect data, it is necessary to detect and respond promptly to attacks that adversaries may launch. Intrusion Detection Systems (IDSs) exist for this purpose.

For this project, you will need to: explore background information on IoT system architectures and existing IDSs to identify a research gap; build a prototype IoT environment; and implement your IDS concept.

General information

- Creditability: Master
- Group size: 3-4
- Knowledge Requirements: C, ESP-IDF, Linux
- Provided Equipment: You will receive access to the software and hardware listed in Table 1.

Equipment	**Quantity**
Raspberry Pi	1
Router MikroTik	1
ESP32 with different sensor modules (button, led, etc.)	2

Table 1: Equipment that will be provided to the students.

Grading:

You can obtain at most 100 points. Table 2 shows how the number of points correlates to this grade.

Point Range	Grade
[0, 50.0[	5.0
[50.0, 55.0]	4.0
]55.0, 60.0]	3.7
]60.0, 65.0]	3.3
]65.0, 70.0]	3.0
]70.0, 75.0]	2.7
]75.0, 80.0]	2.3
]80.0, 85.0]	2.0
]85.0, 90.0]	1.7
]90.0, 95.0]	1.3
]95.0, 100.0]	1.0

Table 2: Mapping of Points to Grades

First steps If you don't have a FIM account yet, create one[1]. Please note that the link is only accessible via the internal university network. After you have created your account, inform your supervisor and tell them your FIM username, with the request to create a FIMGit repository for your group. If you are accepted to this topic, please send an email to the organizer[2] and tell them your campus card number (printed on the front of your campus card underneath the matriculation number). This will give you access to the lab in room ITZ 103 outside of compulsory lab hours.

Tasks

1. Task: Theoretical Background about IoT Systems (12 points)

In this task, you will become acquainted with the background information that is relevant to the topic of this work. Describe the following concepts and terms in a structured PDF file and present your findings, in order to address and get acquainted with the relevant literature.

IoT Systems: General Information.

1. Internet of Things (IoT). What is it? (1 point)
2. Architecture of IoT Systems (layers, mechanisms/protocols used, etc.) (2 points)
3. Vulnerabilities of IoT systems. (1 point)

Existing Attacks Against IoT Systems:

1. List and explain existing attacks against IoT Systems. (2 points)

Intrusion Detection Systems in IoT Domain:

1. Conduct a literature review following the PRISMA approach. (2 points)
2. Based on the identified IDSs, specify their limitations and capabilities. (4 points)

[1] `https://ams.fim.uni-passau.de/pin_request.php`
[2] `martin.schmid@uni-passau.de`

2. **Task:** Test Environment Establishment (29 points)

In this task, you need to implement the foundation of a test environment for subsequent attacks. A Raspberry Pi will act as an access point (AP) that provides Wi-Fi for connecting IoT devices, and will also serve as an IoT hub/gateway and MQTT broker. ESP modules with various sensors will simulate real IoT devices. They must connect to the AP (RPi), be able to communicate using HTTPS and MQTT, and be controllable via the RPi (for example, through a web interface). For the RPi, use Raspberry Pi OS (with a desktop interface). For ESP chips, use native ESP firmware written in C with the ESP-IDF environment.

1. Decide which types of IoT devices will be emulated by the ESP chips and define their functionality based on the available sensors.
2. Implement the AP/hub/gateway/MQTT broker on the RPi. (14 points)
3. Implement firmware for the ESP chips to provide the required functionality. (9 points)
4. Implement a user interface to control the ESP IoT devices and to connect them to the RPi, creating a complete test environment that simulates a real smart home. (3 points)
5. Provide reproducible step-by-step documentation for this task. (3 points)

3. **Task:** Implementation of IDS (51 points)

For this task, you should determine the characteristics your IDS will use and the attacks it will detect, and then implement it.

1. Describe the system and adversary model of the IoT system developed in Task 2. (2 points)
2. Describe the idea of your IDS. (3 points)
3. Implement your IDS. (40 points)
4. Test your IDS. (3 points)
5. Provide reproducible, step-by-step documentation for this task. (3 points)

4. **Task:** Presentations (8 points)

Presentations will be evaluated according to the following criteria:

1. *Clarity & organization*: How well-structured is the presentation? Are the introduction, methods, results, and conclusions presented logically and easy to follow?
2. *Content accuracy*: Are presented data and interpretations correct?
3. *Visuals & ommunication*: Are slides, graphs, or other materials clear, readable, and relevant? Do they enhance rather than distract?
4. *Presentation skills*: Do students explain confidently, speak clearly, manage time well, and engage the audience (e.g., eye contact, voice, handling questions)?

For each presentation, the maximum number of points you can obtain is as follows:

1. Half-time presentation. (4 points)
2. Final presentation. (4 points)

Additional Remarks

Code quality: Use your software engineering skills to write code that is efficient, reliable, and easy to maintain. Organize your code into logical, reusable modules and maintain a clear separation of responsibilities within your implementation. Your code should be well-structured, thoroughly commented, and straightforward to read and understand. For each function, include a short description of its purpose, clearly state its input and output values along with their types, and use a consistent, reasonable naming scheme for variables, functions, and classes.

Final report quality: The final report must be submitted as a PDF file, and all data and code produced must be submitted either as a ZIP archive or via a Git repository. The writing style should follow British academic conventions and include references to all academic sources used.

Tasks required reprodicible documentation: If the solution cannot be reproduced using the provided documentation, 50% of the points awarded for completing the related task will be deducted.

Using AI: While working on the topic, be prepared to answer questions about the technologies used, their functions, and the terminology applied in the work. Excessive or inappropriate use of AI tools will result in a reduction of points and, in exceptional cases, failure to pass the course. Examples of AI misuse include: citing non-existent sources; using inappropriate terminology in the context of IT security; providing overly detailed and lengthy but meaningless code comments; introducing completely unnecessary constructs in the code; giving misleading instructions; or producing obvious errors in relation to the topic.