ARTEMIS KickOff Document

Description

The end product will be an extensible monitoring and intrusion detection tool for IoT systems, that supports plugging in different data sources (IoT devices, networks, protocols) and anomaly detection algorithms. The tool will be demoed with the setup of a realistic IoT scenario and launching intrusions on the system using open source attack tools.

Today's intrusion detection systems (antivirus, firewall, etc.) cannot be used on these devices due to low processor power and memory capacity of IoT devices. To address this issue the program aims to detect anomalies without requiring computing power on the IoT device itself. Through developing an anomaly-based detection approach with machine learning, the developed tool has the goal of detecting intrusions without using attack signatures, which will also prove useful for zero-day attacks.

The target user of this tool is mostly system admins or IT departments of institutions that use IoT devices to transfer sensitive information such as hospitals.

Master feature list

MF-1 Operation on IoT devices with most commonly used IoT-specific protocols such as Zigbee, BLE, CoAP
MF-2 Network connection between the network packet sniffing devices and the anomaly detection server
MF-3 Sniffing the real time streaming data on IoT devices
MF-4 Continuous data gathering on the anomaly detection server
MF-5 Periodically training on gathered data
MF-6 Clustering the data based on the selected features
MF-7 Capability of detecting anomalies using machine learning algorithms in near real time
MF-8 Upon detecting anomalous event, alerting the user with detailed information regarding the detected event
MF-9 Presentation of different machine learning models to choose from
MF-10 Capability of detecting anomalies in the IoT network with at least 90% precision and 90% recall
MF-11 Includes a User Interface for accesing other features
MF-12 (Bonus) Graphical User Interface for visualization of both streaming data and anomalies
MF-13 (Bonus) Detect anomalies when provided a log data

Workpackages

WP #	Term	WP title (this should be as short and as descriptive as possible)	Estimated number of person-months
1	491	Project planning and architecture design	3
2	491	IoT equipment purchase and environment setup	4
3	491	Streaming architecture development for IoT data collection	6
4	492	Identification of anomaly detection algorithms and	6

		implementation	
5	492	Identification of open source IoT attack tools and performance of tests with them	4
6	492	Intrusion detection tests and implementation refinement	7
		Total:	30

Detailed Descriptions of High-Level Workpackages

WP1 - Project planning and architecture design

1. Develop the list of master features of the project.
2. Produce project development plan in accordance with Master Feature List.
3. Design the overall architecture of the project.
4. Analyze risks and make a management plan.

WP2 - IoT equipment purchase and environment setup(MF-1-2)

1. Identify and purchase the components that are needed to set up the environment
2. Prepare a Raspberry Pi to imitate various IoT devices
3. Set up the central anomaly detection server
4. Set up a Raspberry Pi device that will be used as the Sniffer and connect with the server
5. Set up a Raspberry Pi device as the Gateway and connect it with sensors

WP3 - Streaming architecture development for IoT data collection(MF-3-4)

1. Develop and install the necessary software in order to sniff network packets and collect sensor data
2. Transfer the sniffed data continuously to the server
3. Process streaming data on the fly with special software (e.g. Apache Spark) to extract necessary features and store on the server

WP4 - Identification of anomaly detection algorithms and implementation(MF-5-6)

1. Literature search to determine suitable machine learning algorithms
2. Select the features that will be examined by the algorithms
3. Develop anomaly detection algorithms taking advantage of Apache Spark Mllib and Tensorflow
4. Use ML algorithms such as LSTM, Isolation Forests, K-Means clustering, One-Class-SVM, etc.

WP5 - Identification of open source IoT attack tools and performance of tests with them(MF-7)

1. Literature search to determine penetration testing tools for IoT devices for protocols that have been used
2. Learn how to generate attacks using the tools found
3. Test the system to check if the near real time constraints are met
4. Refine the ML algorithms to make them work in near real time

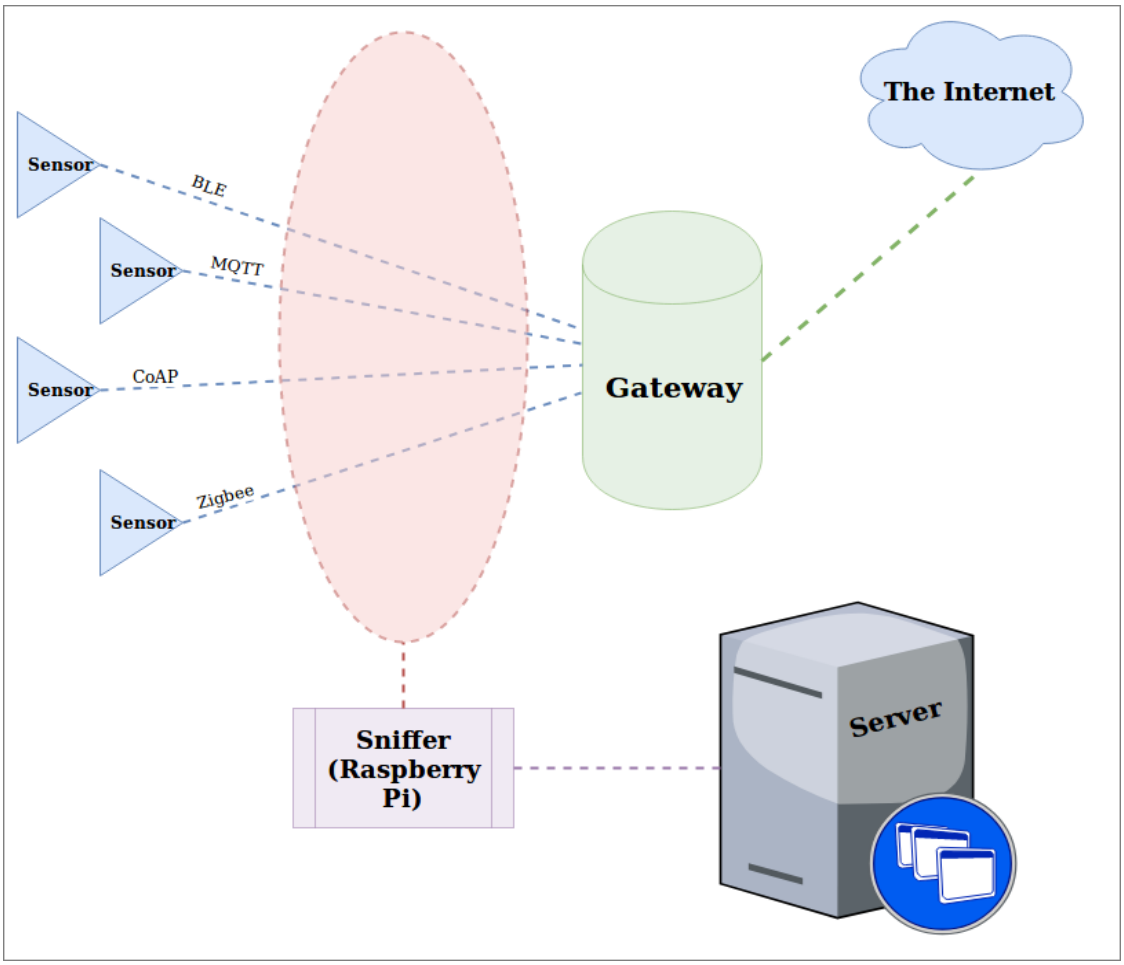
WP6 - Intrusion detection tests and implementation refinement(MF-8-9-10-11-12-13)

1. Test for the required precision and recall rates with the tools found in WP5
2. Run system tests
3. Implement the alert system
4. Make multiple ML models selectable for the user
5. Develop a user interface
6. (Bonus) Develop a Graphical User Interface for visualization of both streaming data and anomalies
7. (Bonus) Add the feature of detection anomalies on data logs

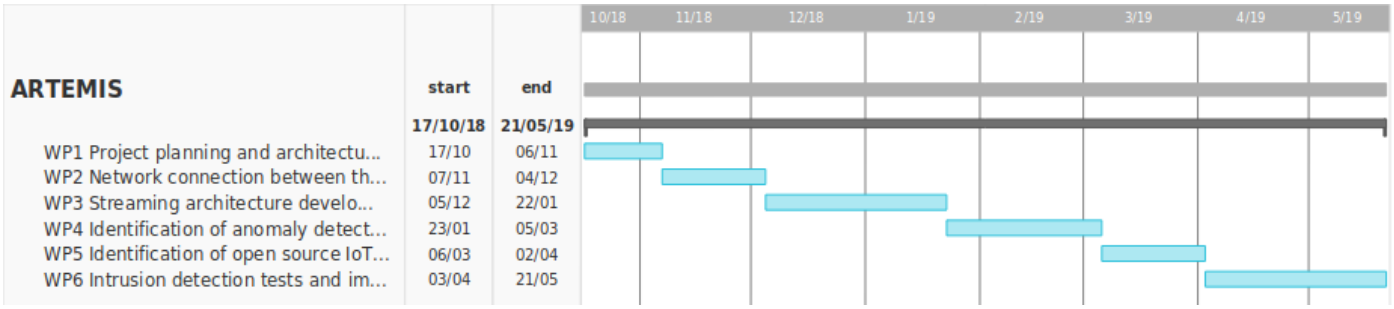
Overall Systems Architecture

The sensor components collect data from the environment. To be connected to the Internet these collected data are forwarded to the gateway over certain protocols including but not limited to BLE, MQTT, and Zigbee.

The sniffer Raspberry Pi intercepts the packet data at this point. The intercepted packet data is then sent to the Server where the anomaly detection algorithm(s) is running. The data is also stored for future training purposes in order to evolve continuously.



TimeLine



Risk Assessment

Risk #	Description	Possible Solution(s)
1	The intended precision/recall rate might not be achieved	Choose a different ML algorithm or try and ensemble learning approach
2	The intended performance in terms of speed, in data processing and intrusion detection may not be achieved due to factors like the large size and dimensionality of data.	Utilize data sampling and dimensionality reduction algorithms to preprocess the gathered data
3	There may be issues using the open source attack generation tools for IoT	The set of attacks with which we will perform our intrusion detection tests can be reduced
4	A chosen protocol might not be well-documented or hard to sniff	It can be replaced with a more favorable protocol.
5	Due to lack of funding the necessary devices might not be acquired	Cheaper alternatives can be used