Intrusion Detection and Prevention for the Internet of Things

Background
The Internet of Things (IoT) is increasing the connectedness of people and things on a scale that once was unimaginable. Connected devices outnumber the world’s population. The pace of IoT market adoption is accelerating because of growth in analytics, cloud computing and increasing interconnectivity of machines and personal smart devices. Use cases like finding an available parking spot e.g. in San Francisco with immediate feedback from sensors are becoming reality. Smart dumpsters in Santander are another great example using the Internet of Things.

However, the technical requirements are very complex. A specialized HTTP-like protocol is defined for communications between very simple electronic devices called CoAP [1,2] (Constrained Application Protocol). CoAP allows them to communicate over the Internet. It is targeted for small low power sensors, switches, valves and similar components that need to be controlled over standard internet networks.
Problem Statement
CoAP messages are commonly secured using lightweight versions of IPsec or DTLS. Yet, these end-to-end security solutions only filter CoAP messages at their final receiver. Thus, attackers can still launch battery exhaustion attacks by injecting large amounts of bogus CoAP messages. Moreover, 6LoWPAN nodes usually do not implement further signature-based intrusion detection. This is because memory on 6LoWPAN nodes is rare and keeping signatures up to date would incur an overhead in energy consumption.

This is a severe problem since the software on 6LoWPAN is typically written in C, which is notorious for its memory vulnerabilities. Thus, attackers may succeed in compromising 6LoWPAN nodes by injecting specially crafted CoAP messages. Once an attacker successfully compromised a 6LoWPAN node, this compromised node may, e.g., report sensor readings to the attacker.

Master project
To solve these issues there is a need for filtering CoAP messages on a CoAP proxy. In fact, CoAP proxies may provide several services such as authentication and authorization of users, HTTP-CoAP proxying, or caching. The focus of this master's project will be on implementing a CoAP proxy that validates CoAP messages. For example, a 6LoWPAN node should never report data to a client that never requested this data. Also, CoAP requests and responses should comply with the respective resource descriptions.

Contact
Chair: Internet-Technologien und -Systeme
Supervisor: Prof. Dr. Christoph Meinel, Konrad-Felix Krentz

References