Compact Object Security for the Internet of Things

Joakim Brorsson, Martin Gunnarsson, *Lund University*

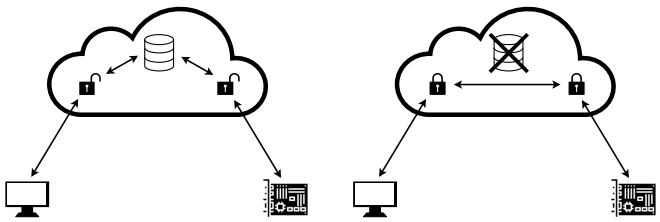
2016-06-29

The Internet of Things (IoT) is coming. With it comes new security challenges from the constrained nature of IoT devices. As a response to the need for efficient security, Ericsson and SICS are collaboratively developing a new protocol, OSCoAP.

Security challenges in IoT

IoT devices often operate on battery power and have restricted computing power. Therefore traditional approaches to communications security, such as the *channel security* protocol DTLS, can be a bad fit for some applications of IoT.

These protocols encrypt everything that is sent over them. If DTLS communication is carried over intermediate proxies, even proxying information will be encrypted. Therefore, decryption capabilities are often given to the proxy. This is called *hop-by-hop* security and is undesired compared to *end-to-end* security where an intermediate can not access data.



(a) *Hop-by-hop* **(b)** *End-to-end*

Figure 1: *End-to-end security compared to Hop-by-hop security.*

In *channel security* protocols, *end-to-end* security in combination with proxying functionality is unobtainable. A new solution is needed in order to provide *end-to-end* security through proxies.

Object Security in IoT

Contrary to *channel security* protocols, *object security* protocols, such as the novel OSCoAP proposal, can provide selective encryption. This means that OSCoAP encrypts confidential parts of a message using keys unavailable to the proxy, while metadata intended for the proxy is sent in plain text. This way of achieving confidentiality minimises the need to trust a proxy, since the proxy can not read confidential data. The purpose of OSCoAP is to provide *end-to-end* security through proxies for IoT devices. To test the feasibility and efficiency of the OSCoAP protocol proposal, this project has implemented a proof of concept. The goal was to test if OSCoAP was implementable and had acceptable performance compared to DTLS.

Results

A number of metrics, e.g. processing-time, memory footprint and network overhead, has been identified as important for IoT devices. In our measurements, OSCoAP proved slightly more network efficient than DTLS. Further, OSCoAP has a longer processing time and a slightly higher memory footprint than DTLS, but the performance is still acceptable. However, the implementation is not yet fully optimised.

Conclusion

OSCoAP shows great promise. It is able to obtain *end-to-end* security through proxies with acceptable performance. If the protocol becomes a standard, *end-to-end* security will be available in more scenarios than today. An *object security* solution designed for IoT is needed, and OSCoAP might just be it.