Demo: Zero Touch Configuration

Nicolai Kuntze, Pedro Larbig, and Carsten Rudolph
{nicolai.kuntze|pedro.larbig|carsten.rudolph}@sit.fraunhofer.de
Fraunhofer Institute for Secure Information Technology - SIT, Darmstadt, Germany

***Abstract*—The deployment of devices at remote or distributed locations is a typical scenario e.g. in large enterprise networks or industrial applications. In the deployment of a device identification of the device and the establishment of security relations (*trust*) between the device and other elements of the infrastructure is crucial. Usually, the process requires either to pre-configure the device or to let administrators physically access the device for configuration. Both options induce costs. For functional configurations and software distribution *zero configuration* solutions are available. One important step hereby is the establishment of trust into the individual device by the device owner. This trust establishment requires in typical schemes a large organizational involvement of the device owner resp. operator. The approach presented in this demo addresses the step of initial trust establishment. The demonstration shows the process of a device being securely configured and provides a visualization through a meta-data graph generated from an IF-MAP server.**

I. Introduction

Deployment of devices at remote locations requires in various cases an initial trust relation to the particular device. This step in nowadays performed using either manufacturer presets or costly physical interaction with the individual device e.g. using USB tokens. The approach shown in this presentation provides a more efficient support for the establishment of security relations. No individual configuration is necessary for the device. Instead, the complete deployment can take place at the location of intended use. Further, no customer data needs to be located in the unit before the automatic initialization. Thus, enabling a direct trust relationship between the unit and the customer's other components only requires off-line registration of the device but no physical interaction with the device.

The following demo exemplifies a protocol for the implementation of such a deployment support within a machine to machine scenario. In this case, the exemplary goal for the demonstration of the concept is showing the equipment can establish a configuration file. Customer secrets that are necessary for this secure channel are transferred onto the equipment during the automatic initialization at the place of deployment and not via a process involving physical interaction, for example by way of a USB stick for the installation. The concept can be used for various machine to machine communication scenarios. So far, in M2M scenarios direct configuration of the units by administrators is the usual approach. To do this, customer-specific secrets enabling a later communication are introduced to the units prior to the actual deployment. This action does not allow for a remote implementation and is thus clearly more expensive. One additional advantage of the presented approach is that it provides a secure identification of a device. Current solutions rely on a combination of insecure hardware identification (e.g. using MAC addresses) and cryptographic keys or other credentials protected by software.

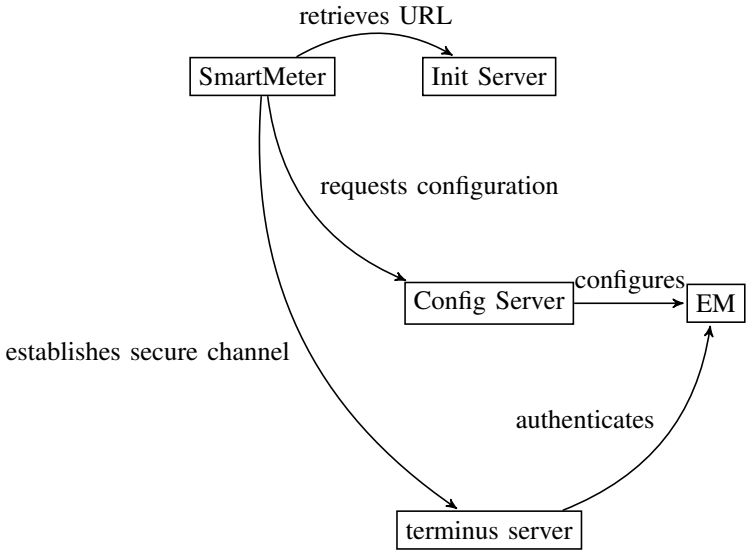

Fig. 1. Components of the Infrastructure

Establishment of trust in devices is done in available products using manual techniques. TOFINO[1], a Belden Group product, transfers the data to the units using a USB storage device, which requires direct physical contact to the unit. In the industry standardization a preliminary protocol was proposed [2] to establish keys and parameters. Recent research offers more developed approaches as introduced in [1]. More detailed information on can also be found in [3].

In the following text, the protocol is explained using a hardware trust anchor, the Trusted Platform Module (TPM) and TPM-specific implementation variants are shown to allow for a high protection and security level. It should be noted that the presented approach can also be implemented with other hardware security solutions or without hardware-based security. Nevertheless, the security level of the implementation will depend on the hardware trust anchor and on how it is integrated into the device.

II. Architectural Overview

A remote device (RD) should access an infrastructure via a secure channel (e.g. SSL/TLS) in order to access various services. The relevant components in this scenario are an init server, a configuration server as well as the secure channel terminus in the form of an e.g. VPN gateway or other

[1] http://www.belden.com/aboutbelden/beldencompanies/tofino-security.cfm

communication server. The components and their respective function are as follows. The required infrastructure components to allow for the configuration and channel establishment are illustrated in Figure 1 with their logical connections. The **RD** is the center of attention and must be configured accordingly before it is actually put into use. The configuration data consist of the terminus address, the special channel settings as well as the required channel secrets (i.e. cryptographic keys and credentials). The **Init Server** is first contacted by the RD in order to determine the config server. The init server will most probably be provided as a service by the manufacturer of RD. It should be noted that in the protocol, the init server will not provide or store any cryptographic keys (except maybe its own private key) or distribute any security-relevant information. Its role is mainly to point the RD to the right place for initial configuration. The **Config Server** (CS) is reached via the address given to the RD by the init server. The main role of the CS is to provide to the RD the special secure channel configuration data as well as the secrets clearly specific to the equipment and necessary for establishing the secure channel in relation to the terminus server. Actual management and storage of secrets is done by another component, namely the **Endpoint Manager**(EM) not directly accessible via the Internet. The actual secure channel is established between the RD and the **Terminus Server**. To do this, secrets established with the CS are used. No further contact to the init or CS is needed for actually establishing a secure channel.

III. The Remote Device Life cycle

The RD's life cycle is divided into the following phases: production, customer registration, installation via the user, operation, and maintenance.

Production: An RD needs to use a securely stored asymmetric keys for communication and signatures. As a technical realization of this requirement, a hardware-protected trust anchor in the form of a TPM guaranteeing protection of the key can be used. The TPM already comes with the endorsement key-pair EK_pub and EK_priv. At the end of the production, the hardware supplier attaches a mark to the RD. This mark contains the equipment's serial number. The fingerprint of EK_{pub} is selected and used later in the registration as well as in the installation. Furthermore, an equipment-specific secret N_{device} is created, delivered separately from the equipment, and stored securely on the RD. This information can be represented by QR codes. Further, the producer establishes on the device a known URL of the init server as well as the certificates of the init servers. This information is the same for all produced devices and not customer-specific.

Registration: The client stores the RD's data (e.g. represented by QR codes) in an appropriate database. The data registered have the following tasks. The fingerprint of the EK_{pub}, i.e. $H(EK_{pub})$, is used for the identification of RD. The EK_{priv} is protected by the TPM. Thus, a challenge-response protocol can be used for identification. The secret N_{device} is stored on the individual RD and only known to the customer (the owner). N_{device} is used as proof of the CS's authorization enabling it to configure the RD. A similar approach was presented in [2].

Installation: The installation phase is meant to establish a trust relationship between the RD and the customer's infrastructure. Here a central goal is establishing and securely storing the secure channel secrets on the RD The protocol for this installation consists of identification of the CS, install information on the terminus server, establish trust, and establish cryptographic keys. In the first step the RD is notified by the Init Server to which CS it should connect. The CS the actually executes the security configuration. This first creates a trust relationship between the RD and the Terminus server. This trust is based on the identity of the device as well as its attested software configuration. Once the trust has been established, the channel secrets are transferred onto the RD and securely saved.

IV. Demo

The presentation of the implementation shows the individual steps of the security protocol developed for the specific needs of the deployment use case. An online version is available at http://www.trustedcomputing.eu/cms/prototypes-demonstrators/zero-touch-configuration/. The source code is available on request and will be released under the GPL in the near future. To allow for a better presentation of the protocol and its individual steps, a graphical presentation was developed using the capabilities of an IF-MAP server (IROND) and the IRONGUI for presentation[2]. Details on the IF-MAP protocol are available at the web pages of the Trusted Computing Group[3].

During the demonstration the interaction between the different servers introduced can be seen using the graphical presentation as well as the output of the servers and client directly. In fact, the graphical presentation uses the already integrated event management added to the core protocols to allow for integration in target applications and systems.

Due to the chosen presentation, the audience can understand the different duties of the individual components and their interaction. Furthermore, the extensible design and the already integrated interface to event processing allows to show how this approach can be integrated in the targeted application scenarios like industry automation, industry control systems, SmartGrid equipment, and general networking equipment.

V. Conclusions

Roll out processes are typically the most costly step in most life cycle designs. The presented protocol and concept allows for an optimized and cost reduced process for initial deployment of devices at remote locations.

References

[1] Leung A. and Mitchell C.J. Towards secure zero configuration. In *Proceedings of Western European Workshop on Research in Cryptography (WeWoRC 2005)*, 2005.

[2] Stephen Hanna. Configuring Security Parameters in Small Devices, July 2002.

[3] Nicolai Kuntze and Carsten Rudolph. On the automatic establishment of security relations for devices. In *Integrated Network Management (IM), 2013 IFIP/IEEE International Symposium on*, may 2013.

[2]http://trust.inform.fh-hannover.de/joomla/index.php/projects/iron

[3]http://www.trustedcomputinggroup.org/developers/trusted_network_connect