Keywords: smart meter, security, smart grid, stuxnet, meter security, manufacturing security, life cycle, SCADA

TUTORIAL 5486

Securing the Life Cycle in the Smart Grid

By: Kris Ardis, Business Director
Nov 27, 2012

Abstract: Investment in smart meters and smart grid end equipment continues to grow worldwide as countries try to make their electric delivery systems more efficient. However, as critical as the electric delivery infrastructure is, it is normally not secured and thus subject to attack. This article describes the concept of life-cycle security—the idea that embedded equipment in the smart grid must have security designed into the entire life of the product, even back to the contract manufacturer. We also talk about how life-cycle security applies to embedded equipment in the smart grid. Potential threats are discussed, as are potential solutions to mitigate the risks posed by those threats.

A similar version of this article appeared on *Power Systems Design*, September 7, 2012.

Security is an increasingly critical subject in the smart grid. With regular attacks on smart grid infrastructure, there is a clear threat—the stable supply of electricity in every nation is at risk of being compromised by unfriendly forces. In response, there is a great focus on IT security; many solutions support the end-to-end encryption between embedded data collection devices on the smart grid and the supervisory control and data acquisition (SCADA) systems that analyze and react to the data. This focus on IT security is undoubtedly critical as data "in flight" must be protected with standards-based cryptography. However, even with the strongest end-to-end encryption, there is a severe shortcoming in the security of the smart grid: the embedded device itself is highly vulnerable to attack.

Encryption Is Security, Right?

While cryptographic tools are critical for ensuring the privacy and authenticity of transmitted data and commands, it is important to note that they are only one part of the solution. Encryption's greatest value is to protect data when it is in transit or in storage to prevent deciphering or forgery. While there are some who

believe that a complex RF or powerline carrier that relies on frequency-hopping provides enough security to "obscure" data, this is a protection easily broken by attackers. Imagine if an attacker could create an arbitrary command to open the remote disconnect switch in a smart meter: electric service could be disrupted for a large number of people and the utility would be swamped with service requests. Not only could this result in a loss of significant revenue and a severe inconvenience to those affected, it could be life threatening in regions where air conditioning is a necessity.

But what happens to the data before it enters the pipe and after it exits? There are encryption keys at either side of communication pipes that encrypt, decrypt, authenticate, or validate the data in transit. While the encryption of the data in the pipe is critical to secure information as it passes from embedded sensor to the control system, the protection of the secret keys used in the encryption is even more important. If those keys are compromised, the security of the network can be compromised. Embedded endpoints in the smart grid must consider key security to provide a more complete security solution. Secure financial terminal technology emphasizes key protection, using multiple layers of protection to protect secret keys on chip from physical and analytical attacks.

The validity of data and commands flowing in the smart grid is not the only avenue of attack to disrupt the supply of electricity. Clever viruses such as Stuxnet have proven the danger of attacks that change the fundamental behavior of embedded equipment in a manner that is difficult to detect. A class of threats called "zero-day attacks" exploits systems that can be erased or reprogrammed, thus breaking a system in undetectable ways. We need not only worry about equipment when it is deployed, but anytime when it is vulnerable to improper programming (such as during manufacturing).

But What Could Go Wrong?

Designing for security is difficult, time consuming, and requires security expertise. Is the investment really worth it? For a moment let's consider a deployed smart meter. Since meters are not generally protected as they sit on our homes, it is easy for an outsider to gain access. If a conventional microcontroller is used for applications and communication processing in that meter, it is likely that there is an attack path through the programming interface, such that the attacker could reprogram the meter or even read out its contents. With enough resources and time, someone could even create a program that behaves exactly like the previous meter program, but with hidden viruses that collect key data or alter the reporting of electricity consumption.

Deployed meters must be protected to ensure their functions cannot be altered. However, if we look backwards in time, we see a moment when the meter is even more vulnerable—the manufacturing floor. There is always the possibility that "social engineering" can give attackers access to your IP and the manufacturing flow. Armed with a few thousand dollars, and attacker could procure your software, reverse engineer it, alter it, and provide a new program to the manufacturing flow. Additionally, the attacker could sell the software to a competitor, giving another company an unfair benefit from your research and design expenses.

How Do We Secure the Life Cycle?

A conscientious life cycle design will consider threats at every step of product development and manufacturing, and determine if those threats warrant countermeasures. To implement a secure life cycle, consider the following:

1. Make sure you procure valid silicon. Purchasing through authorized or direct channels can help with this, but there are cryptographic techniques as well. Maxim sells secure microcontrollers and smart grid products that can be preprogrammed with a customer's key or certificate, ensuring that only the intended customer can unlock and program that IC.

2. Protect your IP. Deliver signed, encrypted code to your manufacturing operation. This requires cooperation from a secure bootloader inside your system microcontroller to decrypt and authenticate the software once delivered to the chip. The encryption protects against reverse engineering or cloning.
3. Only run the code you intended to run. A secure bootloader can use the digital signature on your software to validate the authenticity of the code before loading or running the application.
4. Trust who you are communicating with. New configurations, firmware updates, and commands should be encrypted and signed to validate that they are issued from a trusted source.
5. Protect your keys in the field. Don't store encryption keys in a separate IC from where you will use the key, such as an external EEPROM. If you have a separate secure microcontroller and applications processor, keep the keys in the secure chip and never send them anywhere else. Keys transmitted across PCB traces are easy for an attacker to extract.
6. Protect your keys inside your company. Use development keys for engineering to design security features into your products. Protect access to the production keys by requiring multiple users to authorize the use of production keys. A high-security module (HSM) can help implement some of these policies.
7. Don't rely on a single point of failure. If an attacker only needs to extract keying material from one meter to break the system, they can invest more time and money into that attack knowing that they can then break the entire system. Sophisticated attackers might even decapsulate the IC package and microprobe memories in search of keying material. Use unique keys or use asymmetric cryptographic schemes like elliptic curve digital signatures.

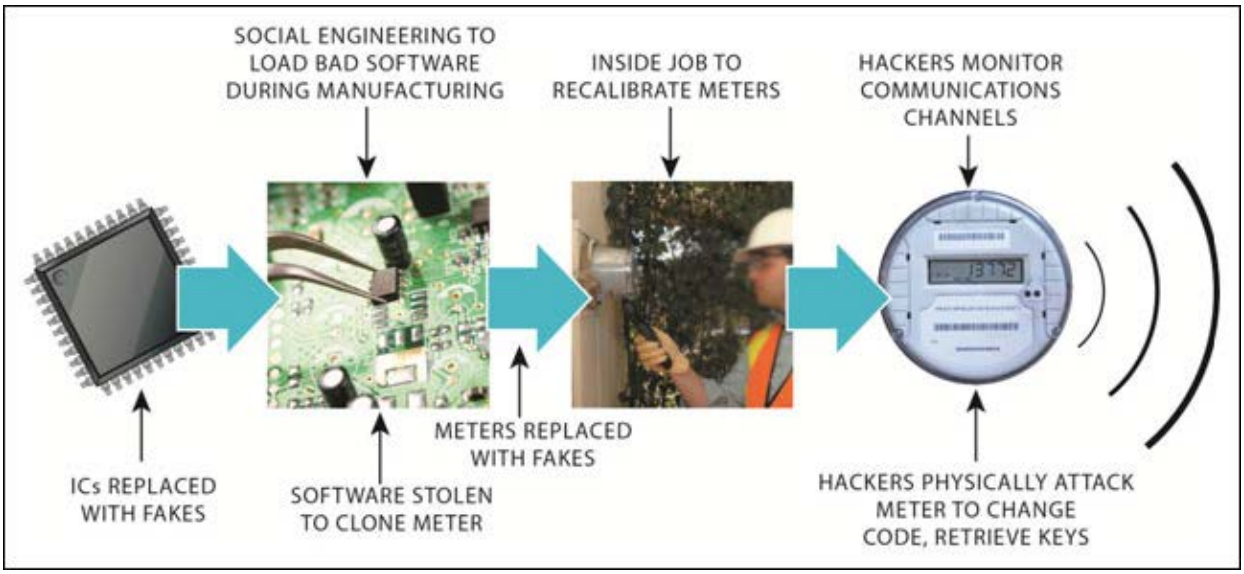

Figure 1. A conscientious life-cycle design will consider threats at every step of product development and manufacturing.

Given today's focus on IT-only security, we are leaving a wide opportunity for attackers to exploit the smart grid. By securing the life cycle of this embedded equipment, we will improve the security of the grid and provide a greater challenge for attackers who wish to disrupt the flow of electricity.

Related Parts

71M6541F	Energy Meter ICs	Free Samples
71M6542F	Energy Meter ICs	Free Samples
71M6543F	Energy Meter ICs	Free Samples
MAXQ1050	DeepCover Secure Microcontroller with USB and Hardware Cryptography	

More Information
For Technical Support: http://www.maximintegrated.com/support
For Samples: http://www.maximintegrated.com/samples
Other Questions and Comments: http://www.maximintegrated.com/contact

Application Note 5486: http://www.maximintegrated.com/an5486
TUTORIAL 5486, AN5486, AN 5486, APP5486, Appnote5486, Appnote 5486