Recapturing Compromised Smart Grid Devices in Large-Scale Attacks

Sebastian Götte <srsrst@jaseg.net> @HIIG Berlin

October 14 2019

1 Problem Statement

After much excitement in both academia and industry, the rollout of "smart" electricity meters is well underway today. From online materials we observe that these systems tend to be country-specific systems which are rolled out at massive scale. Often, this results a near-monoculture. All of these systems contain highly complex communications interfaces such as powerline communications (PLC), DSL or cellular. Many of these "metering" systems additionally include a load switch to disconnect non paying subscribers. Since smart meters are fairly expensive at O(€100) for the device in addition to high installation costs, their expected lifetime is measured in decades.

To a security researcher, these circumstances pose a conundrum. What one has is an IP-connected device that can turn off someone's electricity, that is produced by a small to medium-sized business and that is supposed to run for decades without being hacked.

Experience shows that even large megacorporations have difficulty maintaining software for just a couple of years. At the same time, flawless software does not exist. Even with utmost care and ulimited resources, and in comparatively simple firmware, serious security flaws cannot be ruled out. As a case in point, Apple recently had to see itself confronted with a very embarassing bug inside the first ROM bootloader stage of the secure boot chain used in most iPhones currently in use. This bug allows a full compromise of the system on boot. When even Apple with all its resources cannot manage to secure such a fairly unsophisticated component underpinning the security of the entire iPhone ecosystem, what is the Mittelstand to do trying to secure hundreds of kilobytes of code? If Apple cannot afford or manage to secure a few hundred bytes worth of highly critical firmware, how should anyone else?

From a security point of view the systems employed in this "smart" grid infrastructure are too complex for their makers to handle by several orders of magnitude. Their (in internet terms) extremely long life spans make them likely to outlast their manufacturers. The potential for mayhem caused through their load switches makes them an attractive target for state-sponsored attackers.

From a security expert's point of view given the circumstances outlined above taken over tens of years a large-scale compromise of smart grid infrastructure in at least one of the 24 countries participating in the synchronous grid of Continental Europe is likely as long as there is someone trying.

2 Our Solution

Given the inevitability of serious compromise outlined above, and assuming industry and government inertia in continuing the rollout of the current generation of smart meter architecture, the only thing we can still do is damage control. How can we regain control after a large-scale smart grid compromise?

In this project we propose a hardware measure that can be integraded with any smart meter regardless of manufacturer and technology that allows a grid operator to restore large numbers of compromised meters to a known-good firmware image. The grid operator transmits a cryptographically secured reset signal through a modulation of mains frequency that is picked up by the hardware reset controller. The hardware reset controller then resets and re-programs the meter's main application microcontroller with a known-good factory image. This could be either the meter's original factory firmware or a more minimal bootloader designed to allow the electricity companies to re-gain control of the meter outside their usual software update channels.

3 Project Scope and Open Questions

This project consists of three major steps in addition to a nice specification of attacker model and attack scenarios.

> **Q**1 *How would realistic attackers and attack scenarios look like?*

3.1 Figuring out signal transmission

First, we need to assess feasibility and parameters of our proposed signal transmission method.

> **Q**2 *What control do grid operators have over variables such as mains frequency and phase? How does this compare against normal variations?*

With this knowledge, we will calculate the parameters of our communication channel. Given these channel parameters, we will define details such as modulation scheme and error correction. To aid in validation, we will test a mockup of this system on a simulated channel.

> **Q**3 *How robust would this system be against an advanced active attacker, in particular one that has already pwned a couple million smart meters with load switches?*

3.2 Specifying the transmission protocol

After specifying modulation and FEC parameters we need to specify communication protocol and cryptographic details. We will likely have a highly constrained bitrate, so our overall protocol and cryptographic implementation must be highly efficient in transmission size. All interfaces (modulation, protocol and cryptography included) must be carefully specified and validated to reduce the likelihood of errors at this step. The cryptographic protocol should ideally be formally proven. The overall system of modulation, FEC and cryptographic protocol should be analyzed w.r.t. bit error rate and the resulting expected failure rate.

3.3 Building a hardware prototype

To demonstrate overall viability, a hardware prototype will be constructed. This will be based on a smart meter reference design of a major semiconductor manufacturers.

> **Q**4 *How can we simulate the electric grid, as well as our proposed modulation thereof in this demo setup? Gasoline generator? DDS signal generator plus car audio amplifier plus toroidal halogen transformer?*