Humboldt-Universität zu Berlin
Mathematisch-Naturwissenschaftliche Fakultät
Institut für Informatik

FIXME

Masterarbeit

zur Erlangung des akademischen Grades
Master of Science (M. Sc.)

eingereicht von: Jan Sebastian Götte
geboren am: Aus datenschutzrechtlichen Gründen nicht abgedruckt
geboren in: Aus datenschutzrechtlichen Gründen nicht abgedruckt

Gutachter/innen: Prof. Dr. Björn Scheuermann
FIXME

eingereicht am: ..... verteidigt am: .....

Selbständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und noch nicht für andere Prüfungen eingereicht habe. Sämtliche Quellen einschließlich Internetquellen, die unverändert oder abgewandelt wiedergegeben werden, insbesondere Quellen für Texte, Grafiken, Tabellen und Bilder, sind als solche kenntlich gemacht. Mir ist bekannt, dass bei Verstößen gegen diese Grundsätze ein Verfahren wegen Täuschungsversuchs bzw. Täuschung eingeleitet wird.

Berlin, den 31.03.2020 .....

Contents

Chapter 1

Introduction

1.1 Structure and operation of the electrical grid

1.1.1 Structure of the electrical grid

Generators and loads

Transformers

Tie lines

1.1.2 Operational concerns

Modelling the electrical grid

Generator controls

Load shedding

System stability

Power System Stabilizers

Smart metering

1.2 Smart meter technology

Common components

Smart meters usually are built around a standard microcontroller.

Cryptographic coprocessors

Physical structure

Physical installation

1.3 Regulatory frameworks around the world

1.3.1 International standards

1.3.2 Regulations in Europe

1.3.3 The regulatory situation in Germany

1.3.4 The regulatory situation in France

1.3.5 The regulatory situation in the UK

1.3.6 The regulatory situation in Italy

1.3.7 The regulatory situation in northern America

1.3.8 The regulatory situation in Japan

1.3.9 Common themes

1.4 Security in smart grids

The smart grid in practice is nothing more or less than an aggregation of embedded control and measurement devices that are part of a large control system. This implies that all the same security concerns that apply to embedded systems in general also apply to most components of a smart grid in some way. Where programmers have been struggling for decades now with input validation[41], the same potential issue raises security concerns in smart grid scenarios as well[46, 39]. Only, in smart grid we have two complicating factors present: Many components are embedded systems, and as such inherently hard to update. Also, the smart grid and its control algorithms act as a large (partially-)distributed system, making problems such as input validation or authentication difficult to implement[3] and adding a host of distributed systems problems on top[38].

Given that the electrical grid is a major piece of essential infrastructure in modern civilization, these problems amount to significant issues in practice. Attacks on the electrical grid may have grave consequences[39] all the while the long maintenance cycles of various components make the system slow to adapt. Thus, components for the smart grid need to be built to a much higher standard of security than most consumer devices to ensure they live up to well-funded attackers even decades down the road. This requirement intensifies the challenges of embedded security and distributed systems security among others that are inherent in any modern complex technological system.

A point we will not consider in much depth is theft of electricity. A large part of the motivation of the introduction of smart meters seems to be to reduce the level of fraud by consumers. Academic papers tend to either focus on other benefits such as generation efficiency gains through better forecasting or try to rationalize the funamentally anti-consumer nature of smart

metering with strenuous claims of "enormous social benefits"[48]. We will entirely focus on grid stability and discard electricity theft in the context of this paper for two reasons: One, billing inaccuracies of electricity companies are of very low urgency compared to grid stability, and the one is a precondition for the other. Two, utility companies can already put strong bounds on the amount of theft by simply cross-refrencing meter readings against trusted readings from upstream sections of the grid. This capability works even without smart meters and only gains speed from smart meters, just as the old exploit of bypassing the meter with a section of wire can't be prevented like this.

Due to these bounds on its volume, electricity theft using smart meter hacking would not scale. Hackers would simply be rooted up one by one with no damage to consumers and very limmited damage to utility companies. Damage in these scenarios would be a far cry from the efficiency of an exponentially growing botnet.

1.4.1 Smart grid components as embedded devices

A fundamental challenge in smart grid implementations is the central role smart electricity meters play. Smart meters are used both for highly-granular load measurement and (in some countries) load switching[57]. Smart electricity meters are effectively consumer devices. They are built down to a certain price point that is measured by the burden it puts on consumers and that is generally fixed by regulatory authorities. This requirement precludes some hardware features such as the use of a standard hardened software environment on a high-powerded embedded system (such as a hypervirtualized embedded linux setup) that would both increase resilience against attacks and simplify updates. Combined with the small market sizes in smart grid deployments [1] this produces a high cost pressure on the software development process for smart electricity meters.

1.4.2 The state of the art in embedded security

Embedded security generally is much harder than security of higher-level systems. This is due to a combination of the unique constraints of embedded devices (hard to update, usually small quantity) and their lack of capabilities (processing power, memory protection functions, user interface devices). Even very well-funded companies continue to have serious problems securing their embedded systems. A spectacular example of this difficulty is the recently-exposed flaw in Apple's iPhone SoC first-stage ROM bootloader[2], that allows a full compromise of any iPhone before the iPhone X. iPhone 8, one of the affected models, is still being manufactured and

[1] Most vendors of smart electricity meters only serve a handful of markets. For the most part, smart meter development cost lies in the meter's software There exist multiple competing standards applicable to various parts of a smart electricity meter. In addition, most countries have their own certification regimen[32]. This complexity creates a large development burden for new market entrants.

[2] Modern system-on-chips integrate one or several CPUs with a multitude of peripherals, from memory and DMA controllers over 3D graphics accelerators down to general-purpose IO modules for controlling things like indicator LEDs. Most SoCs boot from one of several boot devices such as flash memory, ethernet or USB according to a configuration set e.g. by connecting some SoC pins a certain way or set by device-internal write-only fuse bits.

Physically, one of the processing cores of the SoC (usually one of the main CPU cores) is connected such that it is taken out of reset before all other devices, and is tasked with switching on and configuring all other devices of the SoC. In order to run later intialization code or more advanced bootloaders, this core on startup runs a very small piece of code hard-burned into the SoC in the factory. This ROM loader initializes the most basic peripherals such as internal SRAM memory and selects a boot device for the next bootloader stage.

sold by Apple today[3]. In another instance, Samsung put a flaw in their secure-world firmware used for protection of sensitive credentials in their mobile phone SoCs in If both of these very large companies have trouble securing parts of their secure embedded software stacks measuring a mere few hundred bytes in Apple's case or a few kilobytes in Samsung's, what is a smart electricity meter manufacturer to do? For their mass-market phones, these two companies have R&D budgets that dwarf some countries' national budgets.

Since thorough formal verification of code is not yet within reach for either large-scale software development or code heavy in side-effects such as embedded firmware or industrial control software[47] the two most effective measures for embedded security is reducing the amount of code on one hand, and labour-intensively checking and double-checking this code on the other hand. A smart electricity manufacturer does not have a say in the former since it is bound by the official regulations it has to comply with, and will almost certainly not have sufficient resources for the latter.

1.4.3 Attack avenues in the smart grid

If we model the smart grid as a control system responding to changes in inputs by regulating outputs, on a very high level we can see two general categories of attacks: Attacks that directly change the state of the outputs, and attacks that try to influence the outputs indirectly by changing the system's view of its inputs. The former would be an attack such as one that shuts down a power plant to decrease generation capacity. The latter would be an attack such as one that forges grid frequency measurements where they enter a power plant's control systems to provoke increasing oscillation in the amount of power generated by the plant according to the control systems' directions.

Communication channel attacks

Communication channel attacks are attacks on the communication links between smart grid components. This could be attacks on IP-connected parts of the core network or attacks on shared busses between smart meters and IP gateways in substations. Generally, these attacks can be mitigated by securing the aforementioned communication links using modern cryptography. IP links can be protected using TLS, and more low-level busses can be protected using more lightweight Noise-based protocols. Cryptographic security transforms an attackers ability to manipulate communication contents into a mere denial of service attack. Thus, in addition to cryptographic security safety under DoS conditions must be ensured to ensure continued system performance under attacks. This safety property is identical with the safety required to withstand random outages of components, such as communications link outages due to physical damage from storms, flooding etc. In general, attacks at the meter level may be hard to weaponize since meters are used mostly for billing and forecasting purposes and for more critical grid control purposes there exist several additional layers of sensors above smart meters that limit how much an attacker can falsify smart meter readings without the manipulation being obvious. In order for an attack to have more far-reaching consequences the attacker would need to compromise additional grid infrastructure[35, 36].

Apple's ROM loader performs some authorization checks, to ensure no unauthorized software is loaded. The present flaw allows an attacker to circumvent these checks, booting code not authorized by Apple on a USB-connected iPhone, compromising Apple's chain of trust from ROM loader to userland right at its root.

[3] i.e. at the time this paragraph was written, on

Exploiting centralized control systems

The type of smart grid attack most often cited in popular discourse, and to the author's knowledge the only type that has so far been conducted in practice, is a direct attack on centralized control systems. In this attack, computer components of control systems are compromised by the same techniques used to compromise any other kind of computer system such as exploiting insecure services running on internet-exposed ports and using one compromised system to compromised other systems connected with it through an ostensably secure internal network. These attacks are very powerful as they yield the attacker direct control over whatever outputs the control systems are controlling. If an attacker manages to compromise a power stations control computers, they may be able to influence generation output or even cause an emergency shutdown.

Despite their potentially large impact, these attacks are only moderately interesting from a scientific perspective. For one, their mitigation mostly consists of a straightforward application of security practices well-known for decades. Though there is room for the implementation of genuinely new, application-specific security systems in this field, the general state of the art is lacking behind the rest of the computer industry such that the low-hanging fruit should take priority.

In addition, given political will these systems can readily be secured since there is only a comparatively small number of them and driving a technician to every one of them in turn to install some security update is perfectly feasible.

Control function exploits

Control function exploits are attacks on the mathematical control loops used by the centralized control system. One example of such an attack would be resonance attacks as described in Wu et al. [55]. In this kind of attack, inputs from peripheral sensors indicating grid load to the centralized control system are carefully modified to cause a disproportionally large oscillation in control system action. This type of attack relies on complex resonance effects that arise when mechanical generators are electrically coupled. These resonances, coloquially called "modes" are well-studied in power system engineering[49, 31, 29]. Even disregarding modern attack scenarios, for stability electrical grids are designed with measures in place to dampen any resonances inherent to grid structure. Still, requiring an accurate grid model these resonances are hard to analyze and unlikely to be noiticed under normal operating conditions.

Mitigation of these attacks is most easily done by on the one hand ensuring unmodified sensor inputs to the control systems in the first place, and on the other hand carefully designing control systems not to exhibit exploitable behavior such as oscillations.

Endpoint exploits

One rather interesting attack on smart grid systems is one exploiting the grid's endpoint devices such as smart electricity meters[4] These meters are deployed on a massive scale, with several thousand meters deployed for every substation. Thus, once compromised restoration to an uncompromised state can be potentially very difficult if it requires physical access to thousands of devices hidden inaccessible in private homes.

By compromising smart electricity meters, an attacker can trivially forge the distributed energy measurements these devices perform. In a best-case scenario, this might only affect billing

[4]Though potentially this could also aim at other kinds of devices distributed on a large scale such as sensors in unmanned substations.

and lead to customers being under- or over-charged if the attack is not noticed in time. However, in a less ideal scenario the energy measurements taken by these devices migth be used to inform the grid centralized control systems and a falsification of these measurements might lead to inefficiency.

In some countries and for some customers, these smart meters have one additional function that is highly useful to an attacker: They contain high-current load switches to disconnect the entire household or business in case electricity bills are left unpaid for a certain period. In countries that use these kinds of systems, the load disconnect is often simply hooked up to one of the smart merter's central microcontroller's general-purpose IO pins, allowing anyone compromising this microcontroller's firmware to actuate the load switch at will.

Given control over a large number of network-connected smart meters, an attacker might thus be able to cause large-scale disruptions of power consumption by repeatedly disconnecting and re-connecting a large number of consumers. Combined with an attack method such as the resonance attack from Wu et al. [55] that was mentioned above, this scenario poses a serious danger to grid stability.

1.4.4 Attacker models in the smart grid

1.4.5 Practical attacks

1.4.6 Practical threats

1.4.7 Conclusion, or why we are doomed

We can conclude that a compromise of a large number of smart electricity meters cannot be ruled out. The complexity of network-connected smart meter firmware makes it exceedingly unlikely that it is in fact flawless. Large-scale deployments of these devices under some circumstances such as where they are used with load disconnect relays make them an attractive target for attackers interested in causing grid instability. The attacker model for these devices very definitely includes enemy states, who have considerable resources at their disposal.

For a reasonable guarantee that no large-scale compromises of hard- and software built today will happen over a span of some decades, we would have to radically simplify its design and limit attack surface. Unfortunately, the complexity of smart electricity meter implementations mostly stems from the large list of requirements these devices have to conform with. Additionally, standards have already been written and changes that reduce scope or functionality have become exceedingly unlikely at this point.

A general observation with smart grid systems of any kind is that they comprise a zealous departure of the decentralized control structure of yesterday's dumb grid and the advent of centralization at an enormous scale. This modern, centralized infrastructure has been carefully designed to defend against malicious actorsand all involved parties have an interest in keeping it secure. Still, like in any other system this centralization also makes a very attractive target for attackers since an attacker can likewise employ this centralized control to their goals. Fundamentally, decentralized systems tend to make attacks of any kind a lot more costly and one might question whether security has truly been gained during smart grid rollout.

Chapter 2

Restoring endpoint safety in an age of smart devices

If as layed out in the previous paragraph we cannot rule out a large-scale compromise of smart energy meters, we have to rephrase our claim to security. If we cannot rule out exploitation, we have to limit its impact. If we assume that we cannot strip any functionality from smart meters since it may be required by standards or for enormous social benefits[48] all we can do is to flush out an attacker once they are in.

In a worst-case scenario an attacker would gain unconstrained code execution e.g. by exploiting a flaw in a network protocol implentation. Since smart meters use standard microcontrollers that do not have advanced memory protection functions (see pg. 1.2), at this point we can assume the attacker has full control over the main microcontroller. With this control they can actuate the load switch if present, transmit data through the device's communication interfaces or use the user interface components such as LEDs and the LCD. Using the self-programming capabilities of modern flash microcontrollers, an attacker may even gain persistency without much trouble. Note that in systems separating cryptographic functions into some form of cryptographic module such as systems used in Germany we can be optimistic and assume the attacker has not in fact compromised this cryptographic co-processor yet and does not have access to any cryptographic secrets yet.

Given that the attacker has complete control over the meter's core microcontroller and given that due to cost constraints we are bound to use whatever microcontroller the meter OEM has chosen for their design, we cannot rely on software running on the core mircocontroller to restore system integrity.

Our solution to this problem is to add another, very small microcontroller to the smart meter design. This microcontroller will contain a small piece of software to receive cryptographically authenticated commands from utility companies and on demand reset the meter's core microcontroller to a known-good state. We have to assume the code in the core controller's flash memory has been compromised, so our only option to flush out an attacker is to re-program the core microcontroller in its entirety. We propose using JTAG to re-program the core microcontroller with a known-good firmware image read from a sufficiently large SPI flash connected to the reset controller. JTAG is supported by most microcontrollers complex enough to end up in a smart meter design and given adequate documentation JTAG programming functionality can be ported to new microcontrollers with relatively little work.

On the microcontroller side our solution requires the JTAG interface to be activated (i.e. not fused-shut) and for our solution to work core microcontroller firmware must not be able

to permanently disable the JTAG interface from within. In microcontrollers that do not yet provide this functionality this is a minor change that could be added to a custom microcontroller variant at low cost. On most microcontrollers keeping JTAG open should not interfere with code readout protection. Code secrecy should be of no concern[51] here but besides security manufacturers have strong preferences about this due to fear of copyright infringement.

2.1 The theory of endpoint safety

In order to gain anything by adding our reset controller to the smart meter's already complex design we must satisfy two interrelated conditions.

1. SECURITY means our reset controller itself does not have any remotely exploitable flaws
2. SAFETY menas our reset controller will perform its job as intended

Note that our SECURITY property includes only remote exploitation, and excludes any form of hardware attack. Even though most smart meters provide some level of physical security, we do not wish to make any assumptions on this. In the following section we will elaborate our attacker model and it will become apparent that sufficient physical security to defend against all attackers in our model would be infeasible, and thus we will design our overall system to remain secure even assuming some number of physically compromised devices.

2.1.1 Attack characteristics

The attacker model these two conditions must hold under is as follows. We assume three angles of attack: Attacks by the customer themselves, attacks by an insider within the metering systems controlling utility company and lastly attacks from third parties. Examples for these third parties are hobbyist hackers or outside cyber-criminals on the one hand, but also other companies participating in the smart grid infrastructure besides the utility company such as intermediary providers of meter-reading services.

Due to the critical nature of the electrical grid, we have to include hostile state actors in our attacker model. When acting directly, these would be classified as third-party attackers by the above schema, but they can reasonably be expected to be able to assume either of the other two roles as well e.g. through infiltration or bribery. Fraunholz, Duque Anton, and Schotten [30] in their elaboration of their generalized attacker model give some classification of attackers and provide a nice taxonomy of attacker properties. In their threat/capability rating, criminals are still considered to have higher threat rating than state-sponsored attackers. The New York Times reported in 2016 that some states recruit their hacking personnel in part from cyber-criminals. If this report is true, in a worst-case scenario we have to assume a state-sponsored attacker to be the worst of both types. Comparing this against the other attacker types in Fraunholz, Duque Anton, and Schotten [30], this state-sponsored attacker is strictly worse than any other type in both variables. We are left with a highly-skilled, very well-funded, highly intentional and motivated attacker.

Based on the above classification of attack angles and our observations on state-sponsored attacks, we can adapt Fraunholz, Duque Anton, and Schotten [30] to our problem, yielding the following new attacker types:

1. **Utility company insiders controlled by a state actor** We can ignore the other internal threats described in Fraunholz, Duque Anton, and Schotten [30] since an insider cooperating with a state actor is strictly worse in every respect.

2. **State-sponsored external attackers** A state actor can obviously directly attack the system through the internet.

3. **Customers controlled by a state actor** A state actor can very well compromise some customers for their purposes. They might either physically infiltrate the system posing as legitimate customers, or they might simply deceive or bribe existing customers into cooperation.

4. **Regular customers** Though a hostile state actor might gain control of some number of customers through means such as voluntary cooperation, bribery, infiltration, they are limited in attack scale since they do not want to arouse premature attention. Though regular customers may not have the motivation, skill or resources of a state-sponsored attacker, potentially large numbers of them may try to attack a system out of financial incentives. To allow for this possibility, we consider regular customers separate from state actors posing as customers in some way.

2.1.2 Overall structural system security

Considering overall security, we first introduce the *reset authority*, a trusted party acting as the single authority for issuing reset commands in our system. In practice this trusted party may be part of the utility company, part of an external regulatory body or a hybrid setup requiring both to cooperate. We assume this party will be designed to be secure against all of the above attacker types. The precise design of this trusted party is out of scope for this work but we will list some practical suggestions on how to achieve security below.

Using an asymmetric cryptographic design centered around the *reset authority*, we rule out all attacks except for denial-of-service attacks on our system by any of the four attacker types. All reset commands in our system originate from the *reset authority* and are cryptographically secured to provide authentication and tamper detection. Under this model, attacks on the electrical grid components between the *reset authority* and the customer device degrade into man-in-the-middle attacks. To ensure the SAFETY criterion from 2.1 holds we must make sure our cryptography is secure against man-in-the-middle attacks and we must try to harden the system against denial-of-service attacks by the attacker types listed above. Given our attacker model we cannot fully guard against this sort of attack but we can at least choose a commmunication channel that is resilient against denial of service attacks under the above model.

Finally, we have to consider the issue of hardware security. We will solve the problem of physical attacks on some small number of devices by simply not programming any secret information into these devices. This also simplifies hardware production. From consideration in this work we explicitly rule out any form of supply-chain attack as out-of-scope.

2.1.3 Complex microcontroller firmware

The SECURITY property from 2.1 is in a large part reliant on the security of our reset controller firmware. The best method to increase firmware security is to reduce attack surface by limiting

external interfaces as much as possible and by reducing code complexity as much as possible. If we avoid the complexity of most modern microcontroller firmware we gain another benefit beyond implicitly reduced attack surface: If the resulting design is small enough we may attempt formal verification of our security property. Though formal verification tools are not yet suitable for highly complex tasks they are already barely adequate for small amounds of code and simple interfaces.

2.1.4 Modern microcontroller hardware

Microcontrollers have gained enormously in both performance/efficiency as well as in peripheral support. Alas, these gains have largely been driven by insatiable customer demand for faster, more powerful chips and for a long time security has not been considered important outside of some specific niches such as smartcards. Traditionally a microcontroller would spend its entire lifetime without ever being exposed to any networks. Though this trend has been reversing with the increasing adoption of internet-of-things things and more advanced security features have started appearing in general-purpose microcontrollers, most still lack even basic functionality found in processors for computers or smartphones.

One of the components lacking from most microcontrollers is strong memory protection or even a memory mapping unit as it is found in all modern computer processors and SoCs for applications such as smartphones. Without an MPU/MPU some mitigations for memory safety violations cannot be implemented. This and the absence of virtualization tools such as ARM's TrustZone make hardening microcontroller firmware a big task. It is very important to ensure memory safety in microcontroller firmware through tools such as defensive coding, extensive testing and formal verification.

In our design we achieve simplicity on two levels: One, we isolate the very complex metering firmware from our reset controller by having both run on separate microcontrollers. Two, we keep the reset controller firmware itself extremely simple to reduce attack surface there.

2.1.5 Regulatory and economical constraints

2.1.6 Safety vs. Security: Opting for restoration instead of prevention

2.1.7 Technical outline of a safety reset

2.2 Communication channels on the grid

2.2.1 Powerline communication systems and their use

2.2.2 Proprietary wireless systems

2.2.3 Landline IP

2.2.4 IP-based wireless systems

2.2.5 Frequency modulation as a communication channel

The frequency dependance of grid frequency

Control systems coupled to grid frequency

Avoiding dangerous modes

Overall system parameters

An outline of practical implementation

2.3 From grid frequency to a reliable communications channel

2.3.1 Channel properties

2.3.2 Modulation and its parameters

2.3.3 Error-correcting codes

2.3.4 Cryptographic security

Chapter 3

Practical implementation

3.1 Cryptographic validation

3.2 Data collection for channel validation

3.2.1 Frequency sensor hardware design

3.2.2 Frequency sensor measurement results

3.3 Channel simulation and parameter validation

3.4 Implementation of a demonstrator unit

3.5 Experimental results

3.6 Lessons learned

Chapter 4

Future work

4.1 Technical standardization

The description of a safety reset system provided in this work could be translated into a formalized technical standard with relatively low effort. Our system is very simple compared to e.g. a full smart meter communication standard and thus can conceivably be described in a single, concise document. The much more complicated side of standardization would be the standardization of the backend operation including key management, coordination and command authorization.

4.2 Regulatory adoption

Since the proposed system adds significant cost and development overhead at no immediate benefit to either consumer or utility company it is unlikely that it would be adopted voluntarily. Market forces limit what long-term planning utility companies can do. An advanced mitigation such as this one might be out of their reach on their own and might require regulatory intervention to be implemented. To regulatory authorities a system such as this one provides a powerful primitive to guard against attacks. Due to the low-level approach our system might allow a regulatory authority to restore meters to a safe state without the need of fine-grained control of implementation details such as application network protocols.

A regulatory authority might specify that all smart meters must use a standardized reset controller that on command resets to a minimal firmware image that disables external communication, continues basic billing functions and enables any disconnect switches. This system would enable the *reset authority* to directly preempt a large-scale attack irrespective of implementation details of the various smart meter implementations.

Cryptographic key management for the smart reset system is not much different to the management of highly privileged signing keys as they are used in many other systems already. If the safety reset system is implemented with a regulatory authority as the *reset authority* they would likely be able to find a public entity that is already managing root keys for other government systems to also manage safety reset keys. Availability and security requirements of safety reset keys do not differ significantly from those for other types of root keys.

4.3 Practical implementation

4.4 Zones of trust

In our design, we opted for a safety reset controller in form of a separate micocontroller entirely separate from whatever application microcontroller the smart meter design is already using. This design nicely separates the meter into an untrusted application (the core microcontroller) and the trusted reset controller. Since the interface between the two is simple and logically one-way, it can be validated to a high standard of security.

Despite these security benefits, the cost of such a separate hardware device might prove high in a mass-market rollout. In this case, one might attempt to integrate the reset controller into the core microcontroller in some way. Primarily, there would be two ways to accomplish this. One is a solution that physically integrates an additional microcontroller core into the main application microcontroller package either as a submodule on the same die or as a separate die in a multi-chip module (MCM) with the main application microcontroller. A full-custom solution integrating both on a single die might be a viable path for very large-scale deployments, but will most likely be too expensive in tooling costs alone to justify its use. More likely for a medium- to large-scale deployment (millions of meters) would be a MCM integrating an off-the-shelf smart metering microcontroller die with the reset controller running on another, much smaller off-the-shelf microcontroller die. This solution might potentially save some cost compared to a solution using a discrete microcontroller for the reset controller.

The more likely approach to reducing cost overhead of the reset controller would be to employ virtualization technologies such as ARM's TrustZone in order to incorporate the reset controller firmware into the application firmware on the same chip without compromising the reset controller's security or disturbing the application firmware's operation.

TrustZone is a virtualization technology that provides a hardware-assisted privileged execution domain on at least one of the microcontrollers cores. In traditional virtualization setups a privileged hypervisor is managing several unprivileged applications sharing resources between them. Separation between applications in this setup is longitudinal between adjacent virtual machines. Two applications would both be running in unprivileged mode sharing the same cpu and the hypervisor would merely schedule them, configure hardware resource access and coordinate communication. This longitudinal virtualization simplifies application development since from the application's perspective the virtual machine looks very similar to a physical one. In addition, in general this setup reciprocally isolates two applications with neither one being able to gain control over the other.

In contrast to this, a TrustZone-like system in general does not provide several application virtual machines and longitudinal separation. Instead, it provides lateral separation between two domains: The unprivileged application firmware and a privileged hypervisor. Application firmware may communicate with the hypervisor through defined interfaces but due to TrustZone's design it need not even be aware of the hypervisor's existence. This makes a perfect fit for our reset controller. The reset controller firmware would be running in privileged mode and without exposing any communication interfaces to application firmware. The application firmware would be running in unprivileged mode without any modification. The main hurdles to the implementation to a system like this are the requirement for a microcontroller providing this type of virtualization on the one hand and the complexity of correctly employing this virtualization on the other hand. Virtualization systems such as TrustZone are still orders of magnitude more complex to correctly configure than it is to simply use separate hardware and secure the

interfaces in between.

Chapter 5

Conclusion

Appendix A

Acknowledgements

Appendix B

References

Bibliography

[1] Damminda Alahakoon and Xinghuo Yu. "Smart Electricity Meter Data Intelligence for Future Energy Systems: A Survey". In: *IEEE Transactions on Industrial Informatics* (2015). DOI: 10.1109/TII.2015.2414355. URL: http://ieeexplore.ieee.org.sci-hub.tw/abstract/document/7063262 (visited on).

[2] Saurabh Amin et al. "Game-Theoretic Models of Electricity Theft Detection in Smart Utility Networks". In: *IEEE Control Systems Magazine* 35 (Feb. 2015). DOI: 10.1109/MCS.2014.2364711. URL: https://cloudfront.escholarship.org/dist/prd/content/qt3658w184/qt3658w184.pdf (visited on).

[3] Matt Blaze et al. "The role of trust management in distributed systems security". In: *Secure Internet Programming*. Springer, 1999, pp. 185–210.

[4] Stuart Borlase, ed. *Smart Grids: Advanced Technologies and Solutions*. Electric Power and Energy Engineering. CRC Press, 2017. ISBN: 978-1-4987-9955-3. URL: http://libgen.is/book/index.php?md5=54E49C790BF4ABE66857D6A86E60A196 (visited on).

[5] Bundesamt für Sicherheit in der Informationstechnik. *Marktanalyse zur Feststellung der technischen Möglichkeit zum Einbau intelligenter Messsysteme nach § 30 MsbG*. Tech. rep. Jan. 2019. URL: https://web.archive.org/web/20190919124052/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/SmartMeter/Marktanalysen/Marktanalyse_nach_Para_30_MsbG.pdf?__blob=publicationFile&v=8 (visited on).

[6] Bundesamt für Sicherheit in der Informationstechnik. *Technische Richtlinie BSI TR-03109*. Bundesamt für Sicherheit in der Informationstechnik. Nov. 2015. URL: https://web.archive.org/web/20190919102010/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR03109.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=3 (visited on).

[7] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage I: CMS-Datenformat für die Inhaltsdatenverschlüsselung und -signatur*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919104234/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1%5C_Anlage%5C_CMS.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2%5C_cid351?%5C_%5C_blob=publicationFile%5C&v=2 (visited on).

[8] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage II: COSEM/HTTP Webservices*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2012. URL: https://web.archive.org/web/20190919104234/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-

1%5C_Anlage%5C_CMS.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2%5C_cid351?%5C_%5C_blob=publicationFile%5C&v=2 (visited on).

[9] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage III: Feinspezifikation "Drahtlose LMN-Schnittstelle" Teil a: "OMS Specification Volume 2, Primary Communication"*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919110054/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Feinspezifikation_Drahtlose_LMN-Schnittstelle.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=2 (visited on).

[10] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage III: Feinspezifikation "Drahtlose LMN-Schnittstelle" Teil b: "OMS Technical Report Security"*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919110101/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Feinspezifikation_Drahtlose_LMN-Schnittstelle-Teil2.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=2 (visited on).

[11] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage IV: Feinspezifikation "Drahtgebundene LMN-Schnittstelle" Teil a: "HDLC für LMN"*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919110101/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Feinspezifikation_Drahtlose_LMN-Schnittstelle-Teil2.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=2 (visited on).

[12] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage IV: Feinspezifikation "Drahtgebundene LMN-Schnittstelle" Teil b: "SML Smart Message Language"*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919110756/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1%5C_Anlage%5C_Feinspezifikation%5C_Drahtgebundene%5C_LMN-Schnittstelle%5C_Teilb.pdf%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2%5C_cid351?%5C_%5C_blob=publicationFile%5C&v=2 (visited on).

[13] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage VI: Betriebsprozesse*. Bundesamt für Sicherheit in der Informationstechnik. Mar. 2013. URL: https://web.archive.org/web/20190919111203/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Betriebsprozesse.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=1 (visited on).

[14] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1 Anlage VII: Interoperabilitätsmodell und Geräteprofile für Smart-Meter- Gateways*. Bundesamt für Sicherheit in der Informationstechnik. Jan. 2019. URL: https://web.archive.org/web/20190919111350/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-1_Anlage_Interop-

Modell-Geraeteprofile.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=2 (visited on).

[15] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-1: Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems*. Bundesamt für Sicherheit in der Informationstechnik. Jan. 2019. URL: https://web.archive.org/web/20190919102217/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR03109-1.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=3 (visited on).

[16] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-2 Anhang A: Smart Meter Gateway Sicherheitsmodul Use Cases*. Bundesamt für Sicherheit in der Informationstechnik. Dec. 2014. URL: https://web.archive.org/web/20190919111540/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-2-Sicherheitsmodul_Use_Cases.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=2 (visited on).

[17] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-2 Anhang B: Smart Meter Mini-HSM Anforderungen an die Funktionalität und Interoperabilität des Sicherheitsmoduls*. Bundesamt für Sicherheit in der Informationstechnik. June 2017. URL: https://web.archive.org/web/20190919111832/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-2_Anhang_B_Smart_Meter_Mini_HSM.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=3 (visited on).

[18] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-2: Smart Meter Gateway - Anforderungen an die Funktionalität und Interoperabilität des Sicherheitsmoduls*. Bundesamt für Sicherheit in der Informationstechnik. Dec. 2014. URL: https://web.archive.org/web/20190919102644/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-2-Anforderungen_an_die_Funktionalitaet.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=3 (visited on).

[19] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-3: Kryptographische Vorgaben für die Infrastruktur von intelligenten Messsystemen*. Bundesamt für Sicherheit in der Informationstechnik. Apr. 2014. URL: https://web.archive.org/web/20190919102648/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-3_Kryptographische_Vorgaben.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=1 (visited on).

[20] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-4: Public Key Infrastruktur für Smart Meter Gateways*. Bundesamt für Sicherheit in der Informationstechnik. Aug. 2017. URL: https://web.archive.org/web/20190919102649/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-4_PKI.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=3 (visited on).

[21] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-6: Smart Meter Gateway Administration*. Bundesamt für Sicherheit in der Informationstechnik. Nov. 2015. URL: https://web.archive.org/web/20190919102651/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-6-Smart_Meter_Gateway_Administration.pdf;%20jsessionid=BD197BE4CB44C76EE792_cid351?__blob=publicationFile&v=4 (visited on).

[22] Bundesamt für Sicherheit in der Informationstechnik. *TR-03109-TS-1: Testkonzept zu BSI TR-03109-1*. Bundesamt für Sicherheit in der Informationstechnik. Jan. 2015. URL: https://web.archive.org/web/20190919112310/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03109/TR-03109-TS-1_Testkonzept.pdf;%20jsessionid=BD197BE4CB44C76EE7945640B8703844.2_cid351?__blob=publicationFile&v=1 (visited on).

[23] Bundesamt für Sicherheit in der Informationstechnik. *TR-03116-3: Intelligente Messsysteme*. Bundesamt für Sicherheit in der Informationstechnik. Jan. 2019. URL: https://web.archive.org/web/20190919112052/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03116/BSI-TR-03116-3.pdf;%20jsessionid=CB56FC0D3137C5624CA697AB9E57671F.1_cid360?__blob=publicationFile&v=9 (visited on).

[24] Bundesamt für Sicherheit in der Informationstechnik. *TR-Prüfstellen: Anforderungen an Antragsteller zur Anerkennung als Prüfstelle im Bereich Technischer Richtlinien*. Bundesamt für Sicherheit in der Informationstechnik. Jan. 2019. URL: https://web.archive.org/web/20190919112552/https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/TR-Pruefstellen.pdf;%20jsessionid=A6B4CB8AD2C038741C656276CE874B2_cid369?__blob=publicationFile&v=10 (visited on).

[25] Bundesamt für Sicherheit in der Informationstechnik and Bundesministerium für Wirtschaft und Energie. *Standardisierungsstrategie zur sektorübergreifenden Digitalisierung nach dem Gesetz zur Digitalisierung der Energiewende*. Jan. 2019. URL: https://web.archive.org/web/20190919100713/https://www.bmwi.de/Redaktion/DE/Downloads/S-T/standardisierungsstrategie.pdf?__blob=publicationFile&v=4 (visited on 09/19/2019).

[26] Bundesnetzagentur. *Smart Meter*. 2019. URL: https://web.archive.org/web/20190919100204/https://www.bundesnetzagentur.de/DE/Sachgebiete/ElektrizitaetundGas/Verbraucher/NetzanschlussUndMessung/SmartMetering/SmartMeter_node.html (visited on 09/19/2019).

[27] R. Czechowski and A. M. Kosek. "The most frequent energy theft techniques and hazards in present power energy consumption". In: *2016 Joint Workshop on Cyber- Physical Security and Resilience in Smart Grids (CPSR-SG)*. IEEE. Apr. 2016, pp. 1–7. DOI: 10.1109/CPSRSG.2016.7684098. URL: https://project-sparks.eu/wp-content/uploads/2016/04/czechowski-cpsr-sg-paper-four.pdf.

[28] Mathias Dalheimer. *Smartin Meter-Einführung Deutschland*. URL: https://entropia.de/images/2/2c/GPN14-SmartMeterEinf%C3%BChrung.pdf.

[29] ENTSO-E System Protection Dynamics and WG. *Oscillation Event 03.12.2017*. Tech. rep. Mar. 2018. URL: https://docstore.entsoe.eu/Documents/SOC%20documents/Regional_Groups_Continental_Europe/OSCILLATION_REPORT_SPD.pdf.

[30] *Introducing GAMfIS: A Generic Attacker Model for Information Security*. IEEE, Nov. 2017. URL: https://doi.org/10.23919/SOFTCOM.2017.8115550.

[31] *Low Frequency Oscillations in the Interconnected System of Continental Europe*. IEEE, Aug. 2010. DOI: 10.1109/PES.2010.5589932{\textperiodcentered}.

[32] The CEN/CENELEC/ETSI Joint Working Group Standards Smart on for Grids. *Final report of the CEN/CENELEC/ETSI Joint Working Group on Standards for Smart Grids*. Tech. rep. May 2011.

[33] Vehbi C. Güngör et al. "Smart Grid Technologies: Communication Technologies and Standards". In: *IEEE Transactions on Industrial Informatics* 7.4 (Nov. 2011), pp. 529–539. URL: https://www.researchgate.net/profile/Salih_Ergut/publication/224257498_Smart_Grid_Technologies_Communication_Technologies_and_Standards/links/56ccb4e508ae85c8233bc062/Smart-Grid-Technologies-Communication-Technologies-and-Standards.pdf.

[34] Yasin Kabalci. "A survey on smart metering and smart grid communication". In: *Renewable and Sustainable Energy Reviews* 57 (2016), pp. 302–318. DOI: 10.1016/j.rser.2015.12.114. URL: https://www.researchgate.net/profile/Yasin_Kabalci/publication/289504234_A_survey_on_smart_metering_and_smart_grid_communication/links/5a6105aaaca272a1581745c1/A-survey-on-smart-metering-and-smart-grid-communication.pdf.

[35] Jinsub Kim and Lang Tong. "On Topology Attack of a Smart Grid: Undetectable Attacks and Countermeasures". In: *IEEE Journal on Selected Areas in Communications* 31.7 (July 2013). DOI: 10.1109/JSAC.2013.130712.

[36] Oliver Kosut et al. "Malicious Data Attacks on the Smart Grid". In: *IEEE Transactions on Smart Grid* 2.4 (Nov. 2011), pp. 645–658.

[37] Andrew E. Kramer. *How the Kremlin Recruited an Army of Specialists for Cyberwar*. 2016.

[38] Leslie Lamport, Robert Shostak, and Marshall Pease. "The Byzantine Generals Problem". In: *ACM Transactions on Programming Languages and Systems* 4.3 (July 1982), pp. 382–401. URL: https://www.microsoft.com/en-us/research/publication/byzantine-generals-problem/?from=http%3A%2F%2Fresearch.microsoft.com%2Fen-us%2Fum%2Fpeople%2Flamport%2Fpubs%2Fbyz.pdf;%20https://doi.org/10.1145%2F357172.357176.

[39] Robert M. Lee, Michael J. Assante, and Tim Conway. "Analysis of the cyber attack on the Ukrainian power grid". In: *Electricity Information Sharing and Analysis Center (E-ISAC)* (2016).

[40] Javier Leiva, Alfonso Palacios, and José A. Aguado. "Smart metering trends, implications and necessities: A policy review". In: *Renewable and Sustainable Energy Reviews* 55 (2016), pp. 227–233. URL: http://kchbi.chtf.stuba.sk/upload_new/file/Miro/Proc%20problemy%20odovzdane%20zadania/Cyprichov%C3%A1/SmartMetering.pdf;%20http://dx.doi.org/10.1016/j.rser.2015.11.002 (visited on).

[41] Nancy G. Leveson and Clark S. Turner. "An Investigation of the Therac-25 Accidents". In: *IEEE Computer* 26.7 (July 1993), pp. 18–41. URL: https://doi.org/10.1109/MC.1993.274940;%20https://web.archive.org/web/20041128024227/http://www.cs.umd.edu/class/spring2003/cmsc838p/Misc/therac.pdf.

[42] Jaime Lloret et al. *An Integrated IoT Architecture for Smart Metering*. IEEE, 2016. (Visited on).

[43] G. Lopez et al. "Paving the road toward Smart Grids through large-scale advanced metering infrastructures". In: *Electric Power Systems Research* (2014). DOI: 10.1016/j.epsr.2014.05.006. URL: http://www.sciencedirect.com.sci-hub.tw/science/article/abs/pii/S0378779614001862 (visited on).

[44] Anzar Mahmood, Nadeem Javaid, and Sohail Razzaq. "A review of wireless communications for smart grid". In: *Renewable and Sustainable Energy Reviews* 41 (2015), pp. 248–260. DOI: 10.1016/j.rser.2014.08.036. URL: http://www.sciencedirect.com.sci-hub.tw/science/article/abs/pii/S1364032114007126 (visited on).

[45] Heise Medien. *checkm8: Boot-Exploit soll neuere iPhones knacken*. URL: https://www.heise.de/mac-and-i/meldung/checkm8-Boot-Exploit-soll-neuere-iPhones-knacken-4542075.html.

[46] Yilin Mo et al. "Cyber-Physical Security of a Smart Grid Infrastructure". In: *Proceedings of the IEEE* 100.1 (Jan. 2012), pp. 195–209. URL: http://ieeexplore.ieee.org.sci-hub.tw/abstract/document/6016202;%20https://ieeexplore.ieee.org/abstract/document/6016202;%20https://www.researchgate.net/profile/Yilin_Mo/publication/224257991_Cyber-Physical_Security_of_a_Smart_Grid_Infrastructure/links/004635395d2f66a584000000.pdf (visited on).

[47] Dillon Pariente and Emmanuel Ledinot. *Formal verification of industrial C code using Frama-C: a case study*. Tech. rep. 2010, pp. 205–219.

[48] McDaniel Patrick and McLaughlin Stephen. "Security and Privacy Challenges in the Smart Grid". In: *Secure Systems* (May 2009).

[49] Graham Rogers. "Power System Oscillations". In: Kluwer, 2000.

[50] Benjamin Schäfer et al. "Decentral Smart Grid Control". In: *New Journal of Physics* 17 (Jan. 2015). DOI: doi:10.1088/1367-2630/17/1/015002.

[51] Bruce Schneier. *Secrecy, Security, and Obscurity*. May 2002. URL: https://www.schneier.com/crypto-gram/archives/2002/0515.html.

[52] Konark Sharma and Lalit Mohan Saini. "Performance analysis of smart metering for smart grid: An overview". In: *Renewable and Sustainable Energy Reviews* 49 (2015), pp. 720–735. DOI: 10.1016/j.rser.2015.04.170. URL: http://www.sciencedirect.com.sci-hub.tw/science/article/abs/pii/S1364032115004402 (visited on).

[53] ST Microelectronics. *STSAFE-J100-BS Data brief. Security module of a smart meter gateway as defined by the BSI*. ST Microelectronics, June 2018. URL: https://www.st.com/resource/en/data_brief/stsafe-j100-bs.pdf (visited on).

[54] Chih-Che Sun, Adam Hahn, and Chen-Ching Liu. "Cyber security of a power grid: State-of-the-art". In: *International Journal of Electrical Power & Energy Systems* 99 (2018), pp. 45–56.

[55] Yongdong Wu et al. "Resonance Attacks on Load Frequency Control of Smart Grids". In: *IEEE Transactions on Smart Grid* 9.5 (Sept. 2018), pp. 4490–4502. DOI: 10.1109/TSG.2017.2661307.

[56] Ye Yan et al. “A Survey on Smart Grid Communication Infrastructures: Motivations, Requirements and Challenges”. In: *IEEE Communications Surveys & Tutorials* (2012). DOI: 10.1109/SURV.2012.021312.00034. URL: http://d-scholarship.pitt.edu/12508/1/Smart_Grid_Infrastructure_Final.pdf.

[57] Jixuan Zheng, David Wenzhong Gao, and Li Lin. “Smart meters in smart grid: An overview”. In: *2013 IEEE Green Technologies Conference (GreenTech)*. IEEE. 2013, pp. 57–64.

[58] Bin Zhou et al. “Smart home energy management systems: Concept, configurations, and scheduling strategies”. In: *Renewable and Sustainable Energy Reviews* 61 (2016), pp. 30–40. URL: http://www.sciencedirect.com.sci-hub.tw/science/article/abs/pii/S1364032116002823.

Appendix C

Demonstrator schematics and code

Appendix D

Economic viability of countermeasures

D.1 Attack cost

D.2 Countermeasure cost