Developing Cyber-Physical Experimental Capabilities for the Security Analysis of the Future Smart Grid

Béla Genge* and Christos Siaterlis

***Abstract*—During the evolution of today's power grid to a Smart Grid it is expected that IP-based communication protocols including Supervisory Control And Data Acquisition (SCADA) systems, will form the basis of communications architecture for substation and distribution automation, advanced metering and home area networking applications. However, this will lead to many Smart Grid security challenges - a forecast that is supported by the vulnerability of current SCADA systems. In this paper we examine how our experimental framework that has been developed for the modeling and simulation of local power plants can be extended and efficiently used for the study of complex wide area environments such as the future Smart Grid. We show that our framework is flexible enough to be easily extended with components for satisfying the requirements of a complex environment as the future Smart Grid. The main contribution of the paper is that it proposes a framework for experimenting with the Smart Grid that can be used by researchers to recreate an experimentation environment for measuring and understanding the consequences of cyber attacks on the Smart Grid. The paper also presents the study of a cyber attack involving compromised control hardware and the IEEE 9-bus system. The results confirm that we can experimentally recreate and study oscillations in the power grid caused by adversaries that attack the system through its IP-based control subsystem.**

***Index Terms*—Cyber-physical, security, experimentation, framework, Smart Grid.**

I. Introduction

THE Smart Grid is evolving rapidly from a relatively isolated environment to an opened one. The adoption of Information and Communication Technologies (ICT) has led to cost optimization as well as greater efficiency, flexibility and interoperability between components. It is forecasted that IP-based networks with IPv6 and Supervisory Control And Data Acquisition (SCADA) [1] will provide the communications architecture for substation and distribution automation, advanced metering and home area networking applications for the future Smart Grid [2]. This will lead to many challenging issues in the security of Smart Grid as current SCADA systems are exposed to significant cyber-threats; a fact that has been highlighted by many studies [3], [4]. For example, the recently discovered Stuxnet worm [5] is the first malware that is specifically designed to attack SCADA systems. Its ability to reprogram the logic of control hardware in order to alter physical processes demonstrated how powerful such threats can be; it has served as a wakeup call for the international security community. Stuxnet raised many open questions, but most importantly it reminded us that we still lack an efficient way to conduct experiments that measure the impact of such threats against cyber-physical systems.

The Smart Grid is a complex system where both physical and cyber realms are present. The study of such systems becomes even a greater challenge as recent studies [8] have shown that the complexity of the future Smart Grid is likely to expand as more requirements are identified. From a technical point of view the study of complex systems such as the Smart Grid could be carried out by experimenting with real systems, software simulators or emulators. Experimentation with production systems suffers from the inability to control the experiment environment in order to reproduce results. Furthermore if the study intends to test the resilience or security of a system, there are obvious concerns about the potential side effects (faults and disruptions) to mission critical services. On the other hand the development of a dedicated experimentation infrastructure with real components is often economically prohibitive and disruptive experiments on top of it could be a risk to safety. Software based simulation has always been considered an efficient approach to study physical systems, mainly because it can offer low-cost, fast and accurate analysis. Nevertheless it has limited applicability in the context of cyber security due to the diversity and complexity of computer networks. Software simulators can effectively model normal operations, but fail to capture the functionality of protocols and computer systems in general. Moreover, with the large number of communication protocols and technologies foreseen to be applicable for the future Smart Grid, such as IPv6 [1], smart objects [22], or even SIP [23], the use simulators might prove to be unfeasible.

In our previous work [6] we have proposed a hybrid approach in between the two extremes of pure simulation and experimentation with only real components. The developed framework uses simulation for the physical components and an emulation testbed based on Emulab [9], [10] in order to recreate the cyber realm, e.g., SCADA servers, corporate network, etc. The models of the physical systems are developed in Matlab Simulink from which the corresponding 'C' code is generated using Matlab Real Time Workshop (Matlab RTW). The generated code is then executed in real time and is able to interact with the real components of the emulation testbed.

In this paper we examine the applicability of our previously

B. Genge and C. Siaterlis are with the Institute for the Protection and Security of the Citizen, Joint Research Centre, European Commission, Italy Via E. Fermi, 2749, Ispra (VA), 21027, Italy
e-mail: {bela.genge, christos.siaterlis}@jrc.ec.europa.eu
* Contact person.

developed framework in the study of Smart Grid security. Matlab Simulink is used to construct a detailed model of the physical realm, e.g. generation, transmission and distribution, while Emulab is used to emulate the cyber realm, e.g. control logic code, network communication protocols. The main contribution of the paper lies in the fact that based on the proposed framework scientists can recreate an experimentation environment for measuring and understanding the consequences of cyber attacks on the Smart Grid while using real cyber components and real malware in a completely safe way, i.e. without the risk of bringing the system into an unstable state. The applicability of the approach is proven through the analysis of a synchronized cyber attack against the IEEE 9-bus system.

The paper is structured as follows. After a short overview of related work in Section II, our experimental framework is presented in Section III. Then, we continue with a discussion on adapting our framework for conducting security studies on the Smart Grid in Section IV. In Section V we present a study of a cyber attack involving the IEEE 9-bus system and we conclude in Section VI.

II. Related Work

This section provides a brief presentation of the most relevant papers on experimentation with cyber-physical systems. An approach that uses real components for the physical parts and partly simulated ones for the cyber parts has been proposed by Chunlei, *et al.* [11]. In this approach the only simulated element is the enterprise network, while all the other components (servers, Programmable Logic Controllers - PLCs, etc.) are real. Although from one point of view such a testbed would provide reliable experimental data, since almost everything is real, it would be hardly able to support tests on large infrastructures such as the *Distribution* or *Transmission* systems of the Power Grid, because that would require a complete implementation of the system to experiment with. Other researchers have focused on simulating both SCADA and field devices. For example, Chabukswar, *et al.* [12] used the Command and Control WindTunnel (C2WindTunnel) [13] multi-model simulation environment, based on the High-Level Architecture (HLA) IEEE standard 1.3 [14], to enable the interaction between various simulation engines. With this approach, analyzing the cyber-physical effects of malware is not a trivial task, as it requires a detailed description of all ICT components and more importantly a detailed knowledge on the dynamics of malware. Davis, *et al.* [15] used PowerWorld [16], to model an entire power grid and run it in real time. The PowerWorld server is connected to a proxy that implements the Modbus protocol and transmits Modbus packages to client applications. However, the approach does not include typical units such as PLCs and SCADA Masters, that are key components in cyber-physical experimental scenarios. Another approach where the PowerWorld server was used to simulate a power grid was proposed by McDonald, *et al.* [17]. Instead of a real network, the authors used the OPNET system for simulating computer networks and real stations for running malware. Although this does not require a simulation of malware, it does require, however the simulation of the interactions between malware and simulated networks, that is not a trivial task.

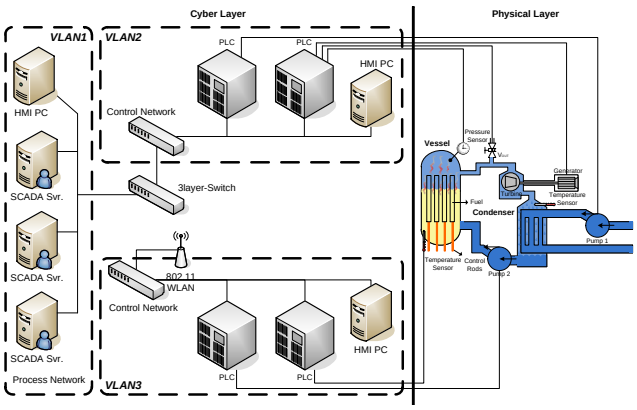

Fig. 1: Example architecture of a typical Industrial Control System

III. Framework Overview

After providing a brief description of a typical Industrial Control System (ICS) architecture, this section presents a short overview of our previously developed experimentation framework.

A. Process Control Architecture Overview

In modern ICS architectures (see Fig. 1), one can identify two different control layers: (i) the physical layer composed of all the actuators, sensors, and generally speaking hardware devices that physically perform the actions on the system (e.g. open a valve, measure the voltage in a cable); (ii) the cyber layer composed of all the ICT devices and software which acquire the data, elaborate low level process strategies and deliver the commands to the physical layer. The cyber layer typically uses SCADA protocols to control and manage the physical devices within the cyber layer. The "distributed control system" of the cyber layer is typically split among two networks: the *control network* and the *process network*. The process network usually hosts all the SCADA (also known as SCADA Masters) and HMI (Human Machine Interface) servers. The control network hosts all the devices which, on the one side control the actuators and sensors of the physical layer and on the other side provide the "control interface" to the process network. A typical control network is composed of a mesh of PLCs (Programmable Logic Controller). From an operational point of view, PLCs receive data from the physical layer, elaborate a "local actuation strategy", and send back commands to the actuators. PLCs execute also the commands that they receive from the SCADA servers (Masters) and additionally provide, whenever requested, detailed physical layer data.

B. Framework Architecture

The previously developed framework [6] follows a hybrid approach, where the Emulab-based testbed recreates the control

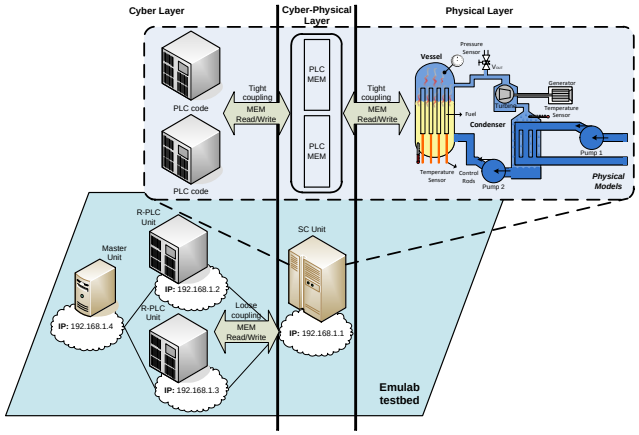

Fig. 2: Experimental framework architectural overview

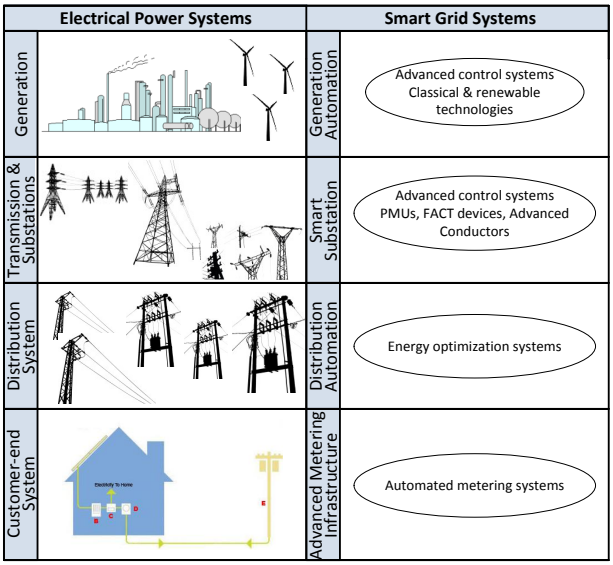

Fig. 3: Major components of the Smart Grid and Electrical Power systems

and process network of SCADA, including PLCs and SCADA servers, and a software simulation reproduces the physical processes. The architecture, as shown in Figure 2, clearly distinguishes 3 layers: the cyber layer, the physical layer and a link layer in between. The cyber layer includes regular ICT components used in SCADA systems, while the physical layer provides the simulation of physical devices. The link layer (i.e. cyber-physical layer) provides the "glue" between the two layers through the use of a shared memory region.

The physical layer is recreated through a soft real time simulator that runs within the SC (Simulation Core) unit and executes a model of the physical system. The simulator's execution time is strongly coupled to the timing service of the underlying operating system (OS). The cyber layer is recreated by an emulation testbed that uses the Emulab architecture and software [7] to automatically and dynamically map physical components (e.g. servers, switches) to a virtual topology. Besides the process network, the cyber layer also includes the control logic code that in the real world is run by PLCs. The control code can be run sequentially or in parallel to the physical model. In the sequential case, a *tightly coupled* code (TCC) is used, i.e. code that is running in the same memory space with the model, within the SC unit. In the parallel case a *loosely coupled* code (LCC) is used, i.e. code that is running in another address space, possibly on another host, within the R-PLC unit (Remote PLC). The main advantage of TCCs is that these do not miss values generated by the model between executions. On the other hand, LCCs allow running PLC code remotely, to inject (malicious) code without stopping the execution of the model, and to run more complex PLC emulators. The unit that implements global decision algorithms based on the sensor values received from the R-PLC units is also present in the proposed framework as the *Master* unit. The cyber-physical layer incorporates the PLC memory, seen as a set of registers typical of PLCs, and the communication interfaces that "glue" together the other two layers. Memory registers provide the link to the inputs (e.g. valve position) and outputs (e.g. sensor values) of the physical model.

Prototypes of SC, R-PLC and Master Units have been developed in C# (Windows) and have been ported and tested on Unix-based systems (FreeBSD, Fedora and Ubuntu) with the use of the *Mono* platform. Matlab Simulink was used as the physical process simulator (physical layer). From Simulink models the corresponding 'C' code is generated using Matlab RTW. The communication between SC and R-PLC units is handled by .NET's binary implementation of RPC (called *remoting*) over TCP. For the communication between the R-PLC and Master units, we used the Modbus over TCP protocol.

IV. Adapting the Framework for Smart Grid Experimentation

In the past, our previously developed framework has been successfully used to study the security of SCADA systems. Nevertheless, its flexibility allows us to extend it with other ICT components, that can lead to the recreation of Smart Grid components shown in Fig. 3. Within this context simulation is used for physical processes (e.g. nuclear reactor, PMU, FACT devices, distribution systems, consumer), while emulation is used for cyber components (e.g. control logic code, network communication protocols).

The result of the integration of Smart Grid components into our framework is illustrated in Fig. 4. Physical components of systems such as *Generation*, *Transmission & Substations*, *Distribution* and *Customer-end* are simulated. Models of the previously mentioned systems are constructed in Simulink, from which the corresponding 'C' code is generated and integrated into our framework. Cyber components such as control logic code and network communication protocols are emulated instead. The emulated components are able to interact with the physical components through the SC unit. This way, researchers can also experiment with real components that interact with emulated and simulated ones.

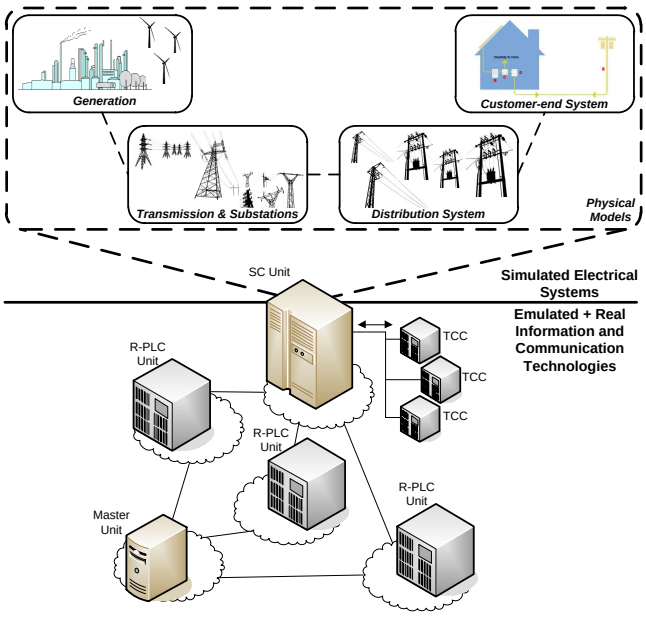

Fig. 4: Integration of Smart Grid components with our framework

The architecture shown in Fig. 4 is not exhaustive in terms of components that the future Smart Grid will incorporate. However, as the Smart Grid is still evolving, identifying all possible components is not possible and also not feasible. Nevertheless, the framework allows designers to expand it easily with new components, making it a great candidate for the study of the future Smart Grid.

From a technical point of view running power grid components in real time requires time-domain analysis capabilities. Matlab packages such as Matpower [18] are well suited for power flow calculations, but do not provide time-based execution. As previously stated, our approach is based on the execution of Simulink models in real time, an approach that enables the interaction with real software components. Fortunately, within the open source community we can already find Matlab packages that enable the real-time execution of power grid models. For our purposes we have found that the MatDyn [19] package is best suited.

MatDyn [19] is an open source Matlab package to perform dynamic analysis of electric power systems. It uses Matpower [18] for power flow calculations and can be easily integrated into Simulink. Considering the limited number of open source solutions we could have adopted [21], the alternative would have been PSAT [20]. PSAT is known to be one of the most complete open source toolboxes for power system analysis. However, the large number of functionalities it provides also represents a challenge for users that need to make changes to the code or need to isolate specific algorithms. In contrast, MatDyn comes with one specific functionality, i.e. time-domain model execution, that already suits our needs while providing a simple code that can be easily extended.

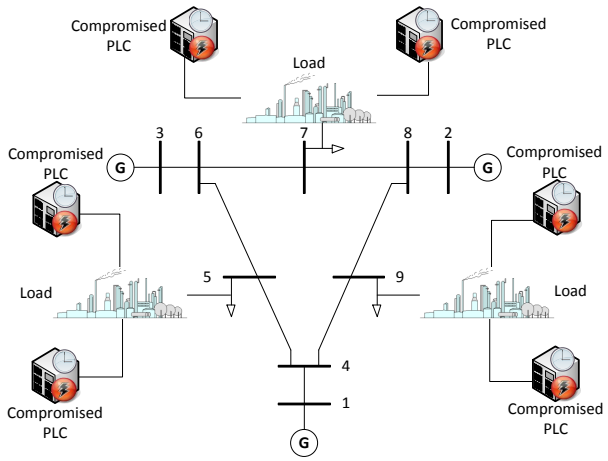

Fig. 5: Attack scenario on the IEEE 9-bus system

V. Study of Synchronized Cyber Attacks Against the Smart Grid

In this section we demonstrate the applicability of our framework to the study of stability, security & resilience of power grids with a view to analyze future Smart Grid implementation options. For this purpose we have implemented a scenario in which the attacker is able to increase the load on a power grid through compromised hardware. The attacker inserts a *logic bomb* into compromised software to initiate the attack once the time conditions are met. This way, the start of the attack can be synchronized and launched at several locations at once, increasing the overall damages to the power grid. The attack is similar to a Distributed Denial of Service attack, already well-known in the field of ICT, in which the attack is launched at a large scale, targeting multiple locations and possibly using thousands of infected stations. Although simple in its implementation, we used it as a first feasibility study and we consider more realistic scenarios as part of our future work.

For the simulated power grid we have used the IEEE 9-bus test system. The attacker is able to compromise PLCs and start physical processes simultaneously at different locations. This in turn causes an increase in the load of the power grid, disturbing its normal operation. The attack scenario is depicted in Fig. 5.

We start our study with the non-synchronized scenario in which PLCs start the attack independently, at different time steps. Then, we continue with the synchronized scenario in which PLCs start the attack at the same time using synchronized clocks. In both scenarios the attack lasts for 10s and consists in increasing the load to 300MVA.

The experimental scenario has been implemented in the Joint Research Centre's (JRC) Experimental Platform for Internet Contingencies (EPIC) laboratory. The Emulab testbed included nodes with the following configuration: FreeBSD OS 8, AMD Athlon Dual Core CPU at 2.3GHz and 4GB of RAM. Compromised PLCs have been implemented as R-PLC units controlling the loads on buses 5, 7 and 9. The implementation

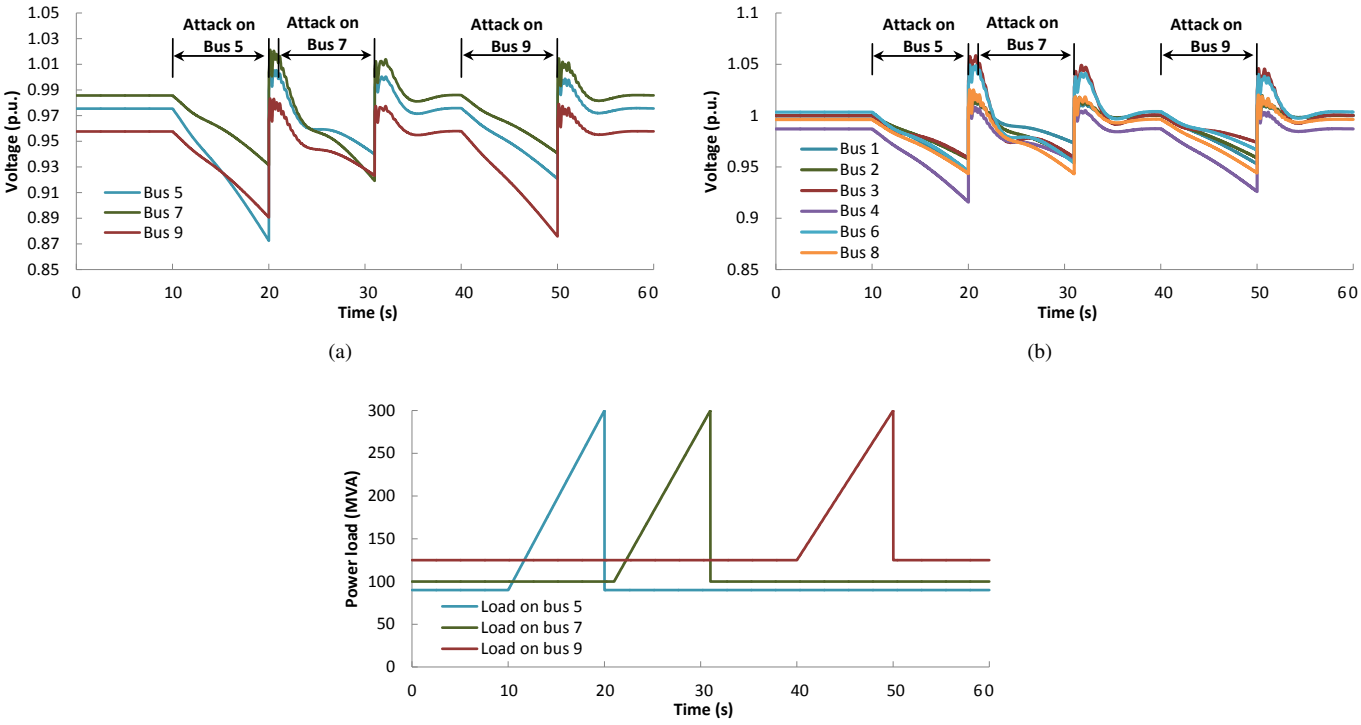

Fig. 7: Non-synchronized attack: (a) effect on attacked buses – 5, 7 and 9 (b) effect of non-attacked buses – 1–4, 6 and 8 (c) Attack load

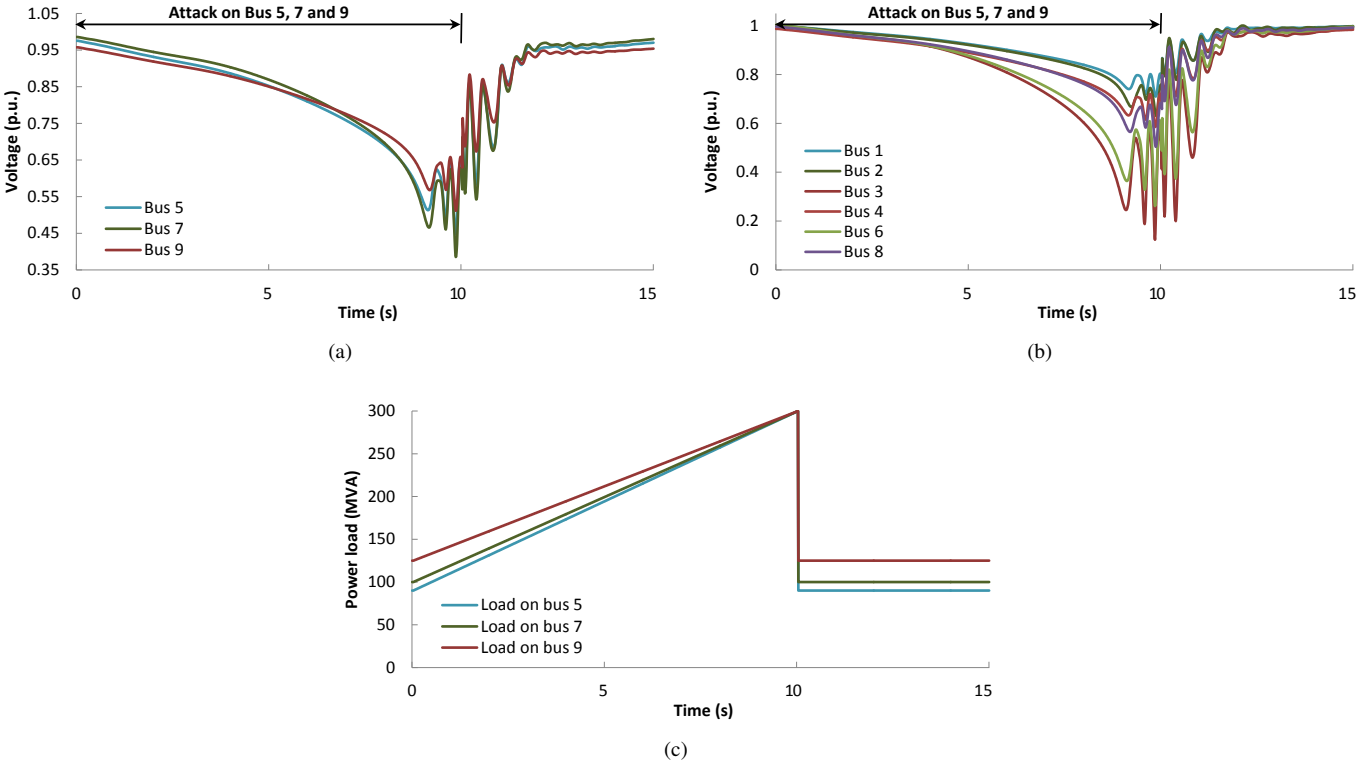

Fig. 8: Synchronized attack: (a) effect on attacked buses – 5, 7 and 9 (b) effect of non-attacked buses – 1–4, 6 and 8 (c) Attack load

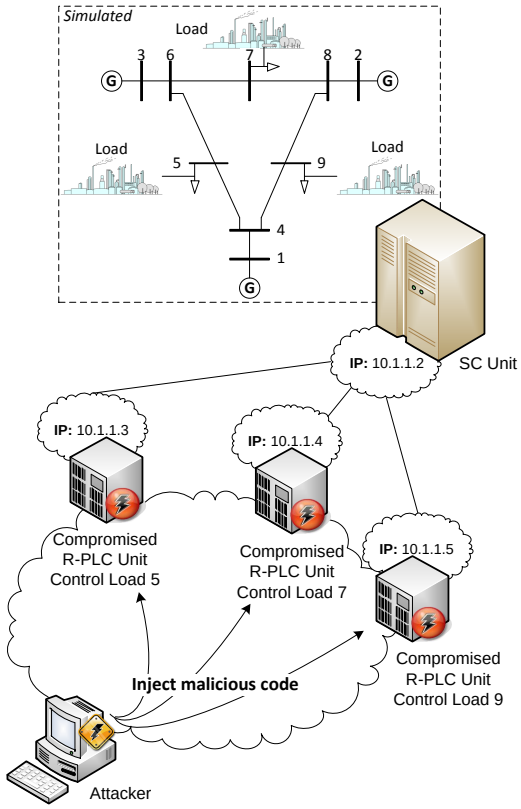

Fig. 6: Attack scenario implementation

of the attack scenario is depicted in Fig. 6.

In the first scenario the attack is started on bus 5, causing the voltage on the same bus to drop from 0.9754 (pu) to 0.8725 (pu). Because of missing synchronization, the attack on bus 7 is started after the previous attack is ended. In this case the voltage on bus 7 drops from 1.013 (pu) to 0.9191 (pu). Finally, the attack on bus 9 is started after 9 seconds the previous one, causing the voltage on bus 9 to drop from 0.9578 (pu) to 0.8758 (pu). These results are also shown in Fig. 7 (a). The effects on the other non-attacked buses should also not be neglected. For this purpose we have measured the voltage changes on the remaining buses with the results shown in Fig. 7 (b). The effect is much smaller than on the directly affected buses because of compensated power flows. The most significant perturbation can be observed on bus 4, where the voltage drops from 0.987 (pu) to 0.9157 (pu).

Next, we have implemented a synchronized attack launched at the same time on buses 5, 7 and 9. As shown in Fig. 8 (a), the attack causes the voltages to drop to 0.544 (pu) on bus 5, 0.3942 (pu) on bus 7 and 0.5362 (pu) on bus 9. The voltage oscillations seen in the same figure are a sign that the total load approaches the total generated power (i.e. 3×300MW). This effect is even more obvious on the other buses, as shown in Fig. 8 (b). Here, the voltage drops to almost 0.1 (pu), that is also a sign of an approaching voltage collapse.

By using a synchronized attack launched from multiple locations the attacker is able to cause major oscillations to the power grid. The oscillations are 7-8 times larger than the ones caused by a non-synchronized attack, which shows the major impact of cyber attacks on physical systems such as the power grid. The implemented scenarios have further shown that our framework can be used to conduct security studies on the Smart Grid. With this approach more complex scenarios can also be implemented, including several types of malware, Internet Service Providers, corporate stations and networks or even more complex power grid models such as the IEEE 24 or 118-bus systems.

VI. Concluding Remarks

In this paper we have extended our previously developed experimentation framework, that was developed for the analysis of localized Industrial Control Systems, so that it can be applied in the security study of the future Smart Grid. We have shown that our framework provides a flexible environment that can be easily extended with components in order to satisfy the requirements for experimenting with a complex environment as the future Smart Grid. Using the presented framework researchers can simulate physical systems such as *Generation* or *Transmission* while using emulated cyber components such as control logic code and network communication protocols. The main contribution of the paper is that it proposes a framework for experimenting with the Smart Grid that can be used by researchers to recreate an experimentation environment for measuring and understanding the consequences of cyber attacks on the Smart Grid. The feasibility study from this paper enabled the experimental recreation of a synchronized cyber attack, similar to a Distributed Denial of Service attack, against a power grid modeled with the IEEE 9-bus system. The experimental results confirmed that our previously developed framework is flexible enough to support security studies on the Smart Grid. As future work we intend to further integrate with cyber components more complex models, including FACTs devices, PMUs, sensor networks and to evaluate the applicability of existing command and control algorithms and protocols for ensuring the security of the future Smart Grid.

References

[1] Internet Protocols for the Smart Grid, RFC6272, June 2011, http://tools.ietf.org/rfc/rfc6272.txt

[2] Pike Research's report, "Smart Grid Communications Architecture," April, 2011.

[3] I. Nai Fovino, A. Carcano, M. Masera, and A. Trombetta, "An experimental investigation of malware attacks on SCADA systems," *International Journal of Critical Infrastructure Protection*, vol. 2, no. 4, pp. 139–145, 2009.

[4] S. East, J. Butts, M. Papa, and S. Shenoi, "A Taxonomy of Attacks on the DNP3 Protocol," in *Proc. 2009 IFIP Advances in Information and Communicaiton Technology Conf.*, Vol. 311, pp. 67–81.

[5] The Symantec Stuxnet Dossier, 2010, http://www.wired.com /images_blogs/threatlevel/2010/11/w32_stuxnet_dossier.pdf

[6] B. Genge, I. Nai Fovino, C. Siaterlis, and M. Masera, "A Framework for Analyzing Cyber-Physical Attacks on Networked Industrial Control Systems," presented at the IFIP Int. Conf. on Critical Infrastructure Protection, Hannower, New Hampshire, USA, 2011.

[7] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed systems and networks," in *Proc. 2002 Symposium on Operating Systems Design and Implementation*, pp. 255–270.

[8] NIST Internal Reports, "Guidelines for Smart Grid Cyber Security: Vol. 3, Supportive Analyses and References," Aug. 2010.

[9] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed systems and networks," in *Proc. 2002 Fifth Symposium on Operating Systems Design and Implementation*, pp. 255–270.

[10] M. Guglielmi, I. N. Fovino, A. P. Garcia, and C. Siaterlis, "A preliminary study of a wireless process control network using emulation testbed," in *Proc. 2010 2nd International Mobile Lightweight Wireless Systems Conf.*, pp. 268–279.

[11] W. Chunlei, F. Lan, and D. Yiqi, "A Simulation Environment for SCADA Security Analysis and Assessment," in *Proc. 2010 International Measuring Technology and Mechatronics Automation Conf.*, pp. 342–347.

[12] R. Chabukswar, B. Sinopoli, G. Karsai, A. Giani, H. Neema, and A. Davis, "Simulation of Network Attacks on SCADA Systems, First Workshop on Secure Control Systems," *Cyber Physical Systems Week*, April, 2010.

[13] S. Neema, T. Bapty, X. Koutsoukos, H. Neema, J. Sztipanovits, and G. Karsai, "Model Based Integration and Experimentation of Information Fusion and C2 Systems," in *Proc. 2009 12th International Information Fusion Conf.*, pp. 1958–1965.

[14] J. O. Calvin and R. Weatherly, "An introduction to the high level architecture (HLA) runtime infrastructure (RTI)," in *Proc. 1996 14th Workshop on Standards for the Interoperability of Defence Simulations*, pp. 705–715.

[15] C. M. Davis, J. E. Tate, H. Okhravi, C. Grier, T. J. Overbye, and D. Nicol, "SCADA Cyber Security Testbed Development," in *Proc. 2006 38th North American Power Symposium*, pp. 483–488.

[16] PowerWorld Corporation, "PowerWorld," http://www.powerworld.com

[17] M.J. McDonald, G.N. Conrad, T.C. Service, R.H. Cassidy, "Cyber Effects Analysis Using VCSE," Sandia technical report, SAND2008-5954, 2008.

[18] R.D. Zimmerman, C.E. Murillo-Sánchez, and R.J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Trans. on Power Systems*, vol. 26, no. 1, pp. 12–19, Febr. 2011.

[19] S. Cole and R. Belmans, "MatDyn, A New Matlab-Based Toolbox for Power System Dynamic Simulation," *IEEE Trans. on Power Systems*, vol. 26, no. 3, pp. 1129–1136, Aug. 2011.

[20] F. Milano, "An Open Source Power System Analysis Toolbox," *IEEE Trans. on Power Systems*, vol. 20, no. 3, pp. 1199–1206, Aug. 2005.

[21] Task Force on Open Source Software for Power Systems, http://ewh.ieee.org/cmte/psace/CAMS_taskforce/software.htm

[22] Smart Objects for the Smart Grid, http://www.wired.com/beyond_the_beyond/2011/05/spime-watch-cisco-and-the-ipv6-internet-of-things/

[23] J. Wang and V.C.M. Leung, "A Survey of Technical Requirements and Consumer Application Standards for IP-based Smart Grid AMI Network," in *Proc. 2011 Int. Information Networking Conf.*, pp. 114–119.

Béla Genge received his BSc in Computer Science in 2005 from the "Petru Maior" University of Tîrgu Mureş, Romania and his PhD in September 2009 in Computer Science from the Technical University of Cluj Napoca, Romania. His main research interests are critical infrastructure protection, design methods and composition of security protocols. Currently, he is a Post-Doctoral Grantholder at the Joint Research Centre of the European Commission. Before joining the Joint Research Centre he worked as an Assistant Professor at the "Petru Maior" University of Tîrgu Mureş and as a consultant focusing on control software design for electrical distribution networks and security of video surveillance systems.

Christos Siaterlis is an Electrical & Computer Engineer and a project officer at the Joint Research Centre of the European Commission. His research interests include various aspects of the resilience, stability and security of complex systems, and specifically critical information infrastructures like the Internet. He has a PhD in the area of Internet security management from the National Technical University of Athens and a Master of Science in Computer Science from the University of Southern California, Los Angeles. Before joining the European Commission he worked as a network engineer focusing on network monitoring, measurement and security in WANs.