Title: Simulation Platform for Cyber-Security and Vulnerability Analysis of Critical Infrastructures

Author: Massimo Ficco Michał Choraś Rafał Kozik

PII: S1877-7503(17)30360-5
DOI: http://dx.doi.org/doi:10.1016/j.jocs.2017.03.025
Reference: JOCS 647

To appear in:

Received date: 15-11-2016
Revised date: 15-3-2017
Accepted date: 31-3-2017

Please cite this article as: Massimo Ficco, Michal Choraś, Rafal Kozik, Simulation Platform for Cyber-Security and Vulnerability Analysis of Critical Infrastructures, *<![CDATA[Journal of Computational Science]]>* (2017), http://dx.doi.org/10.1016/j.jocs.2017.03.025

Simulation Platform for Cyber-Security and Vulnerability Analysis of Critical Infrastructures

Massimo Ficco[a,*], Michał Choraś[b], Rafał Kozik[b]

[a] *Dep. of Industrial and Information Engineering, Universita' degli Studi della Campania "Luigi Vanvitelli", Aversa (CE), Italy.*
[b] *Dep. of Teleinformatics Systems, University of Science and Technology Bydgoszcz, Poland.*

Abstract

The progressive advances in information and communication technology have lend modern critical infrastructures to become more and more complex and interconnected, and in continuous evolution. The increasing complex interrelation among such critical systems creates new security vulnerabilities, which can be exploited by malicious users to compromise sensible data and other systems also very far from the impact zone. Identifying and analyzing these complex interactions represent a challenge to the evaluation of the real vulnerability of each critical system. On the other hand, the evaluation of this complex and large-scale systems requires expensive and sophisticated modeling practices, simulation tools, and experimentation infrastructure. Therefore, we present a hybrid and distributed simulation platform for cyber-security analysis of large-scale critical infrastructure systems. It enables testers to assemble complex and distributed experimental scenarios in the cloud, by integrating different simulated environments, on which perform sophisticated vulnerability analysis, by exploiting penetration testing and monitoring facilities.

Keywords: Critical infrastructures, cyber-security, simulation, HLA, penetration testing, cloud computing.

*Corresponding author: Massimo Ficco, Department of Industrial and Information Engineering, Universita' degli Studi della Campania "Luigi Vanvitelli", via Roma 29, I-81031 Aversa (CE), Italy - Email: massimo.ficco@unicampania.it

1. Introduction

Critical Infrastructure Systems (CISs) involve sectors such as energy, finance, transportation, oil, gas and water distribution, health, government and emergency services. These systems from the point of view of security are of increasing importance in both industrial and public domains. They have to be highly resilient against cyber attacks and malicious activities, in order to reduce the risk of severe failures, compromising of sensitive data. The criticality of such systems poses new challenges for computer engineers, which must develop more robust systems to ensure a high level of protection, and at the same time they must keep low costs and development time [1].

On the other hand, the security of CISs is hard to achieve for their dynamic and structural complexity. First, they are designed as the composition of several off-the-shelf and, often, open-source components. In general, such components are originally designed for different application domains, which potentially may introduce unpredictable security vulnerabilities when used in other contexts. Second, their size has significantly grown on large-scale, and their operational environment, originally planned to be 'closed', becomes more and more 'open' to allow interoperability and remote control. It may potentially introduce new security threats exploitable by malicious intruders, which may compromise the system operation and access to confidential and sensitive data. Third, their dynamic complexity that manifests through the emergence of unexpected system behaviors in response to changes in the environmental and operational conditions of its components, as well as the unpredictable fluctuations both in workloads and in computational and communication resources needed to support the execution of these workloads, can introduce additional complexity in monitoring and detection of cyber attacks.

Moreover, such systems are not isolated, but highly interconnected and mutually interdependent, which may increase the potential for cascading attacks and amplify the impact of both large and small scale initial intrusions into events that may produce large consequences throughout regional or national

scales [2]. The huge complexity of such critical and distributed systems makes the integration task among components extremely challenging. It may introduce unexpected vulnerability that usually manifest late, during systems operation time, thus affecting systems security and resilience. In short, ensuring the trustworthiness of individual system does not guarantee that the whole system will be secure. Therefore, identifying, understanding and analyzing complex interactions, and interdependent represent a challenge to the evaluation of the real vulnerability of each system in consequence of a malicious event [3].

Therefore, the huge complexity of such CISs, even spread among different hosting infrastructures, often geographically distributed, as well as the continuing evolution and the extreme dynamism of modern organizations implies an evolution of the traditional concept of monitoring and protection. Traditional security defenses, including gateway appliances, antivirus, firewalls, and intrusion detection systems, could not be sufficient to detect and mitigate new emergent sophisticated attacks. Indeed, it is not unusual for intruders to circumvent these defenses, by performing stealthy attacks, that is, low-rate attack patterns designed to look like normal traffic performed by authorized users [4, 5]. At the same manner, a malicious individual, for example, employee and business partner, with legitimate access to sensitive data, can conduct a slow internal attack stealing sensitive data [6].

On the other hand, evaluating the security of a system before deploying it on-site is as much important as difficult. In this direction, the main challenge is to be able to reproduce such complex and distributed systems locally, in order to gain knowledge about their real behavior in-factory as it would be on-site. To address these issues, we propose *hybrid simulation*, which combines emulation and simulation. The emulation can be adopted to reproduce the execution of components or subsystems of the real target system deployed over virtual infrastructures, whereas simulation can be used to reproduce the behavior of the external systems (e.g., sensors, radar, web cam, and sw COTS), used to generate the experimental workload, as well as malicious behaviors and cyber attacks performed by compromised internal components or external systems.

In particular, in this work, we propose a platform for security and vulnerability analysis of CISs in factory, through the integrated use of hybrid and distributed simulation techniques. The simulation inter-operation is achieved by using the High-Level Architecture (HLA) based solution proposed in our previous work [7, 8], which is an architecture for simulation of large-scale critical systems. Moreover, in order to built loosely-coupled large-scale simulation systems, an Infrastructure-as-a-Service (IaaS) paradigm has been adopted. The presented solution enables testers to quickly assemble complex and distributed simulation scenarios in the cloud, by integrating different emulated and simulated environments, on which perform sophisticated vulnerability analysis. In particular, OpenVAS agents [9] have been exploited to perform penetration testing, as well as virtual resources introspection [10] to enable indirect inspection of the target virtual components.

The rest of this paper is organized as follows. Background and related work are presented in Sec. 2. The adopted hybrid simulation approach is presented in Sec. 3. The security scanner and monitoring agents are presented in Sections 4 and 5, respectively. Sec. 7 presents the conclusions and future work.

2. Background and related work

2.1. Critical infrastructure cyber threats

For the critical infrastructures, the integration of ICT technologies with physical elements has introduced new threats. There are already several examples of the successful attempts to compromise complex systems, by finding and exploring their vulnerable elements. In many cases, those attacks have had direct impact on physical elements or physical processes (e.g., generation of energy or industrial process in manufacturing). There are also past cases of cyber attacks on SCADA (Supervisory Control And Data Acquisition), ICS (Industrial Control System) and DCS (Distributed Control Systems) systems. Therefore, there is strong need to embrace the cyber aspects of critical infrastructures with

comprehensive tools and methodologies that could analyze the wide spectrum of technical and non-technical aspects.

In this section, we present some examples of case studies that reveal the complex nature of such systems and huge amount of interconnections across different technical and organizational levels of critical infrastructures. For example, due to the fact that the energy sector is quickly evolving and it is widely adapting different ICT technologies, we are able to identify many high profile cyber incidents at various levels of the energy grid. One of the successful cyber-attack happened in 2012 [11] in Iran. Computers controlling one of the nuclear processing facilities had been infected with malicious software called Stuxnet. It was the clear example of the industrial equipment being a target of computer attack. Since that date, the cyber community has realized that cyber weapon can be used "*... to create physical destruction [...] in someone else's critical infrastructure...*" [12] .

Another example of cyber attack on the energy grid was reported on the 23rd December 2015. The Ukrainian power distribution operator Prykarpattya Oblenergo suffered the successful cyber attack on their ICT infrastructure. In effect of this breach, operation of a number of power substations were interrupted and about 80 thousands of customers from Ivano-Frankivsk region suffered the blackout for few hours. The operator informed publicly about other technical failure related to the operation of the call centre infrastructure. In such situation, the customers could not contact the operator for information which deepened the crisis. In this case, the energy operator faced the well-coordinated attack consisting of three steps: malware attack, a denial of service attack targeted at the call center functionality, and the opening of substation breakers to cause the outage. Firstly, the attackers infected the main servers controlling the electricity distribution process, they infiltrated the operator network (using a malware backdoor) and issued a command to open breakers of various substations. The goal of the cyber criminals was to enter into the power grid system, by infecting the machines with malware software. They used macro script in Excel files to drop the malware. The infected Excel spreadsheets have been

distributed during a spear-phishing campaign that targeted IT staff and system administrators working for multiple companies responsible for distributing electricity in Ukraine. After the power was cut off, DoS attacks were launched to limit the awareness of the consequences of the attack. Error messages did not reach service personnel what prevented the proper reaction and delayed the recovering of the infrastructure operation. The Ukrainian blackout case can be seen as the one of the first significant and publicly reported cyber attacks aimed at civil infrastructure and directly impacting civil population (e.g., in opposition to the Stuxnet, the Iranian case, where industry/military premises were infected). The Ukrainian case shows that cyber attackers are able to cause serious damages to the economy, safety and homeland security.

We can also find examples of cyber incidents in the water sector, which show a real and severe impact of the cyber world on physical infrastructures. Similarly as for energy sector, the cyber components for both drinking water and waste-water facilities include control systems known as SCADA. Cyber attacks on such utilities may cause cascading effect on a public health, safety and economics. An example presented in [13] shows how the attacker can influence the water treatment plants. According to IBTimes [13], attackers infiltrated the water plant and were able to change the amount of chemicals that were used to treat drinking water. Healthcare industry is also an important part of critical infrastructure often targeted by cyber criminals. As examples show [14, 15], cyber attacks at this sector can slow down hospitals and expose patients to danger. Also the financial sector is struggling with cyber attacks. According to [16], the activity of cyber criminals increased by 41% in recent years. Recent example of Bangladesh bank [17] show that attackers have effective tools and skills to infiltrate banking systems and steal much money. According to the [18], also the growth of the Internet of Things and complexity of industrial control systems will lead to more vulnerabilities in hardware systems. Companies from cyber security sector [19] have identified serious vulnerabilities in automotive systems and home-automotive systems. This shows that not only CISs, but also citizens and societies can be directly influenced by the cyber attacks.

The cases described above give the short glance at the possible impact of the successful cyber attack launched at the critical infrastructures.

2.2. Cyber threats modeling

Depending on the goal of the modeling process there could be different approaches used to address different cyber security aspects, such as *risks*, *cyber attacks*, *system behaviors*, *networks*, etc.

Several tools and methods used for modeling the cyber risk have been proposed. For example, the aim of tools like Haruspex [20], is to evaluate the likelihood of scenario in which an attacker can implement successful cyber-attack against the evaluated system. Haruspex implements the simulation as model comprising of threat agents and the attacks they convey. The system is modeled as a set of components interacting through channels. As a final result, the tool collects relevant statistical data from the simulations. Similar approach to probability-based risk evaluation is presented in [21]. The authors have used ontology to model the system, its key components and interaction between them. Main concepts (general abstract classes), which compose the taxonomy are *assets* (elements that have value to the organization), *vulnerabilities* (defined as weaknesses of an asset or group of assets, which can be exploited by adversaries implementing the attack), *threats* (defined as potential cause of an unwanted incident, which may result in harm to a system or organization), and *safeguards* (practices, procedures or mechanisms that reduce or eliminate vulnerabilities).

A different group of analyzed techniques are tools and mechanisms for cyber attacks or system behavior modeling. These tools commonly adapt solutions used in the area of artificial intelligence, data mining, and data classification. Those techniques allow for automated data analysis, novelty and anomaly detection without extensive understanding of the underlying data content [22, 23].

As regards network, on the market, there are different modeling tools (both proprietary and open), such as NS3, NS2, OPNET, NetSim. With these tools, it is possible to analyze selected impacts of cyber attacks on the modelled network. For example, in [24] authors used NS2 simulator to predict the impact

of malware propagation, Denial of Service and Man-In-The-Middle attacks on SCADA systems. The authors measure the impact in terms of loss of control, Quality of Service (QoS), and number of dropped packets. With tools like NS3 and OPNET, it is possible to model the network with different granularity and using different formalism. For instance, in the NS3, the topology and the configuration of the simulation are provided either in *.py (python) or in *.cc (c/c++) files. The topology definition file commonly contains information about nodes (names, types, positions, etc.), communication lines (data rates and delays), topology, and IP stack configuration (IP, routing table, etc.).

2.3. CIS simulation approaches

Cloud simulation paradigm has been widely proposed in the recent literature. By exploiting Software-as-a-Service (SaaS) paradigm, simulation software can be offer as services (SimSaaS) into the cloud [25], incorporating multi-tenancy and scalability features. Such paradigm has been adopted to support analysis and simulation in several application domains, such as cloud simulation in manufacturing [26], scheduling parallel simulation jobs [27], traffic and transportation [28].

In this direction, several extensions of HLA in cloud have been proposed. For example in [29], a SysML (Systems Modelling Language dialect of UML) model driven approach and a cloud-based framework are presented to support the implementation of a distributed simulation system over a public cloud infrastructure. Moreover, a HLA solution integrated with a Service Oriented Architecture (SOA) is proposed to support the simulation of smart building projects [30].

Moreover, in order to manage the complexity of CISs, agent-based approaches can be adopted. For example, Rybnicek et *al.* [31], propose an agent-based modeling and simulation approach with game-theoretic elements to implement the behavior of three specific agents used to model both CISs and Distributed Denial of Service attacks (DDoS), including critical infrastructure-, consumer- and threat-agents.

However, each presented work proposes a tool or a framework to implement software services for supporting distributed simulation of complex systems. The current work extends the SimSaaS paradigm, by exploiting hybrid simulation to support more realistic security analysis of large-scale critical infrastructures [7].

3. Simulation of realistic CIS scenarios

Simulation process can be a valuable support to improve system representativeness and coverage at low cost. It can be exploited to perform security vulnerability analysis by accurate modeling of the reality and by the quantification of defined security metrics. Being able to simulate the behavior of the involved systems accurately, would allow engineers to drastically reduce the time to evaluate system security and resilience, enabling a more extensive test, early detection of vulnerabilities, evaluation of alternative security design decisions, and so on. On the other hand, the system representation and modeling is such that the characteristics of the system under analysis cannot all be fully captured, and large uncertainty is always present and impossible to represent and quantify in a reliable way, due to the heterogeneity, structural and dynamic complexity of involved subsystems [32].

A simulation service for CISs would present several serious challenges. The complexity of systems and the large number of involved entities can lead to very high cost and simulation time. It would require sophisticated modeling practices, as well as simulation tools, experimental platforms and real sub-systems need to interact in a coordinated way within a distributed environment. Thus, to be effective, simulation should manage these complexity aspects and be, at the same time, realistic, time-optimal, cost-effective, which, of course, are objectives contrasting to each other.

Therefore, specific hybrid and distributed modeling strategies represent a viable alternative to design simulation platform needed to support the security evaluation of such complex CISs [7]. In particular, in the hybrid simulation, the emulated parts are the system or the components under test, e.g, the SCADA

system, the software for air traffic control), as well as the compromised components, for example, a system node in which either a malware has been injected, or a security policy has been violated, or critical data have been manipulated maliciously, etc. The simulation parts can be, for example, the entities present within the tested scenario, including the final-users, the sensors (e.g., solar panels, radars), mobile objects (e.g., ships, cars, trains), which can be used to generate the experimental workload, as well as components used to simulate (internal or external) malicious behaviors. Other simulated elements are the components or sub-systems that can not be directly used in the simulation, for example, because they are not accessible or available for the simulation activity. Finally, the real systems are all the other sub-systems, which are not the objective of the security analysis, but from which it is possible to obtain, in real-time, operational workloads needed to reproduce a real experimental scenario.

In order to implement the adopted hybrid simulation approach, we exploit HLA [33]. As a framework for advanced distributed interactive simulation, HLA meets the needs of the proposed simulation platform engine, such as simulation tools re-usability, interoperability, and extensibility. Specifically, in order to support hybrid simulation, a specific gateway has been implemented, which operates as a bridge among the simulation parts and the emulation environments. Moreover, given the very complex nature of the considered network-centric system-of-systems, it is necessary to adopt a distributed network emulation solution to reproduce reality with a high degree of verisimilitude. In particular, network emulation enhances the overall HLA-based architecture with network services needed for a hybrid simulation.

The proposed platform has been structured as a distributed environment able to manage multiple heterogeneous systems and emulated networks over the same shared virtual infrastructure. The logical representation of the adopted platform is represented in Fig. 1. In particular, in order to cope with the complexity and scale of the considered scenarios to test, and to drastically reduce the set-up cost of testing, the testers should be able to easily setup scenarios without caring about details of the underlying infrastructure and platform. Therefore, hybrid

and distributed simulation services, which integrate simulation and emulation environments, supported by novel technologies for resources virtualization and working environment reproduction, represent the most promising way to define the needed strategies to actually support such paradigm shift [7, 34]. This can be achieved by designing the simulation platform on a top of a cloud computing infrastructure, namely, by implementing the simulation environment as a service [25]. Specifically, assuming simulation as a black box and not implemented with explicit cloud distribution in mind (e.g., in the case of an already existing simulation system), it is more feasible to use the IaaS approach. It does not restrict the type of applications deployable on the cloud infrastructure, but it abstracts the user only from the details of the physical hardware. Virtual resources are provided to the tester (i.e., virtual machines), allowing fine-grained control of the software stack, such as operating systems, which is a key requirement to test the actual vulnerability of a real system.

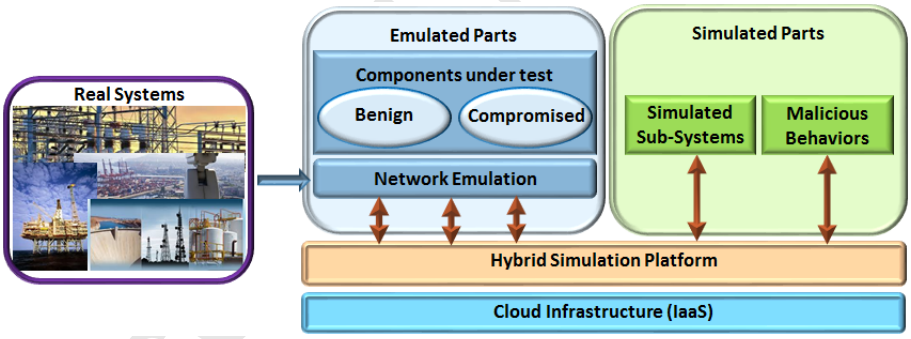

Figure 1: Cyber-security simulation environment.

3.1. HLA-based platform

The standardization of simulation interoperability resulted in the High Level Architecture (HLA), an IEEE standard for modeling and simulation [33]. The HLA is a technical architecture developed to facilitate the reuse and interoperability of different simulation tools, systems and assets. It provides a general

framework, which can be used by developers to implement their simulation systems and to interoperate them with other independent simulation systems. In fact, instead of implementing large monolithic simulation systems from scratch, it allows the integration of multiple independent and heterogeneous simulation environments, each with its own features, languages, and operating systems, within a more complex federated simulation system, by enabling the reuse of already existing solutions for new purposes. In particular, a federation is a distributed simulation system designed for a particular purpose, which consists of several interactive members. Each member that participates in a simulation is called federate, which can be a different simulated component or system, also implemented by using different tools and simulation environments, such as Network Simulator, Matlab, and LabView. The interface specification of the HLA describes how to communicate within the federation, and is implemented by the Run-Time Infrastructure (RTI). Federates can assume the role of Publisher to publish information within the federation, and the role of Subscriber to receive information created by other federates.

The platform adopted in this work exploits a distributed and interactive hybrid simulation approach. Specifically, RTI services have been exploited to manage federations of emulated and simulated components, as well as to specify synchronization and data exchange among the federation members. Each federation member is represented by an HLA simulation object model (SOM), which specifies the types of information that a federate can provide to the federation, as well as information that it can receive from the others. The interactions among the federates are described by the federation object model (FOM), that represents the language of the federation. In order to be able to correctly exchange simulation data among federates, which evolve according to a different temporal model (emulated and simulated), the RTI mechanisms have been exploited to coordinate how fast the simulators advance in their logical and emulation scenarios, in which the correct delivery of data is based on timestamps. Finally, data exchange in the federation is implemented by exploiting the publish/subscribe paradigm provided by the RTI.

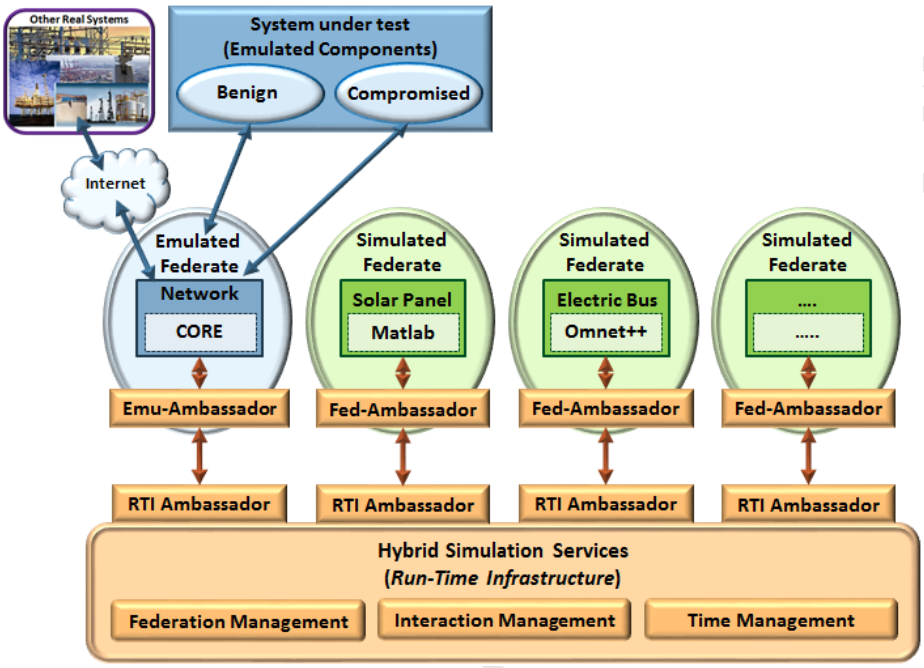

Figure 2: Hybrid Run-Time Infrastructure

As Fig. 2 shows, in order to enable the interoperability among the emulated parts with the simulation components, through the emulated network, a specific network *Emu-Ambassador* has been implemented. In particular, assuming that the emulated and real systems run on virtual and physical hosts, respectively, with specific IP addresses assigned, the Emu-Ambassador operates as a bridge, that, on the one hand, performs the publication and subscription in the federation of messages to be sent to or coming from the emulated/real systems, on the other hand, communicates with the emulated/real systems by using IP sockets. Specifically, it publishes real system messages over RTI, and wrap each incoming message from the RTI, forwarding it directly to the emulated/real system components. Message delivery can be managed according to a store and forward policy based on dedicated application queues. When the simulation starts, the Emu-Ambassador registers itself to the federation, and symbolic name is associated to the IP address of each registered emulated/real application, which is

used to identify its in the federation. The interaction and time synchronization among the involved simulated and emulated entities is implemented by specific functionality described in our previous work [7, 8].

Finally, the simulation infrastructure is based on PoRTIco, which is an open-source cross-platform HLA/RTI implementation [35].

4. Scanning agents for penetration testing

In order to perform vulnerability analysis of the experimental scenarios, a penetration testing approach is adopted. In particular, we adopted OpenVas, an open-source security scanner, which provides sophisticated scanning agents and configurable management interfaces [36]. It enables the scanning of a target both from the outside and from the inside, by installing the agent directly on the monitored node. Its vulnerability database are weekly updates of new vulnerabilities, and specific penetration tests are enabled in order to exploit the vulnerabilities themselves and used to verify whether they affect or not the target system. Specific interfaces can be used to configure and tune the tests and the monitoring policies. The OpenVAS architecture shown in Fig. 3 includes the following main components:

- *Scanner*: It performs the vulnerability assessment of components under test (by using tests downloaded form the vulnerability database with daily updates, through the OpenVAS NVT feed) on multiple target machines. It adopts the OpenVAS Transfer Protocol (OTP), which is provided with SSL support.

- *Manager*: It manages the OpenVAS Scanner through the OTP protocol and offers an interface accessible through the XML-based OpenVAS Management Protocol (OMP). Lightweight clients can be developed to implement filter of scanning results. It allows managing a SQLite-based database for storing configuration data and scanning results.

- *Client*: It offers a user Web interface, which uses an XSLT (Extensible Stylesheet Language Transformation) to convert OMP XML-based responses to HTML, as well as a command line tool that allows to create batch processes and operate the Manager.

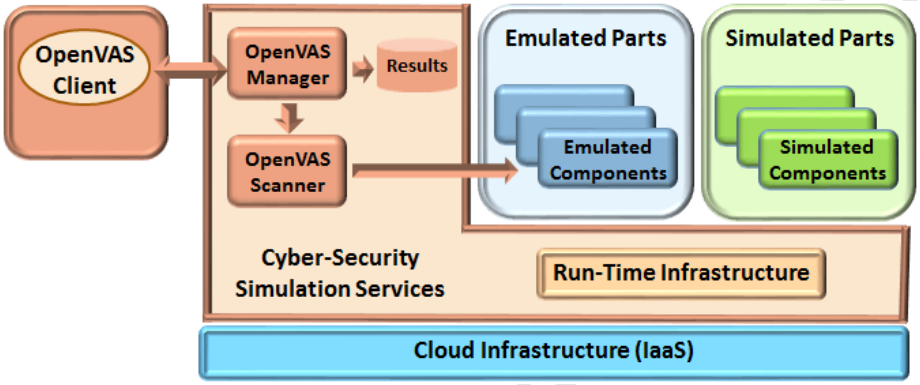

Figure 3: OpenVas-based penetration test service

In order to make OpenVAS available on-demand in a cloud infrastructure, we adopted an OpenVAS adapter and cookbook needed to install and configure respectively the Scanner and the Manager on the target machines [9]. The cookbook is the fundamental unit for the configuration and deployment of content on a target machine by the Chef framework [37]. Specifically, we adopted the Chef framework for the automation deployment of the simulation platform and the emulated/simulated components on the cloud resources. Chef uses recipes to specify the infrastructure and the related configuration tasks. Recipes use building blocks called resources, representing pieces of an infrastructure, included in a cookbook. Chef cookbooks, as well as other configuration data, are stored in the Chef server, which is the main component of the Chef architecture and may be either installed on local machines or invoked as a remote SaaS service. Chef clients are installed on the resource nodes. Finally, the Chef workstation allows to communicate with the Chef server and to execute all operations needed to configure and execute Chef components. The communication between the Chef workstation and the Chef server is performed by the Knife command line

tool, which offers management functionality for nodes, cookbooks, roles, cloud resources and others.

5. Monitoring facilities

In order to identify and analyze possible security vulnerabilities, facilities are provided as services for monitoring the target system. In particular, we adopted a virtual machine introspection (VMI) approach, which allows indirect inspection of the VMs state. VMI mechanisms offer isolation properties that are quite attractive for applications dealing with security and debugging in virtual machine environments [38]. They can be used to observe the behaviors of the single component under test and the interactions with other components, including emulated system and network entities.

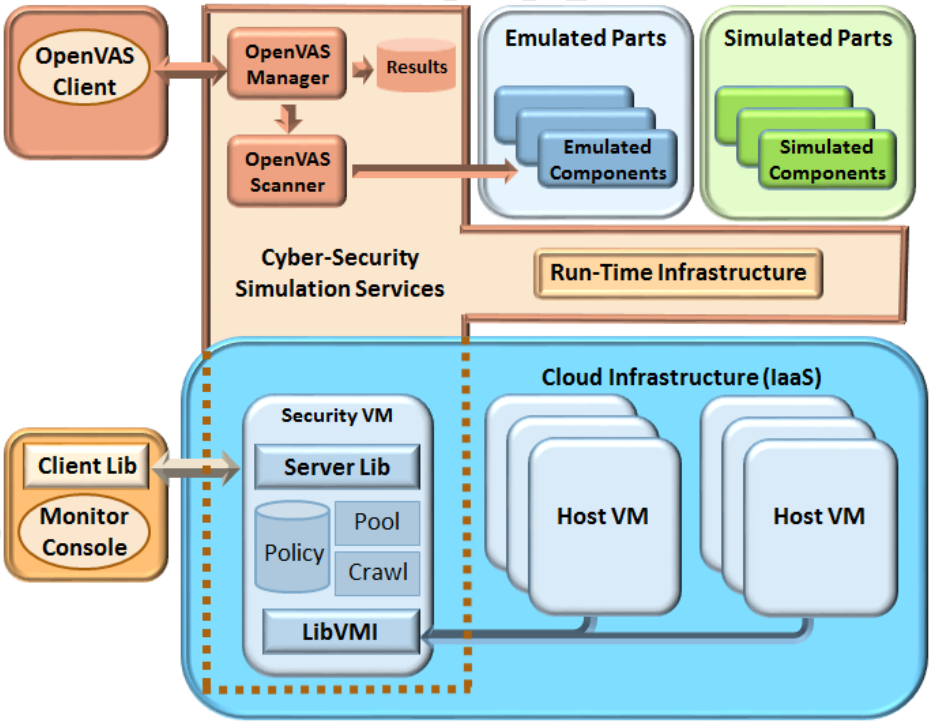

Figure 4: System architecture of security monitoring services

As Fig. 4 shows, a security monitor service is hosted on a security VM, which acts as a back-end of VMI service for monitoring and tracking execution states of hosted VMs. The service has been implemented by LibVMI, an open source implementation of VMI supporting hypervisor, such as Xen and KVM [10], which allows to infer raw information needed to monitor the VMs state. A management console acts as front-end of VMI, and used to enable testers to configure monitoring policies on VMs. It exploits an API implemented by Linux RPC calls to invoke the libVMI on the security VM, which enables to read files, process list, system call tables, dll list, memory buffer, as well as capture packets of network and handle events from VM kernel.

However, such data represent aggregated raw information, which needs to be correlated with the information collected at simulation level, usable for example to discriminate whether the monitored overload is related to a normal user peak load or a simulated cyber attack. In this direction, specific probes can be installed and configured to acquire additional data. In particular, in the proposed solution, the probes are implemented by mobile agents, which can be automatically deployed on the involved components (e.g., simulation and emulation components, scanning agents), from which collect and send the measures to the Monitor console. The monitoring agents are implemented by the JAVA Agent DEvelopment Framework (JADE) [39].

6. Cyber risk evaluation

Complex network-based systems typically contain variety of software platforms, components and facilitate, as well as involve different communication models and protocols. Therefore, when addressing the cyber security of computer networks, apart from considering isolated vulnerabilities of particular elements (such as applications, hardware, operating systems, etc.), much effort should be also devoted to understanding the overall cyber situational picture. In real cyber crime scenarios, adversaries use set of vulnerabilities to break into a network. This is also phrased as a 'vulnerability chaining', which is a very

important technique adapted in the process of breaching the IT systems security.

In order to analyze such situation, a network attack graph can be used. The network attack graphs describe different ways of possible penetration scenarios of computer systems. The penetration scenario is a chain of actions that is supposed to let the adversary to reach particular goal (e.g., service disruption, data theft, host infection, etc.). For well constructed network model, the attack graph can be a valuable tool providing broad operational picture and increasing the situational awareness of the decision makers.

In this paper, we consider a hypothetical (yet realistic) use case (still under the development), where we use the results from penetration testing scenario in order to provide detailed analysis of cyber security risk. This idea extends our previous research [40, 41] with respect to the deployment in the HLA-enabled environment. As it is depicted in Fig. 5, it can use information feeds directly from the hybrid simulation platform.

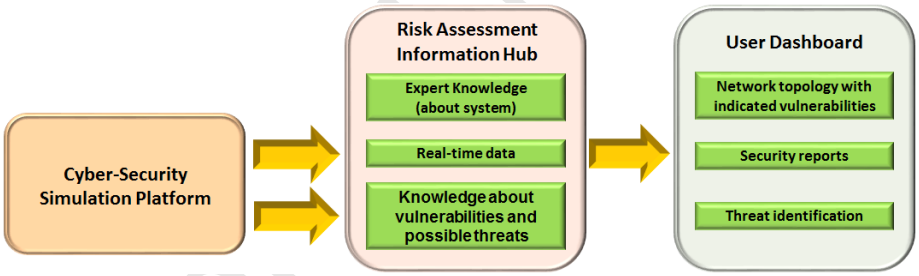

Figure 5: Risk assessment framework combined with hybrid simulation platform

Currently, the created tool is facilitated with GUI (graphical user interface - User Dashboard), which allows the user to identify and analyze vulnerabilities and possible threats (attack scenarios). The following components are provided: (*i*) a topology visualization module depicting network topology annotated with security analysis results, (*ii*) security reporting providing information on the identified threats, and (*iii*) threat report, with the list of possible attack scenarios prioritized by the risk value. What is more, this tool also allows the user to analyze the results of different chains of actions.

In our approach, the knowledge about the system is maintained as the ontology in OWL (Ontology Web Language) format. In this ontology, network assets and components have weak points named vulnerabilities. These vulnerabilities can be exploited by threats, leading to attacks. Each attack can be mitigated by applying appropriate countermeasures. However, the tool allows the user to go far beyond isolated vulnerabilities analysis by means of attack graphs. These are also maintained in the ontology in form of the codified security rules, which allow the user to model complex scenarios and relationships between different elements in the analyzed network. More implementation details are provided in [40, 41].

7. Conclusion

In this paper, a cyber-security simulation platform has been presented to support vulnerabilities analysis of critical systems. In particular, we adopted a HLA-based solution for complex and distributed hybrid simulation. AThe proposed solution can be used for performing penetration testing and security analysis of large-scale complex network applications, in particular for critical infrastructures, such as complex energy grids.

The federated simulation of interconnected railway network, ICT network and energy grid (using Open Track, SINCAL, and ns3 respectively) has been performed in CIPRNet project [42]. However, in CIPRNet, HLA was not used, and we believe that the application of the approach presented in this paper could be beneficial for improving the simulation. Therefore. it is one of the objectives of further work.

Acknowledgment

This work has been partially supported by the Italian Ministry for Education, University, and Research (MIUR) under Project PON02_00485_3487784 “DISPLAY” of the public-private laboratory “COSMIC” (PON02_00669).

References

[1] W. ElMaraghya, H. ElMaraghya, T. Tomiyamac, L. Monostorid, Complexity in engineering design and manufacturing, CIRP Annals - Manufacturing Technology 61 (2) (2012) 793–814.

[2] E. Zio, E. Ferrario, A framework for the system-of-systems analysis of the risk for a safety-critical plant exposed to external events, Reliability Engineering & System Safety 114 (2013) 114–125.

[3] P. Pederson, D. Dudenhoeffer, S. Hartley, M. Permann, Critical infrastructure interdependency modeling: A survey of us and international research, Idaho National Laboratory, Idaho Falls INL/EXT-06-11464 (2006) 1–126.

[4] M. Ficco, M. Rak, Stealthy denial of service strategy in cloud computing, IEEE Transactions on Cloud Computing 3 (1) (2015) 80–94.

[5] M. Ficco, F. Palmieri, Introducing fraudulent energy consumption in cloud infrastructures: a new generation of denial of service attacks, IEEE Systems Journal (2015) 1–11.

[6] J. Oltsik, Data-centric security: A new information security perimeter, https://www.informatica.com/content/dam/informatica-com/global/amer/us/collateral/analyst-report/en_esg-data-centric-security_analyst-report_2876.pdf (2015) 1–4.

[7] M. Ficco, G. Avolio, F. Palmieri, A. Castiglione, An hla-based framework for simulation of large-scale critical systems, Concurrency Computation 28 (2) (2016) 400–419.

[8] M. Ficco, R. Pietrantuono, S. Russo, Using multi-objective metaheuristics for the optimal selection of positioning systems, Soft Computing 20 (7) (2016) 2641–2664.

[9] Vulnscan-openvas adapter, available at: https://cbless.de/posts/en/2015/Nov/vulnscan-openvas/.

[10] H. Xiong, Z. Liu, W. Xu, S. Jiao, Libvmi: A library for bridging the semantic gap between guest os and vmm, in: Proc. of the 12th IEEE Int. Conf. on Computer and Information Technology, 2012.

[11] Stuxnet, http://security.blogs.cnn.com/category/middle-east/iran/stuxnet/.

[12] Stuxnet computer worm, http://www.cbsnews.com/8301-18560_162-57460009/stuxnet-computer-worm-opens-r

[13] Hackers hijacking water treatment plant controls shows how easily civilians could be poisoned, http://www.ibtimes.co.uk/hackers-hijacked-chemical-controls-water-treatment- plant-utility- company-was-using-1988-server-1551266.

[14] Attack targeting health care., http://www.ucdmc.ucdavis.edu/welcome/features/2010-2011/08/20100811_cybe

[15] Cyber attack on health care institution., http://www.bakersfield.com/news/kern-medical-center-battling-vi

[16] Financial institutions on high alert for major cyber attack., http://www.computerweekly.com/news/4500272926/Financial-institutions-on-high-alert-for-major-cybe

[17] How cyber criminals targeted almost 1bn usd in bangladesh bank heist, https://www.ft.com/content/39ec1e84-ec45-11e5-bb79-2303682345c8.

[18] 2016 emerging cyber threats report, http://www.iisp.gatech.edu/2016-emerging-cyber-threats-report.

[19] L. Robert, Internet of things security check: How 3 smart devices can be dumb about the risks, http://www.pcworld.com/article/2884612/security/internet-of-things-security-check-how-3-smart-dev

[20] F. Baiardi, E. Telmon, M. Minichino, D. sgandurra, Haruspex - simulation-driven risk analysis for complex systems, ISACA JOURNAL VOLUME 3 (2012) 1–6.

[21] M. Choraś, A. Flizikowski, R. Kozik, R. Renk, W. Holubowicz, Ontology-based reasoning combined with inference engine for scada-ict interdependencies, vulnerabilities and threats analysis, In Pre-Proc. of 4th International Workshop on Critical Information Infrastructures Security, CRITIS'09, Bonn, Germany (2009) 203–2014.

[22] R. Kozik, M. Choraś, R. Renk, W. Holubowicz, Burduk r. (et. al.): Proceedings of the 9th international conference on computer recognition systems cores 2015, Advances in Intelligent Systems and Computing vol. 403 (2016) 767–776.

[23] R. Kozik, M. Choraś, W. Holubowicz, Evolutionary-based packets classification for anomaly detection in web layer, Security and Communication Networks, vol. 9, Issue 15 (2016) 2901–2910.

[24] E. Ciancamerla, M. Minichino, P. S., Modeling cyber attacks on a critical infrastructure scenario, Information, Intelligence, Systems and Applications (IISA), 2013 Fourth International Conference on, Piraeus (2013) 1–6.

[25] W. Tsai, W. Li, H. Sarjoughian, Q. Shao, Simsaas: Simulation software-as-a-service, 2011, pp. 77–86.

[26] S. Taylor, T. Kiss, G. Terstyanszky, P. Kacsuk, N. Fantini, Cloud computing for simulation in manufacturing and engineering: Introducing the cloudsme simulation platform, in: Proc. of the Annual Simulation Symposium (ANSS 2014), 2014, pp. 89–96.

[27] X. Liu, X. Qiu, B. Chen, Q. He, K. Huang, Scheduling parallel discrete event simulation jobs in the cloud, Mathematical Problems in Engineering 2012 (604) (2012) 1–18.

[28] J. Harri, M. Killat, T. Tielert, J. Mittag, H. Hartenstein, Demo: Simulation-as-a-service for its applications, in: IEEE 71st Vehicular Technology Conference, 2010, pp. 1–2.

[29] P. Bocciarelli, A. D'Ambrogio, A. Giglio, D. Gianni, A saas-based automated framework to build and execute distributed simulations from sysml models, in: Proc. of the 2013 Int Conf. on Simulation: Making Decisions in a Complex World (WSC 2013), 2013, pp. 1371–1382.

[30] M. Dragoicea, L. Bucur, W. Tsai, H. Sarjoughian, Integrating hla and service-oriented architecture in a simulation framework, in: Proc. of the 12th IEEE/ACM Int. Symposium on Cluster, Cloud and Grid Computing (CCGRID 2012), 2012, pp. 861–866.

[31] M. Rybnicek, S. Tjoa, R. Poisel, Simulation-based cyber-attack assessment of critical infrastructures, in: Proc. of the 10th Int. Workshop on Enterprise and Organizational Modeling and Simulation (EOMAS 2014), Vol. Lecture Notes in Business Information Processing, vol. 191, 2014, pp. 135–150.

[32] T. Aven, Interpretations of alternative uncertainty representations in a reliability and risk analysis context, Reliability Engineering and System Safety 96 (3) (2011) 353–360.

[33] Ieee standard for modeling and simulation (m&s) high level architecture (hla), federate interface specification, ieee std. 1516.2-2000, in: New York: Institute of Electrical and Electronics Engineers, Inc.

[34] M. Ficco, G. Avolio, L. Battaglia, V. Manetti, Hybrid simulation of distributed large-scale critical infrastructures, in: Proc. of the Int. Conf. on Intelligent Networking and Collaborative Systems (INCoS 2014), 2014, pp. 616–621.

[35] portico rti tool, Available at: http://www.porticoproject.org/index.php?title=Main_Page.

[36] The world's most advanced open source vulnerability scanner and manager, available at: http://www.openvas.org/.

[37] Automate and manage your it infrastructure, available at: https://cfengine.com/product/.

[38] H. Baek, A. Srivastava, J. Van der Merwe, Cloudvmi: Virtual machine introspection as a cloud service, in: Proc. of the 14th IEEE Int. Conf. on Cloud Engineering (IC2E 2014), 2014, pp. 153–158.

[39] Java agent development framework (jade), Available at: http://jade.tilab.com/.

[40] M. Choraś, A. Flizikowski, R. Kozik, W. Holubowicz, Decision aid tool and ontology-based reasoning for critical infrastructure vulnerabilities and threats analysis, In E. Rome and R. Bloomfield (Eds.): Critical Information Infrastructures Security, LNCS 6027 (2010) 98–110.

[41] M. Choraś, R. Kozik, Flizikowski, Inspire decision aid tool: a support for risk management and cyber protection of critical infrastructures, Telecommunications Review, vol. 8-9 (2012) 1215–1221.

[42] Ciprnet - critical infrastructures preparedness and resilience research network eu project, `https://www.ciprnet.eu/home.html`.

Highlights (for review)

- Hybrid simulation of large-scale distributed critical systems;
- Analysis of the real security vulnerability of complex systems;
- Risk assessment by penetration testing and simulation;

Biographies

Massimo Ficco is Assistant Professor at the Università degli Studi della Campania "Luigi Vanvitelli". His research interests include security, cloud computing and pervasive systems. He has a Ph.D. in computer engineering from the University of Napoli "Parthenope", Italy. He serves as the editor-in-chief of an international journal, participates to the editorial board of several journals, and has been conference chair and member of international conference committees. Contact him at massimo.ficco@unicampania.it.