BPIDS – Attack Impact Assessment

Olga Carvalho
Instituto Superior Técnico (IST), Universidade de Lisboa, Lisboa, Portugal
sofia.berens@ist.utl.pt

Abstract—**As adversaries continue to develop new attack techniques to undermine organizations' business goals, there is an increase necessity for defenders to understand how a cyber-incident can impact those goals, which has motivated research in mission impact assessment (MIA). This paper presents BIA (Business Impact Assessment), an integrated approach for understanding the mission impact of cyber-threats. BIA was developed to offer a mission-oriented evaluation model to profile the organization, and, upon it, a simulation platform to simulate mission impact of a user-chosen exploited threat. Our experimental evaluation has shown BIA is successful in generating a relevant report on mission impact for several organizational settings.**

Keywords— Impact Assessment, Mission Impact, Evaluation Model, Business Process Modelling, Simulation, Cyber-threats, Threat Impact, Security

I. Introduction

Nowadays most organizations have information and communications technology (ICT) embedded into the core of their business-processes, as a means to increase their operational efficiency, exploit automation and/or improve decision quality. An attack to the ICT infrastructure of an organization could significantly impact the business-objectives they support. This can be clearly observed when the organization under attack is an essential services provider, such as transportation, energy supply and distribution. The 2016 Industroyer [1] malware is a notorious example that targeted the Ukrainian power grid and caused a power outage that was able to deprive part of Ukraine's capital of power for an hour, during its characteristic mid-December cold weather. On the one hand, leaving vulnerabilities unattended may indeed lead to significant damage; on the other hand, removing all vulnerabilities of a system is usually impractical [2]. Established security mechanisms, such as antivirus software, log analyzers and intrusion-detection systems (IDS), generally focus on low-level events and report them independently, which leaves to the system defender the entire decision-making process of determining, in a timely fashion, whether a cyber-incident has any current or future negative impact on the organization's monitored network and goals, and to respond quickly and accurately to minimize the impact [3]. To aid that decision process, research in mission impact assessment (MIA) tries to estimate the impact of a cyber-incident on the organization's goal (i.e. mission) which typically requires a great level of detailed knowledge about the organization under assessment, including the organization's mission and the organization's cyber infrastructure, consisting of all organization's ICT resources that carry out the mission, and how they interact, condition and depend on each other, which is often difficult to obtain.

Grounded in the idea that the needed MIA data does exist in a single digital format, but in disparate locations and formats, this paper presents BIA (Business Impact Assessment), an integrated approach which contributions can be summarized as follows:

- based on a multi-layered information structure, a mission-oriented evaluation model is put forward. This model includes four layers to represent threats, assets, services and the organization mission, consisting of business-processes and their activities;
- based on the model, an incident propagation simulation platform was designed accordingly, and a bottom-up computation methodology is proposed to detect the business-processes that are potentially impacted by the simulation of desired threat landscapes;
- it has been demonstrated that the proposed approach can be used to successfully generate a report on mission impact of several organizational settings, that gives an overview of situations of risk.

The rest of this paper is structured as follows: Section II briefly surveys the literature on the subject in matter, highlighting required features of MIA. Section III describes the basic principles of the proposed approach. Section IV discusses the implementation of BIA, and Section V presents the evaluation process of the outlined solution. Finally, Section VI gives a conclusion and outlook to future work.

II. Literature Review

Research in MIA typically follows three main stages:

(1) the *modelling* stage that aims to discover and model all the organization's entities involved in accomplish the organization's mission, and the dependencies among them ([4]–[19]);

(2) the *propagation* stage, to assess how the impact may propagate through those modelled entities and compromise the organization's mission ([4], [6]–[9], [19], [20], [13]–[16], [17], [18], [21]–[25]);

(3) the *measurement* stage, where metrics are integrated within the model to numerically evaluate the mission impact ([4]–[6], [8], [12], [18], [20]–[22], [26]–[31]).

Current approaches on impact modelling ([4]–[6], [12]–[14], [17]–[19]) employ an entity dependency graph [20] to model mission performers (i.e. organization's entities involved in the mission) as abstraction layers, and the interactions and dependencies between mission performers as the links among each individual layer and between layers. Among others, Jakobson's cyber terrain (CT) [5], illustrated in Figure 1, is a distinguished example of this methodology, that provides a high-level reference model [31] to model typical information technology organizations.

In this paper, our proposed model retains the hierarchical and multi-layered structure proposed by Jakobson's CT and builds a security abstract layer, that represents cyber-threats, to make the evaluation model more complete.

Generally, to explore vertical and horizontal interdependencies among layers, a model-based analysis is used to evaluate the impact's propagation, which can be categorized as *logic-based models* ([4], [6]–[8], [17]–[21]), *probabilistic-based models* ([7], [9], [22], [23]) and *sensitivity-based models* ([13]–[16], [24], [25]).

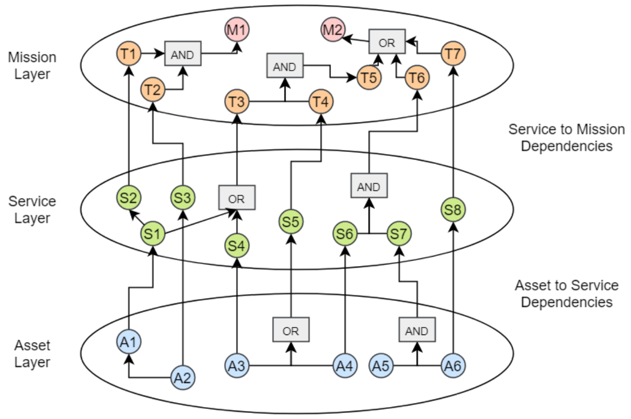

Figure 1 - Jakobson's CT model.

Logic-based analysis approaches are based on an attack graph model ([4], [6]–[8], [17]–[21]), that uses a sequential and explorative process to gradually identify and assess the system under evaluation. Probabilistic-based approaches are mostly based on Bayesian Networks (BN), to represent cause-and-effect relationships based on the assumption that all data can be conveniently represented by probability functions. Sensitivity-based models use active perturbation to measure how sensitive a model is to changes in its control parameter values and infer consequences in the system [13].

A further analysis of their features has concluded both probabilistic-based and sensitivity-based propagation methodologies require a high modelling overhead: the first requires to fully specify the conditional dependencies between random variables, whereas a sensitivity-based approach requires a great level of a priori information to learn a list of candidate control parameters to perturb. In lieu of this, this work adopts a logic-based approach by extending the base knowledge base of MulVAL [32] attack graph tool.

Ultimately, the prime focus of the present work is on the impact modelling and propagation aspects of MIA, nonetheless, the importance of the impact quantification aspect is recognized. A classification of current impact metrics is put forward, following the MIA layered model – *mission* ([5], [6], [8], [12], [18], [22], [31]), *service* ([5], [12], [18], [29], [31]), *asset* ([4], [5], [8], [12], [15], [18], [21], [26]–[29], [31], [33]–[35]) and *security* level metrics ([4]–[6], [12], [18], [20]–[22], [26], [27], [29], [31], [33]). Any of these metrics can be used to determine the impact of a cyber-incident, however, by itself, a unique metric may not be sufficient to qualify and quantify the impact. Yet, taken together, the resulting value may give a good representation of the impact.

III. Business Impact Assessment (BIA)

The goal of BIA approach is to provide a solution to understand how cyber-threats can be leveraged to impact the organization's mission, and identify the business-goals and processes compromised by an exploited threat. BIA is envisioned to be easily integrated with current approaches, tools and standards, and its design is two-fold: (1) to create a multi-layered evaluation model for MIA that can be easily integrated with current information sources and (2) to put forward a simulation platform that allows to reproduce how the impact of exploited cyber-threats propagate throughout the organization's infrastructure and to assess the impact on the organization's mission.

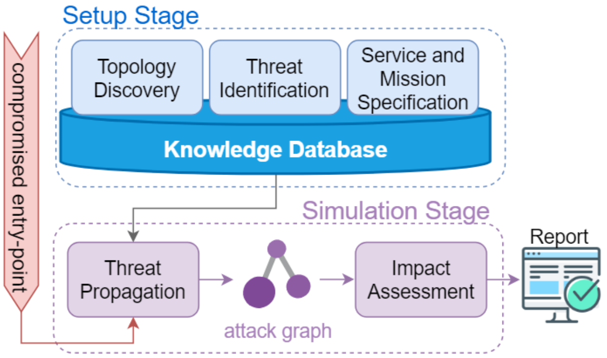

Figure 2 - BIA's architecture.

To do so, this work proposes a two-stage approach for MIA and is architected as illustrated in Figure 2. BIA's general idea is to first create a knowledge database with the organization's cyber infrastructure and mission profile – the *Setup* stage – to then be used to simulate the impact of a user-chosen compromised entry-point on the organization's mission – the *Simulation* stage.

The approach takes a set of three knowledge units as input during the Setup stage and a compromised entry-point during the Simulation stage to generate a MIA report as the output.

A. *Setup Stage*

The central idea of this stage is to capture the cyber infrastructure and business information, and consolidate it in an integrated data representation to be interpretable by the simulation. The data representation proposed to map the organization's cyber infrastructure onto the business-objectives is based on a four-layer evaluation model, as depicted in Figure 3.

To populate the evaluation model based, BIA's Setup comprises three knowledge units that mine different data sources to extract the required information: a *Topology Discovery* unit, a *Threat Identification* unit and a *Service and Mission Specification* unit, as outlined in **Erro! A origem da referência não foi encontrada.**.

1) *Topology Discovery*

This unit aims to gather information about the asset layer by receiving two types of inputs: (1) network packet captures and (2) firewall configuration, which are handled using two

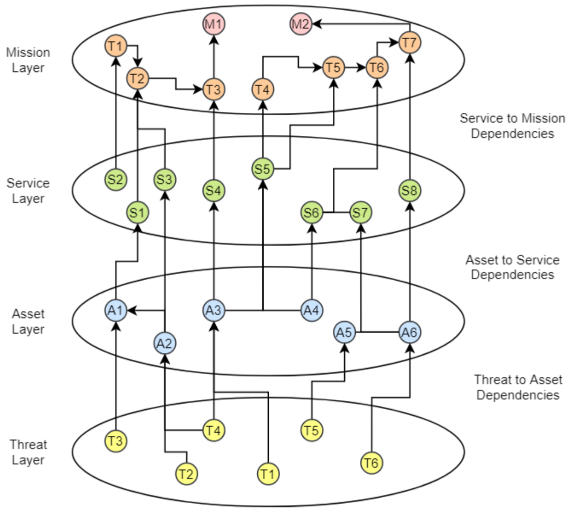

Figure 3 - BIA's impact evaluation model.

different components, a *Network Discovery* component and a *Connectivity Discovery* component.

At the end, this knowledge unit stores its findings about discovered assets and reasoned connectivity between assets, in the knowledge database to be used by the next knowledge units.

a) Network Discovery

This component resorts to a network analyzer tool that receives packet captures containing network communications exchanged between the IT components of the infrastructure under evaluation. Using basic dissection techniques[1], those packet captures are parsed to extract information about the infrastructure's assets, such as Internet Protocol (IP) addresses, and their connectivity, such as network protocols and ports used.

b) Connectivity Discovery

Even though network captures provide a wide perspective of the network topology, non-frequent communications may be missing from packet captures. To complement connectivity information previously gathered using packet captures, this component inspects firewall configuration, given as input, to infer missing allowed communications.

It is important to note that, even if a firewall allows a type of communication, it does not mean this communication is not filtered along the way to its destination, or even completely stopped, by other firewalls, as firewall hierarchies are often in place. Figure 4 illustrates an example of this hierarchy.

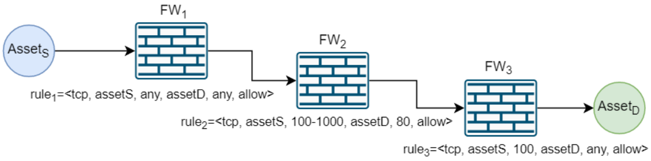
FW$_1$
FW$_2$
FW$_3$
Asset$_S$
Asset$_D$
rule$_1$=<tcp, assetS, any, assetD, any, allow>
rule$_2$=<tcp, assetS, 100-1000, assetD, 80, allow>
rule$_3$=<tcp, assetS, 100, assetD, any, allow>

Figure 4 - Example of firewall hierarchy filtering.

In this example, it is possible to infer that the rule that reflects the connectivity that is effectively allowed by the hierarchical policy would result as:

$$rule_{allowed} = < tcp, asset_S, 100, asset_D, 80, allow >$$

Hence, to assess the communications that are effectively allowed by the firewall policy environment, this component comprises two algorithms: the *Comparing Algorithm* to first assess allowed connectivity by each individual firewall, and a *Filtering Algorithm* to address firewall hierarchy and assess which rules survive the filtering action.

- ***Comparing Algorithm***. When a packet arrives at a firewall it is tested against each rule sequentially, meaning the firewall rules are order sensitive and the sequence of the firewall rule's list is to be taken into consideration when trying to understand which communication packets are effectively allowed. The proposed algorithm is designed to work as follows: first, take each *deny*-rule and compare it to the *allow*-rules that come next. It is possible to arrive to four possibilities, as suggested by previous work [36] and outlined in Figure 5; next, remove from the *allow*-rules the parts in common with the *deny*-rules (red zones in Figure 5).

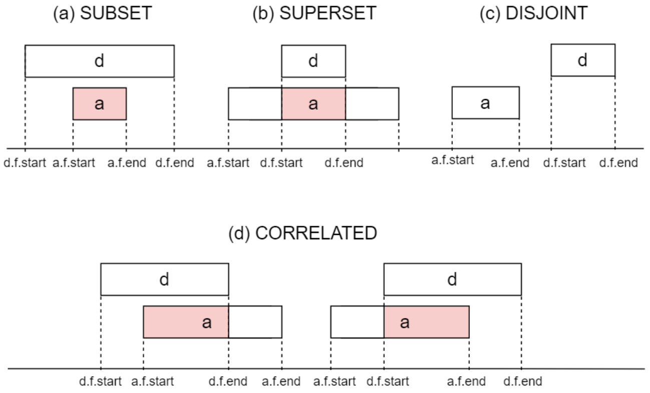

Figure 5 - Comparison possibilities between one field (f) of a deny-rule (d) and an allow-rule (a).

Applying this algorithm to all *deny*-rules in a firewall's configuration results in a list with only *allow*-rules (*allow*-list) that represent all the possible communications that may pass through the firewall.

- ***Filtering Algorithm***. The second algorithm was designed to inspect each firewall *allow*-list, obtained by the previous algorithm, to assess which rules survive the filtering hierarchy, and how. While traversing the network and the firewall infrastructure that constitute it, three distinct actions are proposed to be taken: (1) First, classify each allow-rule from the firewall allow-list according to its source and destination to understand which rules should be submitted to the other firewall's policies; (2) Next, rules with destination outside their firewall's domain are propagated to adjacent firewall's to be compared with their configurations and filtered accordingly; (3) Lastly, repeating this process to every firewall's rule list, results in a list of all allowed communications in the infrastructure.

The inclusion of the missing connectivity reflects how the current firewall policy allows connectivity that may be leveraged by an attack to move within the network. This *allow*-list is translated to possible connectivity and is used to populate further the asset layer.

2) Threat Identification

The second knowledge unit imports the information about the network topology already stored in the knowledge database and identifies the threats affecting the organization's cyber assets. With that objective in view, the unit's input is threefold: (1) a user chosen asset classifications to classify the organization's assets, (2) for each possible asset classification, a list of possible threats affecting that classification is provided and (3) to reduce the number of possible threats, a threat's classification is used to classify each identified threat, for instance, the STRIDE[2] framework for threat classification. The Threat Identification unit then proceeds to map threats with the corresponding assets, according to the user specified asset classification, threat identification and threat classification, and stores that information in the knowledge database.

3) Mission and Service Specification

The third step of the Setup stage aims to bridge the assets found by the Network Discovery unit, to the organization's business goals, specifically, the organization's business-processes, represented by a collection of activities to be

[1] https://www.wireshark.org/docs/wsdg_html_chunked/ChapterDissection.html

[2] https://www.microsoft.com/security/blog/2007/09/11/stride-chart/

accomplished, which are provided by services running on assets. Thus, this unit is envisioned to receive business-processes specification and map this information to the assets already stored in the knowledge database.

B. *Simulation Stage*

Following the Setup stage, which results in a fully populated knowledge database based on the proposed layered model, the Simulation stage proceeds to simulate the impact of a user-chosen entry-point to the system and perform MIA. This is proposed to be achieved by two modules as outlined in **Erro! A origem da referência não foi encontrada.**: a *Threat Propagation* module that aims to propagate the threat at the entry-point, throughout the organization's cyber infrastructure to reach the mission; and an *Impact Assessment* module to interpret the simulation's outcome and produce a report of MIA relevant information.

1) *Threat Propagation*

The Threat Propagation module takes the main stage for the impact propagation simulation, based on an attack graph model, where the goal is to determine whether a compromised asset is likely to deleteriously affect any of the business-goals of the organization. To this end, this module is designed as a simulation platform which is configured with the organization's infrastructure and mission identified and modelled by the Setup phase.

The simulation begins with a user-chosen entry-point (a specific asset and exploited threat) and ultimately tries to determine which organizational business-objectives would be affected if that asset became unreliable or unavailable. Starting from that entry-point, the simulation performs a bottom-up analysis, searching for attack paths by leveraging the organization's model interdependencies to propagate the initial threat. If an asset is accessible and has a threat, then is exploitable and the simulation advances to that asset. Additionally, if an asset runs a service that has a role in the mission, then the threat's impact is propagated towards the mission's activity (or activities) the service supports, and, from there, to the business-process(es) that rely on those impacted activities.

This threat propagation is achieved resorting to logic programming to express how the propagation advances with a set of series of Horn clauses, a logical formula that takes a particular rule-like form: $L_0 \leftarrow L_1, \ldots, L_n$, where $L_i\ \forall i \in N$ are literals, and if $L_1, \ldots, L_n$ are true then L_0 is also true.

In the design of this module, the threat propagation was defined using the four following Horn clauses presented in Table 1.

Table 1 - Horn clauses used to define threat propagation.

#	Description	$\boldsymbol{L_0}$	$\boldsymbol{L_1, \ldots, L_n}$
1	Entry-point compromised	compromisedAsset(*A*)	attackerLocated(*A*), threatExists(*A*, *Threat*)
2	Attack propagated to another asset	compromisedAsset(*A2*)	compromisedAsset(*A1*), connectivity(*A1*, *A2*), threatExists(*A2*, *Threat*)
3	Attack propagated to the service	compromisedService(*S*)	compromisedAsset(*A*), runsService(*A*, *S*)
4	Attack propagated to the business-process	compromisedProcess(*P*)	compromisedService(*S*), runsActivity(*S*, *A*), runsProcess(*A*, *P*)

The procedure of combining the organization's multi-layered modelled entities and their dependencies and iteratively validating the clauses defined creates an attack graph depicting all the possible threat propagation paths found from the simulated entry-point to organization's business-processes. The resulting graph is then output by this module.

2) *Impact Assessment*

After the attack graph is constructed, a careful reading of the graph is necessary to understand relevant mission impact information. Hence, the attack graph produced by the Threat Propagation module is traversed by the Impact Assessment module to identify the compromised assets and exploited threats, the explored connectivity between assets, and the business-processes compromised, and the propagation steps the simulation followed to advance throughout threatened susceptible assets towards the mission. This final analysis highlights relevant information and assembles it in a compact report for impact assessment.

IV. Implementation of BIA

Following the two-stage architecture described earlier, the technical implementation of BIA was also addressed in two sections: the Setup stage and the Simulation stage.

A. *Setup Stage*

To consolidate the proposed assessment model in a data representation, BIA employs the Neo4j[3] database, a graph database that offers a data model optimized for graph operations to address the adopted multi-layered architecture.

Regarding the knowledge units used to populate the database – *Topology Discovery*, *Threat Identification* and *Service and Mission Specification* – as each one leverages different data sources, five components were implemented according to the level of data granularity needed for each particular data source. The components were implemented using a combination of *Python* programming language due to its versatility, and shell scripting for its simplicity.

1) *Network Discovery*

The first component was architected to resort to a network analyzer tool, specifically for inspecting network traffic files (in *PCAP* format) to extract information about assets and their connectivity. It was implemented by a shell script that invokes network protocol analyzer *Tshark*[4], with a custom configuration, to analyze the network traffic recorded in the packet capture file. Any packet that contains TCP (Transmission Control Protocol) or UDP (User Datagram Protocol) layer information is checked for its source and destination's IP address and the network ports used by that communication. The extracted information is stored in a data log file (in *CSV* format) that is parsed to remove duplicated information. Discovered assets and connectivity is uploaded to the database to represent the asset layer, and also given as feedback to the user.

2) *Connectivity Discovery*

To further consolidate the asset layer, the Connectivity Discovery component was developed to receive network infrastructure documentation that identifies existing firewalls and firewalls' domains (hosts protected by the same firewall)

[3] https://neo4j.com/

[4] https://www.wireshark.org/docs/man-pages/tshark.html

and firewall's policies (the list of rules that define what kind of traffic is allowed or blocked). Motivated by *IPTABLES*[5] rule format, a firewall rule was modelled as:

rule = <*protocol*, *source*$_{IP}$, *source*$_{port}$, *destination*$_{IP}$, *destination*$_{port}$, *policy*>,

where *policy* can take the value allow or deny, to indicate if a communication related to this rule is allowed or blocked, respectively. Upon receiving the required input, this component uses the proposed *Comparing algorithm* to determine allowed communications from each firewall configuration, and *Filtering Algorithm* to determine allowed asset connectivity from firewall hierarchies in place. At the end, it stores resulting inferred connectivity in the database, according to the assets discovered by Network Discovery, and gives the result as feedback to the user.

3) Asset Classification

This component receives as input a file with a list of asset IP address mapped to a user chosen role classification (any desired semantical description) represented by data tuples in the following format:

<*asset*$_{IP}$, *classification*>,

where *asset*$_{IP}$ can be associated with multiple role classifications. Next, the component queries the database for the stored assets discovered by the Network Discovery component and proceeds to upload the given roles to the database.

4) Threat Identification

The Threat Identification component follows the Asset Classification component to map the organizational roles to the threats they are most vulnerable to. It receives two files: one with a list of asset classifications mapped to threats that can impact them, and another with the threat classifications STRIDE counterparts. The Threat Identification component then proceeds to map the asset's classification to the threat classification model using the following the format:

<*asset classification*, *STRIDE classification*>,

which finally is uploaded to the database, constituting the threat layer.

5) Service and Mission Specification

To gather information about the service and mission layer, a Service and Mission Specification component was created to interact with *BP-IDS* [37] database through its API to receive business-process information. The API returns business-process information in a *JSON* format, which is reformatted and uploaded to the database to form the service and mission layer.

B. Simulation Stage

BIA's Simulation stage was conceived as a simulation platform that leverages MulVAL [32] to perform MIA. Two components were implemented to achieve this purpose: a *Threat Propagation* component to convert the proposed Horn Clauses to into MulVAL's knowledge base. These clauses are then used by MulVAL as rules to be validated by the organization's evaluation model and produce an attack graph; and a second component, the *Impact Assessment* component to extract relevant MIA information from the attack graph and present it to the user.

1) Threat Propagation

MulVAL acts as a processor of Datalog rules to generate attack graphs, however, its original rules do not consider a threat and mission layer, hence, BIA reformulates MulVAL knowledge base by expressing the proposed four Horn Clauses for threat propagation as new Datalog rules, implemented as a part of *interaction rules* in MulVAL

Interaction rules are based on *primitive* and *derived* facts to represent the preconditions and postconditions, respectively, of Horn Clauses. BIA transforms the organization's infrastructure and mission information identified in the Setup stage into primitive facts. MulVAL then applies the interaction rules towards the primitive facts and, if all preconditions are met, produces derived facts.

In addition to facts and rules, MulVAL requires an initial point to start its verification process, and a target to direct and conclude that process. BIA defines MulVAL's target as the business-processes identified in the Setup stage that MulVAL will try reach, while the initial point is provided as an external input to this component and defined as the entry-point to the system by a <*asset*, *threat*> tuple.

The entry-point is then transformed and combined with the rest of the primitive facts, which completes the required input to run MulVAL, and effectively triggers the start of the simulation. Furthermore, the entry-point is chosen by the user, which can choose to run the simulation several times for different entry-points independently of the Setup Stage. Here lies another main features of BIA's MulVAL extension, where every time the user chooses an entry-point, MulVAL's required input is automatically changed accordingly.

The attack graph generated is output in *PDF* format (optional), together with two *CVS* files, one with the nodes and the other with all arcs present in the attack graph, and a *TXT* file with all this information combined. Since the graphically representation of the attack graph (in *PDF*) often results in an image difficult to digest at naked-eye, and it is the option that takes longer to produce results, BIA's Threat Propagation component only outputs the two *CSV* files for the next component, to assess relevant MIA information.

2) Impact Assessment

When performing MIA, often users want to quickly assess which organization's business-processes are impacted. A further analysis may then be required to understand how the attack may have propagate through the organization's infrastructures. As such, this component is implemented to parse the attack graph produced by the previous component and retrieve relevant information about the compromised performers, and the threats and connectivity exploited to that effect. This information is then presented to the user in *JSON* format, for its readability, and versatility to be further extended and integrated.

V. Evaluation

BIA was deployed in an Ubuntu 18 virtual machine with 9 GB of RAM and 100% access to the resources of two of the four cores of an Intel Core i7-7500U CPU 2.70-2.90 GHz processor, where a series of experiments were conducted on an ICS dataset called EPIC (Electric Power Intelligent Control) from iTrust labs[6] that contains the essential elements of a fully operational critical infrastructure for power supply in a scaled-down replica capable of generating up to 72kVA power.

EPIC's network architecture is illustrated in Figure 6 and is comprised of four main stages – Generation, Transmission,

[5] https://linux.die.net/man/8/iptables

[6] https://itrust.sutd.edu.sg/itrust-labs-home/itrust-labs_epic/

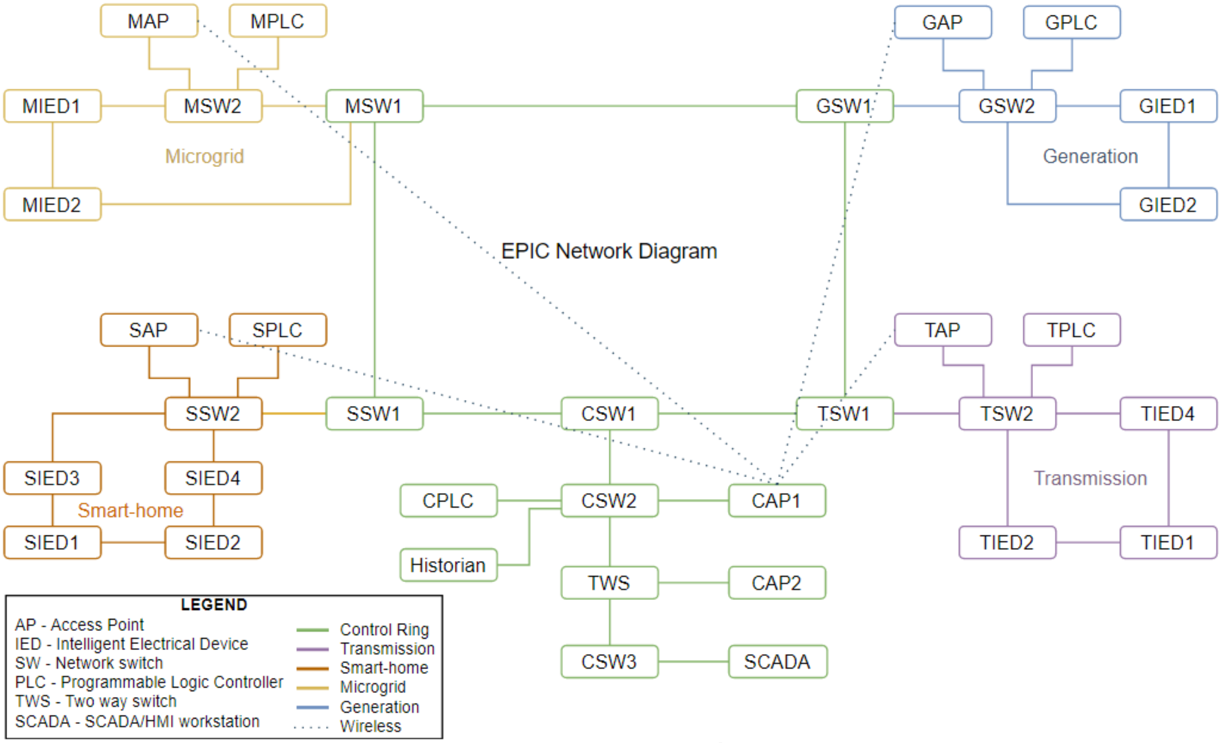

Figure 6 - EPIC's Network Diagram.

Micro-grid, and Smart-home. There is a total of 36 assets that are classified by five asset types:

- Supervisory Control And Data Acquisition (SCADA) workstation
- Programmable Logic Controllers (PLCs)
- Intelligent Electronic Devices (IEDs)
- Access points (APs)
- Switches (SWs)

All assets are prefixed with G, T, S and M, respectively, for Generation, Transmission, Smart-home and Micro-grid stages. For instance, the PLC in Generation is represented as GPLC. These four stages are connected to a control network (with C prefix).

A. *Evaluation Setup*

Besides asset documentation, asset classification and packet capture files provided by EPIC, BIA requires an additional set of information for minimal functionality: the organization's mission, comprised of business-processes (BPs) and threat landscape the user desires to simulate, as well as an entry-point to the system.

1) *Mission*

The following sample of 3 BPs were configured according to EPIC's description of undergoing physical processes and running software [38]:

- ***BP1 - Power supply to Smart-home***. This business-objective, as the name suggests, aims to supply electrical power to load banks at the smart-home stage of EPIC. To achieve this, it is suggested that SCADA sends a command to close (1st activity) the circuit breaker that was open and interrupting the current flow. This command is sent to the SPLC (2nd activity) that in turn sends it to the IED responsible for the circuit breaker (3rd activity).
- ***BP2 - Power supply in grid-connected mode.*** EPIC's generators can produce the power required for the system along with power drawn directly from the main grid. Accordingly, a business-goal is defined to operate the system in grid-connected mode: (1) SCADA sends GPLC a close command of the main circuit breaker to connected EPIC with the main grid, (2) GPLC sends that command to GIED1 responsible for the main circuit breaker and (3) in turn, GIED1 closes the circuit breaker.
- ***BP3 - Read electrical voltage***. The Transmission stage is representative of a distribution grid, supplying power to Smart-home stage. A transformer is used to control the voltage to the Smart-home. A business-process to read the current value of the voltage is proposed, where CPLC sends a read request to TPLC (1st activity) that sends the request to the TIEDs (2nd activity).

In total, the mission layer is thus comprised of 8 activities providing 3 BPs (BP1, BP2 and BP3) and mapped to 7 different services running on 7 different assets.

2) *Threat landscape*

Unlike for the mission layer, there is no benchmark of threats affecting EPIC dataset, however, previous research on feasible attacks [38] and emulated threat scenarios [39] on EPIC and research on typical threats affecting ICSs [40] can be leveraged to define some possible scenarios for threats present on the testbed. Accordingly, the threat distribution proposed for BIA's evaluation results in the following *STRIDE* (spoofing, tampering, repudiation, information disclosure, denial of service, escalation of privilege) landscape in Table 2. An exploited threat in SWs or routers can impact the entire organization. For evaluation purposes, let us consider SWs and routers are not exposed to any threats, to be able to simulate the impact propagation of other type of threats.

Table 2 - Threat distribution.

	S	T	R	I	D	E
IEDs	-	-	-	-	✓	-
PLCs	-	✓	-	-	-	✓
Historian	-	✓	-	✓	-	-
APs	✓	✓	-	✓	-	-
SCADA WS	✓	✓	✓	✓	✓	✓
Routers/SWs	-	-	-	-	-	-

3) *Entry-point*

In the previous chapter it was defined that the SCADA WS was the asset exposed to the most threats, including all 6 STRIDE classifications, as it was the asset most threat scenarios focus on [39]. One of this threat scenarios correspond to a *Power Supply Interruption* attack [38], where the attacker gives a false indication, i.e. "the circuit breaker is closed" (where in reality it is open) to the operator through SCADA workstation. The malicious control code is then uploaded to the SPLC and the correct command to close the breaker was disabled, impacting the Power Supply to Smart-home (BP1).

Another attack scenario is based on compromising PLCs to trigger the protection functions in IEDs, so they open the circuit breakers, interrupting the electrical current, and resulting in a N*uisance Tripping* attack.

Therefore, let us simulate mission impact with entry-point on the SCADA WS, given as a $< SCADA, tampering >$ tuple, to evaluate how tampering with SCADA WS integrity can detect the impact of both attack scenarios.

B. *Evaluation Process*

In the next sections, a series of experiments is conducted to test and analyze BIA's features and limitations.

1) *Discovering organizational topology*

The organization's topology and connectivity constitute the asset layer of the organization's profile, and are firstly handled by the Network Discovery component.

A first series of experiments is undertaken to evaluate the overall accuracy of this component on discovering the organization's infrastructure topology (assets), using packet capture files of all EPIC's 8 scenarios.

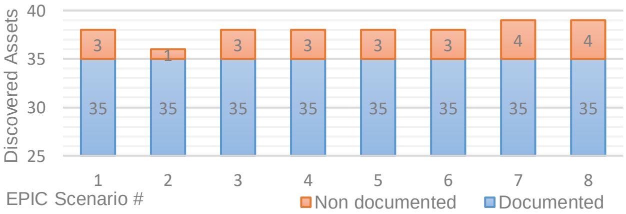

Figure 7 - Accuracy evaluation of Network Discovery.

Taking into account that EPIC documents a total of 36 assets, Figure 7 shows Network Discovery is successful in discovering 97.2% of the infrastructure's assets: for all scenarios 35 out of 36 documented assets were discovered. The asset that was systematically unobserved for all scenarios is documented as the default gateway for the master PLC (CPLC) to connect to SCADA WS, and, as such it does not appear in the network capture.

Moreover, this component also reported undocumented assets for all scenarios, namely assets respective to the IP addresses 224.0.0.252, 237.1.2.19, 239.255.255.250 and 172.16.8.12. The first three belong to known IP multicast address ranges[7], and, as expected, are not associated with a specific asset. The fourth undocumented IP address may be associated with external operators' laptops or other testbeds connected to EPIC. In the same way, this external IP address may also belong to an attacker that has successfully intruded the internal network. Either way, feedback on the discovered assets is given to the user to allow the validation of the organization topology being modelled.

2) *Using packet captures*

Since it was seen the number of documented assets discovered is the same for every packet capture, BIA was validated using EPIC's first scenario packet capture to evaluate how asset connectivity, as a key precondition for BIA's threat propagation methodology, influences BIA's results.

The respective packet capture file contains 449177 packets, from which BIA's Network Discovery component extracts 5770 unique *connections* entries to represent EPIC asset connectivity. Although 5770 *connections* entries are a significantly refined number compared to the number of packets recorded in the packet capture file, it is a high number that can condition BIA's simulated impact to propagate everywhere and result in a rough report for MIA, especially when considering EPIC's topology consists of 35 assets.

This vast connectivity panorama extracted from the packet capture can be explained by the presence of ephemeral network ports in the client side of client-server type of exchanges, to connect with a well-known port in the server side. Each newly allocated port will create a new *connection* entry, which means asset connectivity is increasing as new ports are used for the same asset-to-asset connectivity.

Before describing how this issue can be mitigated, let us simulate mission impact using the extracted asset topology and connectivity from EPIC's packet capture, and the evaluation setup described earlier in Evaluation Setup section.

BIA's simulated impact is depicted in Figure 8, that shows 24 impacted assets (all assets with associated threats) which results in a fully impacted mission, where all BPs, activities and services could be compromised.

A further analysis of the result shows there is propagation of the impact from SCADA WS to all other asset, but there is no propagation among other assets. This suggests packet capture was done at SCADA level, and not at a network device which typically intercepts more communication's packets.

Moreover, it is possible to observe that even if an asset does not run any service supporting the mission (for instance Microgrid assets) they are being reported as part of propagation paths to the mission, since they communicate back with the entry-point at SCADA that directly supports BP1 and BP2. This propagation behavior causes *propagation cycles*, as the impact propagates to already impacted assets, which constitutes a limitation of BIA's threat propagation heuristic that could be addressed by future work.

Nevertheless, according to the attack scenarios *Power Supply Interruption* and *Nuisance Tripping*, the impact should propagate from PLCs to IEDs which is not observable using the extracted asset connectivity. To improve MIA results, BIA's Connectivity Discovery component can be leveraged to consider firewall policy to infer new connectivity between the testbed assets, and is described in the next section.

3) *Using firewall policy*

In the previous case-study it was seen how BIA aids in discovering the organization's topology and connectivity from parsing packet captures, and how it affects the MIA result. At the same time, relying exclusively on the captured communications to profile the organization's connectivity raised some issues: (1) a vast connectivity panorama accentuates propagation cycles that leads the impact to propagate to every possible exploitable asset and subsequently, to the mission, and (2) since the packet capture was done at the SCADA workstation, only communications

[7] Internet Assigned Numbers Authority RFC 5771 guideline

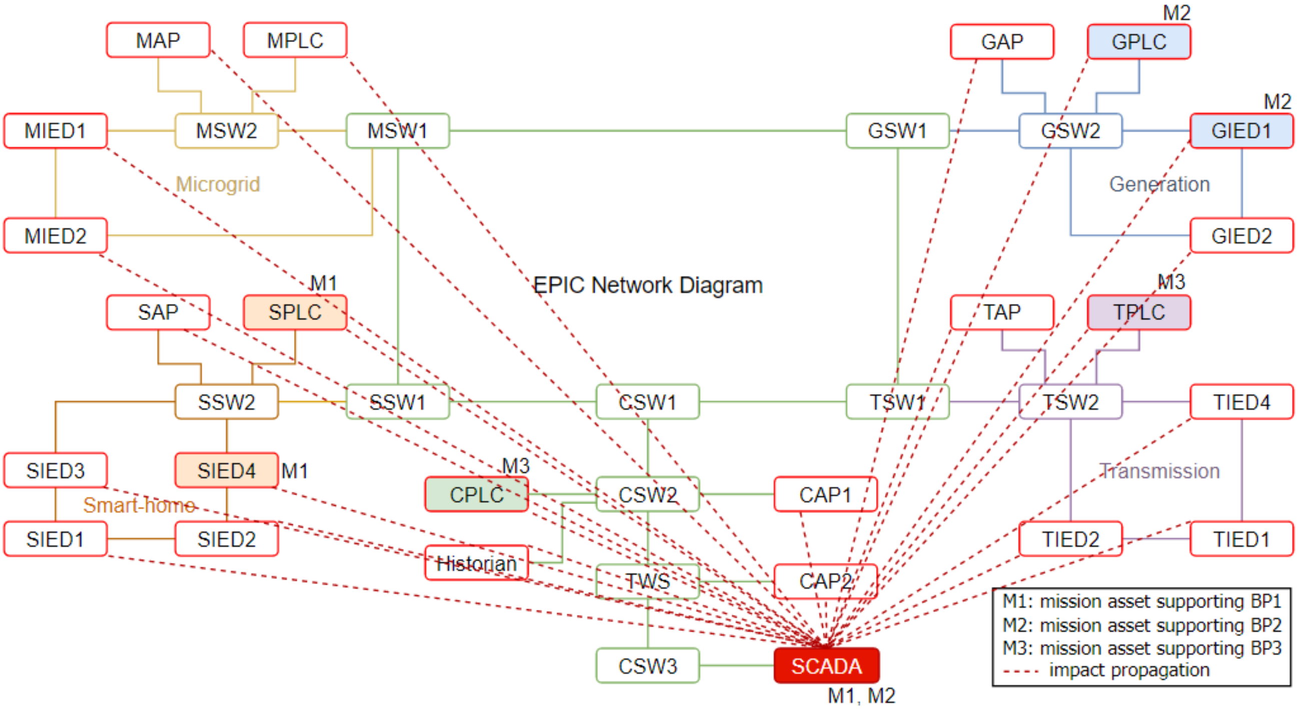

Figure 8 – BIA's MIA for EPIC topolgy extracted from packet capture files.

coming from or to the SCADA were taken into account. This, however, does not reflect the typical connectivity that does exist within a network with multiple assets where additional cross-communication exist. This is indeed the case for EPIC, where it is known that the PLCs also communicate with IEDs trough MMS, among others.

The aforementioned issues can be mitigated by leveraging BIA's Connectivity Discovery component to infer asset connectivity, either exclusively from firewall policy, or in addition to the connectivity already discovered. To study how this feature influences BIA's MIA result, two routers with firewall functionality are introduced to the testbed to create the following high-level network infrastructure, depicted in Figure 9.

The previously discovered assets are grouped into 24-bit netmasks according to EPIC's process stages and architecture. Even without firewall rules in place, this component automatically infers connectivity between assets on the same subnet upon the assumption they communicate freely (any protocol, using any network ports).

Additionally, rules can be added to restrict connectivity between subnets. To study how this feature influences BIA's MIA, a set of 20 *deny* and *allow* rules, to block and allow communications respectively, is introduced for both firewalls. In this way, in addition to the direct connectivity coming from the SCADA to all other assets, the resulting cross connectivity among other assets will be based on:

- Each stage's PLCs, IEDs and APs communicate with each other;
- CPLC communicates with all other PLCs;

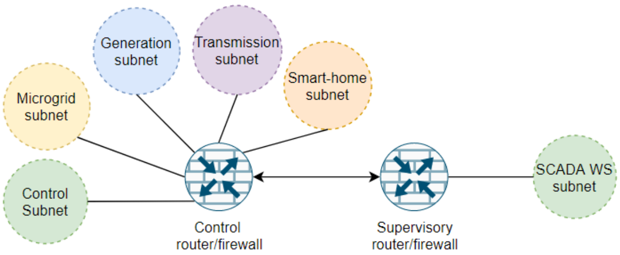

Figure 9 - High-level network infrastructure (routers and subnets) added to testbed.

- control assets (CPLC, Historian, CAP1 and CAP2) communicate with each other;
- main AP (CAP1) communicates with all other APs.

The Connectivity Discovery component first uses the *Comparing Algorithm* to determine the communications that are effectively allowed and then the *Filtering Algorithm* to classify each rule according to its source and destination and apply them. From the 20 rules given as input, 200 *connection* entries were inferred: 190 from assets on the same subnet connecting freely with each other, and 10 from assets on different subnets that successfully represents the connectivity panorama described above.

With firewall policy in place and new connectivity inferred, let us simulate mission impact from the direct connectivity originated on SCADA to the CPLC and Historian, to understand how these control assets, if successfully compromised, can propagate the impact throughout the organization's infrastructure.

This simulation results in the impact propagation illustrated in Figure 10 where it is possible to observe all BPs (and the activities and services supporting them) can be impacted, as well as 20 assets.

Even though the mission is equally impacted as in the first simulation, a careful read of the report shows how this simulation shows new propagation paths to impact the mission. This happens because in the previous simulation there was no log of cross communications between assets other than with SCADA, which BIA's Connectivity Discovery was able to infer from firewall policy. Additionally, it shows the wireless network can also be leveraged to propagate the impact to the mission.

While an attacker is required to execute only a single attack path that leads to his objective, the defender is required to secure all possible paths. Therefore, recognizing available attack paths is especially important for MIA.

Furthermore, these results show how BIA's Connectivity Discovery can be used to better reflect organization's asset connectivity considering firewall policy: indeed, with this simulation, both *Power Supply Interruption* and *Nuisance Tripping* attack scenarios were correctly detected as well as their impact in the defined BPs.

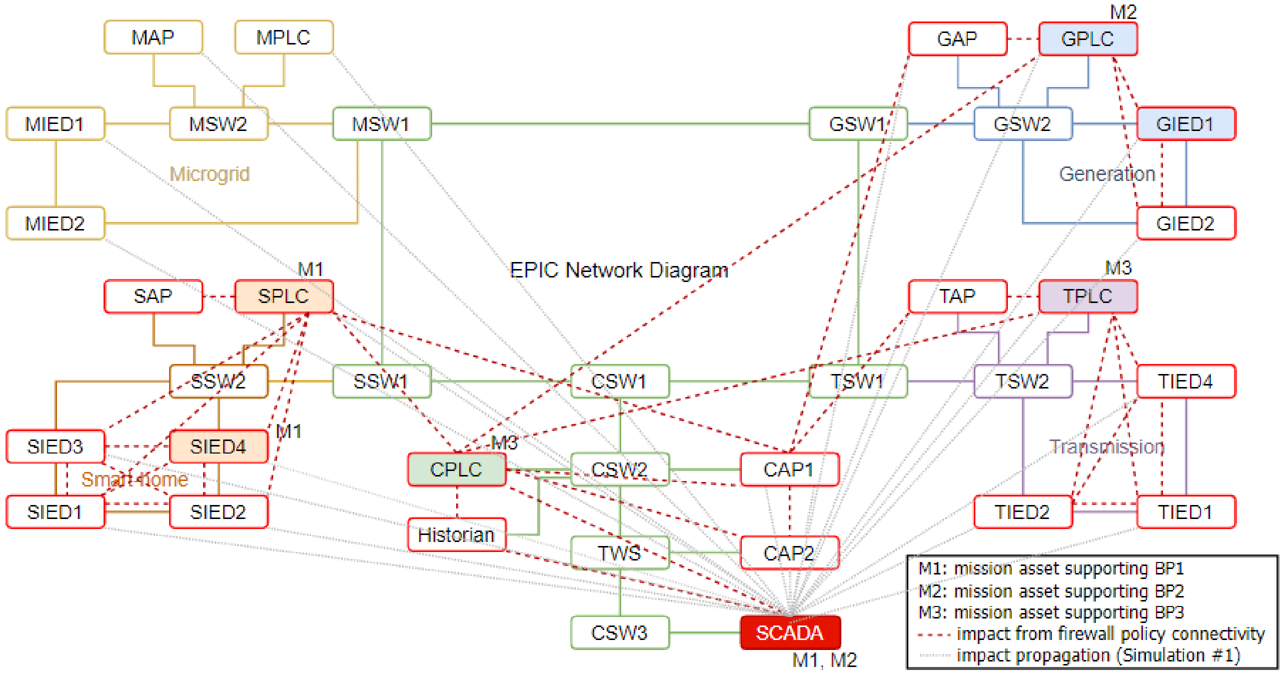

Figure 10 - MIA for asset connectivity inferred from firewall policy.

VI. Conclusion

This article has presented a novel approach for MIA. To do so, a comprehensive survey of current approaches to MIA was performed that led to the identification of three main stages of MIA: impact *modelling*, impact *propagation* and impact *measurement*.

Regarding the modelling stage, it was found that several works resort to an entity dependency graph with a multi-layered structure to profile the organization with various abstract layers, however most kept to a more conceptual approach, not identifying how their models can be populated which supports the motivation that the required knowledge for profiling an organization is often difficult to obtain.

Next, a review of propagation methodologies drove the classification of three types of model-based propagation approaches: logic-based, probabilistic-based and sensitivity-based, which a further analysis of their features drove to the decision of employing a logic-based propagation method by the present work.

As a result, this paper presents BIA, a two-stage approach to address the modelling and propagation aspects of MIA. In a first stage, BIA proposes profiling an organization with four abstraction layers that is able to map cyber-threats to the mission business-processes. In a second stage, BIA concretizes a simulation platform that allows to simulate mission impact, caused by an exploited cyber-threat chosen by the user.

To prove BIA's effectiveness in accomplishing its goal, a series of experiments was developed upon ICS iTrust EPIC testbed, to assess BIA's capability of detecting the impact of realistic attack scenarios on EPIC. Applying BIA has shown how its features can be leveraged and limitations mitigated, and how it can successfully generate a relevant report on mission impact.

A. Achievements

In reaching its goal, this dissertation accomplished two important achievements. The first achievement is the construction of a four-layer evaluation model for MIA, that offers a way to profile an organization and model the impact, which included a rarely considered threat layer that allows mapping cyber-threats onto the organization's assets. The second achievement is the concretization of a simulation platform that allows to simulate the mission impact caused by an exploited cyber-threat. Moreover, from the development and application of BIA, some other contributions can be noted:

- BIA is capable of converting disparate information about assets, threats, firewall policy and business-processes into an impact assessment report.
- Its implementation incorporated existing and established tools, such as *Tshark* network analyzer, MulVAL attack graph which knowledge base was reformulated, *Neo4j* graph database and *BP-IDS* intrusion detection system, as well as known standards, as *IPTABLES* for the organization's asset connectivity and firewall policy, and *STRIDE* to classify threats. Additionally, previous work on inspecting firewall policies was leveraged to infer asset connectivity allowed by firewall hierarchies.
- BIA was built in a way that is independent from the organization's domain (military, business or ICS), however, its application was done on an ICS, where not only a cyber network can become a target, but the physical network can also be impacted, which reinforces the need for MIA.

B. Future Work

From its accomplished achievements it can be concluded that BIA serves as consolidate baseline tool for MIA, and, as such, numerous options exist towards further development and improvement of the approach. In regard to BIA's assumptions and limitations, the main contributions would be based on:

- *Solving propagation cycles.* BIA's most important challenge to address would be the propagation cycles generated by bidirectional asset connectivity. This can be approached either before the simulation takes place, by automatically refining asset connectivity considered, during the simulation with additional Horn Clauses, or upon the attack graph generated by MulVAL by transforming the resulting directed cyclic graph to a tree of attack paths.

- *Integrating new horizontal dependencies*. Other great contribution to BIA is to integrate horizontal dependencies on other assessment layers to create a more authentic simulation.
- *Refining threat propagation heuristics*. The proposed heuristic for threat propagation was based on if an asset is accessible and has a threat associated with it then it can be compromised. This is not always the case, where other conditions must be present for a threat to be exploited. One immediate possibility would be to map threats onto the services the assets run, and only propagate the impact to other assets if there is connectivity to the port the service runs on.

Moreover, during BIA's design and application some interesting opportunities arisen for future work to address as an extension:

- *Inclusion of impact metrics*. The most compelling contribution would be to integrate impact metrics in BIA's model and simulation. This could be done with qualitative and quantitative metrics at any assessment layer.
- *Visualization of results*. An interesting view of BIA's report would be trough visualization, which is currently under development by other works.

REFERENCES

[1] A. Cherepanov and R. Lipovsky, "Industroyer: Biggest threat to industrial control systems since Stuxnet," 2017.

[2] X. He, S. Rass, and H. Meer, "Threat Assessment for Multistage Cyber Attacks in Smart Grid Communication Networks," Fakultät für Informatik und Mathematik Universität Passau, Germany, 2017.

[3] S. Jajodia, S. Noel, P. Kalapa, M. Albanese, and J. Williams, "Cauldron: Mission-centric cyber situational awareness with defense in depth," *Proc. - IEEE Mil. Commun. Conf. MILCOM*, no. May 2017, pp. 1339–1344, 2011.

[4] B. J. Argauer and S. J. Yang, "VTAC: virtual terrain assisted impact assessment for cyber attacks," 2008.

[5] G. Jakobson, "Extending Situation Modeling with Inference of Plausible Future Cyber Situations," in *2011 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), Miami Beach, FL*, 2011.

[6] C. Liu, A. Singhal, and D. Wijesekera, "A layered graphical model for mission attack impact analysis," in *2017 IEEE Conference on Communications and Network Security (CNS)*, 2017, pp. 602–609.

[7] A. Negotiation, "Cyber-Argus: Modeling C2 Impacts of Cyber Attacks," *19th ICCRTS*, pp. 1–17, 2014.

[8] L. Gilbert, C. Henney, L. Alford, A. Khalili, and B. Michalk, "Impact modeling and prediction of attacks on cyber targets," *Cyber Secur. Situat. Manag. Impact Assess. II; Vis. Anal. Homel. Def. Secur. II*, vol. 7709, p. 77090M, 2010.

[9] A. Motzek, R. Möller, M. Lange, and S. Dubus, "Probabilistic mission impact assessment based on widespread local events," *Assess. Mission Impact Cyberattacks*, p. 1, 2015.

[10] A. Barreto, P. Costa, and E. Yano, "A Semantic Approach to Evaluate the Impact of Cyber Actions to the Physical Domain," *Semant. Technol. Intell. Defense, Secur. 2012*, vol. 966, pp. 64–71, 2012.

[11] J. R. Goodall, A. D'Amico, and J. K. Kopylec, "Camus: Automatically mapping cyber assets to missions and users," *Proc. - IEEE Mil. Commun. Conf. MILCOM*, pp. 1–7, 2009.

[12] Y. Sun, T. Wu, X. Liu, and M. S. Obaidat, "Multilayered Impact Evaluation Model for Attacking Missions," *IEEE Syst. J.*, vol. 10, no. 4, pp. 1304–1315, Dec. 2016.

[13] B. Genge, I. Kiss, and P. Haller, "A system dynamics approach for assessing the impact of cyber attacks on critical infrastructures," *Int. J. Crit. Infrastruct. Prot.*, vol. 10, pp. 3–17, 2015.

[14] H. Orojloo and M. A. Azgomi, "A method for evaluating the consequence propagation of security attacks in cyber–physical systems," *Futur. Gener. Comput. Syst.*, vol. 67, pp. 57–71, 2017.

[15] S. Musman, A. Temin, M. Tanner, D. Fox, and B. Pridemore, "Evaluating the impact of cyber attacks on missions," *5th Eur. Conf. Inf. Manag. Eval. ECIME 2011*, pp. 446–456, 2011.

[16] S. Musman and A. Temin, "A Cyber Mission Impact Assessment Tool," in *2015 IEEE International Symposium on Technologies for Homeland Security (HST)*, 2015, pp. 1–7.

[17] S. Noel, E. Harley, K. H. Tam, M. Limiero, and M. Share, *CyGraph: Graph-Based Analytics and Visualization for Cybersecurity*, 1st ed., vol. 35, no. January 2016. Elsevier B.V., 2016.

[18] G. Jakobson, "Mission cyber security situation assessment using impact dependency graphs," *14th Int. Conf. Inf. Fusion*, pp. 1–8, 2011.

[19] S. Noel *et al.*, "Analyzing Mission Impacts of Cyber Actions (AMICA)," *NATO IST128 Work. Cyber Attack Detect.*, no. 15, pp. 1–16, 2015.

[20] C. Cao, L. P. Yuan, A. Singhal, P. Liu, X. Sun, and S. Zhu, "Assessing attack impact on business processes by interconnecting attack graphs and entity dependency graphs," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2018, vol. 10980 LNCS, pp. 330–348.

[21] I. Kotenko and A. Chechulin, "A Cyber Attack Modeling and Impact Assessment Framework," in *2013 5th International Conference on Cyber Conflict (CyCon)*, 2013, pp. 1–24.

[22] P. A. Porras, M. W. Fong, and A. Valdes, "A mission-impact-based approach to INFOSEC alarm correlation," *Int. Work. Recent Adv. Intrusion Detect.*, vol. 2516, pp. 95–114, 2002.

[23] X. Sun, A. Singhal, and P. Liu, "Who Touched My Mission : Towards Probabilistic Mission Impact Assessment," *SafeConfig '15 Proc. 2015 Work. Autom. Decis. Mak. Act. Cyber Def.*, pp. 21–26, 2015.

[24] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "Computing the impact of cyber attacks on complex missions," in *2011 IEEE International Systems Conference*, 2011, pp. 46–51.

[25] M. Lange, M. Krotofil, and R. Möller, "Mission Impact Assessment in Power Grids," in *Proceedings of the NATO IST-128 Workshop: Assessing Mission Impact of Cyberattacks*, pp. 51–59.

[26] R. Sawilla and X. Ou, "Identifying critical attack assets in dependency attack graphs," 2008.

[27] N. Kheir *et al.*, "Assessing the risk of complex ICT systems," *Ann. Telecommun.*, vol. 73, pp. 95–109, 2018.

[28] M. R. Grimaila and L. W. Fortson, "Towards an Information Asset-Based Defensive Cyber Damage Assessment Process," *Proc. 2007 IEEE Symp. Comput. Intell. Secur. Def. Appl. (CISDA 2007)*, no. Cisda, pp. 206–212, 2007.

[29] I. Kotenko and E. Doynikova, "Evaluation of Computer Network Security based on Attack Graphs and Security Event Processing."

[30] J. Holsopple and S. J. Yang, "FuSIA: Future situation and impact awareness," *Proc. 11th Int. Conf. Inf. Fusion, FUSION 2008*, 2008.

[31] A. Kim, M. Kang, J. Luo, and A. Velasquez, "A Framework for Event Prioritization in Cyber Network Defense," 2014.

[32] X. (Simon) Ou, W. F. Boyer, and S. Zhang, "MulVAL: A logic-based enterprise network security analyzer," *14th USENIX Secur. Symp.*, 2013.

[33] A. Kott, J. Ludwig, and M. Lange, "Assessing Mission Impact of Cyberattacks: Toward a Model-Driven Paradigm," *IEEE Secur. Priv.*, vol. 15, no. 5, pp. 65–74, 2017.

[34] P. Cichonski, T. Millar, T. Grance, and K. Scarfone, "Computer Security Incident Handling Guide Recommendations of the National Institute of Standards and Technology."

[35] M. R. Grimaila, L. W. Fortson, and J. L. Sutton, "Design considerations for a cyber incident mission impact assessment (CIMIA) process," 2009.

[36] M. Abedin, S. Nessa, L. Khan, and B. Thuraisingham, "Detection and Resolution of Anomalies in Firewall Policy Rules," in *IFIP Annual Conference on Data and Applications Security and Privacy*, 2006, no. Data and Applications Security XX, pp. 15–29.

[37] INOV, "BPIDS: Using business process specification to leverage intrusion detection in public transportation."

[38] S. Adepu, N. K. Kandasamy, and A. Mathur, "EPIC: An electric power testbed for research and training in cyber physical systems security," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 11387 LNCS, no. November, pp. 37–52, 2019.

[39] A. Siddiqi, N. O. Tippenhauer, D. Mashima, and B. Chen, "On Practical Threat Scenario Testing in an Electric Power ICS Testbed," vol. 7, 2018.

[40] ENISA, *Communication network dependencies for ICS/SCADA Systems*, no. December. 2016.