Calculating Business Impact Assessment of Cyber-Threats

Diogo Alves
Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal
diogo.m.alves@tecnico.ulisboa.pt

***Abstract*—Organizations are becoming increasingly more reliant on information and communication technology to support their day-to-day operations, which includes the storage and access of critical information. Unfortunately, this dependency on ICT systems leaves organizations vulnerable to cyber-attacks, which can cause serious damage to their business-processes. By estimating the impact caused by a given cyber-attack in a particular organization, it is possible to prioritize the mitigation actions and preventative measures to be considered in the risk management procedure. This paper presents the *Business Impact Calculator* (BusICalc) methodology. BusICalc was designed to offer a method capable of quantifying the impact that a cyber-threat would cause, once exploited, to the organization's business-processes. A proof-of-concept of BusICalc was developed for evaluation purposes and integrated with the risk analysis system, BIA (*Business Impact Assessment*). The proposed methodology was evaluated using a dataset corresponding to a *Critical Infrastructure*, and the conducted experiments show that BusICalc is scalable and effective in yielding reasonable values for the impact of cyber-threats.**

***Keywords*—Cyber-Attack, Impact Propagation, Business-Process Modelling, Cascading Effects, Impact Quantification, Security**

I. Introduction

As ICT (Information and Communication Technology) systems become more common in the control and monitoring of *Critical Infrastructures* (i.e., systems considered so essential that their failures would have significant effects on public health, safety, or economic security, which includes, among others, energy, water supply, transport, and communications [1]), the risk of cyber-attacks capable of compromising the operations of such infrastructures increases [2]. Moreover, considering that there are *interdependencies* between different infrastructures, the compromise of the operations of one *Critical Infrastructure* can, in turn, cause failures in other infrastructures that are dependent on the first, in a process known as *Cascading Failures*.

Examples of such attacks include the BlackEnergy and Industroyer [3] malwares. The BlackEnergy and Industroyer were both responsible for cyber-attacks to the Ukrainian power grid, the first in December of 2015 and the second in December of 2016. In the first attack, BlackEnergy was able to exploit remote access software to cut off power to around 250 000 households for six hours. A year later, Industroyer managed to deprive Ukraine's capital, Kiev, of power for an hour by taking control of electricity substation switches and circuit breakers.

This paper presents BusICalc (*Business Impact Calculator)*, an impact assessment tool that is capable of quantifying the impact of the propagation of a cyber-threat across an organization. More specifically, the methodology should calculate the impact that a given cyber-threat can have, once exploited, on the critical business-processes of an organization. For instance, if the considered organization is a *Critical Infrastructure*, then its critical business-processes correspond to the correct delivery of the infrastructure's essential services to its customers, whose disruption would cause severe economic and/or reputational damage (e.g., in an electrical grid, the critical business-process of interest would be the reliable delivery of power to the grid's customers). Hence, this methodology would allow the identification of the most impactful threats to the organization, which in turn would contribute to prioritize the mitigation actions and preventative measures to be taken in the risk management procedure. This paper offers the following contributions:

- Calculation of an impact value (between 0 and 1) that evaluates the level of operationality loss suffered by the business-processes of an organization to a simulated attack. This value is computed by leveraging the structure of a given organization (i.e., the network connections between devices, the threats they are vulnerable to, the services they provide, and the interconnections between the activities that comprise the organization's business-processes) in order to simulate the propagation of a chosen entry-point threat.
- Development of a proof-of-concept that integrates the impact calculation methodology with an existing risk assessment system — BIA (*Business Impact Assessment*) ([4]).

The evaluation process conducted to test the efficacy of BusICalc used a model of a small-scale electric smart-grid as the testbed, and BusICalc was proved successful in computing the impact of cyber-threats on the objectives of such system. Additionally, the application of the methodology to a critical business-process of a *Critical Infrastructure* shows that the compromise of these types of processes can lead to failures in other *Critical Infrastructures* (*Cascading Failures*). For example, a disruption in the delivery of power would affect

the electric pumps, which would make the water distribution system inoperable. It was also shown that the developed proof-of-concept is scalable.

The remainder of this paper is structured as follows: Section II presents a literature review on methods for impact assessment and propagation of cyber-threats; Section III explains the design process behind the development of the BusICalc methodology; Section IV presents the implementation details of the development of this tool; Section V describes the experiments conducted to evaluate BusICalc; and Section VI concludes the paper.

II. Literature Review

The propagation of cyber-threats, and the respective impact assessment, have been studied in the literature by means of Risk Assessment Graphs ([5]), Vulnerability-Asset-Service-Misson models ([4, 6, 7]), logic-based Attack Graphs ([4, 8, 9, 10, 11]), and Bayesian-based Attack Graphs ([12, 13, 14]).

From these methods, one particular work — BIA (Business Impact Assessment) ([4]) — represents a good starting point to achieve the desired functionalities. BIA is a framework for impact assessment that is able to (1) profile an organization using a four-layer model, which includes cyber-threats, assets (i.e., devices), services, and business-process activities; and (2) perform simulations of threat propagation paths across the modelled organization. The main feature missing from this framework is the capability of quantifying the impact of a cyber-threat propagation.

To solve this issue, the work developed by Jakobson [6] proposes an impact quantification algorithm, which is used in the context of a four-layer model (VASM) similar to the one used by BIA. According to this model (Figure 1), the impact is calculated taking into consideration the inter and intra-dependencies between the missions (i.e., business-processes), activities, services, and assets of an organization, as well as the vulnerabilities that affect each asset.

This work will take advantage of both BIA ([4]) and the method proposed by Jakobson [6] in order to construct a methodology capable of simultaneously simulating the propagation of a chosen cyber-threat through an organization, and estimating a metric for the impact of this propagation.

Such a methodology can be used to manage risk, by prioritizing cyber-threats according to the risk they pose to a specific organization, since the risk of a threat is often given by the product of its impact and its probability ([15]). Moreover, a possible area of application of this methodology is in estimating *Cascading Effects*, since the works that study *Cascading Failures* in the literature, both in the context of *Critical Infrastructure* (CI) systems ([16, 17, 18, 19, 20, 21, 22, 23]) and Supply Chains ([24, 25, 26]), require some sort of impact metric for each individual organization in order to simulate the spread of *Cascading Failures* throughout a system of interconnected organizations.

III. Business Impact Calculator (BusICalc)

The proposed approach — Business Impact Calculator (BusICalc) — was designed with the goal of simulating the propagation of a user-chosen cyber-threat throughout an organization, and estimating the impact of that propagation on the organization's business-processes, by yielding an impact metric.

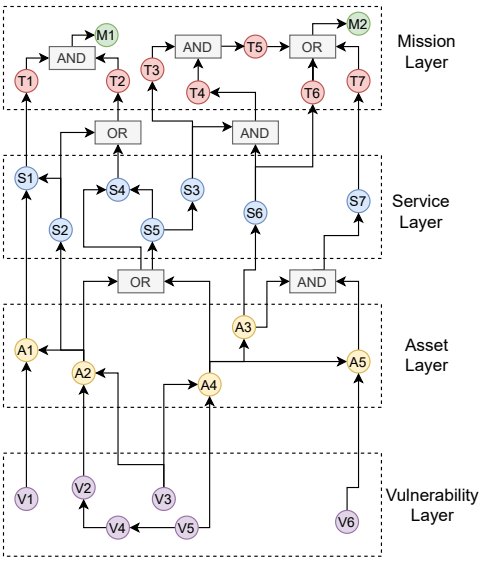

Fig. 1. VASM model used by Jakobson [6].

As a result, BusICalc improves upon the *Business Impact Assessment* methodology ([4]). The main weakness of BIA is not providing a quantitative metric for the impact of a given attack. BusICalc aims at enhancing BIA by computing an estimate of the impact of a BIA simulation.

In this context, it is important to define the concept of *impact*: *impact* refers to the loss of operationality of an organization's business-processes, considering that a specific attack has occurred, characterized by a specific entry-point threat and propagation path through the organization's network. Based on this definition, and considering that BIA uses a version of the VASM model to represent the network, the most adequate method for impact estimation, present in the literature, is the method proposed by Jakobson [6]. Hence, BusICalc will adapt this method to BIA for the purpose of impact calculation.

A. Layered model

Figure 2 presents an example of the layered model used by BIA, composed by the Asset, Service, and Activity Layers.

In the Asset Layer are represented physical devices of the network (yellow circles). Each of these assets contains a set of threats that can be exploited. The Asset Layer also models the connectivity between different assets by allowing each asset to belong to a subnet (green circles) and routers (grey circles) to establish communication between different subnets.

In the Service Layer, each service (blue circles) (e.g., Operating System, Middleware, Applications), is carried out by one or more assets. It is assumed, for simplicity, that if a service is provided by two different assets, each of the assets provides the whole service, such that the multiple assets have the purpose of redundancy.

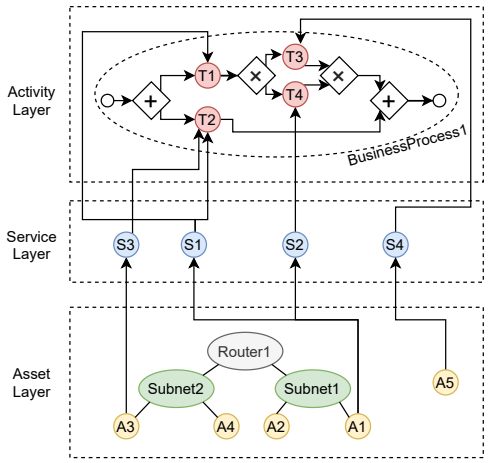

Fig. 2. Example of network model used by BIA.

The topmost layer — Activity Layer — contains the business-processes and corresponding activities (red circles). An activity corresponds to an action that is carried in the context of a business-process. Like before, an activity can be provided by one or more services, and a single service provides the activity entirely.

A business-process is defined as a sequence of activities with a start and an end, and can be modelled through a Business Processes Modelling Notation (BPMN) Diagram [27], depicted in the Activity Layer of Figure 2. Two types of nodes are defined in the business-process diagram — *parallel gateways* (diamonds with "+" inside) and *exclusive gateways* (diamonds with "x" inside). These *gateways* establish the rules for the flow of activities in the business-process, in the following ways:

- A *parallel gateway* functions as an AND, i.e., the activities that belong to the branches leaving the *parallel gateway* must all be executed in order to conclude the execution of the business-process.
- The *exclusive gateway* functions as an OR, i.e., from the branches leaving this type of *gateway*, only the activities belonging to one of the branches need to be executed in order to conclude the execution of the business-process.

B. Propagation Paths

BIA ([4]) is able to identify the individual propagation paths of a threat through an organization. These paths can be referred to as *trivial paths*. They receive this designation because they only contemplate a single route from a threat to an activity belonging to a business-process. In practice, it means that each of these paths will start at a user-selected threat, then it will contain a series of assets through which the threat propagates, and finally a single service and a single activity belonging to a business-process.

In order to discard paths that contain infinite loops, it is also assumed that these trivial paths cannot go through the same node twice, which is particularly relevant for Router and Subnet nodes.

Figure 3 presents the four trivial paths for the example network, in which the entry-point is a threat on Asset A3.

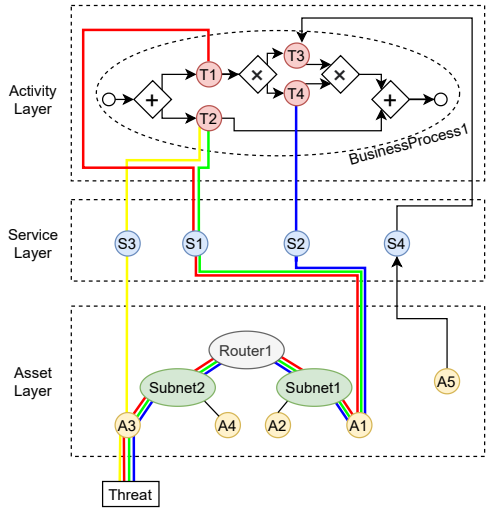

Fig. 3. Four trivial paths (yellow, red, green, blue) for the entry-point in Asset A3.

The main limitation with considering only the trivial paths individually is that it becomes impossible to obtain a single value for the impact of a complex attack. For example, suppose an attacker gains control over a given asset and decides to compromise all the services that are run by that asset. In this scenario, for each compromised service there would be at least one trivial path, since each trivial path only contains one service. This implies that the only way to study the impact of the attack would be to apply the impact quantification algorithm to each of these n trivial paths that correspond to the attack. The result would be n different values of impact, which would be hard to interpret.

This work proposes merging all the trivial paths that result from a given attack scenario into a single merged path. This would allow the impact quantification algorithm to be applied to a single path, and as a result, only one value for the impact would be obtained, making it easier to analyse.

For this reason, BusICalc provides the option of merging a set of user-selected trivial paths into a single merged path. This option can be used either to manually select a set of trivial paths to merge, or to merge trivial paths according to some predefined condition. For example, it can be useful to group all the trivial paths that affect a specific business-process.

C. Impact Calculation

This section will describe the algorithm developed for impact calculation in BusICalc. The goal of the algorithm is to compute a value for the impact of the propagation of the entry-point threat (chosen by the user), which is propagated through a given path $\mathcal{P}$ — $I_{\mathcal{P}}$ — (also referred to simply as the impact of the path $\mathcal{P}$), computed by Equation 8. Here, the concept of *impact* differs from the concepts of *probability* and *risk* (in fact, the *risk* is often given by the product of *impact* and *probability* [15]). The purpose of this metric is

estimating the impact that a given cyber-threat may cause, once exploited, to the business-processes of the organization. Hence, this metric can assume any value between 0 and 1. If a given path $\mathcal{P}$ has an impact of 0 (i.e., $I_{\mathcal{P}} = 0$), then the business-processes were unaffected by the propagation of the entry-point threat through the path $\mathcal{P}$. On the other hand, if the path has an impact of 1 (i.e., $I_{\mathcal{P}} = 1$), then the propagation of the entry-point threat through the organization using path $\mathcal{P}$ has resulted in the complete lack of operationality of the business-processes of the organization.

Considering that:

- T is the set of all threats;
- D is the set of all assets (devices);
- S is the set of all services;
- A is the set of all activities;
- BP is the set of all business-processes;
- $T(d) \subset T$ is the set of threats that affect asset $d \in D$;
- $D(s) \subset D$ is the set of assets that run service $s \in S$;
- $S(a) \subset S$ is the set of services that provide activity $a \in A$;
- $A(bp) \subset A$ is the set of activities that support the business-process $bp \in BP$;
- $E(bp)$ is the set of execution threads belonging to the business-process $bp \in BP$;
- $d_{entrypoint} \in D$ is the entry-point asset (user-chosen);
- $t_{entrypoint} \in T$ is the entry-point threat under analysis (user-chosen);
- $P_i = (D^i, s^i, a^i)$ defines a trivial path, in which:
  - $D^i = \{d_0^i, \ldots, d_K^i\} \subset D$ is the set of affected assets, ordered such that d_0^i is the entry-point asset ($d_0^i = d_{entrypoint}$), d_1^i is the next asset compromised, and so on;
  - $s^i \in S$ is the affected service;
  - $a^i \in A$ is affected activity;
- $\mathcal{P} = \{P_1, \ldots, P_N\}$ represents a generic merged path that aggregates the trivial paths $P_1, \ldots, P_N$. Here, it is a necessary condition that $d_0^1 = d_0^2 = \cdots = d_{entrypoint}$, i.e., the entry-point asset is the same for all trivial paths that comprise the merged path;
  - $\mathcal{S} = \{s^i | \forall P_i \in \mathcal{P}\}$ is the set of services affected by the merged path $\mathcal{P}$ (which corresponds to the set of services affected by each of the trivial paths that comprise $\mathcal{P}$);
  - $\mathcal{A} = \{a^i | \forall P_i \in \mathcal{P}\}$ is the set of activities affected by the merged path $\mathcal{P}$ (which corresponds to the set of activities affected by each of the trivial paths that comprise $\mathcal{P}$);
- IF_t is the Impact Factor of threat t;
- OC_d^{Asset}, $OC_s^{Service}$, $OC_a^{Activity}$, $OC_e^{EThread}$, $OC_{bp}^{BProcess}$ are, respectively, the Operational Capacities of asset d, service s, activity a, execution thread e, and business-process bp;
- $I_{\mathcal{P}}$ is the impact of the propagation of the entry-point threat through the path $\mathcal{P}$.

The algorithm starts by calculating the Impact Factor (*IF*) of each threat. Here, the rationale is that a threat exploits a specific vulnerability in an asset. So, the Impact Factor of the threat, which measures the degree to which it is capable of compromising the attacked asset, is calculated based on the Common Vulnerability Scoring System (CVSS) [28] score of the compromised vulnerability, according to Equation 1, in order to obtain a value between 0 and 1 (since the CVSS score ranges between 0 and 10).

$$IF_t := \frac{CVSS_t}{10}, \forall t \in T \tag{1}$$

The algorithm then assigns to each node — Asset, Service and Activity — an Operational Capacity (*OC*) of 1 (Equation 2). This parameter is a measure of the operationality of the node, that can assume values between 0 and 1, where a value of 1 means the node is fully operational and a value of 0 means the node is completely inoperable. By assigning a value of 1 in the beginning, the assumption is that every node starts fully operational before the simulated attack.

$$\begin{cases} OC_d^{Asset} := 1, \forall d \in D \\ OC_s^{Service} := 1, \forall s \in S \\ OC_a^{Activity} := 1, \forall a \in A \end{cases} \tag{2}$$

Then, for each trivial path that composes the generic path $\mathcal{P}$, the algorithm will update the assets' OCs according to Equation 3 — the Operational Capacity of the asset directly affected by the entry-point threat (i.e., the entry-point asset) is decreased by an amount equal to the Impact Factor of the entry-point threat, whereas the OCs of the remaining assets in the trivial path are either updated to the Operational Capacity of the previous asset in the path, or are lowered by an amount equal to the Impact Factor of their most impactful threat, depending on whichever yields a smaller value. This means that the OC of the previous asset is carried over directly to the next asset, unless the next asset is affected by some threat that would make this value lower, in which case it is assumed that the attacker is able to compromise this threat and lower the OC of the asset.

$$\forall P_i \in \mathcal{P}: \begin{cases} OC_{d_0^i}^{Asset} := max(1 - IF_{t_{entrypoint}}, 0) \\ OC_{d_n^i}^{Asset} := min(OC_{d_{n-1}^i}^{Asset}, \\ \qquad max(min_{t \in T(a_n^i)}(1 - IF_t), 0)), \\ \qquad n = 1, \ldots, K \end{cases} \tag{3}$$

Next, the algorithm will update the Operational Capacities of the services affected by the path $\mathcal{P}$ — $s \in \mathcal{S}$ — according to Equation 4. In practise, it means that the OC of each affected service will be updated to the average of the OCs of the assets that run that service. The reason why the avg operator is used is because it is assumed that each asset runs the full service, as explained in Section III-A, which corresponds to an OR-node in the method described in Jakobson [6]. If, instead, every asset was necessary to run the service, this would correspond

to an AND-node, and the operator min would be used instead of avg.

$$OC_s^{Service} := avg_{d \in D(s)}(OC_d^{Asset}), \forall s \in \mathcal{S} \quad (4)$$

Likewise, for the affected activities — $a \in \mathcal{A}$ — their Operational Capacities are updated according to Equation 5, to the average of the OCs of the services that provide each activity.

$$OC_a^{Activity} := avg_{s \in S(a)}(OC_s^{Service}), \forall a \in \mathcal{A} \quad (5)$$

The next step is computing the OCs of the business-processes. In order to understand how they are computed, it is first necessary to understand the concept of *execution threads*. An execution thread corresponds to a minimum sequence of activities that, once executed, concludes the execution of the business-process. For example, consider the business-process depicted in Figure 2. Since it contains an *exclusive gateway*, only one of either activity $T3$ or $T4$ needs to be executed in a given execution instance. For this reason, the business-process has the following execution threads:

- $e_1 = \{T1, T2, T3\}$;
- $e_2 = \{T1, T2, T4\}$.

In this case, although the set $\{T1, T2, T3, T4\}$ would also conclude the execution of the business-process, it is not considered an execution thread, since it is not a "minimum sequence", i.e., it contains redundant activities (either $T4$ or $T3$ could be removed). Hence, the set of execution threads of business-process bp is solely comprised of e_1 and e_2, i.e., $E(bp) = \{e_1, e_2\}$.

With this, the algorithm will compute, for each business-process, the OCs of all its execution threads, according to Equation 6, i.e., the OC of an execution thread is the product of the OCs the activities that comprise it.

$$\forall bp \in BP : \\ OC_e^{EThread} := \prod_{a \in e}(OC_a^{Activity}), \forall e \in E(bp) \quad (6)$$

The last Operational Capacities computed are the OCs of business-processes. The OC of a business-process is computed by averaging the OCs of the execution threads that belong to it, as defined in Equation 7.

$$OC_{bp}^{BProcess} := avg_{e \in E(bp)}(OC_e^{EThread}), \forall bp \in BP \quad (7)$$

Finally, the impact of the generic path $\mathcal{P}$ on the organization — $I_{\mathcal{P}}$ — (i.e., the impact of the propagation of the entry-point threat through path $\mathcal{P}$), is given by the average of the loss of operationality of the business-processes that belong to the organization, as in Equation 8, where N_{BP} corresponds to the total number of business-processes. This Equation yields a value between 0 and 1 for the impact.

$$I_{\mathcal{P}} = \frac{\sum_{bp \in BP}(1 - OC_{bp}^{BProcess})}{N_{BP}} \quad (8)$$

IV. Implementation of BusICalc

The architecture of BusICalc is illustrated in Figure 4. The system is composed by two main modules — the *Setup Module* and the *Impact Calculation Module*. Due to its resourcefulness, *Python*[1] programming language was used for the development of these modules.

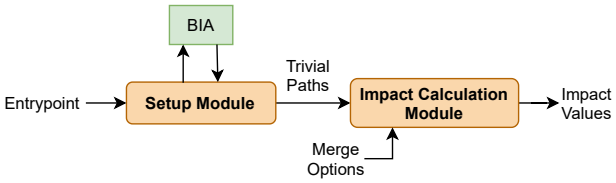

Fig. 4. Architecture of BusICalc.

In the Setup Module, BIA is invoked in order to obtain the propagation paths for a business network given an entry-point. These propagation paths correspond to the *trivial paths* in BusICalc. Then in the second module — Impact Calculation Module — the trivial paths will be merged according to user specification, and the algorithm described in Section III-C will be employed in order to determine the impact of each of the merged paths on the organization.

V. Evaluation

This section describes the dataset used to evaluate BusICalc, and the experiments conducted in order to test its viability.

A. Evaluation Setup

The dataset used to evaluate BusICalc is based on the EPIC (Electric Power Intelligent Control) testbed developed by iTrust Labs[2]. This testbed models a real scaled-down replica of a smart-grid, capable of generating up to 72kVA [29]. This testbed is used for research and experimentation of cyber security mechanisms in the context of *Critical Infrastructures* [30]. The dataset used in this work considers the set of assets in EPIC, as well as the respective connectivities (Figure 5), and a business-process built based on the description of EPIC's processes (Figure 6).

The network architecture containing the assets of the experimental dataset, as in EPIC, is depicted in Figure 5. The dataset contains six types of assets — SCADA (Supervisory Control and Data Acquisition), Historian, PLCs (Programmable Logic Controllers), IEDs (Intelligent Electronic Devices), SWs (Network Switches), and APs (Access Points). Each of the assets is prefixed by a letter — C, G, M, T, S — according to the stage it belongs to — *control*, *generation*, *microgrid*, *transmission*, and *smarthome*.

Besides the connectivity between the different assets, the entry-point vulnerabilities that each type of asset is susceptible to are also required, as well as their CVSS scores. Table I presents a list of the considered vulnerabilities for the asset

[1]https://www.python.org/
[2]https://itrust.sutd.edu.sg/testbeds/electric-power-intelligent-control-epic/

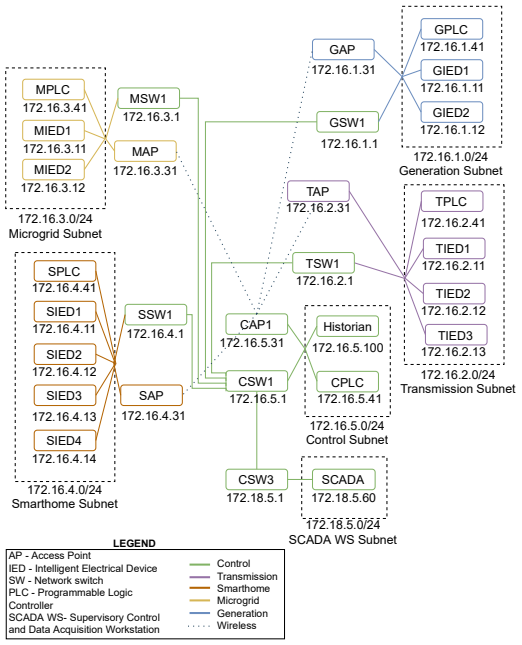

Fig. 5. EPIC's Network Diagram.

types SCADA, PLC, and IED, and their CVSS v3.1[3] Scores, taken from NIST's National Vulnerability Database[4].

TABLE I
LIST OF VULNERABILITIES CONSIDERED FOR EACH ASSET TYPE (SCADA, PLCS AND IEDS).

Type of Asset	Vulnerability ID	Description	CVSS Score
SCADA WS (Win7 machine running PCVue v11)	CVE-2020-26867	Execution of arbitrary code	9.8
	CVE-2020-26868	Denial of Service	7.5
	CVE-2020-26869	Information Disclosure	7.5
	CVE-2019-0752	Remote code execution	7.5
PLC (WAGO PFC200 running CoDeSys)	CVE-2018-5459	Execution of unauthorized commands	9.8
IED (Siemens Relays running SIPROTEC)	CVE-2019-10938	Execution of arbitrary code	9.8
	CVE-2019-19279	Denial of Service	7.5

In the evaluation process, it will be assumed that, by default, the attacker will choose to exploit the vulnerability CVE-2019-0752 in order to perform a *remote code execution* on the SCADA as the entry-point threat. This assumption derives from the fact that SCADA workstations are often the entry-point of cyber-attacks due to them being accessible through the Internet [31].

The business-process that will be considered for the evaluation will be the *power supply to the smarthome* illustrated on the diagram in Figure 6. The power for the *smarthome* can either come from the *generation* and *transmission* stages, or from the *microgrid* stage. The flow of power between the stages is controlled by circuit breakers (CBs) — CB1 (managed by GIED1) controls the output of power from the *generation* stage, and simultaneously input to the *transmission* stage; CB2 (managed by TIED2) controls the output of power from the *transmission* stage; CB4 (managed by MIED2) controls the output of power from the *microgrid* stage; and CB3 (managed by SIED4) controls the input of power to the *smarthome*.

Hence, the business-process diagram contains an *exclusive gateway* (G1) at the beginning, whose top branch corresponds to the supply of power in grid-connected mode, i.e., from the *generation* stage, whereas the bottom branch corresponds to the supply of power from the *microgrid*. In the top branch, there is then a *parallel gateway* (G2), with each of the following branches corresponding to different circuit breakers — CB1, CB2, and CB3. This means that, in grid-connected mode, these three circuit breakers need to simultaneously be closed in order to get power from the *generation* stage to the *smarthome*. Likewise, for the power supply in *microgrid* mode (bottom branch after the *exclusive gateway* G1), both the CB4 and CB3 circuit breakers need to be closed, hence the two branches diverging from the *parallel gateway* G3.

Each of the branches corresponding to the closing of a circuit breaker contains three activities — first, the command that originates from the SCADA to the corresponding stage's PLC, then the relay of this command from the PLC to the IED, and finally the close of the circuit breaker by the IED.

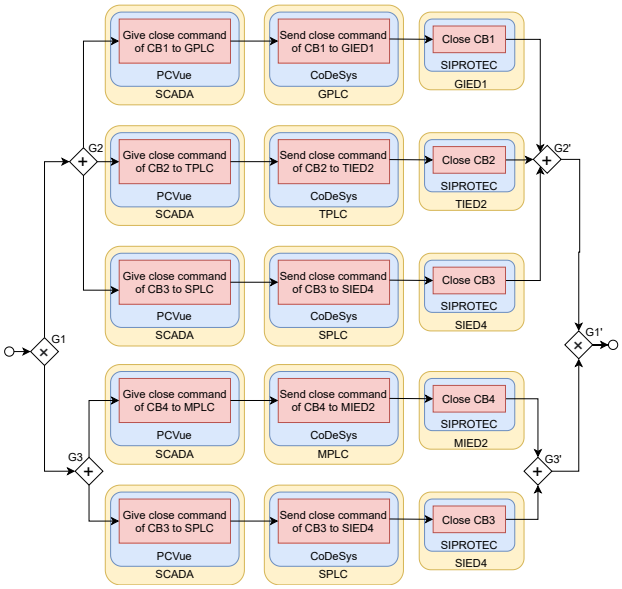

Fig. 6. Business-process diagram of the BP *Power supply to the smarthome*.

B. Evaluation Process

This section has the purpose of demonstrating the experiments performed on BusICalc in order to test its capabilities

[3]https://www.first.org/cvss/v3.1/specification-document
[4]https://nvd.nist.gov/

and limitations. The conducted experiments were designed with the aim of answering the following questions:

1) *How does the placement of the compromised activity(ies) inside the business-process influence the impact?* (Section V-B1)
2) *How does the merging of trivial paths influence the impact?* (Section V-B2)
3) *How does the path taken by the attacker influence the impact?* (Section V-B3)
4) *How does the entry-point threat influence the impact?* (Section V-B4)
5) *How much time does it take to compute the impact? Is the solution scalable?* (Section V-B5)

To answer these questions, a series of experiments were performed on the EPIC dataset, as described in the following sections.

1) Effect of impacted activities: The first experiment is aimed at studying whether BusICalc is able to produce a plausible value for the impact of a threat that is propagated through a given path, considering the significance of the affected activities to the business-process, by analysing whether the relative impact values match with what is expected. Two tests were conducted with this purpose: in the first test the path in Figure 7 was simulated, and in the second test the path in Figure 8.

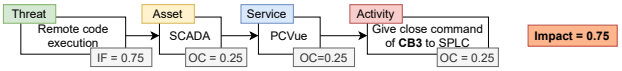

Fig. 7. Path simulated in the first test, with *Give close command of CB3 to SPLC* as the affected activity.

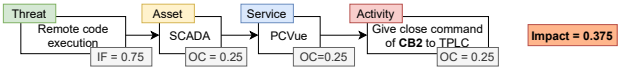

Fig. 8. Path simulated in the second test, with *Give close command of CB2 to TPLC* as the affected activity.

In the two tests, the entry-point is a *remote code execution* threat that affects the SCADA, and subsequently the PCVue service. The difference between the paths lies in the affected activity. In the first path, the affected activity is associated with circuit breaker CB3, while in the second path, the activity is associated with CB2.

According to EPIC's electrical diagram, the impact of the first path should be greater than the impact of the second path, since CB3 directly controls the supply of power to the *smarthome*, and CB2 only controls the output of power from the *transmission* stage, which means that even if CB2 becomes compromised, the supply of power to the *smarthome* is still possible through the *microgrid* stage.

This is in fact confirmed by the simulations — the impact calculated for the first path is 0.75 and for the second 0.375. These values are explained by the fact that the business-process contains two execution threads — the first thread contains all the activities above the *exclusive gateway* G1 in the business-process diagram (Figure 6), and the second thread contains all the activities below the *exclusive gateway* G1. The activity *Give Close Command of CB3 to SPLC* (*smarthome* PLC) belongs to both the execution threads, while the activity *Give Close Command of CB2 to TPLC* (*transmission* PLC) belongs to only one. This means that in the first path, both execution threads see their OC reduced, instead of just one in the second path. Hence, when computing the OC of the business-process (which is the average of the OCs of its execution threads (Equation 7)), it is natural that for the first path this value is smaller, which ultimately results in a greater value for its impact.

2) Effect of merging paths: In the second experiment, the goal is to examine the effect of the aggregation of trivial paths on the overall impact. With this purpose, a series of simulations were performed, first on a set of individual paths, and then on the aggregation of those paths. Figure 9 shows the trivial paths over which the simulations were performed, and Figure 10 shows the path that results from the merging of those trivial paths. All of the trivial paths have the same entry-point (*remote code execution*), and affect the same asset (SCADA) and service (PCVue), but each affects a different activity offered by this service. By merging the trivial paths, a path is obtained in which all the activities provided by the PCVue service are compromised.

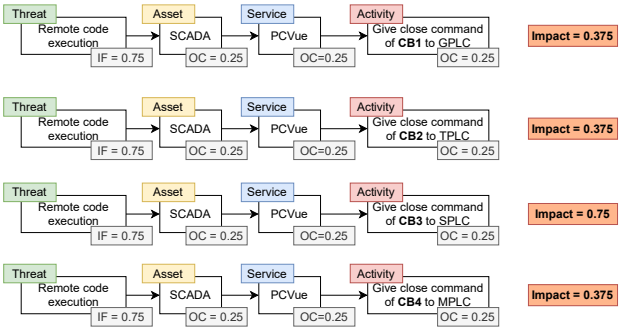

Fig. 9. Set of four paths simulated in the first test, each affecting a different activity.

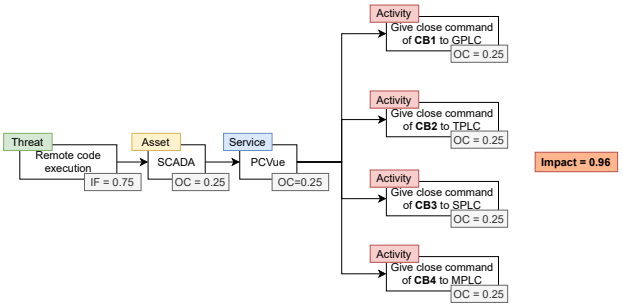

Fig. 10. Path simulated in the second test, that results from merging the trivial paths in the previous test.

Each of the trivial paths has an impact of either 0.375 or 0.75 (the impact of 0.75 corresponds to the path described in the previous section). The resulting impact of the merged

path is 0.96. This value is greater than the impact of any of the trivial paths that comprise the merged path. In fact, this result is a property of the employed algorithm — whenever two or more paths are merged, the impact of the resulting path is greater or equal than the impact of each of the comprising paths. This property is verified when gradually merging the paths in Figure 9, until arriving at the path in Figure 10 — merging the first two trivial paths (both with impact of 0.375), the resulting merged path has an impact of 0.469; merging the first three trivial paths (the first two with impact of 0.375, and the third with 0.75), the resulting merged path has an impact of 0.867; and finally the merging of the four trivial paths (three of them with impact of 0.375 and one with impact of 0.75) results in the path in Figure 10, with impact of 0.96. This behaviour is in fact coherent with reality, since the impact of an attack that compromises several activities of an organization's business-process must take all of the affected activities into consideration, rather than, for instance, only the activity that yields the highest impact.

3) Effect of compromised path: This section aims at analysing how the impact may vary according to the compromised path chosen.

For this experiment, two similar paths were simulated — path in Figure 11 and path in Figure 12. These two paths may not seem very similar at first look, but the activities compromised by each one are in fact equivalent. This means that if the activity *Give Close Command of CB1 to GPLC* on the first path has an OC of x, and the activity *Send Close Command of CB1 to GIED1* also has the same OC of x, then the two paths will have the same impact.

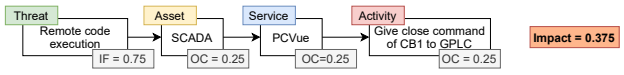

Fig. 11. Path simulated in the first test, only affecting the Asset SCADA.

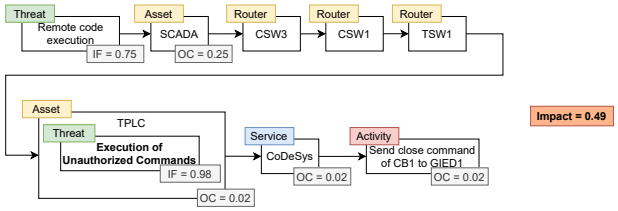

Fig. 12. Path simulated in the second test, affecting the Asset TPLC besides the SCADA.

The actual difference between the two paths that results in them having different impacts lies in the threat exploited in the second path. While the first path is the straightforward path already discussed in previous sections, in the second path the attacker leverages the connectivity of the SCADA in order to move laterally across the network (through routers CSW3, CSW1 and TSW1) until the asset TPLC is reached. In the TPLC, there is a new threat — *Execution of unauthorized commands*, with an Impact Factor of 0.98. Since this new threat has an Impact Factor greater than the original entry-point threat (which has an Impact Factor of 0.75), it is assumed that the attacker will choose to exploit this new threat in order increase the yielded impact on the organization (this behaviour is modelled by Equation 3 in the algorithm).

This exploitation, in turn, causes the OC of the TPLC to decrease below the OC of the SCADA. This ultimately results in the OCs of the service and activity exploited in the second path to be lower than the OCs of the service and activity of the first path, which means that the second path will present a greater value for the impact (0.49) when compared to the impact of the first path (0.375).

4) Effect of Entry-point Threat: In this section, the goal is to study the effect that the chosen entry-point threat might have on the impact. Thus, for the paths shown in the previous sections, a set of simulations was performed with varying Impact Factor for the entry-point threat, from 0 to 1 with a step of 0.01 (Figures 13 and 14), and furthermore from 0.97 to 1 with a step of 0.001 in the case where the first set of experiments was not conclusive (Figure 15).

The first path in which the variation of IF was studied is presented in Figure 7. Figure 13 shows the variation of the impact of this path with the Impact Factor of the entry-point threat. This figure shows that the impact of the path increases as the Impact Factor increases. Moreover, it also shows that this relationship is linear. The reason for this is that this is a trivial path that affects only one asset, which means that the OCs of the asset, service and activity are directly derived from the Impact Factor of the entry-point threat. As a result, the impact of the path is proportional to the Impact Factor.

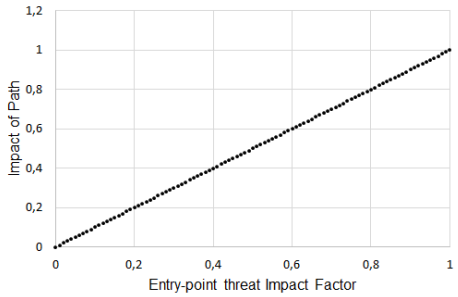

Fig. 13. Relationship between Impact Factor of the entry-point threat and impact of the path in Figure 7.

The next path to be evaluated is the path in Figure 10. This path differs from the previous one because it no longer compromises a single activity. Instead, it compromises a total of four activities, all provided by the same service and asset. As a result, Figure 14 shows that the relationship between the Impact Factor of the entry-point threat and the impact of the path is no longer linear, but instead polynomial. This happens because the OC of each affected activity is derived from the IF of the entry-point threat, as in the previous paths, but in order to compute the impact, these activities' OCs are multiplied by each other according to Equation 6, which results in a polynomial relationship between IF and impact.

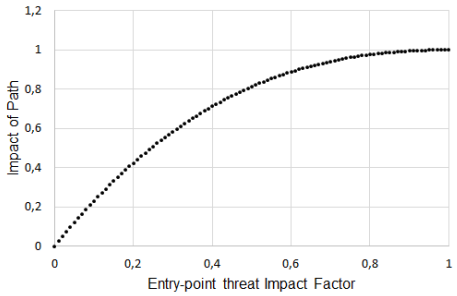

Fig. 14. Relationship between Impact Factor of the entry-point threat and impact of the path in Figure 10.

Next, the path in Figure 12 was evaluated using this method. The resulting relationship between the Impact Factor of the entry-point threat and the impact of the path is presented in Figure 15. According to this figure, the impact of the path is constant and equal to 0.49 for IF lower or equal than 0.98, and then increases linearly with IF, from 0.49 to 0.5, when the IF changes from 0.98 to 1. To understand this behaviour, it is important to comprehend this path. This is a trivial path in which the attacker moves from the entry-point asset (SCADA) to another asset (TPLC). The TPLC contains a threat — *Execution of Unauthorized Command* — with Impact Factor of 0.98. This means that, if the entry-point threat has a lower Impact Factor than 0.98, the attacker will choose to exploit the threat in the TPLC, which yields a higher impact (as explained in Section V-B3), and hence the IF of the entry-point threat will not affect the final impact of the path, which is why Figure 15 shows a constant impact of 0.49 for an IF of the entry-point threat lower than 0.98. On the other hand, if the entry-point threat has a higher IF than the threat in the TPLC, then the attacker will not exploit the threat in the TPLC, and the OCs of the assets, service and activity will be directly derived from the IF of the entry-point threat, resulting in a linear relationship between the impact of the path and the entry-point IF.

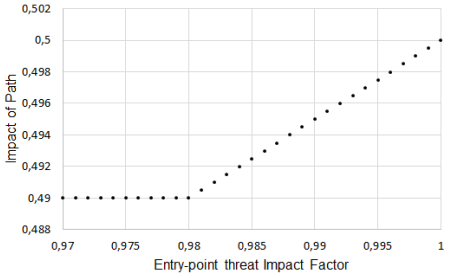

Fig. 15. Relationship between Impact Factor of the entry-point threat and impact of the path in Figure 12.

5) Performance Evaluation: In order to evaluate the performance of BusICalc, several simulations were made using a dataset of variable size.

From these simulations, it was concluded that the majority of the total computational time is spent waiting for the processing of BIA — on average, the Impact Calculation Module only took around 0.027% of the total time of a simulation, while the remaining 99.973% was spent on the Setup Module, of which 81.5% of the time, on average, was spent in the processing of BIA.

In order to evaluate whether the solution for impact calculation is scalable, it is necessary to analyse the relationship between the execution time of the Impact Calculation Module, and the number of assets in the network. Figure 16 presents the results obtained from the simulations.

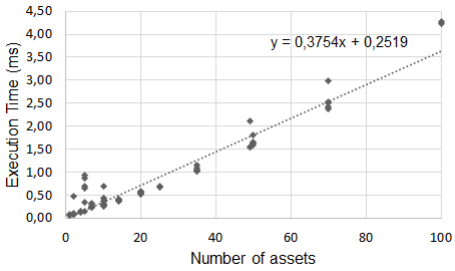

Fig. 16. Relationship between the number of assets of the network and the execution time of the Impact Calculation Module.

This figure shows that there is a linear relationship between the number of assets and the execution time of the Impact Calculation Module. This means that the solution is in fact scalable, since the required time to run the impact calculation algorithm only increases by a constant amount for each asset that is added to the network, which means that the algorithm has a complexity of $O(n)$, where n is the number of assets in the network.

VI. Conclusions

With the goal of providing a methodology capable of quantifying the impact caused by a given cyber-threat in an organization, this paper has studied methods that tackle impact propagation and assessment of cyber-threats, from which two methods have proven especially useful (BIA [4] and Jakobson [6]). Also, the study of methods regarding *Cascading Effects* has emphasized the advantage that this methodology can have in supplying impact metrics to be used in the simulations of *Cascading Failures*.

The BIA (*Business Impact Assessment*) methodology was used as a starting point to achieve the desired functionalities. The main feature missing from BIA is the capability of quantifying the impact of a cyber-threat propagation. The developed approach — BusICalc (*Business Impact Calculator*) — solves this issue by implementing an impact quantification algorithm, based on the work proposed by Jakobson [6]. As a result, BusICalc is able to simulate the propagation of a user-selected cyber-threat across an organization's network to determine which business-process activities have been affected, and subsequently assign an impact value to that scenario.

In order to test BusICalc's efficacy, a set of experiments were conducted, in which the considered testbed was modelled after a smart-grid. These experiments have shown that BusICalc is capable of producing coherent impact metrics

for distinct situations that consider different sets of attack paths and exploited threats, ultimately proving it successful in delivering its primary objective. The experiments have also shown that the developed proof-of-concept is scalable, since it has a complexity of $O(n)$, where n represents the size of the dataset.

With the development of BusICalc, this work has created a methodology capable of quantifying the impact delivered by a simulated attack on the critical business-processes of an organization, with the option of configuring the simulation to better replicate the attacker's behaviour. Additionally, the tool can help in identifying the most impactful threats, that should be considered in the organization's risk management procedure. Moreover, it has also demonstrated that BusICalc is successful in calculating the impact of cyber-threats on physical processes (in this case, delivery of power to a specific section of a smart-grid).

In this context, the methodology could be further improved to study *Cascading Effects* between different organizations/infrastructures by modelling the physical components of the organization (e.g., Circuit Breakers, Power Lines, Generators, Pumps, Valves, Motors, Sensors), and their respective interdependencies, which would allow the simulation of the propagation of failures among the interconnected organizations. For instance, the original EPIC testbed supplies power to a water treatment plant — Secure Water Treatment (SWaT) — and to a water distribution system — Water Distribution (WADI). By modelling the dependency of the physical components of these systems (e.g., pumps, valves) on the energy supplied by EPIC, it would be possible to simulate the *Cascading Effects* that result from a failure on the supply of power.

REFERENCES

[1] Cybersecurity and Infrastructure Security Agency, "Critical Infrastructure Sectors," https://www.cisa.gov/critical-infrastructure-sectors, 2020, accessed: 14-04-2021.

[2] W. Hurst, N. Shone, and Q. Monnet, "Predicting the effects of DDoS attacks on a network of critical infrastructures," in *2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing*. IEEE, 2015, pp. 1697–1702.

[3] A. Cherepanov and R. Lipovsky, "Industroyer: Biggest threat to industrial control systems since stuxnet," *WeLiveSecurity, ESET*, vol. 12, 2017.

[4] C. Köpke, K. Srivastava, L. König, N. Miller, M. Fehling-Kaschek, K. Burke, M. Mangini, I. Praça, A. Canito, O. Carvalho *et al.*, "Impact Propagation in Airport Systems," *Cyber-Physical Security for Critical Infrastructures Protection*, vol. 12618, p. 191, 2020.

[5] N. Kheir, A. R. Mahjoub, M. Y. Naghmouchi, N. Perrot, and J.-P. Wary, "Assessing the risk of complex ICT systems," *Annals of Telecommunications*, vol. 73, no. 1, pp. 95–109, 2018.

[6] G. Jakobson, "Mission cyber security situation assessment using impact dependency graphs," in *14th international conference on information fusion*. IEEE, 2011, pp. 1–8.

[7] Y. Sun, T.-Y. Wu, X. Liu, and M. S. Obaidat, "Multilayered impact evaluation model for attacking missions," *IEEE Systems Journal*, vol. 10, no. 4, pp. 1304–1315, 2014.

[8] R. E. Sawilla and X. Ou, "Identifying critical attack assets in dependency attack graphs," in *European Symposium on Research in Computer Security*. Springer, 2008, pp. 18–34.

[9] 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 *IFIP Annual Conference on Data and Applications Security and Privacy*. Springer, 2018, pp. 330–348.

[10] S. Noel, S. Jajodia, L. Wang, and A. Singhal, "Measuring security risk of networks using attack graphs," *International Journal of Next-Generation Computing*, vol. 1, no. 1, pp. 135–147, 2010.

[11] C.-W. Ten, C.-C. Liu, and M. Govindarasu, "Vulnerability assessment of cybersecurity for scada systems using attack trees," in *2007 IEEE Power Engineering Society General Meeting*. IEEE, 2007, pp. 1–8.

[12] N. Poolsappasit, R. Dewri, and I. Ray, "Dynamic security risk management using bayesian attack graphs," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 1, pp. 61–74, 2011.

[13] M. Frigault and L. Wang, "Measuring network security using bayesian network-based attack graphs," in *2008 32nd Annual IEEE International Computer Software and Applications Conference*. IEEE, 2008, pp. 698–703.

[14] M. Frigault, L. Wang, A. Singhal, and S. Jajodia, "Measuring network security using dynamic bayesian network," in *Proceedings of the 4th ACM workshop on Quality of protection*, 2008, pp. 23–30.

[15] Avital Koren, "Risk Management in ISO 9001," https://isoupdate.com/resources/risk-management-in-iso-9001/, 2019, accessed: 15-09-2021.

[16] D. Rehak, P. Senovsky, M. Hromada, T. Lovecek, and P. Novotny, "Cascading impact assessment in a critical infrastructure system," *International journal of critical infrastructure protection*, vol. 22, pp. 125–138, 2018.

[17] I. B. Utne, P. Hokstad, and J. Vatn, "A method for risk modeling of interdependencies in critical infrastructures," *Reliability Engineering & System Safety*, vol. 96, no. 6, pp. 671–678, 2011.

[18] Y. Y. Haimes and P. Jiang, "Leontief-based model of risk in complex interconnected infrastructures," *Journal of Infrastructure systems*, vol. 7, no. 1, pp. 1–12, 2001.

[19] A. E. Motter and Y.-C. Lai, "Cascade-based attacks on complex networks," *Physical Review E*, vol. 66, no. 6, p. 065102, 2002.

[20] W. Wang, S. Yang, F. Hu, H. E. Stanley, S. He, and M. Shi, "An approach for cascading effects within critical infrastructure systems," *Physica A: Statistical Mechanics and its Applications*, vol. 510, pp. 164–177, 2018.

[21] L. Duenas-Osorio and S. M. Vemuru, "Cascading failures in complex infrastructure systems," *Structural safety*, vol. 31, no. 2, pp. 157–167, 2009.

[22] C. Köpke, J. Schäfer-Frey, E. Engler, C. P. Wrede, and J. Mielniczek, "A joint approach to safety, security and resilience using the functional resonance analysis method," in *8th REA Symposium on Resilience Engineering: Scaling up and Speeding up*, 2020.

[23] O. H. Ramirez Agudelo, C. Köpke, and F. Sill Torres, "Bayesian Network Model for Accessing Safety and Security of Offshore Wind Farms," 2020.

[24] Y. Wang and F. Zhang, "Modeling and analysis of under-load-based cascading failures in supply chain networks," *Nonlinear Dynamics*, vol. 92, no. 3, pp. 1403–1417, 2018.

[25] Q. Yang, C. M. Scoglio, and D. M. Gruenbacher, "Robustness of supply chain networks against underload cascading failures," *Physica A: Statistical Mechanics and its Applications*, vol. 563, p. 125466, 2021.

[26] Y. Zeng and R. Xiao, "Modelling of cluster supply network with cascading failure spread and its vulnerability analysis," *International Journal of Production Research*, vol. 52, no. 23, pp. 6938–6953, 2014.

[27] M. Dumas, M. La Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of business process management*. Springer, 2017, vol. 1.

[28] P. Mell, K. Scarfone, and S. Romanosky, "Common vulnerability scoring system," *IEEE Security & Privacy*, vol. 4, no. 6, pp. 85–89, 2006.

[29] S. Adepu, N. K. Kandasamy, and A. Mathur, "Epic: An electric power testbed for research and training in cyber physical systems security," in *Computer Security*. Springer, 2018, pp. 37–52.

[30] A. Siddiqi, N. O. Tippenhauer, D. Mashima, and B. Chen, "On practical threat scenario testing in an electric power ICS testbed," in *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, 2018, pp. 15–21.

[31] ENISA, "Communication network dependencies for ICS/SCADA Systems," 2016.