Fall 12-2020

Cyber Defense Remediation in Energy Delivery Systems

Kamrul Hasan
Old Dominion University

CYBER DEFENSE REMEDIATION IN ENERGY DELIVERY SYSTEMS

by

Kamrul Hasan
B.Sc. November 2006, Bangladesh University of Engineering & Technology
M.Sc. August 2016, Tennessee State University

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTATIONAL MODELING & SIMULATION ENGINEERING

OLD DOMINION UNIVERSITY
December, 2020

Approved by:

Sachin Shetty (Director)

Chunsheng Xin (Member)

Hong Yang (Member)

Amin Hassanzadeh (Member)

ABSTRACT

CYBER DEFENSE REMEDIATION IN ENERGY DELIVERY SYSTEMS

Kamrul Hasan
Old Dominion University, 2020
Director: Dr. Sachin Shetty

The integration of Information Technology (IT) and Operational Technology (OT) in Cyber-Physical Systems (CPS) has resulted in increased efficiency and facilitated real-time information acquisition, processing, and decision making. However, the increase in automation technology and the use of the internet for connecting, remote controlling, and supervising systems and facilities has also increased the likelihood of cybersecurity threats that can impact safety of humans and property. There is a need to assess cybersecurity risks in the power grid, nuclear plants, chemical factories, etc. to gain insight into the likelihood of safety hazards. Quantitative cybersecurity risk assessment will lead to informed cyber defense remediation and will ensure the presence of a mitigation plan to prevent safety hazards. In this dissertation, using Energy Delivery Systems (EDS) as a use case to contextualize a CPS, we address key research challenges in managing cyber risk for cyber defense remediation.

First, we developed a platform for modeling and analyzing the effect of cyber threats and random system faults on EDS's safety that could lead to catastrophic damages. We developed a data-driven attack graph and fault graph-based model to characterize the exploitability and impact of threats in EDS. We created an operational impact assessment to quantify the damages. Finally, we developed a strategic response decision capability that presents optimal mitigation actions and policies that balance the tradeoff between operational resilience (tactical risk) and strategic risk.

Next, we addressed the challenge of management of tactical risk based on a prioritized cyber defense remediation plan. A prioritized cyber defense remediation plan is critical for effective risk management in EDS. Due to EDS's complexity in terms of the heterogeneous nature of blending IT and OT and Industrial Control System (ICS), scale, and critical processes tasks, prioritized remediation should be applied gradually to protect critical assets. We proposed a methodology for prioritizing cyber risk remediation plans by detecting and evaluating critical EDS nodes' paths. We conducted evaluation of critical nodes characteristics based on nodes' architectural positions, measure of centrality based on nodes' connectivity and frequency of network traffic, as well as the controlled amount of electrical power. The

model also examines the relationship between cost models of budget allocation for removing vulnerabilities on critical nodes and their impact on gradual readiness. The proposed cost models were empirically validated in an existing network ICS test-bed computing nodes criticality. Two cost models were examined, and although varied, we concluded the lack of correlation between types of cost models to most damageable attack path and critical nodes readiness.

Finally, we proposed a time-varying dynamical model for the cyber defense remediation in EDS. We utilize the stochastic evolutionary game model to simulate the dynamic adversary of cyber-attack-defense. We leveraged the Logit Quantal Response Dynamics (LQRD) model to quantify real-world players' cognitive differences. We proposed the optimal decision-making approach by calculating the stable evolutionary equilibrium and balancing defense costs and benefits. Case studies on EDS indicate that the proposed method can help the defender predict possible attack action, select the related optimal defense strategy over time, and gain the maximum defense payoff. We also leveraged software-defined networking (SDN) in EDS for dynamical cyber defense remediation. We presented an approach to aid the selection security controls dynamically in an SDN-enabled EDS and achieve tradeoff between providing security and Quality of Service (QoS). We modeled the security costs based on end-to-end packet delay and throughput. We proposed a non-dominated sorting based multi-objective optimization framework which can be implemented within an SDN controller to address the joint problem of optimizing between security and QoS parameters by alleviating time complexity at $O(MN^2)$. The M is the number of objective functions, and N is the population for each generation, respectively. We presented simulation results that illustrate how data availability and data integrity can be achieved while maintaining QoS constraints.

To the memory of my father, Md.Khalilur Rahaman
My beloved mother, Mrs.Morzina Rahaman
My dear wife, Nusrat Bashunia
My dear sons, Nehan Al Hasan & Adyan Al Hasan
My brothers, Md.Mahbubul Hassan & Md.Imrul Hasan
and my sister Jakia Sultana

ACKNOWLEDGEMENTS

First, I would like to express my most generous gratitude to my advisor Dr. Sachin Shetty for the tremendous support, guidance, and unwavering encouragement of my Ph.D. study. He shares his immense knowledge and experience and cutting-edge research directions, and rigorous working style with me, which inspires me to keep learning and being creative throughout my Ph.D. study for now and will benefit the rest of my life in the future.

Second, I want to thank my other committee members, Dr. Hong Yang, Dr. Chunsheng Xin (Department of Electrical & Electronic Engineering, Old Dominion University), and Dr. Amin Hassanzadeh (Cyber Fusion Lab, Accenture, Arlington, Virginia), for their insightful suggestions and discussions about the research. These suggestions motivate me to reexamine my research from various perspectives and give me many inspirations, which help me complete my Ph.D. research study.

I also wish to recognize the assistance and support from the Department of Modeling and Simulation Engineering and the Virginia Modeling, Analysis, and Simulation Center (VMASC) at Old Dominion University.

Finally, I would like to give my most incredible gratitude to my friends and families for their spiritual support throughout writing this thesis and my life in general.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

This chapter provides the background and motivation for the dissertation followed by the main goals of the research. Finally, we present the basic outline of this dissertation thesis.

1.1 BACKGROUND AND MOTIVATION

1.1.1 BACKGROUND

The integration of Information Technology (IT) and Operational Technology (OT) in Cyber-Physical Systems (CPS) has brought significant efficiencies and facilitated real-time information acquisition, processing, and decision making [2]. Analyzing CPS risks such as those found in the power grid, nuclear plants, chemical factories, etc. is of crucial importance given the hazards linked to these systems (explosion, dispersion, etc.). Organizations have the flexibility to determine the optimal strategies to conduct risk management activities that can be distinguished by the level of rigor, granularity, and information sharing. Organizations utilize risk management methodologies, models, and systems addressing safety and financial risk. The standard CPS, risk management framework, is depicted in Figure 2 [2].

Tier 1 addresses risk from an organizational perspective. It is responsible for considering strategic risk in the risk management program. Strategic risk characterizes the adverse impacts on an organization upon pursuing a particular course of action. Tier 2 addresses risk from a mission/business process perspective. They are informed by the risk context, risk decisions, and risk activities at Tier 1 and the tactical and technical knowledge and activities of Tier 3. Tier 3 addresses risk from a system perspective and is guided by the risk context, risk decisions, and risk activities at Tiers 1 and 2. Tier 3 risk is also known as the tactical risk of an organization.

Traditional industries were based on mechanical devices and sometimes closed systems. Only safety-related risks generated from accidental component failures and human errors

need to be addressed during these industries' risk analysis. However, today, industries are influenced by digital technology development related to instrumentation and Industrial Automation (IA). Supervisory Control And Data Acquisition (SCADA) systems are introduced to monitor and control equipment that deals with critical and time-sensitive materials or events [2]. The rise in the degree of automation increases the degree of complexity and communication among systems that have increased the attack surface and has impacted physical systems' safety. Thus, it is imperative to consider all three layers' risk for influential cybersecurity and safety risk analysis of CPS.

Quantifying and analyzing these significant risks contributes to better decision making and ensures that risks are managed according to defined acceptance criteria. A prioritized cyber defense remediation plan is critical for effective risk management in CPS. The judicious selection of countermeasures (patching, asset redundancy, firewall rules, etc.) at the operational level is crucial for prioritized cyber defense. These low-level selections ultimately determine the upper mission risk and strategic risk. Therefore, it is also crucial to consider negative side-effects of response plans and individual mitigation actions. There are costs associated with any countermeasure geared towards preventing a proactive infiltration of a network and individual nodes. For instance, shutting down a node in a system will inevitably reduce this node's operational resilience with a probability of one.

Further, employing a patch on a node could have an impact on operational resilience. We call this reduction of operational resilience an impact on a node. Moreover, what needs to be considered is that any local impact may spread throughout a network. If a node is highly dependent on receiving information from a node that has been shut down, it will not operate as intended anymore. In the end, it may be worse- from an operational perspective assuring mission success- to defend or eliminate an attack surface by action, i.e., one sacrifices mission success for a false sense of security by a too narrow perspective on the problem.

1.1.2 MOTIVATION FROM LITERATURE REVIEW

Traditionally, safety assessment has been associated with accidental risks caused by component failures, human errors, or any non-deliberate source of hazard. In contrast, security is related to malicious activities that are induced by cyber or physical means. Besides, attacks

targeting safety are considered rare events with low frequency, and security incidents occur more frequently [3]. The highly-publicized Stuxnet worm is an example of intentional attack and of how vulnerabilities within IT systems can be used to target a Programmable Logic Controller (PLC) in Industrial Control Systems (ICS).

Kriaa et al. [4] study the differences and similarities between safety and security in the context of ICS but the authors did not indicate if the cyber threats can also cause safety hazards in ICS. Several researchers [3] [5] also have studied the impact of cyber attacks on safety that lead to significant accidents. Although the authors have considered security and safety risks at the operational level (tactical risk), they did not consider the tactical risk's impact on the business/mission process risk and strategic risk. Risk assessment has to consider the tactical, mission, and strategic risk perspectives to gain a holistic view of the cyber risk to the organization [2].

Granadillo et al. [6] and Motzek et al. [7] have shown how the operational risk propagates to business impact. They have proposed a mitigation plan at the operational level to mitigate the impact on the mission. This approach overcomes the limitation of previous work by taking into account the business/mission impact. However, the authors did not consider the strategic risk and safety risk from the operational perspective [8]. The authors also assessed the effect of compromising a node at an operational level in a qualitative fashion. Using the 2003 North East blackout as an example, Anderson et al. [9] discussed the strategic risk and presented several remediation policies to mitigate the risk. Considering all of those drawbacks, we are using the Energy Delivery System (EDS) as a specific instance of CPS in this dissertation defense. We propose a methodology to quantify the safety and security risk in the EDS infrastructure based on nodes' criticality and model its impact on business/mission risk and strategic risk.

According to MITRE cyber threat mitigation database [10], currently, there are 40 cyber defense remediation options for known cyber-attack. Vulnerability scanning and patching security vulnerabilities are some of the most frequently used remedial options. Vulnerability scanning is used to find potentially exploitable software vulnerabilities to remediate them.

Vulnerability and Patching Management (VPM) continues to be a heavily manual intensive process in the energy sector. Energy companies spend a tremendous amount of human

resources digging through vulnerability bulletins, determining asset applicability, and determining remediation and mitigation actions. The U. S. energy sector faces a unique and formidable challenge in vulnerability and patch management. The NERC patching requirements in $CIP-007-6R2$ [11] heavily incentivize flawless vulnerability mitigation. It is not uncommon for utilities to have several hundred software vendors to monitor, several thousand vulnerabilities to assess, and tens of thousands of patches or mitigation actions to implement. Whereas most companies in other sectors do risk-based patching, electric utilities must address every patch in a short period. Operators have to analyze every vulnerability and determine the corresponding remediation action

Many vulnerability and patch management automation tools have been developed for traditional IT networks, such as Symantec Patch Management, Patch Manager Plus by ManageEngine, Asset Management by SysAid, and Patch Manager by Solarwinds. These VPM solutions mainly address security issues for operating systems such as Windows, Mac, and Linux, and the applications running on these systems. They can automatically discover vulnerabilities and deploy available patches. For example, Symantec Patch Management [12] can detect security vulnerabilities for various operating systems, and Microsoft applications and Windows applications. It can provide vulnerability and patch information to operators, but it cannot analyze vulnerabilities and make decisions about remediation actions by itself. Patch Manager Plus by ManageEngine [13] discovers vulnerabilities and patches, and then automates the deployment of patches for Windows, Mac, Linux, and third-party applications. These solutions are mainly designed for commonly used operating systems and applications in traditional IT systems, but cannot be applied to electric systems mostly for two reasons. On the one hand, they cannot handle vulnerabilities for control system devices such as Programmable Logic Controller (PLC), which are very important and common in electric systems. On the other hand, these solutions mostly deploy all available patches automatically regardless of asset or system differences, which is infeasible in electric systems since it may interrupt the system service.

Some VPM solutions have been explicitly provided for electric systems by companies such as Flexera, FoxGuard Solutions, and Leidos [11]. The primary function of these solutions

is to provide applicable vulnerabilities for electric systems. They ask for software information from utilities, find relevant vulnerabilities and patches for the software, and then send appropriate vulnerability information to utilities. They cannot analyze vulnerabilities against the operating environment and make prioritized decisions on how to address the vulnerabilities. Some public vulnerability databases are also available such as the National Vulnerability Database (NVD) and Exploit Database [14] to help drive VPM automation. NVD publishes discovered security vulnerabilities and provides the information and characteristics of these vulnerabilities [14]. Exploit Database includes information about whether vulnerabilities can be exploited.

To ensure the security and reliability of power systems, NERC developed a Critical Infrastructure Protection (CIP) Cyber Security Reliability Standards to define security controls applying to identified and categorized cyber systems. It represents the requirements for Security Patch Management in $CIP - 007 - 6R2$. It requires the utilities to (1) identify patch sources for all installed software and firmware, (2) identify appropriate security patches every month, and (3) determine whether to apply the security patch or mitigate the security vulnerability. Recognized patching sources must be evaluated at least once every 35 calendar days for applicable security patches. For those patches that are applicable, they must be applied within 35 calendar days. For the vulnerabilities that cannot be patched, a mitigation plan must be developed, and a timeframe must be set to complete these mitigations.

In the research area, some work has been done to analyze vulnerabilities and patches to understand vulnerabilities better. Stefan et al. [15] explored discovery, disclosure, exploit, and patch dates for about 8000 public vulnerabilities. Shahzad et al. [16] studied the evolution of vulnerability lifecycles such as disclosure date, patch date, and the duration between patch date and exportability date, and extracted rules that represent the exploitation of hackers and the patch behavior of vendors. The work in studied software vendors' patch release is such as how quickly vendors patch vulnerabilities and how vulnerability disclosure affects patch release. Li and Paxson [17] investigated the duration of a vulnerability's impact on a codebase, the timeliness of patch development, and the degree to which developers produce safe and reliable fixes. Li et al. [18] evaluated vulnerabilities of the installed software

version and the latest version and then decided whether to update the software based on the value of the Common Vector Scoring System (CVSS) score [19]. Most of these analyzed datasets are retrieved from public vulnerability databases, such as NVD and Open Sourced Vulnerability Database (OSVDB) [20]. Still, they do not combine vulnerability metrics with organizational context to analyze decision making.

A prioritized cyber defense remediation plan is critical for effective risk management in the Energy Delivery System (EDS). Due to the complexity of EDS in terms of heterogeneous nature blending Information Technology (IT) and Operation Technology (OT) and Industrial Control System (ICS), scale, and critical processes tasks, prioritized remediation should be applied gradually to protect critical assets. In this dissertation defense, we propose a methodology for prioritizing cyber risk remediation plans by detecting and evaluating paths to critical nodes in EDS. We suggest critical nodes characteristics evaluation based on nodes' architectural positions, measure of centrality based on nodes' connectivity and frequency of network traffic, and the controlled amount of electrical power.

The majority of the proposed remedial system uses fixed cost or static evaluated cost models [21] [22]. In contrast, a few models have been presented in the dynamic evaluated cost [23] [24] [25]. Since our proposed framework lies in this category, we will subsequently discuss some highly related frameworks. We first consider service dependencies models in remedial actions, initially proposed by Toth and Kregel [23]. They presented a network model that accounts for relationships between users and resources, illustrating that they perform their activities by utilizing the available resources. The response model goal is to keep the usability of a system as high as possible. Each response alternative (which node to isolate) is inserted temporarily into the network model. A calculation is performed to find which one has the lowest negative impact on the services. When the IDS detects an attack coming towards a machine, an algorithm tries to find which firewall/gateway can minimize the response actions' penalty cost. This approach suffers from many limitations. First, they did not consider the positive effect of responses. The evaluation of responses without considering their positive effects leads us to inaccurate evaluation. For example, if the negative impact of response A is greater than response B, it does not mean that response B has to be applied first. Maybe the positive effect of response A is better than B and, if we calculate the response effectiveness,

overall response A is better. Secondly, from a security goals perspective (CIA), there is no evaluation regarding data confidentiality and integrity. Eventually, in the proposed model, only the $blockIP$ response has been considered. Balepin et al. [24] presented a local resource dependency model to evaluate responses in case of attack. Like Toth and Kregel [23], it considers the current state of the system to calculate the response cost. Each resource has common response measures associated with it. They believe that designing a model to assess each resource's value is a difficult task, so they order the resources by their importance to produce a cost configuration. Then, fixed costs are assigned to high priority resources. Thus, costs are inflicted on the resource dependency model when associated resources get involved in an incident. A particular remedial response for a node is selected based on three criteria: (1) response benefit: sum of costs of resources that the response action restores to a working state, (2) response cost: sum of costs of resources which get negatively affected by the response action, and (3) attack cost: the sum of costs of resources that get negatively affected by the intruder. Thus, unlike Toth and Kregel [23], this model considers the positive effects of responses. This approach suffers from many limitations. First, it is not clear how remediation benefits are calculated in terms of confidentiality and integrity. Secondly, restoring the state of resources cannot be the only factor in evaluating the response's positive effect [25]. Finally, the proposed model is applicable for host-based intrusion response systems. To use for network-based intrusion response, it requires significant modifications in its cost model [25]. Jahnke et al. [26] proposed a graph-based approach for modeling the effects of attacks against resources and the effects of the response measures taken in reaction to those attacks. The proposed approach extends the idea put forward in Toth and Kregel [23] by using general, directed graphs with different kinds of dependencies between resources and deriving quantitative differences between system states from these graphs. If we assume that G1 and G2 are the graphs obtained before and after the reaction, respectively, then the calculation of the responses' positive effect is the difference between the availability plotted in the two graphs: G1 and G2. Like Toth and Kregel [23], Balepin et al. [24], these authors focus on the availability impact. Kheir [25] presented a dependency graph to evaluate the confidentiality and integrity impact and the availability impact. The confidentiality and integrity criteria are not considered in Toth and Kregel [23], Balepin et al. [24], and Jahnke

et al. [26]. In Kheir [25], the impact propagation process proposed by Jahnke et al. [26] is extended to include these impacts. Each service in the dependency graph is described with a 3D CIA vector, the values of which are subsequently updated, either by actively monitoring the estimation or by extrapolation using the dependency graph. In the proposed model, dependencies are classified as structural dependencies or as functional dependencies. The only work that considered data confidentiality and integrity to evaluate the negative effect is Kheir [25]. As mentioned in Kheir [25], it is challenging to identify the impact on data confidentiality and other resources' integrity when we apply a remediation response to a resource. A specific problem has been solved in Kheir [25]. The proposed framework assumes using secure protocols. When an attack happens to one of the secure protocols, the framework switches to insecure mode. The authors use a specific response type like allow insecure connections in an open, secure socket layer (SSL) attack. If we use this type of response, it is clear that it affects other resources' data confidentiality and integrity.

On the other hand, the landscape of cybersecurity has been reformed dramatically by the recently emerging Advanced Persistent Threat (APT) [27]. Unlike traditional cybersecurity threats, APT attackers can adopt any *advanced* actions in a *stealthy* manner with a goal of $long-term$ utility gain, instead of any $one-shot$ benefit. Hence, these unique properties render the existing security solutions [27] inapplicable for APT, since they are confined by one or more of the following limitations: i) each attacker has a discrete and limited set of actions for one specified type of attack (e.g., DoS attack and password-based attack), violating the feature of *advanced* actions in APT which could include the combination all possible types of attacks; ii) the security game runs in a discrete-time fashion and the defender and attacker take actions either *concurrently* or *alternately* in each time slot, which are far from the real practice for APT since the attacker/defender cannot be accurately coordinated to make a move as the attacker acts *continuously* (not discretely) and *stealthily*[1]; and iii) the security problem is modeled as a $one-shot$ static game, which cannot characterize the *persistent* interplay among players for their long-term utility gains, or a *repeated* game, whose system status (e.g., how much portion of the system has been compromised) remains static and

[1]There is no way to know the opponent's time to take an action and to react accordingly either concurrently or alternately

cannot be impacted by players' behaviors.

Game theory is a decision-making theory for studying the direct interaction among decision-makers [28], whose goal is to maximize the earnings of players and is suitable for analyzing the strategy selection issue when the behaviors of decision-makers interact directly. It mainly includes player, state, action, information, strategy, payoff, and equilibrium elements. Game theory has the characteristics of objective opposition, non-cooperative, and strategic interdependence, all of which are in line with the essential attributes of cyber attack-defense [29]. Therefore, applying game theory to the model and analyzing the cyber attack-defense process has become a hot research issue in recent years [30]. However, there are still some challenges. To our best knowledge, existing game models for cyber attack-defense are mainly based on the hypothesis of complete rational players [31] [32] [33]. Complete rationality includes many preconditions that are difficult to achieve, such as perfect rational consciousness, the perfectability of analyzing and inferring, identifying and judging, memorizing, and computing. If any of these conditions cannot be reached, it belongs to bounded rationality. The strict requirement of complete rationality is too harsh for the social attacker and defender. Real-world attackers and defenders have different cognizance abilities, which is determined by their interests, such as safety knowledge, skill level, experience, and so on [34]. In a word, the selection of strategy affected by various uncertain factors leads to the bounded rational game. At present, this issue is still assumed as a significant challenge.

Thus, APT calls for a framework that could characterize the continuous interplay of *advanced* defense-attack on system resources with $imperfect/incomplete$ opponent's actions in a long time-span. This study involves (1) a model to accurately capture the continuously evolving process of the system status and how it is influenced by an attacker's and a defender's actions; and (2) dynamic defense/attack strategies that judiciously and continuously take steps to minimize/maximize the long-term system damage without knowing the opponent's behavior.

Considering all of the above-discussed drawbacks in different models, in this dissertation thesis, we are proposing four different situational cyber-defense remediation models that will decrease the cyber risk of the system for better operational resilience while considering the negative impacts of remedial response along with positive impact and maintaining the

quality of the running services.

1.2 PROBLEM STATEMENT AND OUR CONTRIBUTION

Four problem statements that stem from the above-discussed literature review motivate us to solve by our developed algorithm to make a cyber-resilient EDS. Following those, the main contributions of this dissertation are as follows:

- Given an EDS context, how to model tactical risk that includes safety and security risk of a node considering that node's criticality measure and model how that risk propagates to business/mission and strategic risk [35].
- Suppose you have a heterogeneous EDS infrastructure, then how to model different assets' criticality index of this infrastructure to select optimal remediation schemes [36].
- For the evolutionary cyber-defense remediation controls' selection, how the game model provides an optimal decision in countermeasure selection in response to players' rationality and could predict the probable future paths of attacks.
- Again, in the dynamic remediation controls' selection, how to balance the positive and negative impacts of those selections under certain limits of the quality of service (QoS) and security parameters (Confidentiality, Integrity, and Availability (CIA)) [37] [38].

1.3 ORGANIZATION OF THE PROPOSAL DISSERTATION REPORT

The rest of this dissertation organizes as In Chapter 2, we discuss the NIST approved three layer framework of cyber risk management for cyber defense remediation in EDS. In Chapter 3, we describe the cyber defense remediation for tactical risk management in EDS based on criticality indexes of network assets. Chapter 4 depicts the evolutionary game model for cyber attack-defense remediation selections. In Chapter 5, we depict the model, cost of countermeasures in Software-Defined Networking-enabled EDS. Finally, in Chapter 6, we provide the conclusion and future research aspects of this dissertation report.

Chapter 2

CYBER DEFENSE REMEDIATION BASED ON NIST THREE LAYERS ARCHITECTURE OF CYBER RISK MANAGEMENT

This chapter guides us, how to model tactical risk that includes safety and security risk of a node considering that node's criticality measure and model how that risk propagates to business/mission risk and strategic risk in an EDS context. The contributions of this chapter are as follows:

- Model tactical risk considering safety and security risk of a node in the EDS infrastructure considering node criticality and model how they propagate to business/mission risk and strategic risk.
- Propose an optimal resource allocation scheme of a fixed resource budget according to nodes' criticality at operational level and then optimize among tactical risk, business/mission risk, and strategic risk.
- Empirical validation within an ICS test-bed to assess performance of the criticality model and resource allocation scheme.

2.1 ARCHITECTURE

The risk analysis and remediation architecture are composed of Strategic Response Decider (SRD), Attack Graph Generator (AGG), Fault Graph Generator (FGG), Response Operational Impact Assessment ($ROIA$), and Threat Risk Quantifier (TRQ) modules. These modules evaluate individual and combined mitigation actions in financial and operational perspectives to generate the corresponding response plans. The data provided to these modules comprises network inventory, service inventory, reachability, weighted connection matrices, security policies, mitigation actions, and vulnerability inventory.

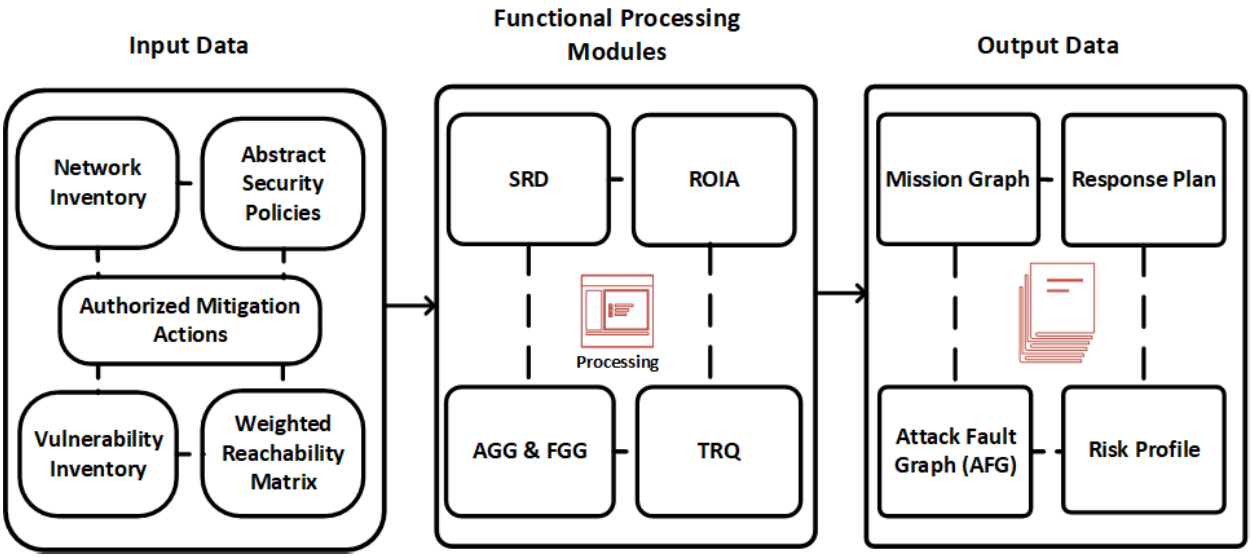


Figure 1. Optimal Cyber Risk Remediation conceptual workflow

We model the cyber attacks using Attack Graphs (AG) provided by MulvaL [39], which calculates the exposure of the monitored system to threats [40]. The AG generation depends on the monitored system's reachability matrix, the vulnerability inventory, and the mission graph (MG). An MG carries information about the probability of global operational impact on the mission originating from widespread effects causing local impacts [41]. The Fault Graph (FG) [5] is generated from inputs provided by the system designer and resource dependency model in the form of a probabilistic graphical model (PGM) [41]. An AG is integrated into a pre-existent FG to extend traditional risk analysis (which captures only safety risks) to include malicious threats [5] and formed attack fault graph (AFG).

2.1.1 THREAT RISK QUANTIFICATION (TRQ)

Organizations conduct risk management activities by using a standard EDS risk management framework, depicted in Figure 1 [2].

TRQ evaluation starts from Tier 3 ($Tacticalrisk$) and uses the AFG to compute the risk profile. TRQ is responsible for communicating the risk profile to the SRD module to derive the best response plan. The MG describes an organization's business model, including the consequences of potential impacts on business processes. TRQ computes elementary

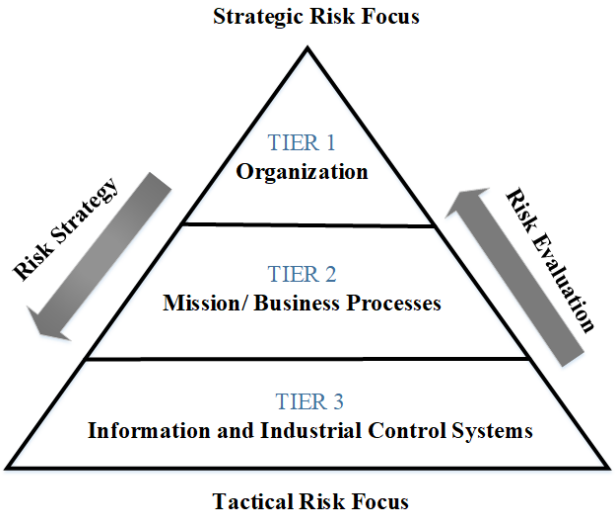


Figure 2. EDS Risk Management Framework

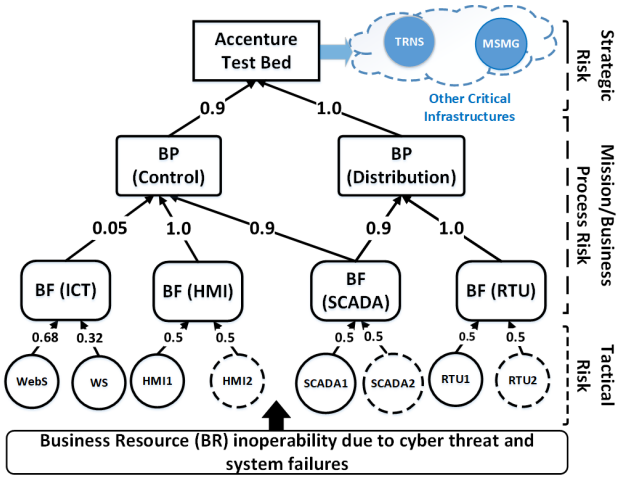


Figure 3. Extended BPMN of Test-bed

risks (ER) based on calculated attack and fault paths. ER is defined as *the quantum of risk inflicted by a single detrimental event to an asset through the exercise of a single attack/fault scenario on one single supporting asset contributing to that asset.* The ER is a three-tuple metric and is defined as:

$$ER_i = [Likelihood_i, Impact_i, f(Likelihood_i, Impact_i)] \tag{1}$$

The likelihood can be calculated from the probability of exploiting a software vulnerability of an asset-based on CVSS base score [40] and cumulative probability [42] for attack path. The chance of random faults occurring in an asset can extract from vendor data sheets [43]. The events caused by cyber-attack or unexpected defects are assumed to be not mutually exclusive but are independent. These events are OR gated at the integration point and can be defined as [5]:

$$P_{out}OR_{GOAL} = 1 - \prod[(1 - P_{in}(AG))(1 - P_{in}(FG))] \tag{2}$$

where $P_{in}(AG)$ indicates the input event from combined malicious attack events and $P_{in}(FG)$ indicates the input event from combined random fault events for final goal OR

gate. We assume that although $P_{in}(AG)$ and $P_{in}(FG)$ stems from two different sources, but the unification of their occurrence can derive from [3] in a standard period.

The impact of each ER depends on the successful execution of a dangerous scenario's terminal step. The inflicted consequences from a terminal node can be calculated from the criticality of that asset. The criticality ranking of an asset and the optimized remediation actions can be calculated from the models described in Chapter 4.

TRQ calculates the Tier 2 business process/mission risk with the help of estimated three tactical risk parameters (likelihood, criticality impact, and tactical risk) from Tier 3. To perform a mission impact assessment, we model business processes using the business process modeling notation ($BPMN$). We also model a business process as a (dependent) collection of tasks [7].

In this work, we extend BPMN and model mission dependencies, as illustrated in Figure 3. An expert models a business function "Industrial Automation" in an Industrial Control System (ICS) and may identify a Supervisory Control and Data Acquisition (SCADA) server as the mission-critical device, i.e., business resource. The business processes of the test-bed are to distribute and control electric power to the small city. In a mission dependency model, every dependency represented by conditional probability describes a probability of impact. These business processes require four business functions provided by eight mission-critical resources. The distribution of electric power requires the business process BP (Distribution) and business function BF (remote terminal units (RTUs)), which are remotely placed actors for switching power. There has two RTUs access via a cable communication link, and individual RTU provides business function BF_R. The two RTUs are redundant. Due to that, the conditional probability $P(BF_R|RTU1\,or RTU2)$ is equal to 0.5. Likewise BF_R, the conditional probabilities $P(BF_H|HMI1\,or HMI2$ or $P(BF_S|SCADA1\,or\,SCADA2$ is equal to 0.5.

Two redundant SCADA servers (business function BF_S) provide the central intelligence between controlling (BP_C) and distributing power (BP_D). Those SCADA servers manage the individual RTUs monitored by human-machine interfaces (BF_H).

Two SCADA servers of the business function BF_S represent the mission dependency model. Business experts identified an "ICT" business function BF_I consisting of one WS

and one WebS deemed non-critical. Therefore, $P(BF_I|WebS) = (0.684/1.009) = 0.68$ and $P(BF_I|WS) = (0.325/1.009) = 0.32$ (Calculated from asset criticality). BF_I has minimal involvement in controlling BP_C and consequently assessed with a probability fragment of $P(BP_C|BF_I = 0.05$. The joint probability distribution of mission in Figure 3 can be calculated as follows:

$$P(Accenture, BP_C, BP_D, BF_I, BF_H, BF_S, BF_{RTU}, WebS, WS, HMI, SCADA, RTU) \quad (3)$$

where we obtain joint conditional probability through the noisy-or assumption that every conditioned event can be independent [41].

We adopt the inoperability input-output model(IIM) [9] to measure Strategic Risk. IIM is capable of quantifying interdependencies among multiple critical economic infrastructure sectors [44] and is formulated as [9] :

$$\mathbf{q} = \mathbf{A}^*\mathbf{q} + \mathbf{c}^* \quad (4)$$

$\mathbf{q}$ is the inoperability vector expressed in terms of normalized economic loss (EL). $\mathbf{c}^*$ is a perturbation vector expressed in terms of normalized degraded final demand. $\mathbf{A}^*$ indicates the degree of coupling of the industry sectors. Here, *EL represents the monetary loss associated with an inoperability value. This loss includes an organization's reputational value, regulatory loss, environmental effects to nature, and safety issues towards living beings.*

2.1.2 RESPONSE OPERATIONAL IMPACT ASSESSOR (ROIA)

$ROIA$ aims to measure mission impact assessment based on inter-dependencies between safety and security. $ROIA$ seeks to assess the operational (tactical) risk condition of resources for different resource budget allocation. $ROIA$ determines the effects on operation due to remediation actions. We only consider software patching (SP) and a node's redundancy as remediation actions in this work. The optimized resource allocation for SP and node redundancy is calculated from Chapter 3.

2.1.3 STRATEGIC RESPONSE DECIDER (SRD)

The goal of SRD is the automated administration of policies, including new rules, removal of extreme conditions, and activation of strategic responses. SRD interacts with the AG, FG, TRQ, and ROIA components to evaluate and select the best answer by applying trade-off among operational resilience (tactical risk), mission risk, and strategic risk.

SRD relies on the response financial impact assessor ($RFIA$) component to quantify the economic benefit. Response plans represent the proposed mitigation (SP and asset redundancy) of the assessed security and safety risks. $RFIA$ calculates the return-on-response-investments ($RORI$) index associated with the mitigation actions composing a risk response plan. RORI index can evaluate optimal plans based on the trade-off between resilience to attacks and faults and EL. Santos et al. [45] show that the EL can represent the function of the inoperability of sector i:

$$\Delta x_i = q_i x_i \tag{5}$$

where q_i is the resulting inoperability to sector i derived from Eq. 4, which combined with the ideal production (x_i) will yield an estimate of the EL (Δx_i). Summing all the individual sector's Els will yield the cumulative economic loss. For a particular response plan j, the cumulative economic loss to the economy, denoted by $\Gamma_{w[j]}$, can be calculated as follows:

$$\Gamma_{w[j]} = \sum_{i=1}^{n} \Delta x_{w[j],i} = \sum_{i=1}^{n} q_{w[j],i} x_i \tag{6}$$

where $\Delta x_{w[j],i}$ is the EL and $q_{w[j],i}$ is the resulting inoperability for sector i for a risk response plan j.

RORI index is calculated for each mitigation risk response plan and defined as:

$$\Phi_j = \frac{\Gamma_{w[0]} - \Gamma_{w[j]}}{B_{Dj}} \tag{7}$$

The model divides the difference between the magnitude of cumulative ELs with the

response plan option j relative to a baseline scenario by the costs associated with implementing that j^{th} option. Φ_j represents a benefit-cost ratio. When the mitigation actions ($SP\&noderedundancy$) apply based on criticality value within a defined resource budget, effective Risk Minimization (RM) is achieved

$RFIA$ components evaluate and select mitigation actions from a pool of candidates by ranking them in terms of $RORI$. The optimal selection of the response plan derives from the Surrogate Worth Trade-off (SWT) method described by [45] provides a technique to address these multiple-objective problems. For two-objective functions, f_1 and f_2, denoting the policy cost and cumulative EL respectively. Policymakers would aim to minimize these objective functions, to minimize EL while investing the minimum resources. The first objective is to minimize the cumulative EL (f_2). Eq. 6 then yields:

$$minf_2 = \Gamma_{w[j]} = \sum_{i=1}^{n} \Delta x_{w[j],i} \tag{8}$$

The second objective is to minimize the investment cost (f_1):

$$minf_1 = B_{Dj} \tag{9}$$

An effective policy option can drive down the value of the expected EL to the allowable risk acceptance level of an organization, subject to the constraint of acceptable implementation cost. To solve the two-objective optimization problem via the SWT method, we convert Eqs. 8 and 9 into the following ε- constraint formulation:

$$minf_1 \, subject \, to \, f_2 \leq \varepsilon_2 \tag{10}$$

We can reformulate Eq. 10 in terms of a Lagrangian function using the ε-constraint approach [46]. The problem then becomes:

$$L(.) = f_1 + \lambda_{12}(f_2 - \varepsilon_2) \tag{11}$$

From this equation a necessary condition for optimality states that:

$$\lambda_{12} = -\frac{df_1}{df_2} > 0 \tag{12}$$

where λ_{12} represents the trade-off, or slope, between the two objective functions. Once these relationships are defined, the decision-maker may interact with the model and evaluate different policy options, subject to preferences regarding constraints such as cost and acceptable risk.

2.2 IMPLEMENTATION, RESULTS, AND ANALYSIS

2.2.1 EDS NETWORK IMPLEMENTATION:

We implemented an EDS network depicted in Figure. 4 in Accenture ICS research test-bed [47]. The user workstations contained the vulnerability of $CVE-2009-1918$ in Internet Explorer (IE). If a user accesses malicious content using the vulnerable IE browser, the machine might be compromised. The web server at the demilitarized zone (DMZ) contained the vulnerability $CVE-2006-3747$ in the Apache HTTP service, resulting in a remote attacker executing arbitrary code on the machine. The redundant $SCADA1$ and $SCADA2$ servers contained the vulnerability $CVE-2018-5313$, allowing privilege escalation up to the administrator level. Every $SCADA$ server had a human-machine interface (HMI) software in it. The $SCADA1/SCADA2$ server controls $RTU1/RTU2$ of the substation distributing 1 Mega Watt (MW) electricity to other connected critical infrastructures Transportation ($TRNS$) and Manufacturing Industries (MFG). We assume that, if an attacker acquires control over the $SCADA$, the respective RTU and HMI may be acquired as well.

2.2.2 RESULT AND ANALYSIS:

The Nessus's scanned data, Qualys's host, scanned logs, and Wireshark's passive traces were collected from the test-bed synchronously for half an hour. An AG and an AFG are depicted in Figure 5 and in Figure 6. From the AFG, we can determine that there were two types of threats in the Accenture test-bed: Lateral movement of a cyber attacker to

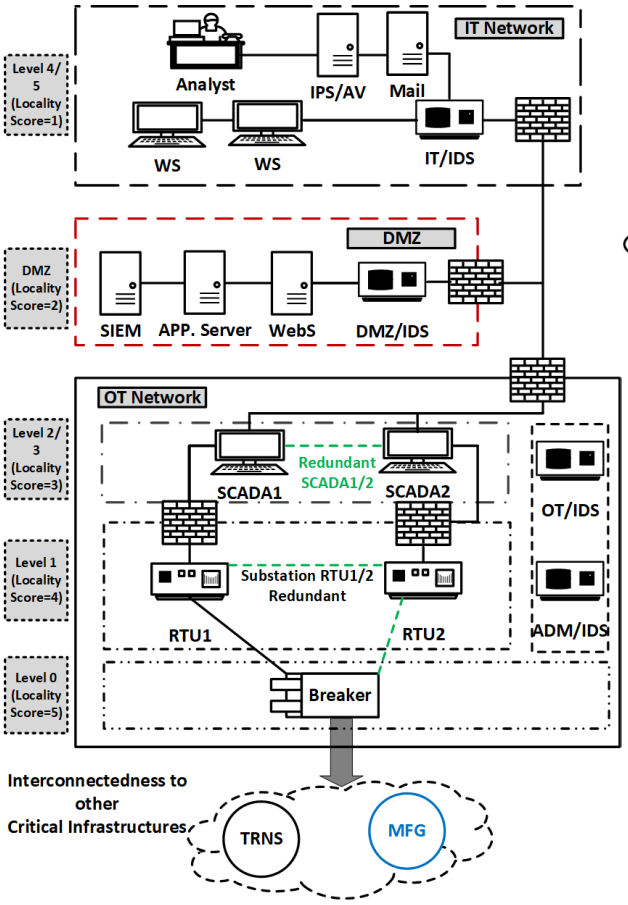


Figure 4. Logical view of EDS operational Test-bed

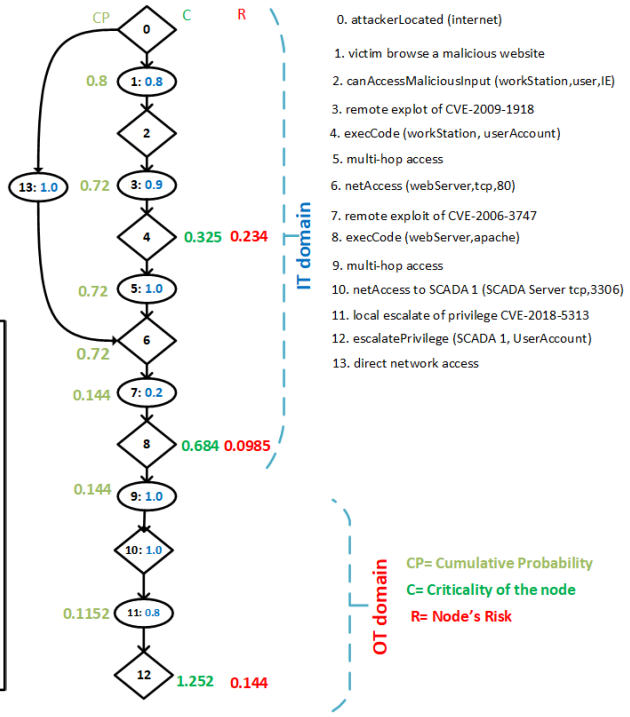


Figure 5. AG of EDS Operational network

compromise critical assets and random operational failure of a critical asset which might impact safety.

TRQ is calculated based on a cyber threat's exploiting the probability of an asset from $CVSS$ the base score [42]. We collect the probability of a random fault of an OT asset (SCADA, HMI, and RTU) from the asset datasheet described in [43]. The unification of a cyber threat exploiting probability and the system's random fault probabilities are derived from [5]. Figure 6 also depicts three different scenarios of node exploitation by a cyber attack and a random failure of an asset. The black-colored box represents the probability when no remediation policy (Base Policy) is applied; the green colored box indicates policy 1 (mitigation action: SP), and the blue color box represents policy 2 (mitigation action: SP+ asset redundancy). The weighted reachability matrix calculated from $TCP/DNP3$

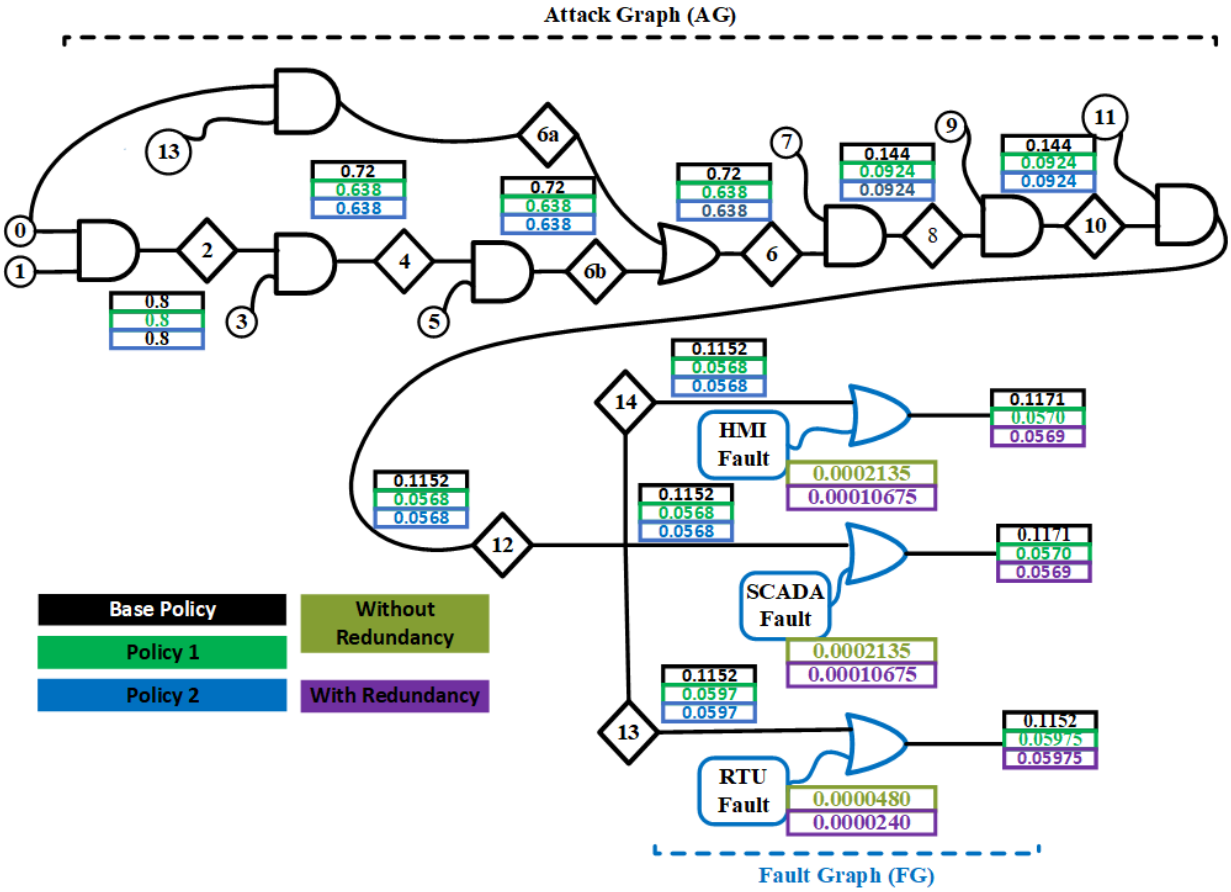


Figure 6. AFG of EDS operational network

dump data is shown in Table 1. Total exchanged packets during the half an hour time was 8006. The weights were calculated by dividing exchanged packets between pairs with the total numbers of packets exchanged during the period among nodes. Then TRQ calculated an asset's criticality to measure the tactical risk of security and safety after following Eqs. 13-17 from Chapter 3 (see Table 2).

For total criticality calculation in Table 1, we set $\alpha = 0.15, \beta = 0.25$, and $\gamma = 0.6$.

The TRQ developed the complete mission dependency graph from tactical risk, the likelihood of exploiting software vulnerability and a random system's fault, and criticality impact of an asset. The test-bed mission dependency graph is depicted in Figure 3.

Table 1. Weighted Reachability Matrix of tactical network

Nodes	WS	WebS	SCADA/HMI	RTU
WS	0	0.125	0	0
WebS	0.125	0	0.125	0
SCADA/HMI	0	0.125	0	0.25
RTU	0	0	0.25	0

Table 2. Operational nodes' total Criticality Calculation

Nodes	l	CEN($\delta = 0.5$)	d	$Criticality(C)$
WS	1	0.5	0	0.325
WebS	2	1.225	0	0.684
SCADA/HMI	3	0.949	1.0	1.252
RTU	4	0.548	1.0	1.18

Table 3. Tactical network's Exponential Cost Resource Allocation

Nodes	C	$maxA$	$\frac{C}{maxA}$	A	$V(\%)$
WS	0.325	4.64	0.070	0.561	63.8
WebS	0.684	4.64	0.147	2.060	9.24
SCADA/HMI	1.252	4.64	0.270	6.3078	5.68
RTU	1.18	4.64	0.241	6.071	5.97

Then, TRQ determined business/mission risk likelihood for identified threats (in our case inoperability of SCADA, HMI, or RTU) of tactical risk, which may include external cyber threats (an attacker's lateral movement to compromise SCADA, HMI, or RTU) or internal system random failures after following Eq. 3. The threat events' likelihoods of every asset and calculated business/mission inoperability is shown in Table 4. The business/mission inoperability puts as an input inoperability to calculate strategic risk from Eq. 4. The calculation of strategic risk and EL is shown in Table 5 after following Eq. 4. The interdependency matrix $\mathbf{A}^*$ [9], which consists of the interconnectedness of EDS distribution

Table 4. Threat Likelihood at every asset & Business Inoperability (I)

Mitigation Actions	Threat Likelihood at assets					Business Inoperability
	WS	WebS	SCADA	HMI	RTU	
No Action	0.72	0.144	0.1152	0.1152	0.1152	1.26×10^{-6}
SP	0.638	0.0924	0.0570	0.0570	0.05975	1.01×10^{-7}
SP&Redundancy	0.638	0.0924	0.0569	0.0569	0.05975	9.28×10^{-8}

Table 5. IIM Output& EL

Policy	Inoperability (I)(Power Base:10^{-5}) & EL (Units)					
	ED(I)	ED(EL)	TRNS(I)	TRNS(EL)	MFG(I)	MFG(EL)
Policy0	0.1522	152.2	0.0005	0.5	0.0027	2.7
Policy1	0.01222	12.22	0.00004	0.04	0.00022	0.22
Policy2	0.01124	11.24	0.00004	0.04	0.00020	0.20

(ED), transportation ($TRNS$), and manufacturing industries (MFG). For simplicity, we assume the yearly economic contribution from each sector towards the city is the same as 10^8 units. The monetary valuation of each unit can be determined from expert judgments from economists. $Policy0$ indicated when there had been no mitigation action, Policy1 mapped with remediation action SP, and $Policy2$ indicated remediation action SP along with system redundancy for system faults.

$ROIA$ evaluated remediation actions at the operational level to minimize the likelihood of tactical risk within a predefined budget resource. In this work, we only considered two types of remediation actions: 1) SP: SP for known vulnerabilities of critical assets, and 2). A redundant critical asset to build resistance against system failure. The system administrator can allocate the SP resource (15 units) after following the optimized resource allocation scheme described in Eq. 26 from Chapter 3. The detailed calculation of exponential cost resource allocation is shown in Table 3.

Table 3 indicates that the optimal allocation ensures when the resource distributes according to the criticality of the node. Likewise, the system's optimal redundancy ensures

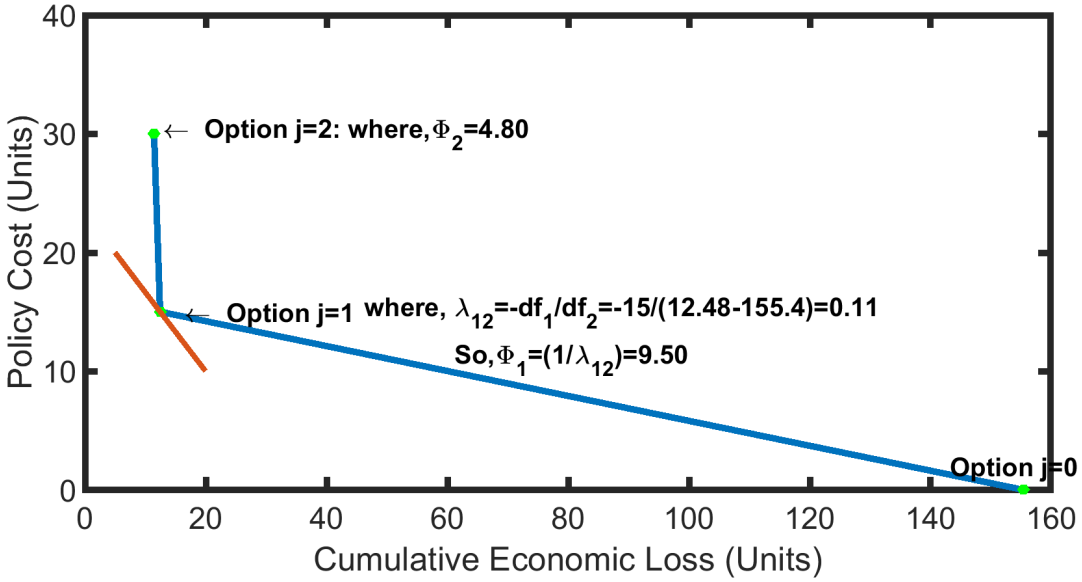


Figure 7. Strategic Risk trade-off analysis

that the redundant system distributes to the node's criticality. We kept the same amount of redundant resources (5 units) for SCADA, HMI, and RTU for simplicity. The threat likelihood snapshot of every asset and business impact after resource allocation is shown in row 2 of Table 4. The operational effect of redundancy also calculated and shown in row 3 of Table 4.

SRD took identified threats (Attacker lateral movement to compromise $SCADA, HMI$, and RTU and random failure of critical assets' operation), authorized mitigation actions (SP and system redundancy), and strategic policies ($Policy0, Policy1, and Policy2$) as inputs and calculated $RORI$ indexes after following Eqs. 6 & 7, which could be used to determine policies by optimizing strategic risk and operational budget. The calculation of optimization scheme after following Eqs. (8-12), is shown in Figure 7.

The resulting graph in Figure. 7 will give decision-makers a sense of the potential returns associated with the level of investment. The trade-offs at each point between policy costs (15 units for patching and 15 units for three critical assets redundancy) and ELs represent by the slope λ_{12}. For this scenario, we find that $\lambda_{12} = 0.12$ at the location where policy option $j = 1$. This value of λ_{12} shows us the ratio of investment at $j = 1$ with a cost of 15 units

(SP cost only) concerning its EL reduction of (12.22-152.2) units. Note that $\lambda_{12} = 0.12$ is the reciprocal of $\Phi_1 = 9.50$, calculated from Eq. 7.

2.3 SUMMARY OF THE CHAPTER:

In this chapter, we introduced a risk remediation response system that generates response plans containing mitigation actions and corresponding financial and operational assessment. The response plan includes system level mitigation actions that can mitigate system security and safety threats according to the system's criticality within budget constraints. The empirical validation in an actual ICS test-bed showed that system-level risk was maximally reduced when the resource was allocated according to node criticality. The model then mapped optimized mitigation actions to the strategic response plans and optimally selected a response plan to mitigate system threats by trading-off between system resilience and EL.

Chapter 3

CYBER DEFENSE REMEDIATION BASED ON CRITICAL NODE ANALYSIS IN EDS

This chapter describes how to model a node's criticality index for the selection of optimal remediation schemes in a heterogeneous EDS infrastructure. The contributions of this chapter are as follows:

- Model criticality of a node in the EDS infrastructure considering network heterogeneity.
- Propose an optimal resource allocation (remediation) scheme of a fixed resource budget according to nodes' criticality that minimizes the network risk.
- Empirical validation within an ICS testbed to assess performance of the criticality model and resource allocation scheme.

3.1 SYSTEM MODEL

Figure 8 depicts the processes and interactions among different modules in the proposed risk analysis and resource allocation model. Here, *resource refers to the deployable items to mitigate exploiting a vulnerability in a system that can be converted to monetary value like working hours for installing new patches, purchasing new patches, and related system downtime cost.* Leveraging network scanning data and host logs, such as TCP/DNP3 dump, the system creates Attack Graph (AG) using [39] to determine nodes' criticality. The risk analysis module calculates the risk of exploiting a node's vulnerability in the AG as a product of the probability of using the vulnerability and potential damages caused by controlling the node. The node criticality for the target EDS gives the quantification of the damage. After calculating a node's risk, the security administrator can filter out the most critical paths and reduce the risk for those paths by selecting appropriate remediations. In the next subsections, we will discuss every module of our system.

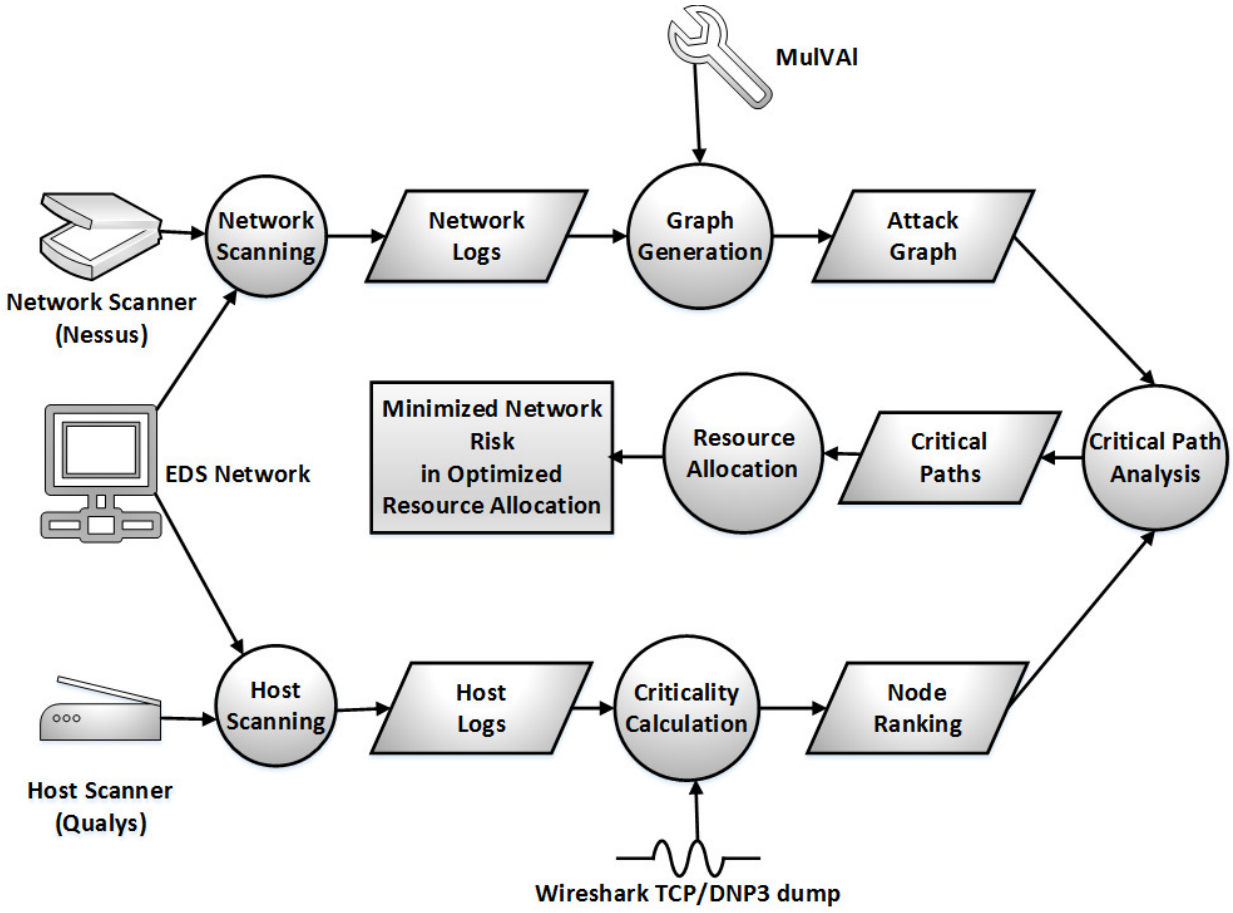


Figure 8. System Model

3.1.1 ATTACK GRAPH GENERATION

The open-source tool MulVal [39] is used in our system to create AG from network scanned data. The semantics of MulVal AG takes from [40] and Figure 10 explains AG with an example. The labels of the graph nodes display on the right-hand side of the AG diagram. The intrinsic probability for exploiting a vulnerability without pre-conditions inside the oval shape takes from $CVSS$ base score [40]. The cumulative probability (CP) derives from this intrinsic probability of a vertex after the following methods in [42]. There is an expected loss of C_i associated with each vertex/node representing the loss value in monetary units when the vertex has been acquired or exploited. This loss value also indicates the $Criticality$ of this node.

3.1.2 HOST SCANNING AND CRITICALITY CALCULATION

To model the criticality of a node in EDS, we primarily focus on three factors which describes the heterogeneous nature of EDS (IT & OT).

$$C(i) = \alpha l(i) + \beta CEN(i) + \gamma d(i) \tag{13}$$

where $C(i)$ is the criticality of the node i, driven by three properties $l(i)$, $CEN(i)$, and $d(i)$ respectively indicate locality, centrality and physical damage properties of critical node i. Each characteristic has a tuning parameter α, β, and γ for administrator's adjustments usages to control the relative importance of three characteristics. For example, in OT networks, γ should have more weight to consider physical damage, whereas, in the control system, β should increase to consider centralized control nodes more.

Locality (l): locality is defined as the relative position of a node according to network layers defined in IEC 62443 standard [48] for EDS. Servers closer to physical assets are considered to be more cyber critical and receive a higher value, for example, in an EDS shown in Figure 9, Supervisory Control And Data Acquisition 1 ($SCADA1$) and $SCADA2$ servers located at levels 2 or 3 are more critical than workstations situated at levels 4 or 5. A higher score assigned to an asset indicates that it is closer to the physical processes—the localization of a node maps from running services and processes in the node. The running services and processes collect from hosts' scan logs.

Centrality (CEN): Centrality is a measure of criticality within the same layer of the IEC 62443 model. Since nodes at the same layer may have different attack propagation opportunities, individual node criticality can vary. Quantifying a single layer's relative centrality is done with a weighted network depicting network connectivity (unique neighbor connections) and traffic load per node. The load measure calculates by enumerating the number of packets (TCP, DNP3, etc.) exchanged between a pair of nodes normalized by the total number of packets traversing the layer during a predefined period. Unique connections count (Degree) of a node i is the number of communicated adjacent nodes in a network [49]:

$$k_i = c_d(i) = \sum_{j=1}^{N} x_{ij} \tag{14}$$

where j represents all other nodes, N is the total number of nodes, and x is the adjacency vector, in which $x_{ij} = 1$, if node i is connected to node j, and $x_{ij} = 0$ otherwise. Degree has generally been extended to the sum of weights when analyzing weighted networks [49] and labeled strength on node. Unique connections weight is formulated as follows:

$$s_i = c_d^w(i) = \sum_{j=1}^{N} w_{ij} \tag{15}$$

where, w_{ij} is defined as the weight of the link from node i to j. The product of count and weight yields the degree indicating the level of involvement of a node within its network. In addition, the tuning parameter, δ, determines the relative importance of the number of links compared to tie weights. More specifically, we propose a degree centrality measure, which is the product of the number of nodes that a focal node is connected to and the average weight to these nodes adjusted by the tuning parameter:

$$CEN(i) = k_i(\frac{s_i}{k_i})^{\delta} = (\sum_{j=1}^{N} x_{ij})^{1-\delta}(\sum_{j=1}^{N} w_{ij})^{\delta} \tag{16}$$

Damage Factor (d): to address global topological properties in the EDS context, we consider potential damage at the physical process level (L2 and L1), which is a function of managing the managed OT physical element. Utilization is a measure of applied electrical current (controlled power) over time within the acceptable range. The higher the current within the range, the more used the device is. Normally this information can be found from the exchanged $DNP3$ messages between $SCADA$ server and substations' Remote Terminal Units (RTUs). RTUs periodically transmit voltage and current level to the $SCADA$ server to control a substation's operation. From the current level, $SCADA$ servers calculate the operational load of a substation. As such, an attack on more utilized $SCADA$ controlled

devices can create more physical damage. Damage is defined as [50]:

$$d(i) = (\frac{P_l(i)}{P_T})^{L^*-1} \tag{17}$$

where L^* indicates the value of the loading level where power flow diverges (P-V curve). SCADA server decides the value of L^* from monitored voltage and current level. $P_l(i)$ is loss of load for the compromised system i and P_T indicates system's total load.

3.1.3 DISCOVER CRITICAL PATH

The system administrator uses a cumulative probability (CP) of every node in AG and $Criticality$ of that node to calculate the critical path. Here, the *critical path is the path that creates maximum damages to the system if an attacker has chosen this path to attain his/her goal*. The most probable attack path may not necessarily always be the same as a critical path. We assume that the critical path is preferable for an extremely skilled and knowledgeable attacker rather than the most probable route.

3.1.4 RISK ANALYSIS

The AG of an EDS network provides the logical representation of the attacker's lateral movements. To analyze the risk of those movements, we need to estimate the complexity of activities per stage in creating an AG. The complexity associated with an attack is a function of the $Criticality$ of that stage as defined above denoted as C_i. The probability of exploiting a vulnerability implies V_i (provided by external repositories indicating the complexity of using such vulnerability), and the likelihood of threat manifestation in that stage is denoted as T_i. Since threat intelligence varies in time according to global threat, we used an equal value for all elements, set to one for our model. As such, the risk of a stage in AG is defined as:

$$R(i) = T_i V_i C_i \tag{18}$$

3.1.5 RESOURCE ALLOCATION/REMEDIATION PLAN

Suppose we have a resource budget, B_D, and the cost to eliminate all vulnerabilities and exploits from node i is $maxA_i$, where A_i is the actual cost invested. The goal is to reduce the number of pre-conditions, vulnerabilities and exploits, denoted as V_i, to zero. The number of remaining vulnerabilities is a function of budget allocation A_i that represents actions performed on a node to remove and remediate such vulnerabilities for every node in AG. The target function is to allocate the correct A_i to each node such that the overall risk may minimized. Namely,

$$min\,\{R\} = \sum_{i=1}^{N} V_i(A_i)C_i \tag{19}$$

subject to

$$\sum_{i=1}^{N} A_i \leq B_D; \sum_{i=1}^{N} maxA_i > B_D; A_i \geq 0 \tag{20}$$

Linear Cost Model: What will be the risk reduction amount if we allocate more resources to critical nodes and fewer resources to less critical nodes? The disproportionate budget B_D allocation to critical nodes and a smaller amount to less critical nodes to reduce overall system risk is known as the linear cost model of risk reduction.

In the linear cost model, the more funds allocated to A_i to protect node i, the less vulnerable is the node up to a maximum investment, $maxA_i$, as follows [49]:

$$V_i(A_i) = 1 - \sigma_i A_i; 0 \leq A_i \leq maxA_i \tag{21}$$

Here, σ_i = slope of straight line such that $0 = 1 - \sigma_i maxA_i$. The slope is determined by the cost of 100% hardening, which is $maxA_i$. Vulnerability is driven to zero when $A_i = maxA_i$,

so $\sigma_i = \frac{1}{maxA_i}$. This leads to the simple linear cost model of risk reduction:

$$min\{R(A)\} = min\sum_{i=1}^{N} C_i max\{(1 - \frac{A_i}{maxA_i}), 0\} \tag{22}$$

subject to

$$\sum_{i=1}^{N} A_i \leq B_D; A_i \geq 0 \tag{23}$$

To calculate the actual optimized budget allocation to each node, we need to know the $maxA_i$ for each node. According to [51], the maximum spending for hardening an asset from cyberattack should not be more than 37% of its criticality value irrespective of Exponential Power Class type attack or Proportional Hazard Class type attack. In our system model, we first determined the $maxA$ based on the most critical node [51]. The next nodes' $maxA$ is determined by sorting the list of nodes according to their consequence values, where i enumerates nodes in ascending order by the product, C_i and $maxA_i$:

$$C_{i1}maxA_{i1} \geq C_{i2}maxA_{i2} \geq .... \geq C_{iN}maxA_{iN} \tag{24}$$

Next, allocate $maxA_{i1}$to the highest, $maxA_{i2}$ to the next highest, and so on, until the remaining budget is less than $maxA_{ik}$. The remaining budget Φ allocates to the k^{th} ranked node, and zero allocates to all remaining nodes. In this way, the nodes that use resources most efficiently are given the highest priority and highest amount possible.

The ranked-order allocation strategy is optimal because it efficiently reduces the risk contribution of the highest risk nodes until the budget depletes. Thus, the ranked-order allocation maintains the rank-order property established by consequences:

$$C_{i1}\frac{maxA_{i1}}{maxA_{i1}} \geq C_{i2}\frac{maxA_{i2}}{maxA_{i2}} \geq .. \geq C_{ik}\frac{\Phi}{maxA_{ik}} \geq 0$$

$$C_{i1} \geq C_{i2} \geq .. \geq C_{ik} \frac{\Phi}{maxA_{ik}} \geq 0; \frac{\Phi}{maxA_k} < 1$$

The exponential cost model:

The linear cost model is unrealistic because it assumes that the vulnerability will be zero with the increased budget allocation, but in reality, the vulnerability attached to a node cannot be zero since the vulnerability landscape evolves and continuously generates new threats. In other words, vulnerability reduction may suffer from diminishing returns. For this reason, researchers prefer the exponential cost model [49]. The exponential cost model is precisely the same as the linear cost model except for the relationship between budget allocation and vulnerability reduction. Moreover, the allocation strategy is the same; the higher-ranked ($\frac{C_i}{maxA_i}$) nodes receive more resources than lower-ranked nodes.

The exponential cost model differs from the linear model in two important ways: (1) the actual resource allocations A_i are different, and (2) network risk is typically higher because an infinite investment is required to eliminate the vulnerability. A simple exponential function for vulnerability reduction is [49]:

$$V_i(A_i) = e^{-\sigma_i A_i}; 0 \leq V_i(A_i) \leq 1 \tag{25}$$

This function asymptotically declines to zero when an infinite budget allocation assigns to this node. Unlike the linear strategy, the exponential cost allocation never completely removes the vulnerability. Allocation of budget B_D to nodes is optimized when objective function R is minimized, with budgetary constraint. The optimized function is [49]:

$$R(A) = \sum_{i=1}^{N} e^{-\sigma_i A_i} C_i - \lambda[\sum_{i=1}^{N} A_i - B_D] \tag{26}$$

where,

$A_i = \frac{ln(\sigma_i C_i) - ln(\lambda)}{\sigma_i}$ and $ln(\lambda) = \frac{\sum_{i=1}^{N} \frac{ln(\sigma_i C_i)}{\sigma_i} - B_D}{\sum_{i=1}^{N} \frac{1}{\sigma_i}}$

In this work, we kept maximum budget allocations for every node the same as the maximum allocation for the most critical node which was ($maxA_{i1}$).

3.2 IMPLEMENTATION, RESULT, AND ANALYSIS:

EDS Network Implementation: We implement an EDS network depicted in Figure 9 in Accenture ICS research test-bed [48]. The entire test-bed connects to a network switch and a router, and the zoning implements using VLAN and firewall rules. There are five subnets created by an external and internal firewall. The IT Workstations (WSs) are located at the IT subnet. A Web Server ($WebS$) locates at the DMZ subnet and is directly accessible from the Internet through an external firewall. $SCADA$ servers (L3/L2), RTUs (L1) are in different subnets under a larger OT subnet that holds critical communication. The SCADA1 servers and SCADA2 servers are only accessible from the $WebS$ of the DMZ zone. The WebS is accessible from user WS and other hosts from levels 4 or 5. The user subnet contains user's WS. The firewalls allow all outbound traffic from the user's subnet. The test-bed also includes an Intrusion Detection System (IDS) running both IT and OT specific rules and a commercial OT Asset Discovery and Management (ADM). They are both connected to the span port of the switch to inspect all ICS traffic. The DNP3/TCP dump can collect from this switch. For simulation, we injected vulnerabilities on the test-bed machines. The user workstations contained the vulnerability of $CVE-2009-1918$ in Internet Explorer (IE). If a user accesses malicious content using the vulnerable IE browser, the machine may be compromised. The web server (DMZ) contains the vulnerability of $CVE-2006-3747$ in the Apache $HTTP$ service, resulting in a remote attacker executing arbitrary code on the machine. The $SCADA1$ and $SCADA2$ server contained the vulnerability of $CVE-2018-5313$, which could allow privilege escalation up to the administrator level. The $SCADA1$ server controls 10 RTUs of substation 1, whereas the $SCADA2$ server controls 7 RTUs of substation 2. We assume that if an attacker acquires control over the $SCADAs$, the RTUs can be acquired.

Result and Analysis:

The Nessus's scanned data, Qualys's host, scanned logs, and Wireshark's passive traces

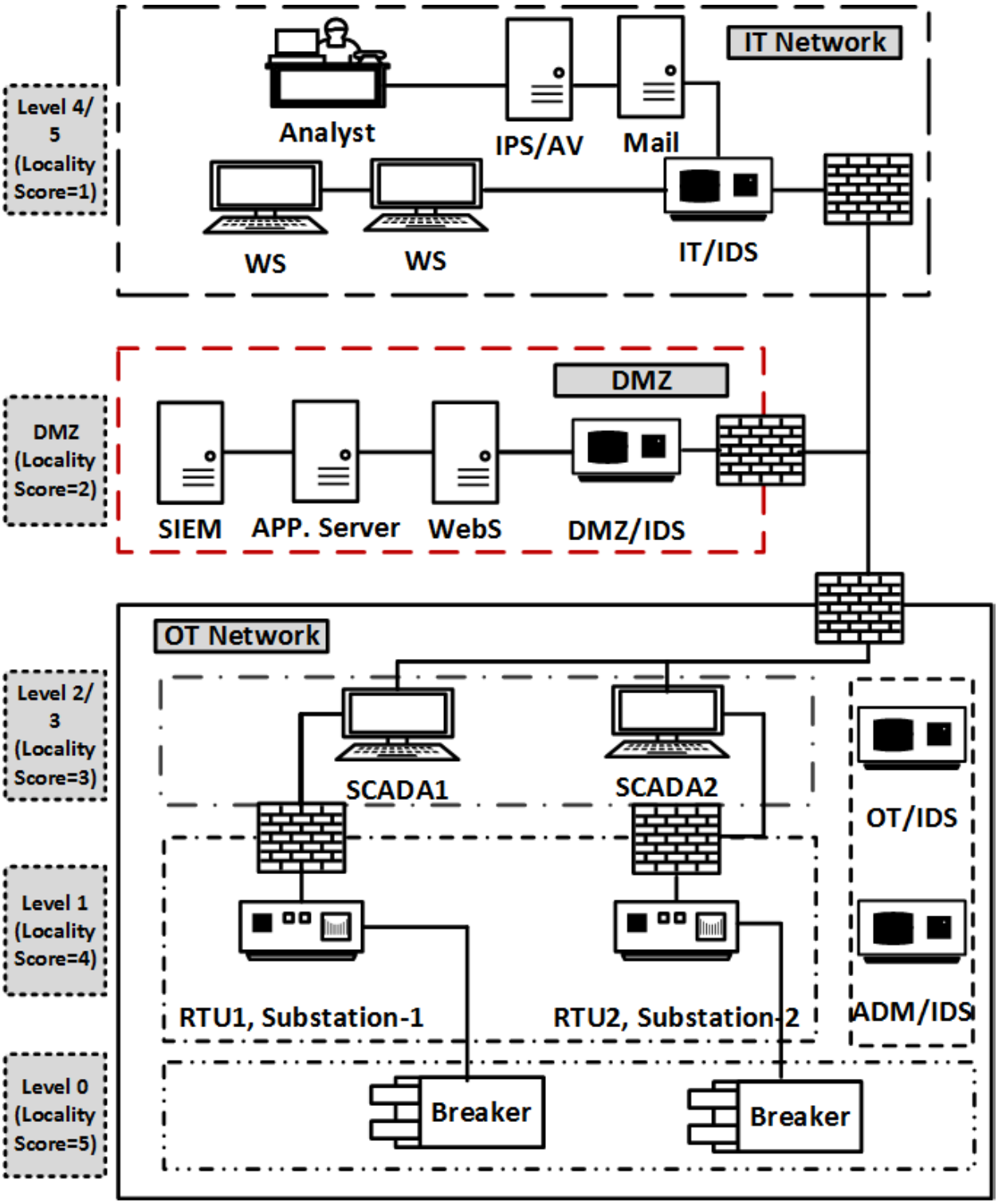


Figure 9. Logical view of EDS Test-bed for Criticality based tactical risk remediation.

collect from the test-bed synchronously for half an hour. $MulVAL$ generates an AG as depicted in Figure 10. This AG contained logical attack paths for the attacker, the cumulative probability of exploiting vulnerabilities starting from a vantage point (Internet) to a target ($SCADA1/SCADA2$), the relevant consequences for exploiting the vulnerability, and the risk of each exploitation.

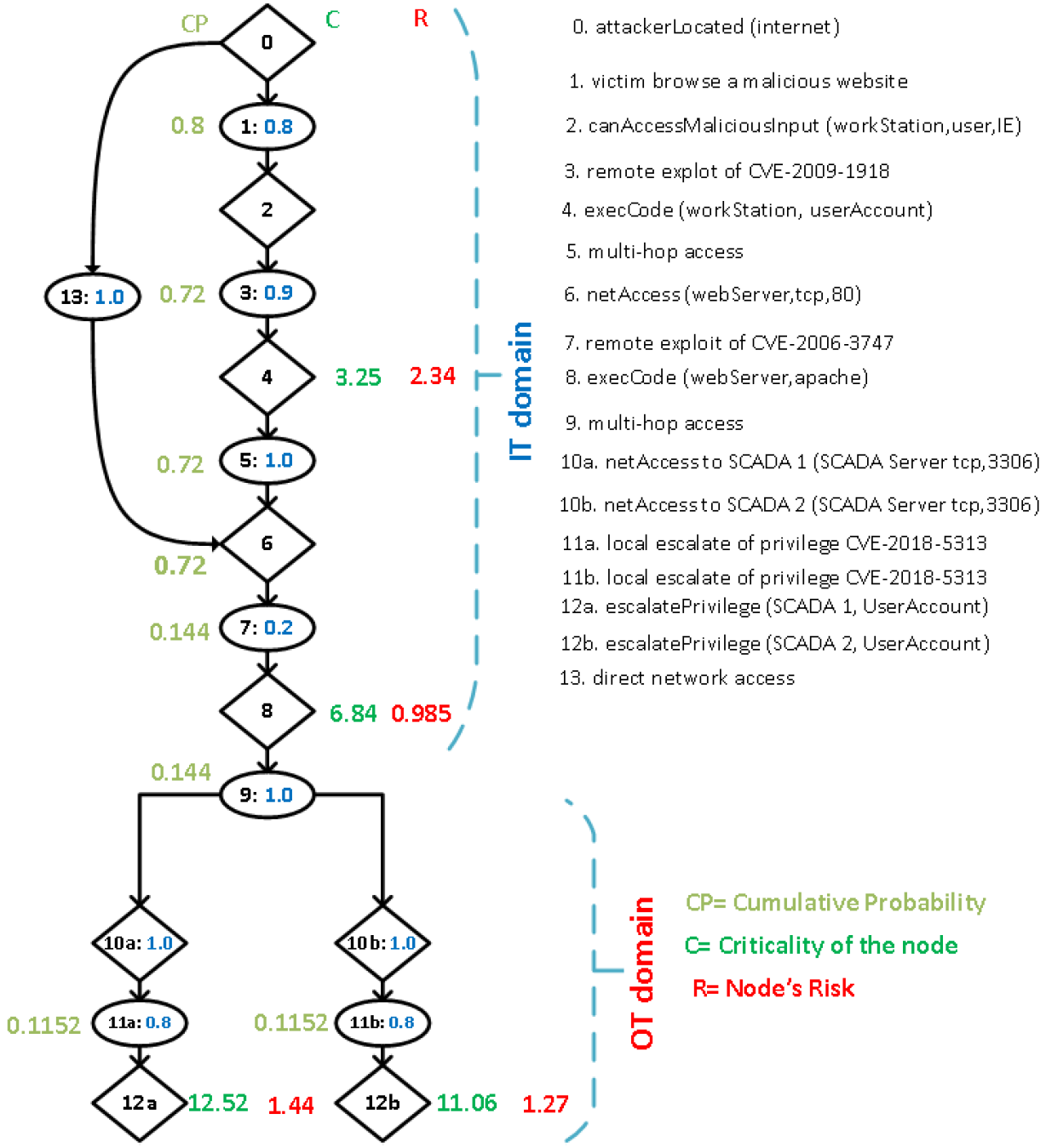


Figure 10. The AG of test-bed based EDS for Criticality based remediation.

The weighted graph derives from TCP/DNP3 dump data is shown in Figure 11. Total exchanged packets during the half an hour time is 8006. We get weights by dividing exchanged packets between pairs with the total numbers of packets exchanged

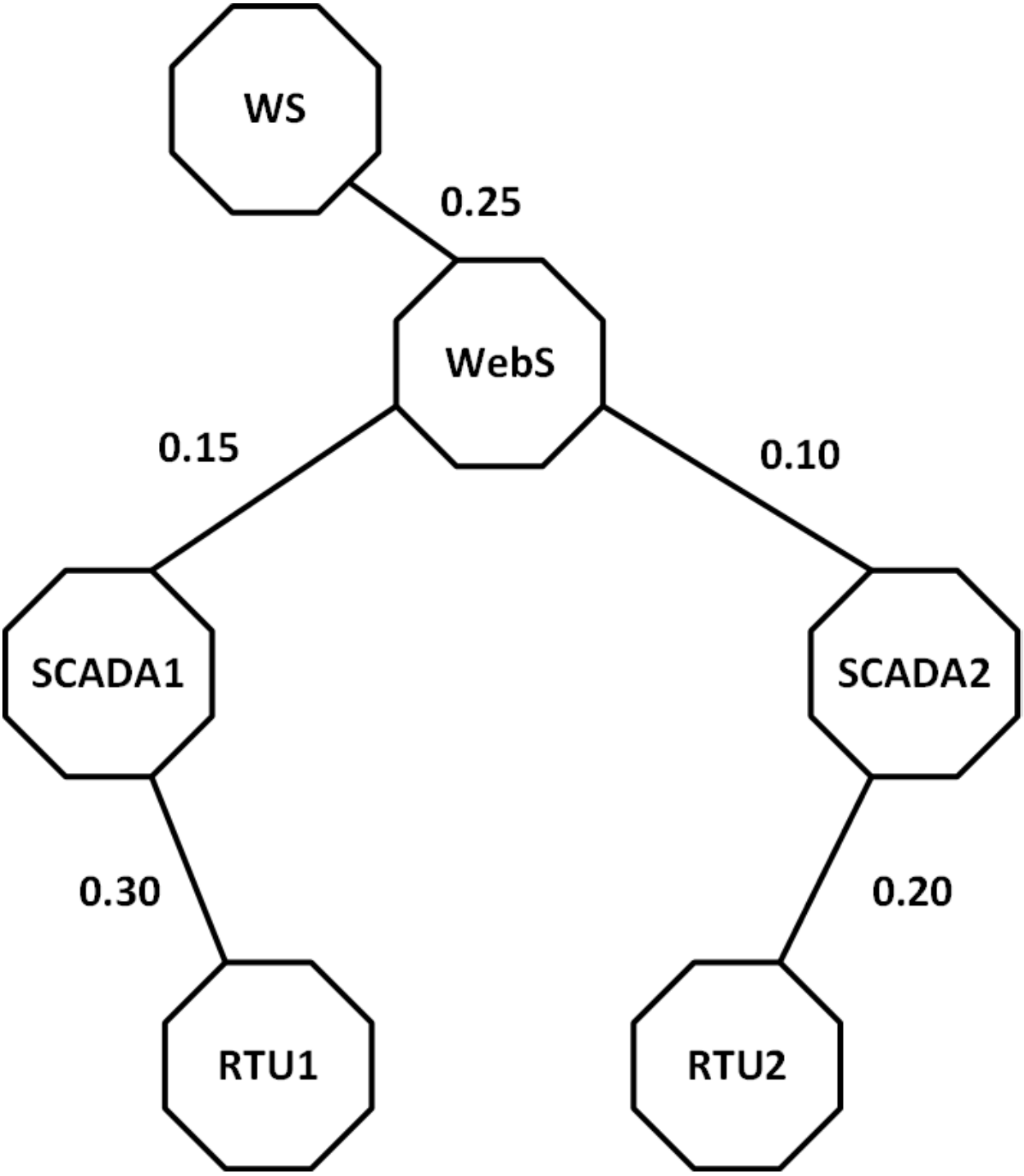


Figure 11. The weighted graph

Subsequently, the centrality of criticality may calculate by plugging the Centrality presented in Figure 11 and calculated in Eq. 16. The details are shown in Table 6 for different values of δ.

Table 6. Degree centrality at different δ.

Node	c_d	c_d^w	$CEN(i) = k_i^{1-\delta} \times s_i^{\delta}$ when $\delta =$			
			0	0.5	1.0	1.5
WS	1	0.25	1	0.5	0.25	0.125
WebS	3	0.5	3	1.225	0.5	0.204
SCADA1	2	0.45	2	0.949	0.45	0.213
SCADA2	2	0.3	2	0.775	0.3	0.116
RTU1	1	0.3	1	0.548	0.3	0.164
RTU2	1	0.2	1	0.447	0.2	0.089

Table 6 illustrates the effect of the δ on the degree of centrality for the nodes in Fig 11. It can be explained logically from this table that when $\delta = 1$ the measure's value is equal to the node's weight (Eq. 16). When $\delta < 1$ and the total node weight is constant, then the number of connections increases the value of the measure. For example, when $\delta = 0.5$, node WebS attains a higher score than node SCADA1, despite having almost the same node weight. Conversely, when $\delta > 1$ and the total node weight is constant, then the number of connections decreases the value of the measure in favor of a greater node weight concentration. Hence, node WebS attains almost the same value of the measure as node $SCADA1$, so logically in our EDS model, we choose the tuning parameter as $0 < \delta < 1$.

The locality of Criticality (l) derives from the running applications (like HMI tick), services (Operation-critical or Non-critical services, etc.), and processes collected from hosts' logs. To calculate the Damage characteristic, we only focus on the messages that regulate the level of 0 sensors and breakers. From DNP3 messages, we determined that the $SCADA1$ is controlling a substation of 3 MW load through 10 RTUs, whereas the SCADA2 is controlling 2 MW substation through 7 RTUs. Plugging those load values into Eq. 17, we determined the Damage characteristic of Criticality for individual $SCADA$. The RTU's damage characteristic of Criticality calculates by dividing individual $SCADA$'s damage characteristics

by the number of RTUs under this $SCADA$. Here, $P_T = 5MW$ and $L = 2$, so the total Criticality of a node derives from the (l), (CEN) from Table 6, and (d) in Eq. 13. Table 7 shows the calculation of total Criticality (C) of an individual node in EDS.

Table 7. Total Criticality Calculation

Nodes	l	CEN($\delta = 0.5$))	d	$Criticality(C)$
WS	1	0.5	0	0.325
WebS	2	1.225	0	0.684
SCADA1	3	0.949	0.6	1.252
SCADA2	3	0.775	0.4	1.106
RTU1	4	0.548	0.06	1.18
RTU2	4	0.447	0.057	1.101

For total criticality calculation in Table 7, we set $\alpha = 0.15, \beta = 0.25$ and $\gamma = 0.6$. The system administrator can now apply nodes' criticality to the AG and determine the most critical path along with most probable paths for a certain attacker goal to achieve.

Figure 10 also shows the cumulative probability and its consequences. Assuming the attacker's goal is $SCADA1/SCADA2$, we can see two paths for the attacker to reach any of these servers. Among those paths: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10a \rightarrow 11a \rightarrow 12a; 0 \rightarrow 13 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10a \rightarrow 11a \rightarrow 12a$ belongs to SCADA1 and $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10b \rightarrow 11b \rightarrow 12b; 0 \rightarrow 13 \rightarrow 6 \rightarrow 7 \rightarrow 8 \rightarrow 9 \rightarrow 10b \rightarrow 11b \rightarrow 12b$ belongs to SACDA2.

Although these two paths have the same exploitation probability from the attacker starting node to the server goals of $SCADA1/SCADA2$, the damage that occurs along the paths is not the same. Consequently, assuming that the $SCADA1/SCADA2$ is not the only goal, taking the attacker can analyze options and not react to the next achievable stage of the AG during an actual attack. A knowledgeable attacker may then select the path where he/she can do the most damage with a different intention than described in [52]. Regardless, before an attack, a cyber-resilient organization will harden the most impactful path prior to an attack to reduce the overall potential damage while protecting the golden target, so, the recommendation should be first to harden the attack path with the highest risk score.

Consequently, the system's security planner needs to propose remediations to potential paths to block future malicious activities. In our model, we considered the allocated operating budget to monetize different security actions to be performed on our network. We considered that a security planner, respectively, could decide every unit of Criticality and resource budget. Total criticality is also scaled up here from $[0 \rightarrow 2]$ to $[0 \rightarrow 20]$ for simplicity. Suppose the system planner has 15 units of such a budget. The system administrator has two options to spend the budget optimally amongst nodes: 1) allocate according to the linear cost model, and 2) allocate according to the exponential cost model.

Initially, before applying any operating budget as remediation, the network's total risk value is 8.62. Applying a linear cost model after following Eq. 21, Eq. 22 and Eq. 23, the risk reduces to 4.30 which is 49.86% of total risk. The details are in Table 8.

Table 8. Linear Cost Resource Allocation

Nodes	C	$maxA$	$\frac{C}{maxA}$	A	$V(\%)$	R
WS	3.25	4.64	0.70	0	72	2.34
WebS	6.84	4.64	1.474	0	14.4	0.985
SCADA1	12.52	4.64	2.70	4.64	0	0
SCADA2	11.06	4.64	2.38	4.64	0	0
RTU1	11.18	4.64	2.41	4.64	0	0
RTU2	11.01	4.64	2.37	1.08	8.83	0.973

Allocating the budget according to the exponential cost model after following Eq. 25 and Eq. 26, the risk reduces to 5.41 which is 62.7% of total network risk. The details of the calculation are shown in Table 9:

The exponential cost model's risk reduction is slightly lower than the linear cost model because the exponential model never reduces vulnerability to zero. However, for both linear and exponential cost models, the optimal allocation is ensured when the budget is distributed according to nodes' rank. Figure. 12 shows budget allocation amongst nodes for linear cost and exponential cost allocation. In both cases, the limited budget (15 units) allocates after ranking their Criticality from highest to lowest: SCADA1, RTU1, SCADA2, RTU2, WebS,

Table 9. Exponential Cost Resource Allocation ($\lambda = 0.53$)

Nodes	C	$maxA$	$\frac{C}{maxA}$	A	$V(\%)$	R
WS	3.25	4.64	0.70	0.561	63.8	2.074
WebS	6.84	4.64	1.474	2.060	9.24	0.632
SCADA1	12.52	4.64	2.70	3.278	5.68	0.711
SCADA2	11.06	4.64	2.38	3.029	6.0	0.663
RTU1	11.18	4.64	2.41	3.051	5.97	0.667
RTU2	11.01	4.64	2.37	3.020	6.01	0.662

and node WS.

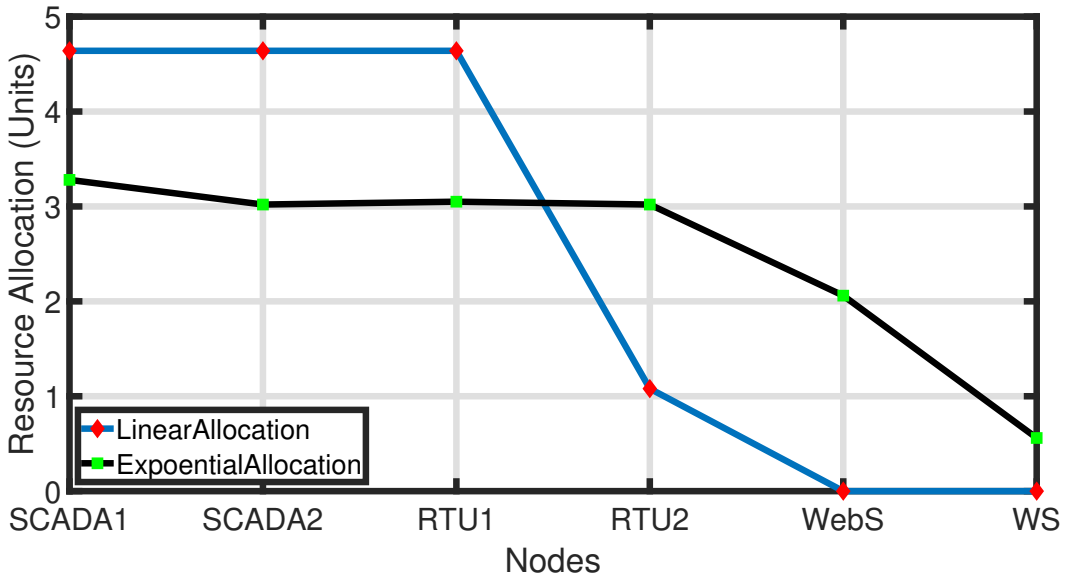


Figure 12. Linear and exponential resource allocation

Figure 13 depicts the allocation priority- from highest to lowest. Linear and exponential allocation obeys the rank-order established by the product of $\frac{C_i}{maxA_i}$- see the columns labeled $\frac{C}{maxA}$ in Table 8 and Table 9. This property is observed in allocation strategies regardless of whether the relationship between allocation and vulnerability reduction is linear, exponential, or a power law. Thus, the most critical nodes of a network are those with the highest $\frac{C}{maxA}$ value.

3.3 SUMMARY OF THE CHAPTER

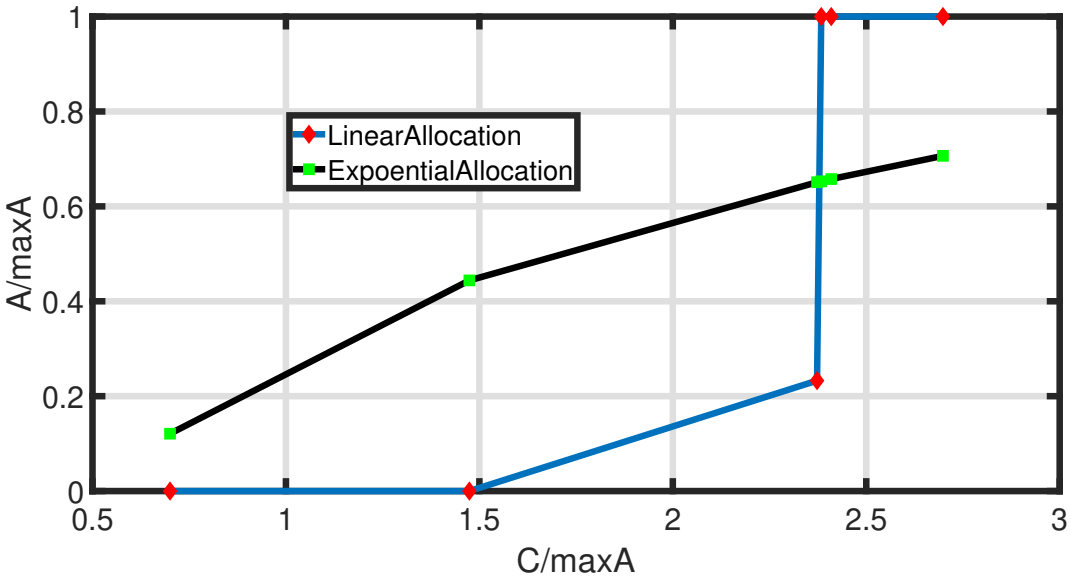


Figure 13. Linear and exponential cost allocation vs criticality

This chapter presented a data-driven model to assess a node's Criticality in a heterogeneous IT/OT/ICS EDS network. We also showed that assets along critical paths are as important as the target when several potential attack paths can be performed. We proposed critical node characteristics evaluation based on architectural location in IEC 62443, a measure of centrality based on node connectivity and frequency of network traffic, and electrical power control. We also examined the relationship between cost models of budget allocation to remove vulnerabilities on critical nodes and the impact on gradual readiness. Empirically validated in an existing network ICS test-bed computing nodes criticality, two cost models were examined. Although varied, we concluded the lack of correlation between cost models' types to the most damageable attack path and critical nodes readiness.

Chapter 4

OPTIMAL EVOLUTIONARY CYBER DEFENSE REMEDIATION AGAINST ADVANCED PERSISTENT THREAT IN ENERGY DELIVERY SYSTEMS

This chapter describes an evolutionary game model for cyber defense remediation against Advanced Persistent Threat in Energy Delivery Systems. The main contributions of this chapter are as follows:

(1). The dynamic stochastic attack-defense game model for describing the evolutionary process of cyber attack-defense remediation is constructed. The complex APT scenario has the characteristics of multistate and multistep. With the penetration of network attacks, the information gained by the attacker will gradually increase. Then, based on the new information, the attacker can implement new strategies. Accordingly, the rational defender needs to adjust the optimal defense strategies to improve the payoff. We use Logit Quantal Response Dynamics ($LQRD$) equation to describe the evolutionary mechanism of attack-defense strategies in an APT. Our model also considers different types of cyber players with different rationalities comparing with other game models. We distinguish different cyber players by measuring and quantifying their rational degrees. Moreover, we simulate the growth of rationality. Our model can guide decision-makers on what actions and decisions they need to take, the maximum payoff, and what impact the strategy has on its adversary.

(2). The improved strategy payoff calculation and formation of the optimal strategies are analyzed. The existing method only considers the direct security payoff, which affects the accuracy of strategy selection. In fact, the APT attack defense is often difficult to achieve by relying only on single remediation. Simultaneously, it requires the combination of various defense remediations to maximize the comprehensive defense payoff. This paper considers the indirect payoff of counterattacks' negative and positive impact. In the classical complete rational game model, the best approach is explained as the best reaction between

adversaries, but no forming process of this strategy is given. The emerging bounded rational game model of replicator dynamic only offers a simple decision-making approach but not a practical mixed strategy of APT like the combination of several defense techniques. This paper aims to analyze the dynamic evolution tracks of an optimal mixed strategy. We first simulate the formation process of player strategy through repeated games. Our approach can predict the possible attack strategy in the future of an APT chain and provide the corresponding best defense plan.

(3) The optimal cyber defense remediation selection approach for a multistep attack is designed. The optimal cyber defense remediation selection is given by solving a stable evolutionary equilibrium. To depict the attacker and defender's interaction process, we employ the evolutionary game to illustrate the two sides' decision interaction and behavior evolution. By calculating the game equilibriums in different game stages, we can calculate the optimal defense strategy arrangements at each moment. Compared with the static Nash equilibrium, the dynamic selections at different evolutionary times of APT and its earnings are depicted visually. The evolution tracks with the best strategies for both sides of the attacker and defender and are exhibited simultaneously. We give the convergence process from evolutionary equilibrium to the Nash equilibrium, which improves the dynamic analysis and situation prediction of attack early warning and defense decision-making.

4.1 GAME MODEL FOR CYBER ATTACK-DEFENSE REMEDIATION

The attacker–defender arms race model assumes both network opponents use the same strategy—and apply the same exponential cost model. Besides, it implies that each player knows the other players' allocation after each round of reallocations. These assumptions are perhaps valid in many situations, but what happens if attacker and defender don't know anything of each other's allocation strategy? Specifically, what is the best allocation strategy when neither party knows the policy of the other party? We turn to game theory to analyze this question. Different from the general replicator dynamics game model, we build the attack-defense game model based on improved $LQRD$ stochastic evolutionary game. By adding the parameter ς, we quantify the cognitive differences of diverse players. Through

this, we develop the previous approach by depicting the inertia, randomness, dynamics, and diversity of real-world biological players. In this section, we first demonstrate our motivation and then construct the Attack-defense Stochastic Evolutionary Game Model (ASEGM) using $LQRD$. Finally, we give the metric of strategy payoff based on cost and benefit analysis.

4.1.1 MOTIVATION

Like most other cyber-attacks, APT follows certain attack phases defined in a cyber kill chain [27]. This framework assumes that every attack sequence starts with a reconnaissance phase, in which an attacker tries to locate gaps and vulnerabilities of a target system. The weaponizing phase follows, during which the uncovered weaknesses are used to develop targeted malicious code. The delivery phase follows the weaponizing phase when the malware transfers to the potential target. When the malware delivers successfully, the exploit phase starts during which the malware triggers the installation of an intruder's code. Afterward, the compromised host system allows the establishment of a command and control channel so that the attacker can initiate malicious actions. The attacker uses different techniques and tools to move from one phase of the cyber kill chain to another phase [53]. Normally the defender collects network logs, host logs, and vulnerabilities to build up the attacker penetration path throughout the network. Based on the path penetration and critical consequences along the path, the defender takes his/her decision for applying control measures for remediation [36].

From the perspective of decision-making, we can abstract the APT security adversary as a stochastic game. If we treat the time of the whole attack-defense process containing a series of time slices as shown in Fig. 14, each time slice corresponds to one security phase of the cyber system, and then the attack-defense actions can be treated as a series of discrete events occurring at discrete times. In this way, we can process the cyber attack-defense process discretely. In each time slice, the player detects the current network state. Taking the time slice t_1 as an example, the player evaluates the present optimal action according to the system state and the adversary's response. The game ends when the network transfers to the security state. During this process, when one side changes its strategy, the game system moves to an unbalanced state. Then, the security state of time slice t_1 transfers to the next

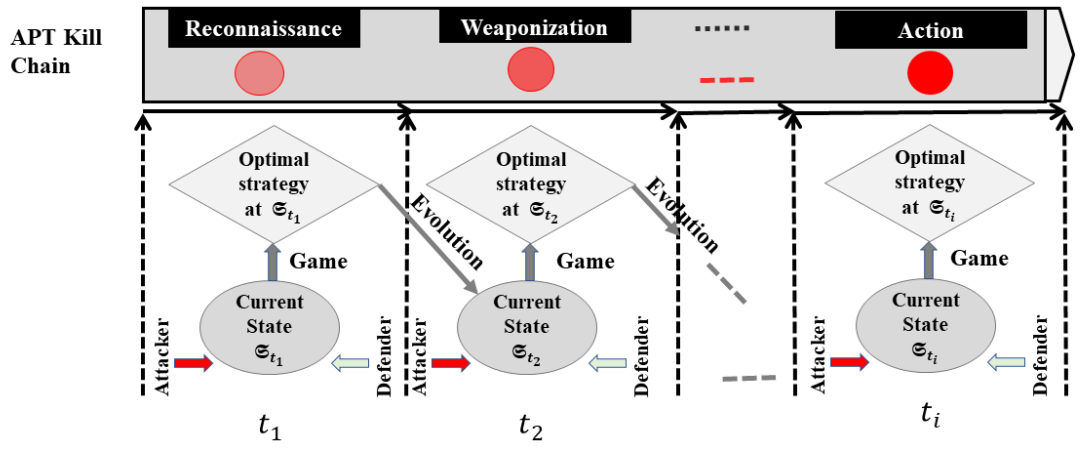


Figure 14. The process of optimal strategy selection for an APT attack-defense game varying with time.

state of time slice t_2 under the actions of both attacker and defender. From the dimension of time, the attacker and defender make a continuous decision and dynamic adversary over time. The game equilibrium strategy is treated as the controller of the track depicting state evolution. By predicting the optimal strategy at each game moment, we can improve the accuracy and timeliness of security decision-making.

Let's explain our motivation with an example as follows:

1) For the bounded rational attack-defense modeling, existing researchers generally adopt the replicator dynamic. The core idea is that the population with lower rationality will gradually take the strategy with a higher average payoff, which essentially reflects the deterministic selecting behavior. In other words, the mechanism of selection determines that players always select high-payoff strategies during evolution. For instance, we assume a population containing three game players, in the first evolution round; the obtained payoffs are respectively $6, 8, 10$. Then we can calculate the average payoff 8. In the second round, due to that, the player with payoff 6 earns less than the average level. He prefers to select the

strategy with payoff 10 and obtains the corresponding payoff level. Then the three payoffs are $10, 8, 10$, and the average payoff is 9.3. In the third round, since the player with payoff 8 earns less than the average level, he adopts the strategy of payoff 10. In this way, all three players obtain the payoff 10 after three rounds of the game. The population realizes the evolution from low payoff to high payoff. This process requires that the game players can always identify and learn the high-payoff strategy accurately. However, due to the influence of the attack-defense players' factors (skill level, safety knowledge, prior experience, etc.), the cognitive abilities of different players are usually different. Not all individuals can correctly calculate the expected strategy payoff. These individuals have specific probabilities of changing planned strategies. We call this a stochastic disturbance. Moreover, for some multi-variant replicator dynamic equations, there are no polymorphic equilibrium solutions, which reduces the operability.

2) The $LQRD$ considers that the player belief is keeping rational, while the limitation of learning capability leads to the gap of achieving an ideal Nash equilibrium. We denote the payoff as $\mathfrak{U} = \mathfrak{V} + \varepsilon$, where $\mathfrak{V}$ means the payoff generated by deterministic factors, while ε denotes the payoff caused by uncertain disturbance. The players make decisions for gaining maximum payoff $\mathfrak{U}$. Compared with the general replicator dynamics, we further consider the individual preferences and cognitive differences. On this basis, we introduce the parameter ς to quantify the degree of player rationality to reflect the diversities of population behaviors. Meanwhile, with the increase of ς by strategy learning and improving, players can obtain indirect decision information by observing their own experience or other players' decisions in similar environments. Meanwhile, the players can also get direct decision-making information on the population's strategy distribution from the observed game history. Through repeated games, the cognition of players is enhanced. We exhibit the best strategy selection varying with time, which improves the interpretation and prediction for strategy formation.

4.1.2 THE SYSTEMS MODEL OF OUR OPTIMAL CYBER DEFENSE REMEDIATION

The architecture of our optimal defense decision-making approach is illustrated in Fig. 15. The input includes evidence such as vulnerability database, Nessus scanned logs, Attack

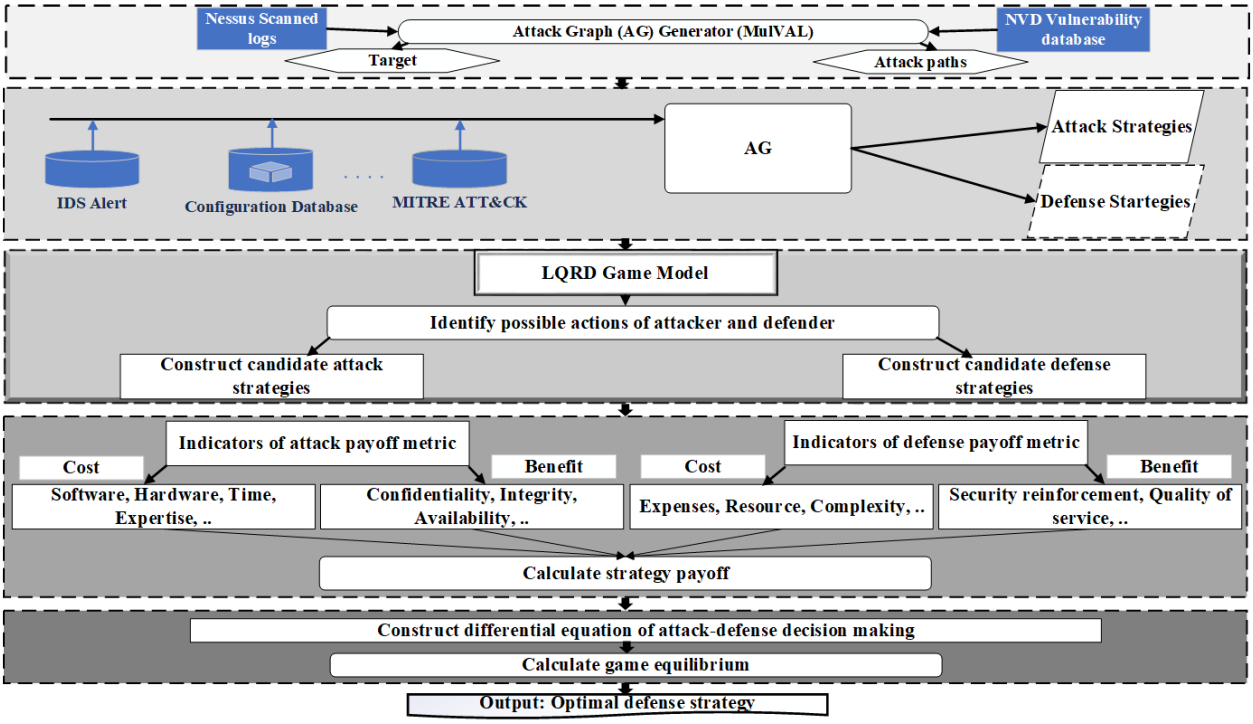


Figure 15. The architecture of our cyber defense remediation method

Graph (AG), IDS real-time alert, security configuration, network topology, and MITRE ATT&CK, and the output is the optimal defense strategy. The decision-making process contains five steps: (1) Determine the targets and critical attack paths to those targets of strategy selection. (2) Extract candidate attack-defense strategies from the input security data according to the enhanced AG of attack evidence and abnormal evidence. (3) Model the process of attack-defense as the stochastic evolutionary game based on the $LQRD$ model. (4) Evaluate the strategy payoff based on cost-benefit analysis. (5) Generate optimal defense strategy.

In addition to that, we consider that the actual cyber adversary scenario usually consists of multiple players. We also extract the set of candidate defense strategies by analyzing the network environment information, including the vulnerability repairs, firewall access rules, security configuration and so on. We further collect alert data of firewall, IDS, and the virus detection system and host audit log. By analyzing the attack behavior information, we can extract the set of candidate attack strategies by referring to the network behaviors database

of MITRE ATT&CK [54].

4.1.3 GAME MODELING OF ATTACK-DEFENSE BASED ON *LQRD*

The evolutionary game model includes four essential elements: player sets, candidate strategy set, belief set, and payoff set.

Definition 1. A four-tuple can denote the model of the Attack-defense Stochastic Evolutionary Game (ASEGM).

(1) $\mathfrak{N} = (\mathfrak{N}_{\mathfrak{A}}, \mathfrak{N}_{\mathfrak{D}})$ is the population set of attack-defense players, where $\mathfrak{N}_{\mathfrak{A}}$ and $\mathfrak{N}_{\mathfrak{D}}$ are the populations of attackers and defenders, respectively.

(2) $\mathfrak{S} = (\mathfrak{S}_{\mathfrak{A}}, \mathfrak{S}_{\mathfrak{D}})$ is the set of candidate attack-defense strategies, in which $\mathfrak{S}_{\mathfrak{A}} = \{\mathfrak{A}_1, \mathfrak{A}_2, \ldots.., \mathfrak{A}_n\}$ is the set of the candidate strategies for attackers, $\mathfrak{S}_{\mathfrak{D}} = \{\mathfrak{D}_1, \mathfrak{D}_2, \ldots.., \mathfrak{D}_m\}$ is the set of candidate strategies for defenders. n and m are the numbers of attack and defense strategies, respectively. where $m, n \epsilon N^+$ and $n, m \geq 2$

(3) $\Theta = (\mathfrak{P}, \mathfrak{Q})$ is the belief set of the attack-defense game, where $p_i \epsilon \mathfrak{P}$ represents the probability that the attacker selects candidate strategy $\mathfrak{A}_i$, $q_i \epsilon \mathfrak{Q}$ represents the probability that defender chooses candidate strategy $\mathfrak{D}_j$, where $1 \leq i \leq n$, $1 \leq j \leq m, \sum_{i=1}^{n} p_i = 1, \sum_{j=1}^{m} q_j = 1$

(4) $\mathfrak{U} = (\mathfrak{U}_{\mathfrak{A}}, \mathfrak{U}_{\mathfrak{D}})$ is the payoff function set. $\mathfrak{U}_{\mathfrak{A}}$ and $\mathfrak{U}_{\mathfrak{D}}$ represent the payoff functions of attack and defense, respectively.

4.1.4 GAME PAYOFF QUANTIFICATION OF ATTACK-DEFENSE STRATEGY

Considering condition 4) of *Definition* 1, the payoff quantification of the attack-defense strategy is the basis of defense strategy selection. Therefore, its accuracy directly affects the selecting results. We summarized the types of different attack-defense strategies and proposed the payoff metric based on cost-benefit analysis.

Definition 2: Attack Benefit (AB) is the earned network resources through a series of attack actions or the level of network damage, which reflects the capability of controlling the targeted network system.

Definition 3: Attack Cost (AC) is the cost or effort that an attacker pays to obtain

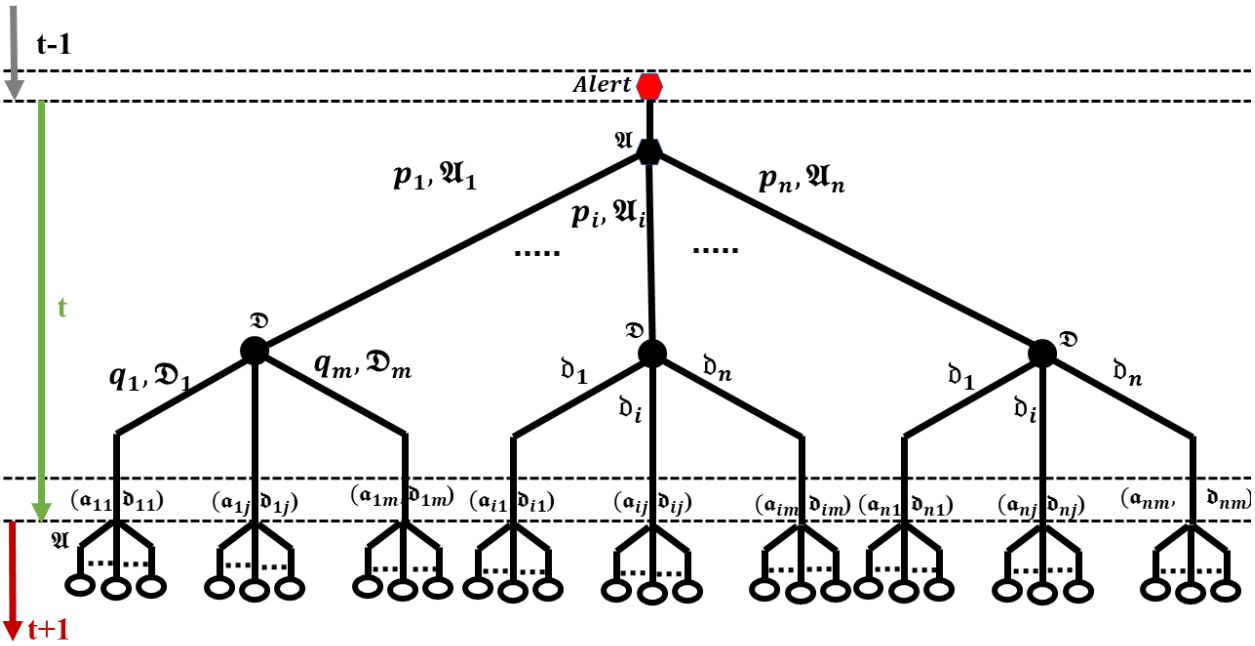


Figure 16. The game progression of attack-defense

network resources or cause losses to the network system.

Definition 4: Defense Benefit (DB) includes direct benefit and indirect benefit. The immediate benefit is the level of security reinforcement through security measures only considers the direct benefits, and we further add the indirect benefits of defender through the counterattack. For example, the electronic evidence of port scanning time, port number, source IP address, and destination IP address can be used to reconstruct the attack chain through which the defender can earn indirect benefits through investigating criminal responsibility.

Definition 5: Defense Cost (DC) is the cost or effort that defenders take against the possible attacks, including the human and time cost of the implementation of security devices, and the economic value of affecting the regular operation of service (a.k.a.negative impact of control measures.).

Definition 6: Attack-defense payoff matrices M are as follows. $\mathfrak{a}_{ij}$ and $\mathfrak{a}_{ij}$ represent the attack and defense payoff of selecting strategy combination (A_i, D_j) respectively, $a_{ij} =$

$AB - AC$, $d_{ij} = DB - DC$. The payoff matrices M is as below:

$$M = \begin{pmatrix} \mathfrak{a}_{11}, \mathfrak{d}_{11} & \mathfrak{a}_{12}, \mathfrak{d}_{12} & \cdots & \mathfrak{a}_{1m}, \mathfrak{d}_{1m} \\ \mathfrak{a}_{21}, \mathfrak{d}_{21} & \mathfrak{a}_{22}, \mathfrak{d}_{22} & \cdots & \mathfrak{a}_{2m}, \mathfrak{d}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathfrak{a}_{n1}, \mathfrak{d}_{n1} & \mathfrak{a}_{n2}, \mathfrak{d}_{n2} & \cdots & \mathfrak{a}_{nm}, \mathfrak{d}_{nm} \end{pmatrix}$$

The attack-defense game tree shown in Fig. 16 depicts the attack-defense payoff with different combinations of candidate strategies visually. When the defender detects attack events, the game analysis begins at time $t - 1$. The details of the process are as follows:

(1) Both sides of the attacker and defender detect the current security state of the network system in the period of t,

(2) Players select their optimal actions according to the candidate strategies and their payoffs in the period of t.

(3) Players earn actual rewards through implementing strategies.

(4) Some players change their strategies through learning and modifying in the period of $t + 1$.

(5) Repeat steps (1-4) until the game system transfers to a stable state. That is, no player can make higher earnings by changing its strategy alone.

4.2 OPTIMAL DEFENSE DECISION MAKING APPROACH

4.2.1 CONSTRUCTION OF EVOLUTION EQUATIONS FOR ATTACK-DEFENSE DECISION MAKING

Evolutionary stable strategy (ESS) is an optimal decision of the game system in long-time strategy evolution. We obtain the best policy, which is balanced and stable and can protect against the invasion of other strategies. The definition of the permanent evolutionary strategy of cyber attack-defense is as follows.

Definition 7. Suppose the attacker population selects the candidate strategy set $\mathfrak{S}_{\mathfrak{A}} = (\mathfrak{A}_1, \mathfrak{A}_2, .., \mathfrak{A}_n)$ with the probability distribution $\mathfrak{P} = (p_1, p_2, .., p_n)$, and the defender

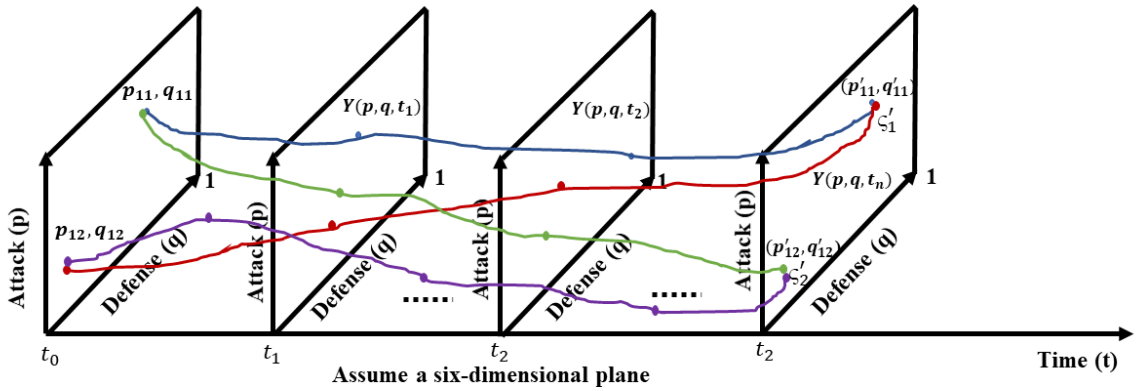


Figure 17. The evolution of strategy selection over time

population selects the candidate strategy set $\mathfrak{S}_{\mathfrak{D}} = (\mathfrak{D}_1, \mathfrak{D}_2, .., \mathfrak{D}_m)$ with the probability distribution $\mathfrak{Q} = (q_1, q_2, .., q_m)$. It means that individuals in the attacker and defense population randomly select and implement their pure strategies with the probability distribution $\mathfrak{P}$ and $\mathfrak{Q}$ in the actual adversary. We call $\sigma^* = (\mathfrak{P}, \mathfrak{Q})$ the stable strategy of attack-defense if the following conditions hold. $\mathfrak{U}(\sigma^*, \sigma^*)$ denotes the payoff when attacker and defender both select σ^*. For any $\sigma \neq \sigma^*$, $\mathfrak{U}(\sigma^*), \sigma)$ is the payoff when either side changes its strategy. Then the following conditions hold.

1). (stability) $\mathfrak{U}(\sigma^*\sigma^*) \geq \mathfrak{U}(\sigma, \sigma^*)$

2). (balance) $\mathfrak{U}(\sigma^*, \sigma^*) = \mathfrak{U}(\sigma, \sigma^*) \Rightarrow \mathfrak{U}(\sigma^*, \sigma) \geq \mathfrak{U}(\sigma, \sigma)$

Condition (1) guarantees that attacker and defender cannot earn more if either side changes strategy. In a policy containing a large number of σ^* and a small number of σ, it is necessary to meet that $\mathfrak{S}$ is the best response to itself; otherwise, other strategies may invade and develop.

Condition (2) guarantees that if there is another optimal strategy, σ, σ^* is required to react better than σ, which ensures that σ cannot develop even if the approach mutates to σ.

To explain Definition 7, Fig. 17 shows a possible dynamic track of the strategy evolved. $Y(p, q, t)$ is the probability function of selecting a strategy, t is the evolution times, and p and q denote the probabilities of selecting attack and defense strategy. The blue, red and green, magenta curves depict the evolution tracks of the game system with different initial states $(p_{11}, q_{11}), (p_{12}, q_{12})$ at two different ς. We can determine that the system can evolve to the two different stable states $(p_{11}^{'}, q_{11}^{'})$ and $(p_{12}^{'}, q_{12}^{'})$ through multiple repeated games. The middle strategy at time t_0, t_1, t_i, t_n can be obtained through intercepting the time surfaces of the game system. The optimal strategy we want to receive meets the Fig. 17; that is, no matter what action the attack-defense players take at the first moment, through strategy learning, imitation, and improvement, we can get the best strategy ultimately. The strategy is stable and anti-jamming, which can defend against the invasion of other approaches. The critical point is how to give the construction of $Y(p, q, t)$.

Definition 7 gives the condition of whether a strategy is an evolutionarily stable strategy, but does not characterize the track of players' selection on this strategy. As described in Section III-A, the attack-defense players search the best approach and are disturbed by stochastic error. This section describes the strategy evolution track by modifying the $LQRD$ equation to indicate the randomness of selection. The $LQRD$ uses Fisher-Tippett (an independent-identical-distribution) to depict the degree of noise influence on different players [55]. That is to say, the player selects the strategy with the exponential probability distribution, which is in line with the law of evolution of most things in the real world. Herein, we first give the deduction of the proposed $LQRD$ equation combined with *Definition 8 - Definition 10*.

Definition 8. The differential equation of the probability of selecting this strategy is [56]:

$$\frac{dp_i}{dt} = \sum_{k=1}^{n} p_k c_{ki} - \sum_{y=1}^{n} p_i c_{iy}$$

where p_i is the probability of selecting strategy $\mathfrak{A}_i$, $\frac{dp_i}{dt}$ is the probability that selects strategy $\mathfrak{A}_i$ varying with time. c_{ki} is the conditional transition probability of the attackers selection from strategy $\mathfrak{A}_k$ to strategy $\mathfrak{A}_i$, which describes the updating rules of strategy selections.

The core of the attack-defense evolutionary game is to study the dynamic change speed of the proportion of individual selecting strategy in the total population. That is, we need to calculate the selecting probabilities of different techniques. For conditional transition probability, we use the $LQRD$ equation to describe the rules of strategy updating and add an extra rationality parameter ς to quantify the cognitive capabilities of different game players. The improved $LQRD$ equation is defined as follows.

Definition 9. Improved transition probability of $LQRD$ is:

$$c_{ki} = \frac{exp(\varsigma \mathfrak{U}_{\mathfrak{A}_i})}{\sum_{k=1}^{n} exp(\varsigma \mathfrak{U}_{\mathfrak{A}_i})}$$

Set the rational parameters $\varsigma(\varsigma \geq 0)$ based on the historical rational degrees of players. The bigger ς is, the higher the degree of rationality is. As described in Section III-A, the payoff is $\mathfrak{U} = \mathfrak{V} + \epsilon$, where $\mathfrak{V}$ is the payoff of observable factors, ϵ is the payoff of uncertain factors. The deduction of c_{ki} in *Definition 9* can be referred to [55]. Take the formula in *Definition 9* into*Definition 8* and get the $LQRD$ equation as *Definition 10*.

Definition 10. After pluging the c_{ki} from definition 9 into definition 8, the equation of LQRD is:

$$\frac{dp_i}{dt} = \frac{exp(\varsigma \mathfrak{U}_{\mathfrak{A}_i})}{\sum_{k=1}^{n} exp(\varsigma \mathfrak{U}_{\mathfrak{A}_i})} - p_i$$

Definition 10 shows that the change rate of the population proportion of player selecting strategy $\mathfrak{A}_i$ is proportional to the difference between the ratio of individual expected payoff to the total gain and the balance of unique numbers of choosing this strategy to the whole numbers.

Definition 10 also shows that in the attacker population composed of bounded rational players, the number change rate of players selecting a specific candidate strategy varies with the proportion of this strategy payoff to the total gain.

To construct the $LQRD$ equations of attack-defense, from condition 3) of *Definition 1*, we denote the strategy of probability vectors $\mathfrak{P}$ and $\mathfrak{Q}$ is the mixed probability of selecting $\mathfrak{S}_{\mathfrak{A}}$ and $\mathfrak{S}_{\mathfrak{D}}$ respectively. The evolution equations are as follows:

(1). Evolution equation of attack strategy over time

The expected payoff $\mathfrak{U}_{\mathfrak{A}_i}$ of an attacker selecting candidate strategy $\mathfrak{A}_i$ is as follows, $i = 1, 2, ..., n$

$$\begin{aligned}
\mathfrak{U}_{\mathfrak{A}_1} &= q_1\mathfrak{a}_{11} + q_2\mathfrak{a}_{12} + ... + q_m\mathfrak{a}_{1m} \\
\mathfrak{U}_{\mathfrak{A}_2} &= q_1\mathfrak{a}_{21} + q_2\mathfrak{a}_{22} + ... + q_m\mathfrak{a}_{2m} \\
\mathfrak{U}_{\mathfrak{A}_i} &= q_1\mathfrak{a}_{i1} + q_2\mathfrak{a}_{i2} + ... + q_n\mathfrak{a}_{im} = \sum_{j=1}^{m} q_j\mathfrak{a}_{ij} \\
&.... \\
\mathfrak{U}_{\mathfrak{A}_n} &= q_1\mathfrak{a}_{n1} + q_2\mathfrak{a}_{n2} + ... + q_m\mathfrak{a}_{nm}
\end{aligned}$$

The changing rate of the proportion of individuals selecting strategy $\mathfrak{A}_i$ in the attacker population overtime is $\frac{dp_i}{dt}$. It reflects the learning and improving selecting strategy $\mathfrak{A}_i$ for bounded rational attacker through repeated games. The $LQRD$ differential equation of change rate is:

$$\frac{dp_i}{dt} = \frac{exp(\varsigma \sum_{j=1}^{m} q_j\mathfrak{a}_{ij})}{\sum_{k=1}^{n} exp(\varsigma \sum_{j=1}^{m} q_j\mathfrak{a}_{kj})} - p_i$$

(2). Evolution equation of defense strategy over time.

The expected payoff $\mathfrak{U}_{\mathfrak{D}_i}$ of an attacker selecting candidate strategy $\mathfrak{D}_j$ is as follows, $j = 1, 2, ..., m$

$$\mathfrak{U}_{\mathfrak{D}_1} = p_1\mathfrak{d}_{11} + p_2\mathfrak{d}_{21} + ... + p_n\mathfrak{d}_{n1}$$

$$\mathfrak{U}_{\mathfrak{D}_2} = p_1\mathfrak{d}_{12} + p_2\mathfrak{d}_{22} + ... + p_n\mathfrak{d}_{n2}$$

$$\mathfrak{U}_{\mathfrak{D}_j} = p_1\mathfrak{d}_{1j} + p_2\mathfrak{d}_{2j} + ... + p_n\mathfrak{d}_{nj} = \sum_{i=1}^{n} p_i\mathfrak{d}_{ij}$$

$$....$$

$$\mathfrak{U}_{\mathfrak{D}_m} = p_1\mathfrak{d}_{1m} + p_2\mathfrak{d}_{2m} + ... + p_n\mathfrak{d}_{nm}$$

The changing rate of the proportion of individuals selecting strategy $\mathfrak{D}_j$ in the defender population overtime is $\frac{dq_j}{dt}$. It reflects the learning and improving selecting strategy $\mathfrak{D}_j$ for bounded rational defender through repeated games. The $LQRD$ differential equation of change rate is:

$$\frac{dq_j}{dt} = \frac{exp(\varsigma \sum_{i=1}^{n} p_i\mathfrak{d}_{ij})}{\sum_{k=1}^{m} exp(\varsigma \sum_{i=1}^{n} p_i\mathfrak{d}_{ik})} - q_j$$

The practical significance of the above evolution equation is as below. Taking the defense strategyD_j as an example, if the number of individuals selecting the pure strategy D_j is smaller than the payoff proportion of individual obtaining from D_j, the growth rate of the defender number choosing D_j is larger than 0. Otherwise, the growth rate is less than 0. If the number proportion is exactly equal to the payoff proportion, then the growth rate of the number of defender selecting strategy D_j is 0. Set $F(p_i = \frac{dp_i}{dt})$, $G(q_j = \frac{dq_j}{dt})$, and then combine the above equations to equate the following condition:

$$Y(p_i, q_j) = \begin{pmatrix} F(p_i) \\ G(q_j) \end{pmatrix} = 0$$

This will give us the stable equilibrium of attack-defense adversary.

Algorithm 1: Optimal Cyber Defense Remediation:

BEGIN

1. *//Initialize attack-defense evolutionary game model//*

Initialize: $ASEGM = (\mathfrak{N}, \mathfrak{S}, \Theta, \mathfrak{U})$

{

1.1. *//Analyze security device configuration information to get the candidate defense strategy set//*

Construct $\mathfrak{S}_{\mathfrak{D}} = \{\mathfrak{D}_j\}, 1 \leq j \leq m$

1.2. *//Collect real-time alert data to get attack behaviors and extract the candidate attack strategy set//*

Construct $\mathfrak{S}_{\mathfrak{A}} = \{\mathfrak{A}_i\}, 1 \leq i \leq n$

1.3. *//Initialize attack believe set* $\mathfrak{P}$ *in which the attacker selects the attack strategy* $\mathfrak{A}_i$ *with the probability* $p_i \epsilon \mathfrak{P}$ *//*

(1.3) Construct $\mathfrak{P} = p_i, 0 \leq p_i \leq 1, \sum_{i=1}^{n} p_i = 1$

1.4. *//Initialize the defense believe set* $\mathfrak{Q}$ *in which the defender selects the attack strategy* $\mathfrak{D}_j$ *with the probability* $q_j \epsilon \mathfrak{Q}$ *//*

(1.4) Construct $\mathfrak{Q} = q_j, 0 \leq q_j \leq 1, \sum_{j=1}^{m} q_j = 1$

}

2. *//Calculate the attack-defense payoffs of different strategies combinations* $\mathfrak{A}_i$ *and* $\mathfrak{D}_j$ *in turn//*

for (i=1;i$\leq n; i++$) **do**

for (j=1;j$\leq m; j++$) **do**

{

Calculate: $\mathfrak{a}ij = AB(\mathfrak{A}_i, \mathfrak{D}_j) - AC(\mathfrak{A}_i, \mathfrak{D}_j)$ and $\mathfrak{d}ij = DB(\mathfrak{A}_i, \mathfrak{D}_j) - DC(\mathfrak{A}_i, \mathfrak{D}_j)$

}

3. *//Set the value of rationality degree* ς*//*

Assign $\varsigma_1, \varsigma_2; \varsigma_1, \varsigma_2 \geq 0$

4. *//Constructing the LQRD equation for selecting attack strategy* A_i*//*

for (i=1;i$\leq n; i++$) **do**

{

$$F(p_i) = \frac{exp(\varsigma \sum_{j=1}^{m} q_j \mathfrak{a}_{ij})}{\sum_{k=1}^{n} exp(\varsigma \sum_{j=1}^{m} q_j \mathfrak{a}_{kj})} - p_i$$

}

5. *//Constructing the LQRD equation for selecting defense strategy* $\mathfrak{D}_j$*//*

for (j=1;j$\leq m; j++$) **do**

{

$$\mathrm{G}(\mathrm{q}_j) = \frac{exp(\varsigma \sum_{i=1}^{n} p_i \mathfrak{d}_{ij})}{\sum_{k=1}^{m} exp(\varsigma \sum_{i=1}^{n} p_i \mathfrak{d}_{ik})} - q_j$$

}

6.*//Calculate the evolutionary stable equilibrium point//*

Calculate $Y(p_i, q_j) = \begin{pmatrix} F(p_i) \\ G(q_j) \end{pmatrix} = 0$

7.*//Find out the optimal strategy//*

$\mathfrak{Q} = \{q_1, q_2, ..., q_m\}$

END

=0

4.3 IMPLEMENTATION, RESULT, AND ANALYSIS:

The Industroyer malware has unleashed a major escalation in cyber-attacks on Industrial Control Systems (ICS) by combining a multi-stage APT attack with in-depth domain knowledge. Industroyer (also referred to as Crash-override) is a malware framework considered to have been used in the cyber-attack on Ukraine's power grid on December 17, 2016. The attack cut a fifth of Kyiv, the capital, off power for one hour and was considered a large-scale test [57]. The Kyiv incident was the second cyber-attack on Ukraine's power grid in less than a year. The first attack occurred on December 23, 2015. Industroyer is the first-ever known malware specifically designed to attack electrical grids. Simultaneously, it is the fourth malware publicly revealed to target industrial control systems, after Stuxnet, Havex, and BlackEnergy.

In this section, we take the invasion and proactive defense against Industroyer in the real-world Energy Delivery System (EDS) network as an example. We analyze the adversarial attack-defense process against Industroyer and verify the proposed approach for optimal defense strategy selection. The results of the two scenarios with different strategy payoffs are compared and analyzed. Besides, we summarize the general evolution rules of the best

defense strategy in the targeted network system. Finally, we compare our methods with the existing research comprehensively.

4.3.1 EDS NETWORK IMPLEMENTATION:

We implemented an EDS network that is shown in Fig. 18 in Accenture ICS research test-bed [48]. The entire test-bed is connected to a network switch and a router, and the zoning is implemented using VLAN and firewall rules.

There are five subnets created by an external and internal firewall. The IT Workstations (WSs) were located at the IT subnet. A Web Server (WebS) is located at the DMZ subnet and is directly accessible from the Internet through an external firewall. Supervisory Control and Data Acquisition (SCADA) servers (L3/L2), Remote Transmit Unit (RTUs) (L1) are in different subnets under a larger OT subnet that holds critical communication. The SCADA1 servers and SCADA2 servers are only accessible from the WebS of the DMZ. The WebS is accessible from user WS and other hosts from levels 4 or 5. The user subnet contains the user's WS. The firewalls allow all outbound traffic from the users' subnet. The test-bed also includes an Intrusion Detection System (IDS) running both IT and OT specific rules and a commercial OT Asset Discovery and Management (ADM). They are both connected to the span port of the switch to inspect all ICS traffic. For the Industroyer attack simulation, we injected vulnerabilities on the test-bed machines. The user workstations contained the vulnerability CVE-2009-1918 in Internet Explorer (IE). If a user accesses malicious content using the vulnerable IE browser, the device may be compromised. The WebS contained the vulnerability CVE-2006-3747 in the Apache HTTP service, resulting in a remote attacker executing arbitrary code on the machine. The SCADA1 and SCADA2 server had the vulnerability CVE-2018-5313, allowing privilege escalation up to the administrator level. The SCADA1 server controls 10 RTUs of substation 1, whereas the SCADA2 server controls 7 RTUs of substation 2. We assume that if an attacker acquires control over the SCADAs, the RTUs can be acquired as well.

As a defender, the network center's administrator is responsible for the security of the EDS's whole intranet. The attacker comes from the external network and attacks the intranet through the Internet. The purpose is to erase system-crucial registry keys and overwrite all

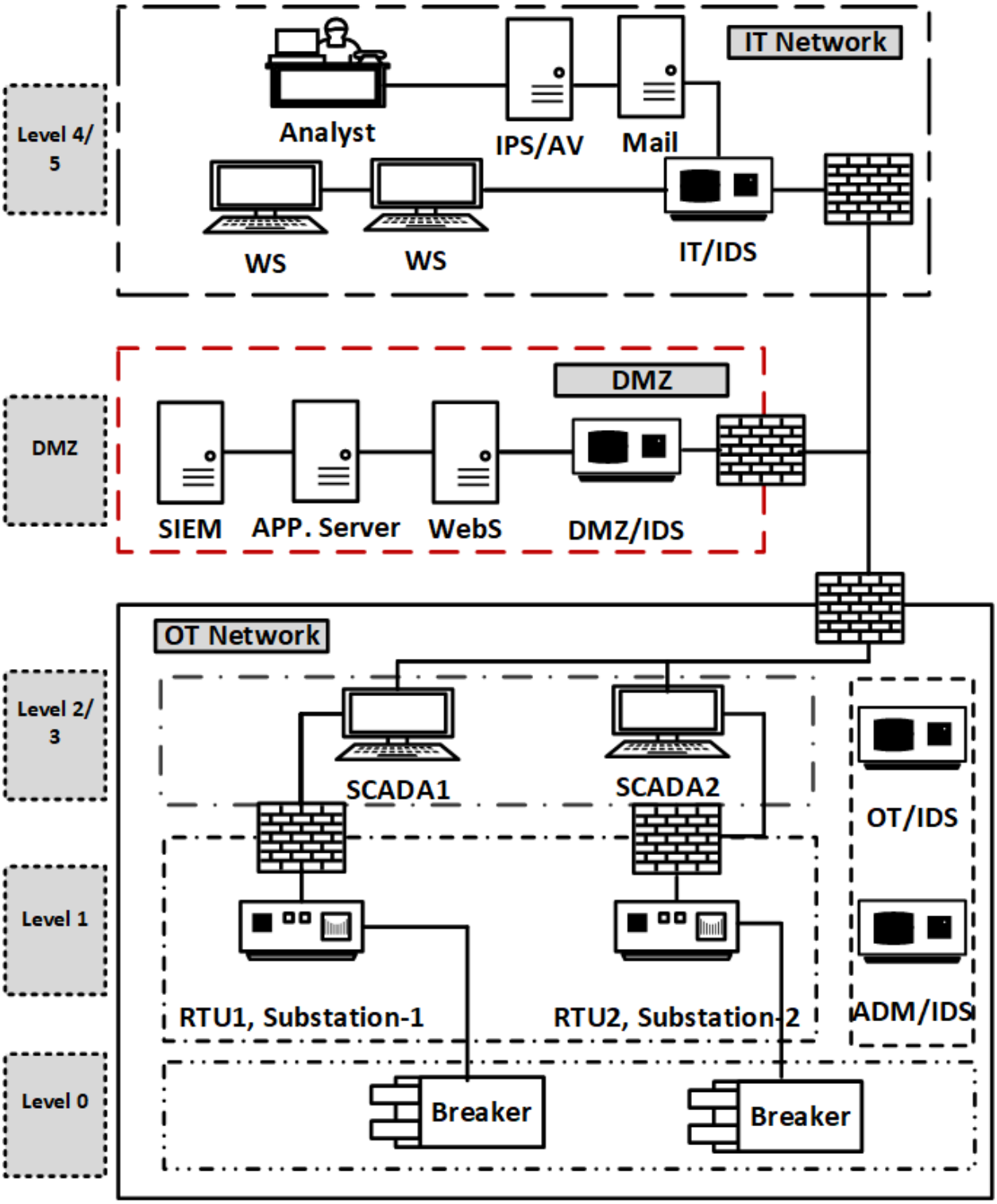


Figure 18. Logical view of EDS Test-bed for evolutionary game model

ICS configuration files to make the system unbootable and recovery from the attack harder. Industroyer attacks can be divided mainly into two steps, the first is to break through the boundary, and the second is to penetrate the intranet horizontally. Due to the firewall rules, external attackers can only communicate with the IT network's WS and mail server but cannot access the OT network. The security protection devices are composed of the firewall, IPS, virus detection system (VDS), and patch management system. We used the *Nessus* scanning tool to scan the EDS network. Table 10 shows the results of the principal vulnerabilities.

Table 10. Network Configuration and Vulnerability Information

Nodes	*Configuration*	*CVE*	*Description*
WS	Microsoft Internet Explorer (IE)	CVE-2009-1918	Allows remote attackers to execute arbitrary code via a crafted HTML document
WebS	Apache Web Server	CVE-2006-3747	allows remote attackers to cause a denial of service (application crash) and possibly execute arbitrary code via crafted URLs
SCADA 1	SCADA Master server	CVE-2018-5313	An attacker can leverage this vulnerability to execute arbitrary code under the context of Administrator
SCADA 2	SCADA Master server	CVE-2018-5313	An attacker can leverage this vulnerability to execute arbitrary code under the context of Administrator

4.3.2 CANDIDATE STRATEGY EXTRACTION AND PAYOFF CALCULATION:

In this experiment, based on the network topology and vulnerabilities, the logical Attack Graph (AG) is created using the open-source tool $MulVAL$ as illustrated in Fig. 19 [36]. The $MulVAL$ is a reasoning toolkit for automatically identifying vulnerabilities in IT and

OT networks [39]. The different shapes represent the network state, and the edge represents the atomic attack action. By referring to the attack-defense behavior database of MITRE for Industrial Control Systems (ICS) [52], we extracted the atomic attack and defense actions that can be launched in the network system. All the possible atomic actions are shown in Table 11.

Table 11. Cyber Attack and Defense Actions

No.	Attack Action	No.	Defense Option
$\mathfrak{A}_1$	Scan Port	$\mathfrak{D}_1$	Close Unused Port
$\mathfrak{A}_2$	Obtain Root Privilege	$\mathfrak{D}_2$	Restart Device
$\mathfrak{A}_3$	Buffer Overflow	$\mathfrak{D}_3$	Offline Network
$\mathfrak{A}_4$	Denial of Service	$\mathfrak{D}_4$	Block unwanted IPs
$\mathfrak{A}_5$	Execute Arbitrary Code	$\mathfrak{D}_5$	Install Patches

We find that the attacker first conducted port scanning action $\mathfrak{A}_1$ through port 25 of the mail server at the IT domain. Furthermore, the attacker collected open service information to prepare for subsequent attacks. Since port scanning is a concealed means of attacking, which is the passive attack virtually, we denote it as $\mathfrak{A}_1$ =*Scan Port*. Based on further detections and analyses of alert information, we find that some adventurous attackers may execute atomic attacks, $\mathfrak{A}_4$, and $\mathfrak{A}_5$ shown in Table 11 along the most critical path from the alert node to a goal SCADA 1/SCADA 2 [36]. The unauthenticated attackers exploit the vulnerability *CVE-2006-3747* of Webs at DMZ to allows remote attackers to cause a denial of service (application crash) and possibly execute arbitrary code via crafted URLs that are not adequately handled using certain rewrite rules. We denote this candidate strategy as $\mathfrak{A}_4$=*Denial Of Service (DoS)*, which is an active attack. After the WebS is compromised as the next stage of an APT, the attacker starts exploiting *CVE-2018-5313* of SCADA 1/SCADA 2. We denote this candidate strategy as $\mathfrak{A}_5$ = *Execute Arbitrary Code*, which is also an active attack, so in this experiment three candidate defense strategies $\mathfrak{D}_1$ = *Close Unused Ports*, $\mathfrak{D}_4$ =*Block Unwanted IP Address*, and $\mathfrak{D}_5$ = *Install Patches* are mapped from Table 11 as an extraction for that critical APT chain.

From *Definition 6*, the payoff matrix of attack-defense is as follows:

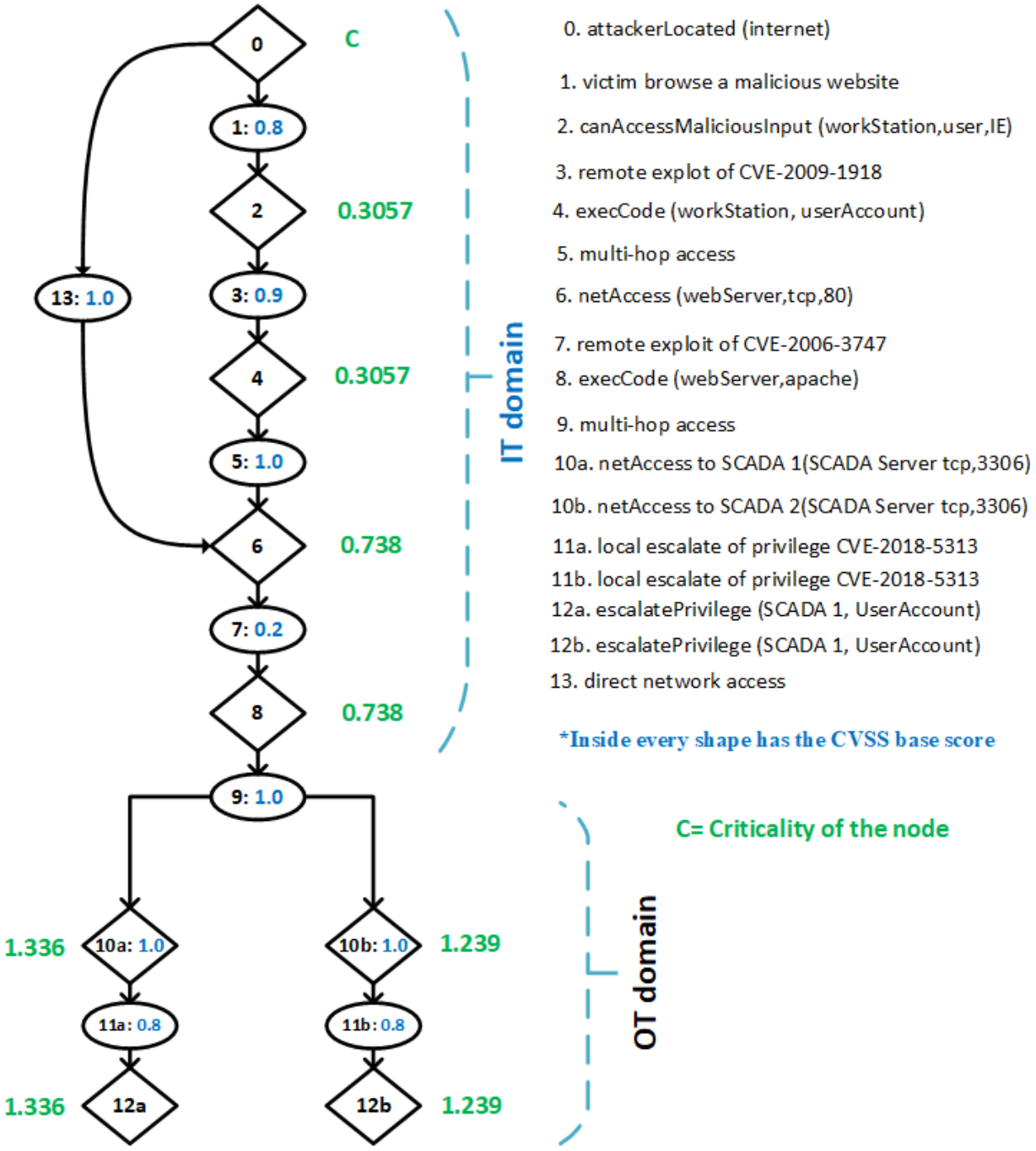


Figure 19. The AG of test-bed based EDS [36]

$$M = \begin{pmatrix} \mathfrak{a}_{11}, \mathfrak{d}_{11} & \mathfrak{a}_{12}, \mathfrak{d}_{12} & \mathfrak{a}_{13}, \mathfrak{d}_{13} \\ \mathfrak{a}_{21}, \mathfrak{d}_{21} & \mathfrak{a}_{22}, \mathfrak{d}_{22} & \mathfrak{a}_{23}, \mathfrak{d}_{23} \\ \mathfrak{a}_{31}, \mathfrak{d}_{31} & \mathfrak{a}_{32}, \mathfrak{d}_{32} & \mathfrak{a}_{33}, \mathfrak{d}_{33} \end{pmatrix}$$

4.3.3 EVOLUTION OF EQUATIONS FOR DECISION MAKING:

Firstly, we set the attackers and defenders with equal degrees of rationality. Furthermore, we assign the proportion of the number of players selecting strategy $\mathfrak{A}_1$, $\mathfrak{A}_4$, and $\mathfrak{A}_5$, in the attacker population as p_1,p_2, and p_3 respectively. Secondly, according to Steps (1.1-1.4) of *Algorithm 1*, we assign the proposition of defender population selecting strategy $\mathfrak{D}_1$, $\mathfrak{D}_4$, and $\mathfrak{D}_5$, as q_1, q_2, and q_3 respectively. Besides, we construct the $LQRD$ equation of attack-defense strategy as follows, respectively.

The expected payoff of the attacker selecting strategy $\mathfrak{A}_1$ = *port scan attack* is $\mathfrak{U}_{\mathfrak{A}_1} = \mathfrak{a}_{11}q_1 + \mathfrak{a}_{12}q_2 + \mathfrak{a}_{13}q_3$, the expected gain of *denial of service* is $\mathfrak{U}_{\mathfrak{A}_4} = \mathfrak{a}_{21}q_1 + \mathfrak{a}_{22}q_2 + \mathfrak{a}_{23}q_3$, and the expected payoff of the attacker selecting strategy $\mathfrak{A}_5$ = *Execute Arbitrary Code* is $\mathfrak{U}_{\mathfrak{A}_5} = \mathfrak{a}_{31}q_1 + \mathfrak{a}_{32}q_2 + \mathfrak{a}_{33}q_3$. Then, according to Step 4) of *Algorithm 1*, we can obtain the evolution equation of strategy $\mathfrak{A}_1$, $\mathfrak{A}_4$, and $\mathfrak{A}_5$ as follows:

$$\frac{dp_1}{dt} = \{\frac{exp(\varsigma(\mathfrak{a}_{11}q_1 + \mathfrak{a}_{12}q_2 + \mathfrak{a}_{13}q_3))}{exp(\varsigma(\mathfrak{a}_{11}q_1 + \mathfrak{a}_{12}q_2 + \mathfrak{a}_{13}q_3)) + exp(\varsigma(\mathfrak{a}_{21}q_1 + \mathfrak{a}_{22}q_2 + \mathfrak{a}_{23}q_3))+ exp(\varsigma(\mathfrak{a}_{31}q_1 + \mathfrak{a}_{32}q_2 + \mathfrak{a}_{33}q_3))\} - p_1$$

$$\frac{dp_2}{dt} = \{\frac{exp(\varsigma(\mathfrak{a}_{21}q_1 + \mathfrak{a}_{22}q_2 + \mathfrak{a}_{23}q_3))}{exp(\varsigma(\mathfrak{a}_{11}q_1 + \mathfrak{a}_{12}q_2 + \mathfrak{a}_{13}q_3)) + exp(\varsigma(\mathfrak{a}_{21}q_1 + \mathfrak{a}_{22}q_2 + \mathfrak{a}_{23}q_3))+ exp(\varsigma(\mathfrak{a}_{31}q_1 + \mathfrak{a}_{32}q_2 + \mathfrak{a}_{33}q_3))\} - p_2$$

$$\frac{dp_3}{dt} = \{\frac{exp(\varsigma(\mathfrak{a}_{31}q_1 + \mathfrak{a}_{32}q_2 + \mathfrak{a}_{33}q_3))}{exp(\varsigma(\mathfrak{a}_{11}q_1 + \mathfrak{a}_{12}q_2 + \mathfrak{a}_{13}q_3)) + exp(\varsigma(\mathfrak{a}_{21}q_1 + \mathfrak{a}_{22}q_2 + \mathfrak{a}_{23}q_3))+ exp(\varsigma(\mathfrak{a}_{31}q_1 + \mathfrak{a}_{32}q_2 + \mathfrak{a}_{33}q_3))\} - p_3$$

The expected payoff of the defender selecting strategy $\mathfrak{D}_1$ = *close unused port* is $\mathfrak{U}_{\mathfrak{D}_1} = \mathfrak{d}_{11}p_1 + \mathfrak{d}_{21}p_2 + \mathfrak{d}_{31}p_3$, the expected gain of *denial of service* is $\mathfrak{U}_{\mathfrak{D}_4} = \mathfrak{d}_{12}p_1 + \mathfrak{d}_{22}p_2 + \mathfrak{d}_{32}p_3$, and the expected payoff of the defender selecting strategy $\mathfrak{D}_5$ = *Execute Arbitrary Code* is

$\mathfrak{U}_{\mathfrak{D}_5} = \mathfrak{d}_{13}p_1 + \mathfrak{d}_{23}p_2 + \mathfrak{d}_{33}p_3$. Then, according to Step 5) of *Algorithm 1*, we can obtain the evolution equation of strategy $\mathfrak{D}_1$,$\mathfrak{D}_4$, and $\mathfrak{D}_5$ as follows:

$$\frac{dq_1}{dt} = \{\frac{exp(\varsigma(\mathfrak{d}_{11}p_1 + \mathfrak{d}_{21}p_2 + \mathfrak{d}_{31}p_3))}{exp(\varsigma(\mathfrak{d}_{11}p_1 + \mathfrak{d}_{21}p_2 + \mathfrak{d}_{31}p_3)) + exp(\varsigma(\mathfrak{d}_{12}p_1 + \mathfrak{d}_{22}p_2 + \mathfrak{d}_{32}p_3))+ exp(\varsigma(\mathfrak{d}_{13}p_1 + \mathfrak{d}_{23}p_2 + \mathfrak{d}_{33}p_3))\} - q_1$$

$$\frac{dq_2}{dt} = \{\frac{exp(\varsigma(\mathfrak{d}_{12}p_1 + \mathfrak{d}_{22}p_2 + \mathfrak{d}_{32}p_3))}{exp(\varsigma(\mathfrak{d}_{11}p_1 + \mathfrak{d}_{21}p_2 + \mathfrak{d}_{31}p_3)) + exp(\varsigma(\mathfrak{d}_{12}p_1 + \mathfrak{d}_{22}p_2 + \mathfrak{d}_{32}p_3))+ exp(\varsigma(\mathfrak{d}_{13}p_1 + \mathfrak{d}_{23}p_2 + \mathfrak{d}_{33}p_3))\} - q_2$$

$$\frac{dq_3}{dt} = \{\frac{exp(\varsigma(\mathfrak{d}_{13}p_1 + \mathfrak{d}_{23}p_2 + \mathfrak{d}_{33}p_3))}{exp(\varsigma(\mathfrak{d}_{11}p_1 + \mathfrak{d}_{21}p_2 + \mathfrak{d}_{31}p_3)) + exp(\varsigma(\mathfrak{d}_{12}p_1 + \mathfrak{d}_{22}p_2 + \mathfrak{d}_{32}p_3))+ exp(\varsigma(\mathfrak{d}_{13}p_1 + \mathfrak{d}_{23}p_2 + \mathfrak{d}_{33}p_3))\} - q_3$$

Then, according to Step 6) of *Algorithm 1*, equalize all six equations to zero. The solution of those six equations is the stable evolutionary equilibrium of attack-defense decision-making, and the defender's optimal defense strategy is selecting strategy $\{\mathfrak{D}_1, \mathfrak{D}_4, \mathfrak{D}_5\}$ with mixed probability $\{q_1, q_2, q_3\}$.

4.3.4 RESULT AND ANALYSIS:

We take two numerical experiments: *Scenario 1* (without considering counterattack payoff) and *Scenario 2* (considering counterattack payoff). The comprehensive comparison and analysis are finally given.

Scenario 1:

According to Step 2) of *Algorithm 1*, we combine *Definition 2 - Definition 5* and security behaviors database and then obtain the game payoff of attack-defense as organized in Table

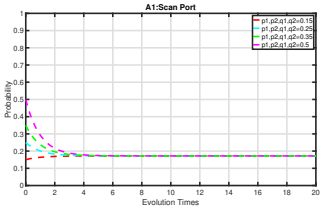
12.

Table 12. Game Pay-off of Scenario 1

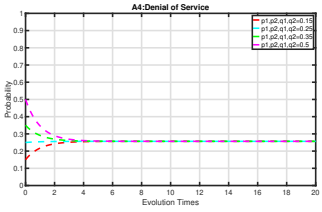
Candidate Attack Strategy	Candidate Defense Strategy		
	$\mathfrak{D}_1$	$\mathfrak{D}_4$	$\mathfrak{D}_5$
$\mathfrak{A}_1$	(0.16,0.06)	(0.16,-0.15)	(0.16,-0.3)
$\mathfrak{A}_4$	(0.24,-0.2)	(0.24,0.39)	(0.24,-0.3)
$\mathfrak{A}_5$	(0.4,-0.2)	(0.4,-0.15)	(0.4,0.7)

In general, the degree of player rationality in the real world is medium, and here we set $\varsigma = 5.0$, and set the initial state of the game system as $p_1 = p_2 = p_3 = q_1 = q_2 = q_3 = 0.33$ according to Step 5 of *Algorithm 1.* That is, the attacker randomly selects a strategy from candidate $\mathfrak{A}_1$,$\mathfrak{A}_4$, and $\mathfrak{A}_5$ with equal probability 0.33 at the initial time. Similarly, the defender randomly selects an action from candidate $\mathfrak{D}_1$, $\mathfrak{D}_4$, and $\mathfrak{D}_5$ with equal probability. With the simulation tool *Matlab* 2020, the stable equilibrium point is calculated by function $fsolve()$ for $\varsigma = 5.0$. The calculated stable equilibrium point is $\{p_1, p_2, p_3\} = \{0.172, 0.257, 0.571\}$ and $\{q_1, q_2, q_3\} = \{0.087, 0.179, 0.734\}$. In this context, the attacker is more likely to select $\{\mathfrak{A}_1, \mathfrak{A}_4, \mathfrak{A}_5\}$ with mixed probability of $\{0.172, 0.257, 0.571\}$. Meanwhile, the optimal defense strategy for the defender is to randomly implement $\{\mathfrak{D}_1, \mathfrak{D}_4, \mathfrak{D}_5\}$ with mixed probability $\{0.087, 0.179, 0.734\}$. The results show that the attacker is more likely to select the aggressive strategy $\mathfrak{A}_5$ = *Execute Arbitrary Code* with probability 0.571. Since the attack of the *Execute Arbitrary Code* is more harmful, to avoid the severe attack influence, the corresponding optimal defense strategy is to select $\mathfrak{D}_5$ = *Install Patch* with a probability of 0.571.

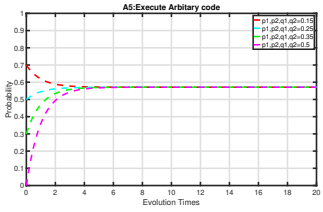
Secondly, to analyze the influence of the system's initial state on strategy selections, we simulate the evolution tracks of strategy selections with different first $p_1, p_2, p_3, q_1, q_2, q_3$ in Fig. 20 and Fig. 21. The abscissa t represents the number of evolutions in decision-making. The ordinate *probability* represents the probability of selecting a strategy. Fig. 20 and Fig. 21 can predict the defender's best strategy selection at different game moments.



$(a.)A_1$ =Scan Port

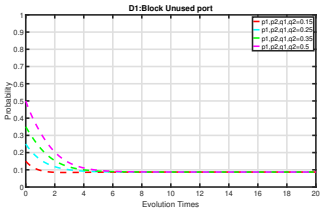


$(b.)A_4$ =Denial of Service

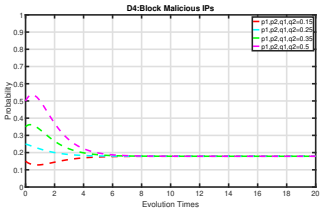


$(c.)A_5$ =Execute Arbitrary code

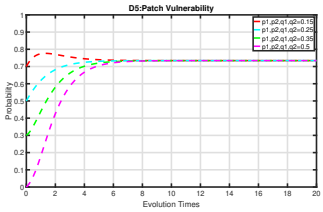
Figure 20. Strategy Evolution of an Attacker



$(d.)D_1$ =Close Unused ports



$(e.)D_4$ =Block malicious IPs



$(f.)D_5$ =Patch Vulnerability

Figure 21. Strategy Evolution of a Defender

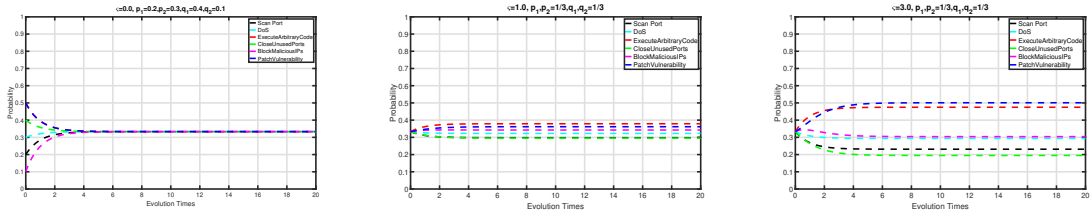
Fig. 20 (a-c) and Fig. 21 (d-f) respectively show the evolution tracks of $\{\mathfrak{A}_1, \mathfrak{A}_4, \mathfrak{A}_5, \mathfrak{D}_1, \mathfrak{D}_4, \mathfrak{D}_5\}$, when the initial states of attacker and defender are the same with $p_1, p_2, q_1, q_2 = \{0.15, 0.25, 0.35, 0.5\}$. From Fig. 20 (a-c) and Fig. 21 (d-f), we assume that the attacker and defender initially select the strategy $\{\mathfrak{A}_1, \mathfrak{A}_4\}$ and $\{\mathfrak{D}_1, \mathfrak{D}_4\}$ with probability $p_1 = p_2 = q_1 = q_2 = 0.5$, when t = 0. Then from the magenta curve of Fig. 21(d), the likelihood of selecting strategy $\mathfrak{D}_1$ is falling over time and stabilize to *probability* = 0.179, when $t = 10$. Also, the possibility of choosing a strategy $\mathfrak{D}_4$ is falling and stabilize to *probability* = 0.017 from the magenta curve of Fig. 21(e). Herein, the optimal defense strategy is selecting $\mathfrak{D}_1, \mathfrak{D}_4, \mathfrak{D}_5$ with mixed *probability* = $\{0.087, 0.179, 0.734\}$. This selection is stable and best when used against different candidate attack strategies.

Moreover, as we assume that the defender selects the strategy $\{\mathfrak{D}_1, \mathfrak{D}_4, \mathfrak{D}_5\}$ with a probability $\{q_1, q_2, q_3\}$=$\{0.5, 0.5, 0.0\}$ initially, namely, the larger the gap between the defender's initial selection and the optimal selection $\{q_1 = 0.087, q_2 = 0.179, q_3 = 0.734\}$, the more evolution times needed to achieve the best strategy. In contrast to the Nash equilibrium

game model [33], our approach can better explain the strategy evolution rules in adversarial attack-defense and have stronger performance of attack prediction.

Again, the higher the probability of selecting a strategy from $\{\mathfrak{A}_1, \mathfrak{A}_4\}$ at the initial time, the later the curve inflection point appears. They are indicating that more repeated games are required for decision-making and it takes a longer time. This is because the attacker selects $\mathfrak{A}_1$ or $\mathfrak{A}_4$ with a very high probability at the initial time. The false signal deceived the defender. It caused the defender mistakenly to assume that the attacker will select the moderate attack strategy about $\mathfrak{A}_1$ and $\mathfrak{A}_4$ while overlooking the ultimate attack purpose $\mathfrak{A}_5$ = *Execute Arbitrary Code*. Therefore, rational defenders need to implement many evolution times to discover the attacker's real purpose and obtain the best defense strategy. For example, when $\{p_1 = 0.5, q_2 = 0.5\}$, the probability of selecting the strategy $\mathfrak{D}_4$ denoted by the magenta curve in Fig. 8(e) first increases to $q_2 = 0.54$ at $t = 0.466$ and then rebounds and finally stabilizes to $q_2 = 0.178$ at $t = 11$. The reason is that the proportion of the defender population selecting strategy $\mathfrak{D}_4$ at the initial time increases to high. With the increase of the $\mathfrak{D}_4$ payoff to the total payoff, the number of individuals selecting $\mathfrak{D}_4$ decreases gradually to ensure that the proportion of population selecting $\mathfrak{D}_4$ to the total population is equal to the proportion of payoff selecting $\mathfrak{D}_4$ to the total payoffs.

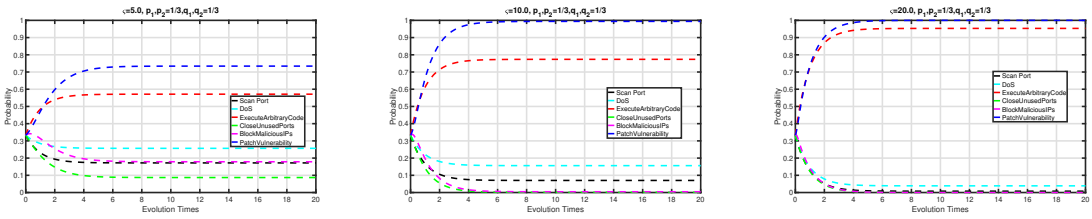
As can be seen from each column in Fig. 20-21, the optimal strategy for both defender and attacker are the same regardless of their initial p_1, p_2, p_3 and q_1, q_2, q_3 selections. It is only related to the candidate strategy set, player, and the strategy pays off. Moreover, the initial state can only affect the stabilization time of the game system.



$(a.)\varsigma = 0, p_1, p_2 = 0.2, q_1, q_2 = 0.1$ $(b.)\varsigma = 1.0, p_1, p_2 = 0.33, q_1, q_2 = 0.33$ $(c.)\varsigma = 3.0, p_1, p_2 = 0.33, q_1, q_2 = 0.33$

Figure 22. The strategy evolution tracks with different rationality ς.

Finally, to analyze the influence of degrees of players' rationality on strategy evolution, some simulations shown in Figs. 22-23 and discussions are as follows:



$(d.)\varsigma = 5.0, p_1, p_2 = 0.33, q_1, q_2 = 0.33$ $(e.)\varsigma = 10.0, p_1, p_2 = 0.33, q_1, q_2 = 0.33$ $(f.)\varsigma = 20.0, p_1, p_2 = 0.33, q_1, q_2 = 0.33$

Figure 23. The strategy evolution tracks with different rationality ς.

1) According to Step 5) of *Algorithm 1*, we assume that the players are irrational and set $\varsigma = 0$, assign initial $p_1 = 0.2, p_2 = 0.3, q_1 = 0.4, q_2 = 0.1$, then obtain the strategy evolution tracks in Fig. 22(a). Herein, the final result is to select the different candidate strategy with the same probability of 0.33. It means that players cannot distinguish the advantages and disadvantages of varying candidate strategies since they have no cognitive abilities. Meanwhile, from the *LQRD* equation of attack-defense in Section 4.2, there is only one solution $\{p_1, p_2, p_3, q_1, q_2, q_3\} = \{0.33, 0.33, 0.33, 0.33, 0.33, 0.33\}$ when $\varsigma = 0$. The results show that when the game players are irrational, regardless of their initial selections, they cannot distinguish each strategy's merits and demerits since they do not have any learning and cognitive capabilities. The candidate strategies are still selected by game players randomly.

2) Suppose that the rational player degree $\varsigma > 0$, we simulate the strategy evolution in Fig. 22 (b-c) and Fig. 23 (d-f). As time goes by, all the players can finally obtain the correct strategy through several times of repeated games. The main difference is that when the players have a high degree of rationality, they can find the optimal strategy more quickly. For example, when $\varsigma = 1$, the game system can reach the stable state through about five times of game evolution (shown in Fig 23 (e)), while when $\varsigma = 2$, they can be stable only through 4 times of game evolution (shown in Fig 23 (f)). The above results demonstrate that when the defenders have a high degree of rationality (have rich knowledge, skilled techniques, etc.), their cognition, learning, and adjustment abilities are strong, which helps the defenders identify the optimal strategy more quickly.

In general, both attackers and defenders gain increased decision-making experience

through adversarial attack-defense. Hence, a rational degree of ς increases during the game process. Fig. 24 illustrates the results under different ς, where the abscissa ς represents the reasonable degree, and the ordinate represents the probability of strategy selection. When $\varsigma = 0$, players have no rationality, so they choose candidate strategies randomly. When $\varsigma = 0.1$, the reasonable degree of the players is very low as the replicator dynamics [33]. From Fig. 24, the probability of a defender selecting strategy $\mathfrak{D}_1$ and $\mathfrak{D}_4$ rapidly decreases to 0 and $\mathfrak{D}_5$ increases to 1, respectively, which reflects the sensitivity of the decision-making system. The corresponding equilibrium solution is $\{p_1 = 0.33, p_2 = 0.33, p_3 = 0.34\}$ and $\{q_1 = 0.33, q_2 = 0.33, q_3 = 0.34\}$. The result corresponds to the replicator dynamic equilibrium [33]. Since the rational degree of dynamic replicator game is very low, its equilibrium solution is pure strategy. When $\varsigma > 0.1$, the player rational degree increases, and both sides of the attacker and defender always approach to complete balanced Nash equilibrium as ς increases. When $\varsigma > 15$, the solution $\{p_1 = 0.0003, p_2 = 0.27, p_3 = 0.97, q_1 = 0.02, q_2 = 0.1, q_3 = 0.88\}$ of $LQRD$ in this paper is very close to the Nash equilibrium solution. It indicates that the player rationality is very close to complete rationality over time, and the difference with the Nash equilibrium decreases gradually through obtaining experience in the game process. It is foreseeable that when ς moves towards infinity, then the proposed $LQRD$ equilibrium will approach Nash equilibrium. Compared with the complete rational Nash equilibrium [33] and the bounded rational replicator dynamic equilibrium, our approach can depict the diversity of rationality of attacker and defender players and reflect the real strategy selection rules.

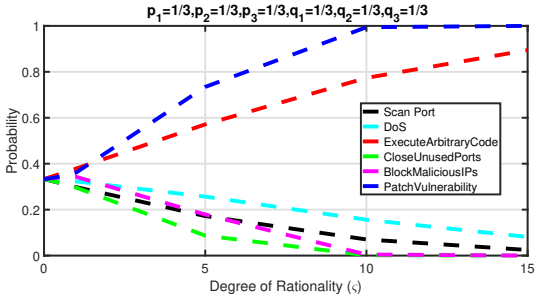


Figure 24. The impact of rationality (ς) on the strategy selections

Scenario 2:

Based on *Scenario 1*, this section further explores the impact of payoff changes (consider/do not consider countermeasure's negative/positive impact) on the selection of attack-defense strategies. Compared to *Scenario 1*, only considering the direct security rewards, in *Scenario 2*, we further discuss the indirect rewards through control measure selection and legal penalty, economic and time rewards (business recovery time, data recovery time, etc.) [58]. Table 13 gives the strategy payoff of *Scenario 2*, which depicts the difference from Table 12.

Similarly, we first consider the low medium rationality $\varsigma = 1$ and set the initial selection as $\{p_1, p_2, p_3, q_1, q_2, q_3 = 0.33\}$. The calculation process is the same as that of the *Scenario 1*. We obtain the evolution equilibrium point $\{p_1 = 0.2981, p_2 = 0.3229, p_3 = 0.3790, q_1 = 0.3661, q_2 = 0.4240, q_3 = 0.2099\}$ in Fig.25(a). Comparing the equilibrium points of *Scenario 1* (see in Fig. 22-23) with those of *Scenario 2* (see in Fig. 25), we can derive that:

1). For the attacker in *Scenario 2*, the probabilities of selecting a strategy are the same as with *Scenario 1* which are $\{p_1 = 0.2981, p_2 = 0.3229, p_3 = 0.3790\}$. That indicates that the attackers' expected pay offs do not depend on the negative impact of the defenders' strategy selections. Subsequently, attackers are not likely to change their strategy from selecting one to others.

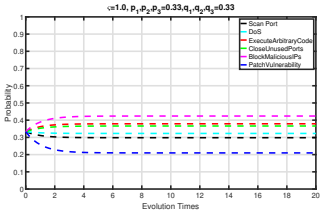
2). For the defender of *Scenario 2*, the probability of selecting strategy $\mathfrak{D}_1$ = *close unused ports* improves from 0.087 to 0.3246, and that of strategy $\mathfrak{D}_4$ = *Block unwanted IPs* from 0.1787 to 0.6664. On the other hand, the probability of selecting strategy $\mathfrak{D}_5$ = *Patch Vulnerability* reduces from 0.7343 to 0.0090 when $\varsigma = 5.0$. Since we consider the indirect negative impact [58] of selecting security control strategy $\mathfrak{D}_5$ = *Patch Vulnerability* dynamically in payoff calculation, the attacker did not force to adopt one moderate action than another among the three attack strategies. Accordingly, the defender was forced to increase the probability of selecting the simpler defense actions $\mathfrak{D}_1$ = *close unused ports* and $\mathfrak{D}_4$ =*Block unwanted IPs* over $\mathfrak{D}_5$ =*Patch Vulnerability*. This optimal selection can reduce the defense cost and mitigate the risk of the total network system.

Secondly, we set different initial $\{p_1, p_2, p_3, q_1, q_2, q_3 = variable\}$ and simulate the evolution tracks of both attacker and defender in Fig. 25 (c). The abscissa denotes the number of repeated games, and the ordinate indicates the probability of strategy selection. Similarly,

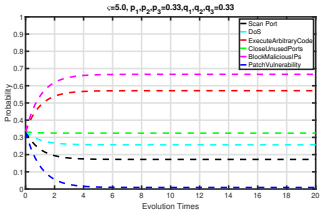
we can derive that the game system's stable equilibrium is not determined by the game system's initial state but is impacted by the candidate strategy set, players' rationality degree and payoff value. Different initial states can only affect the stabilization time; namely, the moment of the curve's inflection point appears, but cannot determine the final trend of the game. Moreover, rational players can always find the optimal strategy through strategy learning and improving the repeated game process. Additionally, the proposed optimal defense strategy has stronger foresee-ability and robustness when facing different candidate attack strategies.

Table 13. Game Pay-off of Scenario 2

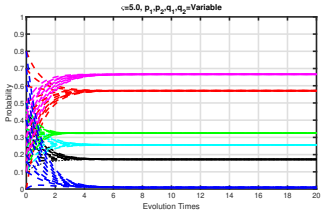
Candidate Attack Strategy	Candidate Defense Strategy		
	D_1	D_4	D_5
A_1	(0.16,0.06)	(0.16,-0.15)	(0.16,-0.3)
A_4	(0.24,-0.2)	(0.24,0.39)	(0.24,-0.3)
A_5	(0.4,-0.2)	(0.4,-0.15)	(0.4,-1.3)



$(a.)\varsigma = 1.0, p_1, p_2, p_3 = 0.33, q_1, q_2, q_3 = 0.33$



$(b.)\varsigma = 5.0, p_1, p_2, p_3 = 0.33, q_1, q_2, q_3 = 0.33$



$(c.)\varsigma = 5.0, p_1, p_2, p_3, q_1, q_2, q_3 = Variable$

Figure 25. The strategy evolution tracks of attack-defense strategies in case study 2

To sum up, through the above two scenarios, we can conclude that:

1). We established an evolutionary game model based on the bounded rationality of both sides of attackers and defenders. We considered that the process of adversarial attack-defense reaches a stable state progressively through repeated games. Figs. 20-23 and Fig. 25 depict the strategy's tracks, which predicts the best strategy selections at different game

moments. It presents the formation of the best defense strategy. It also provides an attack 's early warning and the corresponding security control options for cyberdefense. Our approach helps the security analyst to win the time warfare of cyber attack-defense effectively.

2). We simulate the defense evolution process against Crash Override attack. Since the radical attackers often select the strategy to *execute arbitrary code* at SCADA. Any disruption can happen at OT systems by an attacker that may also trigger safety issues at a power substation. Hence, most ICS power systems choose to avoid that extremely targeted attack at OT. Through our analyses on *Scenario 1*, if we cannot strengthen the defense and counterattack, the best strategy for the defender is to take the action of *Patch Vulnerability* in advance. Although this strategy's cost is high, from the game process of the whole attack-defense, it can ensure the defender maximizes the defense revenue.

3). We quantify the rationality of different attackers and defenders by introducing a flexible parameter of ς. Through this, we can depict the diversities of evolution behaviors. Fig. 24 shows the impact of players' rationality changes in the proposed approach in explaining the attack-defense adversarial process. It cannot only be converted to the dynamic replicator model of bounded rationality [59] but also can be converted to the Nash equilibrium model of complete rationality [32]. Moreover, we explain the dynamic approach process from low rational replicator dynamic equilibrium to a perfect balanced Nash equilibrium.

4). We find that a stable evolution strategy is only related to the candidate strategy and payoff value but not to the attacker's and defender's initial selections. In *scenario 2*, by adjusting the payoff of the candidate strategy, we can change the results of strategy selection to turn around the defense situation. For example, by increasing the payoff reward of attack penalty for defender and strengths on defense and counterattack, it is beneficial to guide attackers and defenders to adopt more moderate candidate strategies, avoid escalating conflicts, and promote cybersecurity governance.

4.3.5 COMPARISONS AND ANALYSIS

The comparisons among ours and others are organized in Table 14, and some discussions are as follows:

[31] [32] [33] [61] assume that attackers and defenders are rational. For instance, the

Table 14. Performances Comparison among Different Models

Ref.	Rationality	Game Type	Game Structure	Strategy Type	Equilibrium Solution	Generality
[31]	Complete	Static	n	Mixed	Nash Equilibrium	Medium
[32]	Complete	Static	n	Mixed	Nash Equilibrium	Medium
[60]	Bounded	Dynamic	2	Mixed	LQRD equilibrium	Low
[33]	Complete	Static	n	Mixed	Nash Equilibrium	Medium
[61]	Complete	Dynamic	n	Pure	Bayesian Equilibrium	Medium
[62]	Bounded	Dynamic	2	Mixed	Replicator dynamics evolutionary equilibrium	Medium
[63]	Bounded	Dynamic	n	Mixed	Replicator dynamics evolutionary equilibrium	High
[64]	Bounded	Dynamic	n	Mixed	Replicator dynamics evolutionary equilibrium	High
[59]	Bounded	Dynamic	n	Pure	Replicator dynamics evolutionary equilibrium	Medium
Ours	Bounded	Dynamic	n	Mixed	LQRD equilibrium	High

Nash equilibrium [32] [33] requires that all attackers and defenders can predict adversary's optimal strategy correctly at the same time. However, different players' cognitive capabilities are quite different, so the hypothesis of complete rationality deviates from reality. In contrast, [62] [63] [64] [59] [60] and ours regard that the players are bounded rational. We analyze

strategy learning and improving the mechanism of game players. Therefore, ours significantly improve the scientificity of modeling of cyber attack-defense.

As the most used bounded rationality game model, the replicator dynamics equilibrium is limited to pure strategy [59], which is a particular case of a mixed approach. The others and ours consider the more general mixed policy. [62] [63] [64] [59] describe the strategy updating rules using the dynamic replicator mechanism of biological evolution, which is still limited to pure strategy selection. Moreover, the depicted player rationality is very low, which deviates from the characteristics of fast learning and cyber warfare improvement. They are inherently deterministic evolutionary behaviors without considering the stochastic disturbance in the real network environment. In this paper, we use $LQRD$ equations to describe different evolutionary responses. Meanwhile, we use the flexible parameter ς to quantify the reasonable degree to reflect the randomness and inertia of population social behaviors of realistic attackers and defenders. With the improvement of rationality, we present the best strategy formation and simulate the approach process from bounded rational replicator dynamic equilibrium to complete the rational Nash equilibrium.

4.4 SUMMARY OF THE CHAPTER:

This paper studies the strategy selection with a maximum payoff in the EDS attack-defense dispute based on the evolutionary bounded rationality game model. Advanced Persistent Threat (APT) becomes more diverse with the complexity of large-scale network information systems, leading the cyber attack-defense situation to change dynamically. How to comprehensively analyze defense costs and benefits, maximize defense revenue, predict the possible attack strategy, select the optimal defense strategy from the candidate strategies and measure the strategy revenue are still assumed be big challenges. Game theory is a useful tool to model the adversarial cyber attack-defense. At present, game modeling of attack-defense with bounded rationality is still in its infancy. There are many limitations, such as player rationality quantification, game structure, strategy type, and equilibrium calculation. To a certain extent, it affects the scientificity and effectiveness of game theory for cybersecurity. For this purpose, we construct a novel evolutionary game model to describe attack-defense using $LQRD$ and expand the strategy set and type of existing game structure. We build the

differential equations of strategy evolution, varying with time for attackers and defenders with customized rational degrees. The strategy evolution tracks are simulated in the real-world attack scenario of CrashOverride to depict the best strategy formation. By analyzing the stable evolutionary equilibrium, we can obtain the optimal defense strategy at different game moments. Our approach is more generalized compared to replicator dynamics and the Nash equilibrium model. Two case studies on Crash Override both show that the proposed method is effective and practical. The performances of attack prediction and defense decision-making are improved significantly for winning cyber attack-defense warfare.

Chapter 5

CYBER DEFENSE REMEDIATION BASED ON SDN-ENABLED DYNAMICAL COUNTERMEASURES SELECTION

This chapter shows, how to balance the positive and negative impacts of the cyber defense remediation selection under certain limits of the quality of service (QoS) and security parameters (Confidentiality, Integrity, and availability (CIA)) [65]. The outlines of this chapter are as follows:

- We present an approach to help select security countermeasures dynamically in an SDN enabled Energy Delivery System (EDS) and achieve a trade-off between providing security and QoS.
- We also present the modeling of security costs based on end-to-end packet delay and throughput. We propose a non-dominated sorting based multi-objective optimization framework which can be implemented within an SDN controller to address the joint problem of optimizing between security and QoS parameters by alleviating time complexity.

5.1 SYSTEMS MODEL

In this section, we present the system model consisting of a system framework and optimization model. Both of them together ensure assurance of QoS while enforcing security countermeasures in the event of a cyber-attack.

5.1.1 FRAMEWORK

The proposed framework illustrated in Figure 26 influenced by the software-defined infrastructure proposed by Song et al. [66]. The difference between the two architectures is the

SDN controller's optimization framework to balance QoS and the deployment of security countermeasures. We assume the EDS consists of one $SCADA$ master (control center) connected with regional $SCADA$ substation slaves via a SDN enabled communication network. We also believe that the communication protocol between master and slaves is $DNP3$, whose ethernet packet size is 1500 bytes, and the shared link capacity is 10 Mbps.

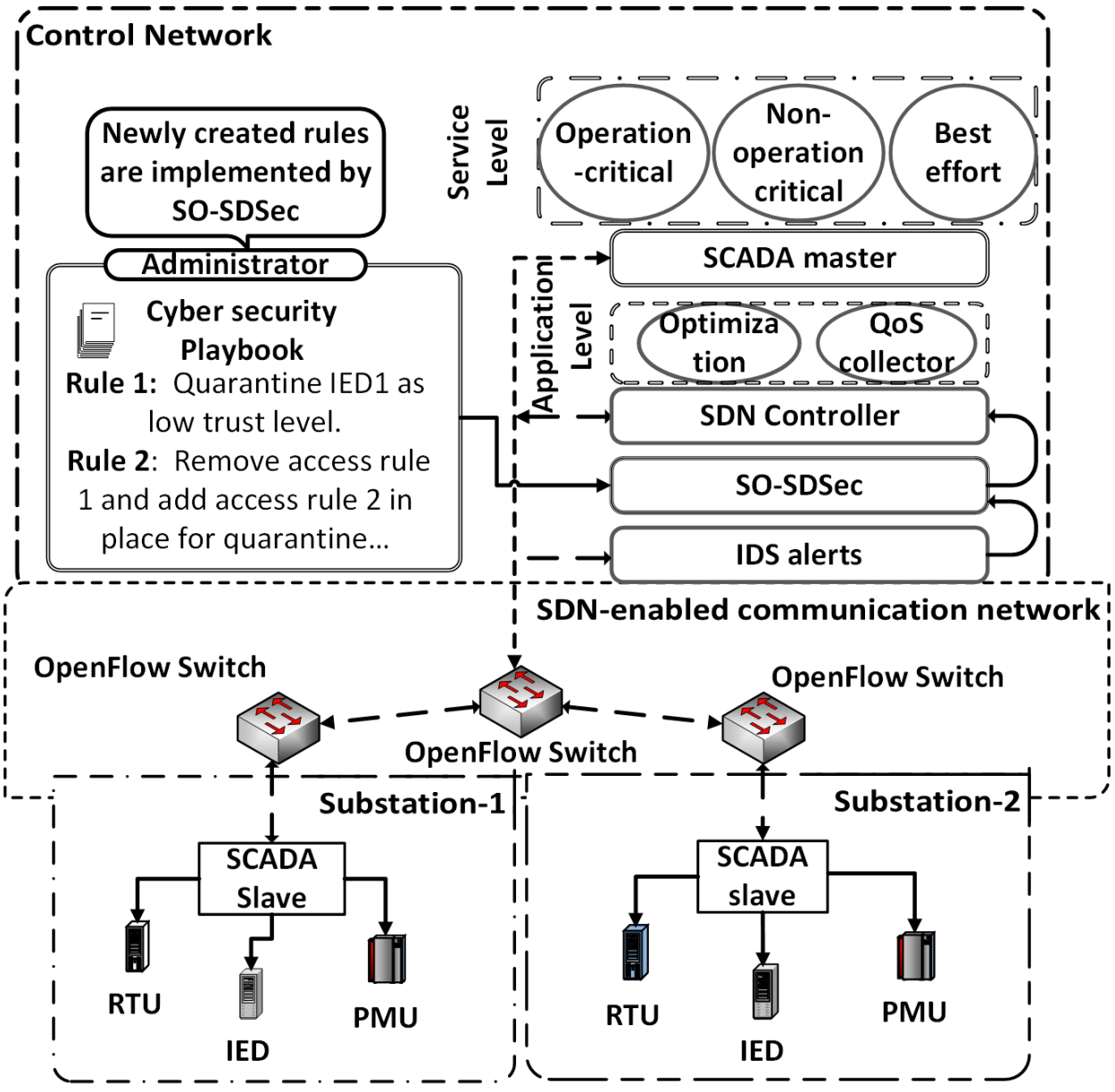


Figure 26. Security and QoS framework for SDN-enabled EDS

- $SCADA$ **master** is responsible for providing grid substation operation-critical service

(configuration and setting updating, etc.), non-operation critical service (power quality monitoring, etc.), and best-effort service (exchanging historical data for mid-term and long-term planning, etc.) [67]. $SCADA$ master collects measurement data and transmits control commands from/to $SCADA$ slaves in the grid via the SDN enabled communication network.

- $SCADA$ **slave** interacts with Intelligent Electronic Devices ($IEDs$) and Remote Terminal Units ($RTUs$), sensors (traditional meters, Phasor Measurement Unit ($PMUs$)) and issues commands to actuators, e.g., circuit breakers, relays, and tap changers.

- **SDN controller** supports applications such as *optimization* and *QoS collector* through *northbound interface* (NBI). The controller communicates with the *openFlow* switch through *southbound interface* (SBI) with *openFlow* protocol.

- SDN**-enabled communication network** is reconfigured by the controller to optimize QoS and security, thereby ensuring resilience support.

- **Playbook** contains operational plans provided by security administrators who should take actions and include alerts from security controls, network monitors, server programs, or any other sensors. The playbook's steps specify service-related operations such as changing the key size of communication service for mutual authentication and messages authentication code (MAC) size for data integrity, updating a security control's service descriptions modifying a security service binding.

- **Service Oriented Software-Defined Security (**$SO-SDSec$**)** converts security functions into abstract security services, with security appliances serving as service providers

5.1.2 OPTIMIZATION MODEL

The objective of the proposed optimization model is to ensure security services while guaranteeing QoS. Figure 26 illustrates the optimization module that can communicate with $SCADA$ master and other SDN applications for coordinated actions. The optimization model consists of four inputs, an optimization module, and one output. The first input

represents IDS alerts, which result in the selection of an effective security countermeasure. The second input is communication services, classified according to its type (operation-critical, non-operation-critical, and best-effort services). The third input is network QoS, representing the network performance assessment, such as throughput, delay, etc. The fourth input is security service settings including the parameters of each security service (the message authentication code (MAC) length, the encryption key in our case). For attack alerts reported by IDS, the playbook rule instructs that, for a particular service attacked, the SO-SDSec must modify the device security service requirements by changing MAC length or Key length. After the rule is applied, the $SO - SDSec$ orchestrator starts monitoring the IDS alerts and automatically requests the security parameters settings to optimize the module through the SDN controller. The optimization module executes the Genetic Algorithm (GA) for each class of service of $SCADA$ control center and assesses the resource availability in terms of network performance parameters. If they are sufficient to implement the security service settings with the highest security level, they will directly send to the output. Otherwise, the optimization module tradesoff between the security level and QoS and calculates the optimal result of security service settings, QoS requirements, and network performance. Then the SDN controller applies the security parameters through the $SCADA$ master for that particular security countermeasure.

5.2 SECURITY RISK LEVELS AND IMPACT ON QOS

Security risk and QoS are two opposite parameters in a communication system. If you want to increase one part, you have to sacrifice in other regions. Thus, proper optimization needs to occur between these two parameters during run-time, which is quite impossible to do in traditional IP enabled $SCADA$ communication networks but an SDN enabled system can easily balance between them. In the following subsections, we discuss the security risk levels and the impact of those security metrics on network QoS.

5.2.1 SECURITY RISK LEVELS

To provide secure data communication in SDN-enabled EDS, we use the combination of three security risk levels (end-to-end authentication level, integrity level, and confidentiality

level). The end-to-end integrity level is described by a Hashed Message Authentication Code ($HMAC$) function based on a shared key between the $SCADA$ master server and regional $RTUs$. The integrity level ensures no one in between the $SCADA$ master and RTU can temper the transmitted messages. On the other hand, the confidentiality level provided by the shared key length between these two entities ensures that transmitted information must disclose to parties for which it's intended. Thus, key management plays a vital role in maintaining $SCADA$ in communication security levels. As per the standard of $DNP3$ protocol [68], the $SCADA$ master only has the authority to assign keys to the $SCADA$ master and the outstation ($SCADA$ slave, $RTUs$, $IEDs$, and $PMUs$, etc.). There are three types of communication in the $SCADA$ system: unicast, multicast, and broadcast. In this work, we only consider unicast, also known as point-to-point communication. The unicast communication means here to communicate between a $SCADA$ master and an RTU or IED, a $SCADA$ slave and an RTU or IED, and an RTU or IED to another RTU or IED. The $SCADA$ master usually generates two types of keys known as Update Key and Session Key for all $SCADA$ communication network devices. By default, Update Keys are pre-shared to the master and outstation and must change by symmetric cryptography or asymmetric (public key) cryptography. Such a mechanism must ensure that the Update Key is kept secret and cannot be obtained by eavesdropping in transit. Usually, Update Key changes in the month to year intervals. On the other hand, the master initializes the Session Keys immediately after communication is established and regularly changes the Session Keys. This practice of periodically changing the Session Keys protects them from being compromised by analyzing the communications link. This Session Key maintains the message authenticity by frequently changing the session key, integrity by applying encryption algorithms (SHA-1-HMAC, SHA-256- HMAC, AES-GMAC) to create $HMAC$ and confidentiality by keeping key length large enough (at least 128 bits) [68]. Thus, security service settings of each security risk level vary according to the SDN-enabled network performance parameters, the class of network service from the $SCADA$ master to RTU, and the number of resources available at a given time. Maximizing the security level implies maximizing the network services. For each security service setting, we evaluate the following security risk levels:

- **Security risk at authentication (A) level :** The authentication risk level depends

on the session key update rate of r_{sk}, which is the arrival rate of authentication requests. During every authentication procedure, devices frequently share session keys in a short period. The higher the frequency of session key update rate, the stronger the security of the service. When the session key update rate increases, the authentication risk level also increases. The authentication risk level is formulated as [68]:

$$AU_l = c_1 \frac{r_{sk}}{r_{sk} + c_2} \tag{27}$$

where both c_1 and c_2 are constants, and can be determined by the range of r_{sk} and authentication risk level settings. In our case, the authentication risk level scaled from 1 to 4. According to the $DNP3$ standard, the session key rate varies from 0.067 to 1 per minute [68]. We can observe that AU_l grows steadily along with r_{sk}. When r_{sk} approaches the maximum values, AU_l slowly approaches the maximum value that represents the highest security level of authentication.

- **Security risk at integrity (I) level :** Data integrity is ensured by checksums generated using cryptographic hash functions with strong collision resistance property. The probability of generating the same hash code for two different messages is higher in lower hash values. The maximum integrity level is defined as [69]:

$$I_l = (2^{\frac{MAC}{K_{min}}} - 1) \frac{c_3}{2^{\frac{MAC}{K_{min}}} + c_4} \tag{28}$$

where MAC represents the length of the checksum digest from hash function, $\frac{c_3}{2^{\frac{MAC}{K_{min}}} + c_4}$ is the impact factor. Both c_3 and c_4 are determined from the range of MAC (16 to 512 bits) and from the range of integrity risk level (1 to 4). Eq. 28 can assure that the security level reaches the maximum four when the checksum approaches infinity. The K_{min} indicates the minimum key length is 128 bits.

- **Security risk at Confidentiality (C) level :** Confidentiality is determined by the

length of the key and the encryption algorithm. The confidentiality level of security is defined as [69]:

$$C_l = \frac{1}{2}[(2^{\frac{K_L}{K_{min}}} - 1)\frac{8 - c_5}{2^{\frac{K_L}{K_{min}}} + c_6}] \tag{29}$$

where, the term $\frac{8-c_5}{2^{\frac{K_L}{K_{min}}}+c_6} + c_5$ indicates the impact factor, K_L represents the length of the key. The value of the c_5 is determined from the specific algorithm and c_6 is decided by the range of K_L (128 to 512 bits) and range of confidentiality risk level (1 to 4) in practical applications.

- **Total security risk level (SL) :** The security level SL can be defined as:

$$SL = w_1 AU_l + w_2 I_l + w_3 C_l \tag{30}$$

where the weights of the security features denoted by w_1, w_2, and w_3 are configurable by the security administrator.

5.2.2 THE IMPACT OF SECURITY ON QOS METRICS

Authentication, data integrity, and confidentiality all bring extra overhead on packet delay, throughput, and packet discard probability. In this section, we present the QoS metrics for EDS.

- **End to End packet delay :** Let's consider d_E, and N represents the total delay and forward devices between source and destination, respectively. The end to end delay is defined as [70]

$$d_E = N(d_{proc} + d_{trans} + d_{prop} + d_{queue}) + d_{proco} \tag{31}$$

In Eq. 31, the terms d_{queue}, d_{prop}, d_{proc} and d_{trans} refer to the queuing, propagation, processing and transmission delay respectively. Queuing delay of a packet is the waiting time in the output buffers of the forwarding devices to be forwarded. Propagation delay is the time that a transmitted packet needs to travel from one end of a link ($SCADA$

master) to the other end (regional RTU). Processing delay of a packet includes time to look up the routing table and to move the packet over the switch fabric, and lastly, the transmission delay is the time it takes to transmit a packet on a link.

If the network is not congested, ($d_{queue} \simeq 0$) and the distance between the source node and destination node is very small ($d_{prop} \simeq 0$). The processing delay, d_{proc}, is often negligible; however, it strongly influences a forwarding device's maximum throughput, which is the maximum rate at which a router can forward packets [70]. In the presence of an uncongested network, Eq. 31 reduces to:

$$d_E = N \times d_{tran} + d_{proco} \tag{32}$$

where d_{proco} indicates the processing overhead because of authentication, integrity, confidentiality, and firewalls rules checking and $d_{tran} = L/R$, where L=packet size (1500 bytes for DNP3 ethernet packet), R=transmission rate out of each forwarding device (bits/sec). The d_{proco} can be defined as:

$d_{proco} = 2 \times d_{AU} + 2 \times d_I + 2 \times d_C$

The terms d_{AU}, d_I and d_C refer to authentication, data integrity and data confidentiality delay respectively.

Authentication delay (d_{AU}) : The authentication delay is proportional to authentication rate r_{au} and can be defined as [68]:

$$d_{AU} = c_7 r_{au} + c_8 \tag{33}$$

where, c_7 and c_8 are constants that are determined by concrete network status.

Data integrity delay (d_I) : The data integrity delay d_I increases linearly with the check sum length and can be defined as [69]:

$$d_I = c_9 MAC + c_{10} \tag{34}$$

where, MAC denotes the length of the check sum, c_9 and c_{10} are determined by different computers and integrity algorithms.

Data confidentiality delay (d_C) : Confidentiality delay d_C is proportional to the encryption key length and can be defined as [69]:

$$d_C = c_{11}K_L + c_{12} \tag{35}$$

where K_L represents the length of key used and c_{11}, c_{12} are determined by different environments. Obviously, the longer the key length, the more the overhead.

Total end to end packet delay (d_E) : Based on the previous analysis, the total end to end delay includes transmission delay, authentication delay, data integrity delay, confidentiality delay. Therefore, the total end to end delay looks like:

$$d_E = (N \times d_{tran}) + (2 \times d_{AU}) + (2 \times d_I) + (2 \times d_C) \tag{36}$$

The constraints of end to end delay for individual network services from the $SCADA$ master to RTU refers to the service availability timing for the respective service.

- **Throughput :** We define throughput as [71]

$$Thr = efficiency \times bitrate = \frac{1500}{1538 + MAC} \times bitrate \tag{37}$$

since the ethernet packet size is maximum 1500 octet payload + 8 octet preamble + 14 octet header + 4 octet trailer + minimum inter-packet gap corresponding to 12 octets = 1538 octets. The maximum efficiency is $\frac{1500}{1538} = 97.53$ % and the physical layer net bit rate depends on the ethernet physical layer standard, and may be 10 Mbit/s, 100 Mbit/s, 1 Gbit/s or 10 Gbit/s.

5.3 OPTIMAL SECURITY COUNTERMEASURE SELECTION PROBLEM FORMULATION

EDS has several network services that can be classified based on the specifications for delay, integrity, and confidentiality. This paper focuses on three types of services: operation-critical service, non-critical services, and best-effort services. An example of an operation-critical service is transmitting a control signal from $SCADA$ to a controlled device. It has a strong delay constraint, and data confidentiality can be ignored. An example of non-critical service is customers' metering data. It requires high throughput and robust data integrity but allows a relatively large delay. The problem of achieving the trade-off among security risk, delay, and throughput can be cast as a multi-objective optimization problem as follows:

$$F(x) = (f_1(x), f_2(x), ..., f_k(x)) \tag{38}$$

where, $f_1, ..., f_k$ are the k objective functions to optimize and $x = (x_1, x_2, ..., x_n)$ is a vector of n decision variables. The goal is to find the vector x that optimizes the k objective functions. In our case, we formulate three objectives:

- **Maximize the security level:** $SL(x)$
- **Minimize the end to end delay:** $d_E(x)$
- **Maximize the Throughput:** $Thr(x)$

subject to,

$d_E \leq 100ms$

$Thr \geq 97\%$

$x \in 16, 32, 64, 128, 256, 512$

where, $x = (MAC, K_L)$ is the vector of the decision variables, which represents the security services settings. By varying this vector, we will determine the optimum or even near-optimum solution of the objective functions.

There is usually no single optimum solution concerning all objectives and constraints in the case of a multi-objective optimization problem. It has a set of optimal or near-optimal solutions known as Pareto optimal solutions or Pareto front. A Pareto front is a set of points

in parameter space (the space of decision variables) with non-inferior fitness function values. In other words, for each point on the Pareto front, you can improve one fitness function only by degrading another [72].

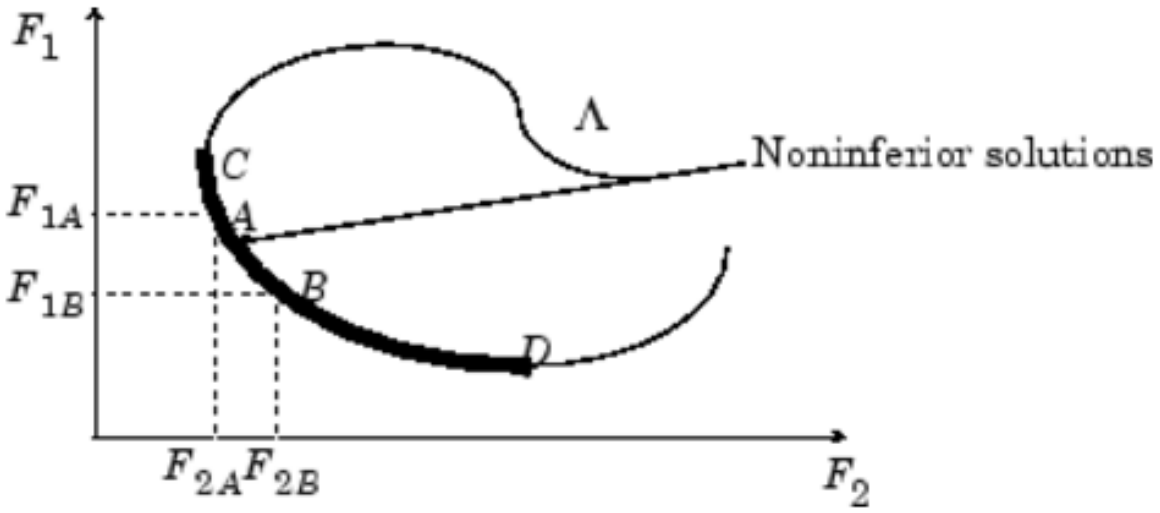


Figure 27. Pareto front [1]

In the Figure 27, A and B are clearly non-inferior solution points because an improvement in one objective, F_1, requires a degradation in the other objective, $F_2, i.e., F_{1B} < F_{1A}, F_{2B} > F_{2A}$.

Since any point in $\wedge$ that is inferior represents a point in which improvement can attain all the objectives, it is clear that such a point is of no value. Multi-objective optimization is, therefore, concerned with the generation and selection of non-inferior solution points. Non-inferior solutions are also called Pareto optima. A general goal in multi-objective optimization is to construct the Pareto optima.

In our problem context, this translates into simultaneously optimizing the objectives: maximize security level, minimize end to end delay and maximize throughput and constraints: an end to end delay $\leq 100ms$ for operation-critical services and throughput ≥ 97 % for non-operation-critical services [67] [73]. In search of a Pareto optimal front, researchers propose many multi-objective optimization methods. In the next section, we present approaches to solve the multi-objective optimization method and our rationale to select the Genetic Algorithm (GA).

5.3.1 MULTI-OBJECTIVE OPTIMIZATION

In the traditional multi-objective optimization approach, the objectives aggregate together to form a single (scalar) fitness function. The classical techniques such as multiple objective linear programming ($MOLP$), multiple attribute utility theory ($MAUT$), random search, simulated annealing, etc. [74] can solve this scalar function. The optimization of the single objective may guarantee a Pareto-optimal solution but results in a single-point solution. In real-world situations, decision-makers often need to evaluate several alternatives during decision making. Moreover, the techniques mentioned above are not effective if some of the objectives are noisy or have discontinuous variable space. Some of these techniques are also expensive as they require knowledge of the individual optimum before vector optimization. Another drawback is the sensitivity towards weights or demand levels [75]. The decision-maker must have a thorough knowledge of the priority of each objective. The solutions obtained largely depend on the underlying weight vector or demand level. Thus, different weight vectors need to be used for different situations, and the optimization process needs to repeat several times. A more effective technique would be one that can find multiple Pareto-optimal solutions simultaneously so that decision-makers may choose the most appropriate solution for a given optimization scenario. The knowledge of many Pareto-optimal solutions is also useful for later use, mainly when the given design has changed and an updated solution is required for implementation. Since GA deals with a population of several points instead of one end, multiple Pareto optimal solutions can capture the people in a single run. Elitism keeps track of reasonable solutions already encountered during optimization to increase the performance significantly of GA. Therefore, using the elitist method is attractive to reduce the delay and increase the SL by the $SCADA$ communication network's optimization process. Furthermore, to choose the best optimization method, we need to search for the optimal security setting solution. We also need to ensure a fair distribution on the Pareto front to discover security setting solutions while guaranteeing the desired trade-offs among the three objectives.

A comparative study of elitist multi-objective optimization methods allows us to evaluate each one's performance in [74]. Results show that the Elitist Non-dominated Sorting Genetic

Algorithm ($NSGA-II$) (A Fast and Elitist multi-objective GA) is the best-suited method to maintain a better spread of solutions and converges better in the obtained non-dominated front. Therefore, we choose the fastest Genetic Algorithm ($NSGA-II$) to optimize our multi-objective problem, whose time-complexity is $O(MN^2)$ where M is the number of objective functions and N is the population size.

In this work, we implement the NSGA-II in $MATLAB$ using the *gamultiobj* function. NSGA-II maintains the diversity of the population for convergence to an optimal Pareto front. Diversity can be maintained by controlling the elite members of the population as the algorithm progresses. Two options, first non-domination sorting ($ParetoFraction$) and crowded distance estimation procedure ($DistanceFcn$), control the elitism. $ParetoFraction$ limits the number of individuals on the Pareto front (elite members). The $DistanceFcn$ helps maintain diversity on a front by favoring individuals who are relatively far away. The algorithm stops if the spread, a measure of the Pareto front's movement, is small [1]. When NSGA-II is adopted to solve our multi-objective problem, an individual will symbolize a possible solution; therefore, the security settings combine the two security services. Thus, the vector of variables $(MAC; K_L)$ can consider as an individual. The $NSGA-II$ implementation illustrated in Figure 28 is applied to get optimal security settings.

The step-by-step procedure shows that the NSGA-II algorithm is simple and straightforward [76]. At first, an initial population P_t of S possible solutions of the security services settings $(x_1, x_2...., x_s)$ is created randomly, where an individual $x_i = (MAC; K_L)$ represents the combination of the two security services parameters. We assign each x_i with the three objective functions, security services level, throughput, and end-to-end delay. The usual binary tournament selection, recombination, and mutation operators are used to create a new population of possible security services settings called child population Q_t of size S. Thereafter, a combined population $R_t = P_t \cup Q_t$ is formed. The population R_t is of size $2S$. Then, the population R_t is sorted according to non-domination. Since all previous and current population members are included in R_t, elitism ensures. Then the total population R_t is sorted according to non-domination and non-dominated fronts $F_1, F_2, ..., F_l$ are obtained. Now, solutions belonging to the best non-dominated set F_1 are the best solutions in the combined population and must emphasize more than any other solution in the combined

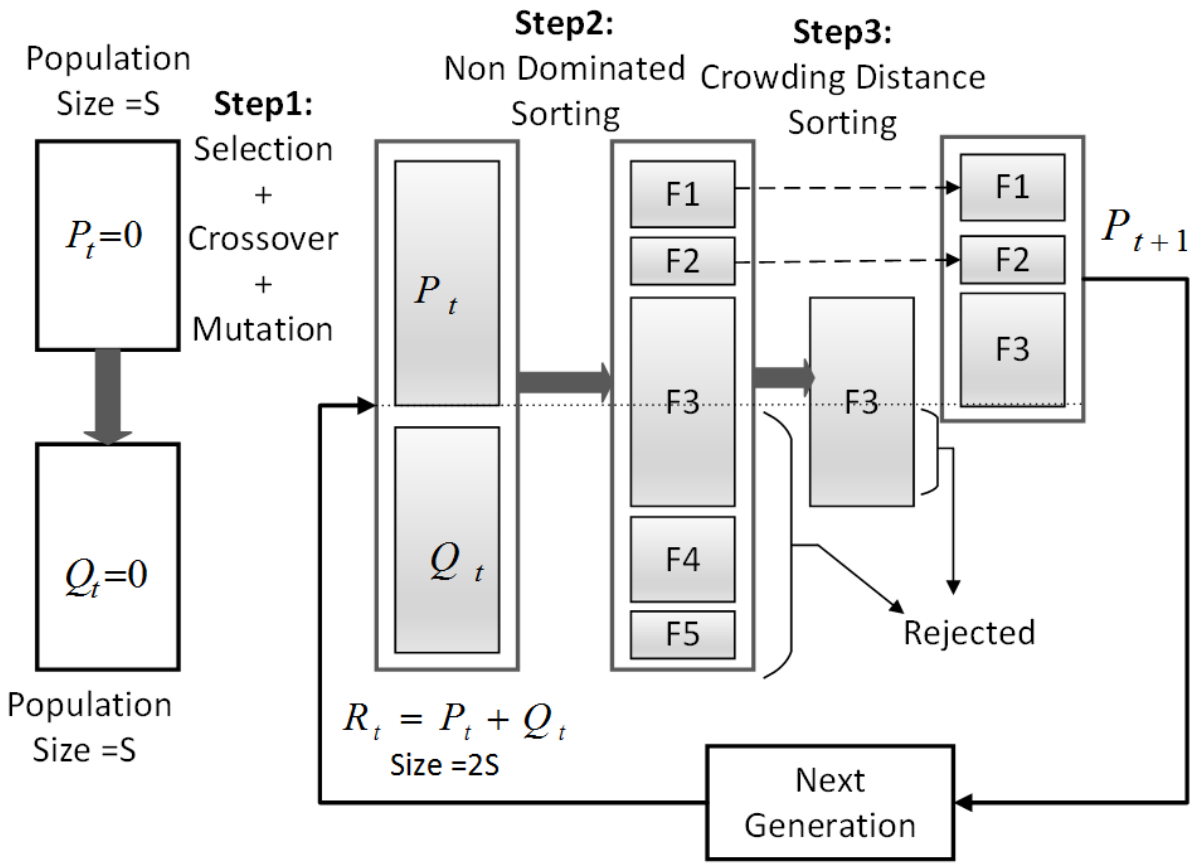


Figure 28. NSGA-II procedure

population. If the size of F_1 is smaller than S, we definitely choose all members of the set F_1 for the new population P_{t+1}. The remaining members of the population P_{t+1} can choose from subsequent non-dominated fronts in the order of their ranking. Thus, solutions from the set F_2 are chosen next, followed by solutions from the set F_3, and so on. This procedure continues until no more sets can accommodate. Say that the set F_l is the last non-dominated set beyond which no other set can accommodate. In general, the count of solutions in all sets from F_1 to F_l would be larger than the population size. To choose exactly population members S, we sort the last front F_l using the crowded distance operator in descending order and choose the best solutions needed to fill all population slots. The new population P_{t+1} of size S now uses for selection, crossover, and mutation to create a new population Q_{t+1} of size S. This procedure will apply until finding the optimal solution and then optimal security configuration.

5.4 SIMULATION RESULTS

We developed the simulation environment in MATLAB 2016a on a Windows 7 Intel(R) Core (TM) i7 -6820 HQ CPU of 2.67 GHz with parameters of NSGA-II for all simulation as follows:

Initial population size : 50

Maximum generation : 150

String length in binary code (n) : 32

Probability of cross-over : 0.8

Probability of mutation : $\frac{1}{n}$

The assigned values to initial population size, maximum generation, and cross over rate ensure diversity and less processing time to get the result. If we select the initial population and maximum generation are high, it gives more accurate diversified solutions to converge, but time complexity increases. In EDS operation, time is also a critical factor for keeping a value to those two parameters. It maintains high diversity and low processing time to get a converged result. The amount to the crossover rate also ensures a better variety to converge. After that, the program is allowed to iterate over several generations, and the final optimized security settings values of the non-dominated solutions resulting from this run have been noted.

Good performance of any type of service running between S$CADA$ and the regional substation depends on security level, delay time, and throughput settings. The trade-off among those objective functions is a multi-objective problem. The parameters for the simulation of all services are given in detail in Tables 15-17.

Table 15. Range of basic parameters for security level

Name	MAC(bits)	K_L(bits)	$r_{sk}(/min)$	$d_{trans}(ms)$
Value	[16,512]	[128,512]	[0.067,1.0]	12

Table 15 contains the range of basic parameters for security. The checksum MAC varies from 16 bits to 512 bits. The key length K_L also ranges from 128 bits to 512 bits. The session key refresh rate r_{sk} changes continuously from 0.067 to 1.0 per minute [68]. The end to end transmission time is 12 ms for the context of 1500 bytes packet size, 10 Mbps end to end shared link, and ten intermediate forwarding devices. Based on the range of Table 15, the values of parameters in Table 16 reflect the normal requirement from Eqs. 27 to 29 and the setting of three security levels. The parameters of Table 17 are calculated based on the ranges of delay factor imposed from Eqs. 33-35 to packets end to end delay in an SDN-enabled $SCADA$ communication network.

Table 16. Parameters for evaluating security level

Name	c_1	c_2	c_3	c_4	c_5	c_6
Value	5.29	0.22	4.07	0.72	8	1.2

Table 17. Parameters for evaluating delay and throughput

Name	c_7	c_8	c_9	c_{10}	c_{11}	c_{12}
Value	0.6	0.21	0.2	8.1	0.2	65

Operation-critical services:

For the operation-critical services (i.e. control command from the SCADA master to RTU/IED), the authentication function operates in a short time and mostly needs the data integrity and smaller part of data confidentiality. Hence, the weights are: $w_1 = 0.3, w_2 = 0.6, w_3 = 0.1$ and Eqs. 30 and 36 look like,

$$SL = 0.3AU_l + 0.6I_l + 0.1C_l \tag{39}$$

$$d_E = (N \times \frac{L}{R}) + 2 \times 0.3 \times d_{AU} + 2 \times 0.6 \times d_I + 2 \times 0.1 \times d_C \quad (40)$$

Figure 29 shows that the trend of the delay increases with the increase of security level up to 100 ms for critical operation services. The session key update rate, number of intermediate nodes, the communication link bandwidth (BW), the message authentication code size (MAC), and key length determine end to end delay. In our case, the session key update rate, number of intermediate nodes, and BW keep constant, but MAC size varies from 16 bits to 512 bits, and key length varies from 128 bits to 512 bits. All 15 points of the Pareto front are optimized. Among these points, one point may give better SL, and another point may ensure a better end to end delay. Like the leftmost point whose co-ordinates $(SL, d_E) = (1.69, 72.3ms)$ when $(MAC, K_L) = (16, 128)$ and rightmost point is $(SL, d_E) = (1.75, 73.7ms)$ when $(MAC, K_L = (32, 128)$ [67]. The leftmost point ensures better delay constraint at a smaller MAC length than the rightmost point, where better SL constraint ensures at the cost of a bigger MAC length. Both points are important for SDN-aware EDS in different situations as per the need for respective operation-critical service. Still, with the slight increment of SL (from 1.69 to 1.75) from the left point to the right point, the system has to increase MAC length from 16 bits to 32 bits, which increases end to end delay from 72.3 ms to 73.7 ms.

According to the time complexity of $NSGA-II$, which is big O mentioned earlier, 7500 functions evaluated in every simulation because our problem domain consists of 3 objective parts and an initial population of 50. The calculated time requirements for every function evaluation in our simulation environment is $10^{-4}s$. The total time overhead added due to this optimization solver to the SDN controller is 0.75 seconds, but this time delay is incurred one time and is only applicable when the attacker attacks the $SCADA$ communication link. The optimization module identifies the optimized security setting parameters to mitigate that attack. After applying new security settings, there will be no additional latency due to the optimization solver until another threat pops up to the $SCADA$ communication networks. In the absence of the optimization solver, the time taken to identify the problem

and apply the appropriate countermeasure will take considerable time. It may not satisfy the timeliness of delivering operational critical messages from the $SCADA$ master to substations. Though the countermeasure applies, the end-to-end delay may not meet, resulting in the substation operating in a pulsating condition and thereby responsible for an unstable grid. By adopting our optimization module, we can implement optimized security settings within $0.75s$, ensuring that the operation critical messages are exchanged in a timely fashion and providing operational resilience of communication network between the $SCADA$ master and substations RTU.

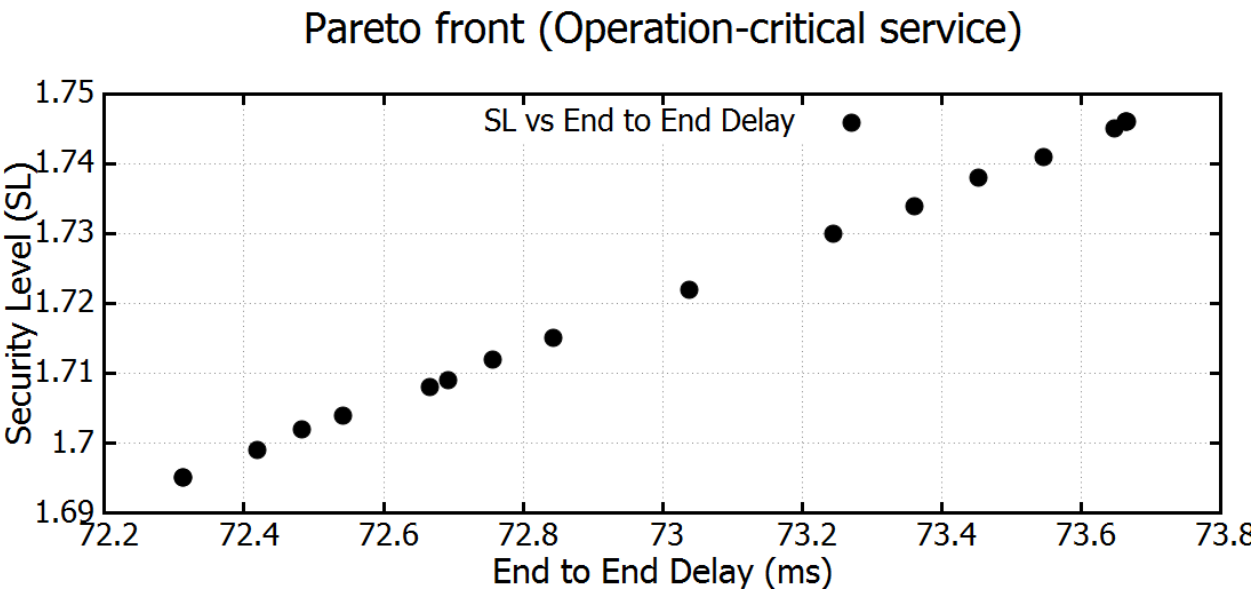


Figure 29. Pareto front to maintain delay $\leq 100ms$ when N=10]

Non operation-critical services

For a defined network condition, the throughput has an inverse relationship with the delay. We can say that the network delay multiplying the throughput is a constant in a known network. Non-operation-critical services deem authenticity, integrity, and confidentiality at equal share [67]. Hence, the weights are: $w_1 = 0.33, w_2 = 0.33, w_3 = 0.33$ and Eqs. 30 and 36 look like,

$$SL = 0.33AU_l + 0.33I_l + 0.33C_l \tag{41}$$

$$d_E = (N \times \frac{L}{R}) + 2 \times 0.33 \times d_{AU} + 2 \times 0.33 \times d_I + 2 \times 0.33 \times d_C \tag{42}$$

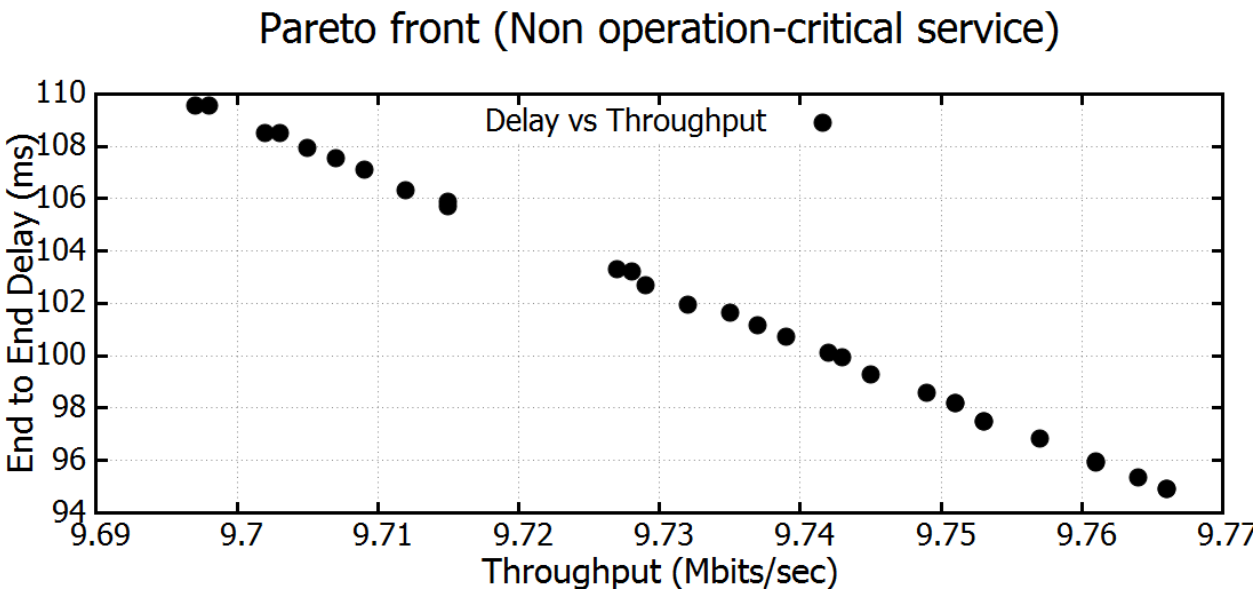


Figure 30. Pareto front with constraint Thr $\geq$ 97 % when N=14

Figure 30 and Figure 31 together indicates that when the security level increases, the throughput decreases, and also end to end delay increases. There are 29 optimized points in each Pareto front. If we consider two points from Figure 31, the leftmost point is $(SL, Thr) = (2.1, 97.10\%)$ when $(MAC, K_L = (128, 128)$ and rightmost point is $(SL, Thr) = (1.7, 97.76\%)$ when $(MAC, K_L = (64, 128)$. Both points are important in SDN-aware EDS in different situations and costs. The leftmost point is important for those non-operation-critical services (power quality monitoring, customer metering data) where data throughput and SL are important, but delays can be considered up to a certain level [67]. In that case, there are extra processing cost with improved SL because of selecting bigger MAC.

On the other hand, the rightmost point applies to those non-operation-critical services

where SL can be considered up to a certain level. Still, data throughput is more important than SL because massive data has to be collected quickly. Collecting remote customers' metering data can be treated as this type of service where MAC are smaller, indicating less processing time.

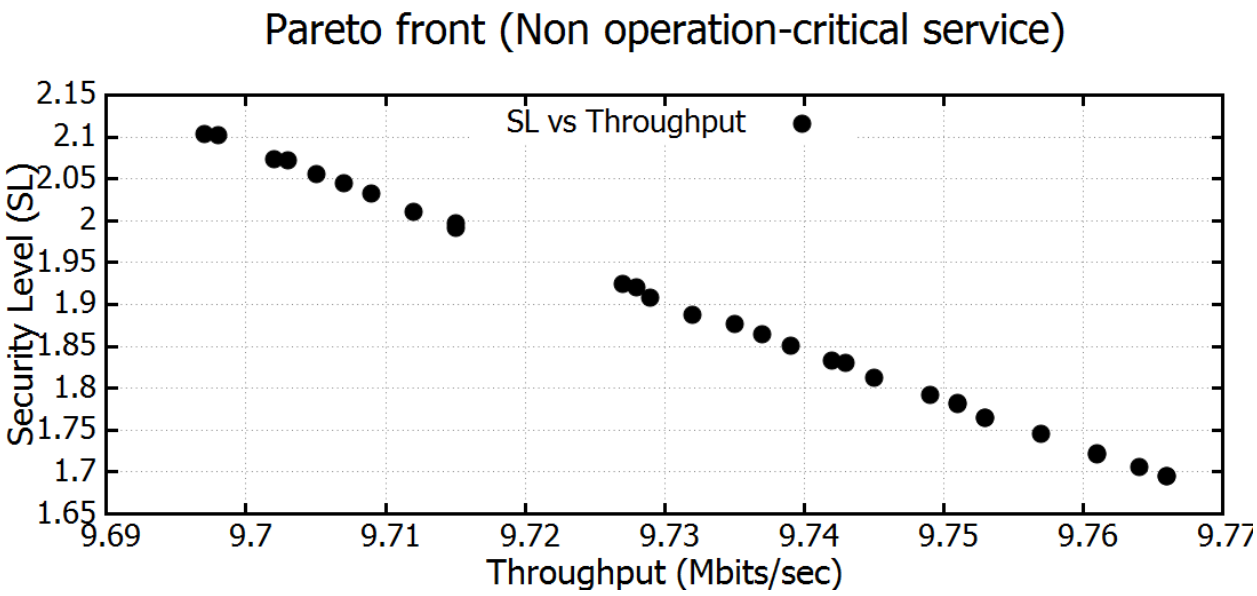


Figure 31. Pareto front with constraint Thr $\geq$ 97 % when N=14

Best effort services

Best effort service has no specific requirements in terms of security features and network quality of service. In that case, the network specialist can run the optimization solver and set security parameters, and QoS parameters depend on the service requirements. Figure 32 shows the Pareto front of optimization output when no constraint specifies as input. In that case, the security level increases with an increasing trend of end to end delay. The leftmost point, where $(SL, d_E) = (1.75, 75ms)$ $(MAC, K_L) = (16, 128)$ is perfectly applicable to collecting data for automation system engineering [73] and troubleshooting in SDN-aware EDS. Those services do not have any hard and fast constraints, but the delay factor is slightly important. On the other hand, the rightmost point $(SL, d_E) = (4, 325ms)$ when $(MAC, K_L) = (512, 256)$ applies to exchanging historical data for midterm and long term planning where data integrity is more important than an end to end delay.

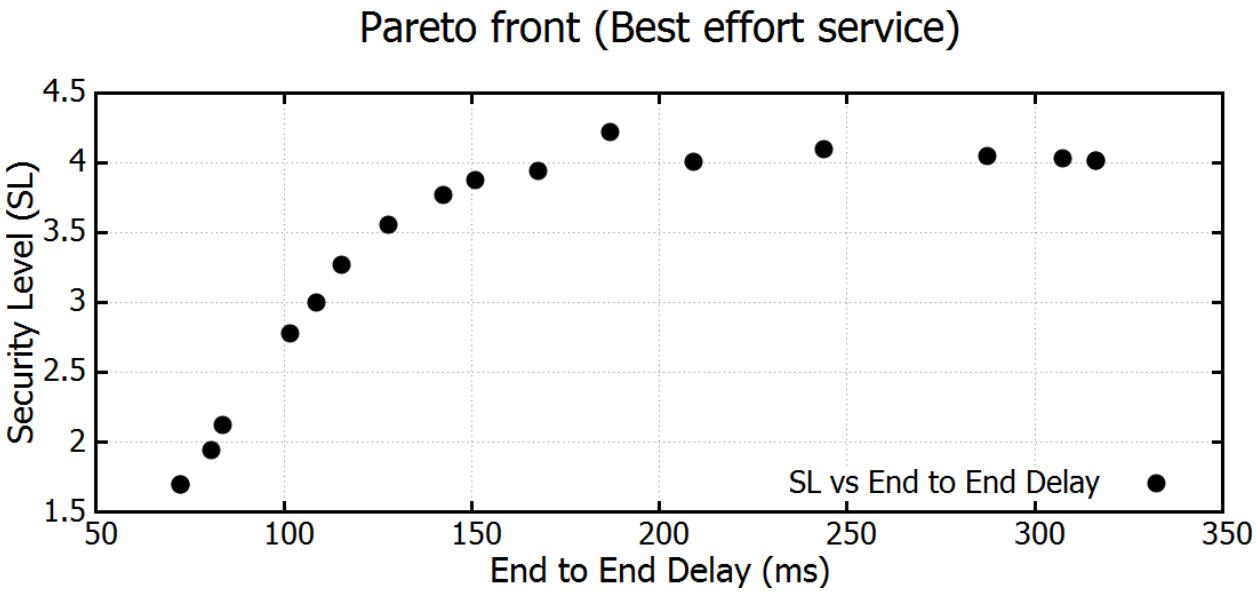


Figure 32. Pareto front with no constraint when N=14

Performance evaluation of the optimized security settings

Different performance metrics propose [77], such as error ratio, to evaluate the performance of the obtained security setting solutions. The error ratio is used to evaluate the percentage of the convergence to the known Pareto-optimal front. It is defined as: $E = \frac{sum_{i=1}^{n} e_i}{n}$ where n is the total number of security setting solutions. $e_i = 0$, if a solution is a member of the Pareto optimal front, otherwise 1. As in Figure 32 there is one point so far that is a little bit out of Pareto front, so the error ratio $E = \frac{1}{29} = 0.034$. That means 96.6 % of the security setting solutions are close to the Pareto front.

5.5 SUMMARY OF THE CHAPTER

This chapter presented an SDN-enabled EDS architecture that provides the ability to enforce security countermeasures to reduce the risk of cyber attack and ensure QoS. We proposed a genetic algorithm-based multi-objective optimization approach to select the optimal security countermeasure, which balances the reduction of security risk and maintains QoS, thereby ensuring EDS resilience. Simulation results indicate that the proposed approach can provide resiliency by balancing the trade-off between reducing security risk and

QoS guarantees.

Chapter 6

CONCLUSIONS AND FUTURE RESEARCH

In this final chapter, we summarize this dissertation's contributions, and we also provide future directions.

6.1 CONCLUSIONS

A prioritized cyber defense remediation plan is critical for effective risk management in the Energy Delivery System (EDS). National Institute of Standards and Technology ($NIST$) proposes a framework that includes three layers: tactical risk, mission impact risk, and organizational risk, to manage cyber risk in EDS [2]. In the literature, researchers have considered a security risk and safety risk together at the operational level (tactical risk). Still, they failed to notice the propagation of tactical risk to business/mission process risk and strategic risk level. Without this complete three layers of risk analysis, one may sacrifice mission success or organizational reputation for a false sense of security by a too narrow perspective on the operational problem. In *Chapter 2* of this thesis, we model tactical risk considering the safety and security risk of a node in the EDS infrastructure considering node criticality and model how they propagate to business/mission risk and strategic risk. We also propose an optimal resource allocation scheme of a fixed resource budget according to nodes' criticality at the operational level and then optimize among tactical risk, business/mission risk, and strategic risk. Finally, we empirically validate within an Industrial Control System (ICS) test-bed to assess the performance of the criticality model and resource allocation scheme.

Chapter 3 further emphasizes the prioritized cyber defense remediation for risk management at the operational level. In this chapter, we model the criticality of a node in the EDS infrastructure considering network heterogeneity. We also propose an optimal resource allocation (remediation) scheme of a fixed resource budget according to nodes' criticality

that minimizes the network risk. Finally, we empirically validate within an ICS test-bed to assess the performance of the criticality model and resource allocation scheme.

The remediation schemes of *Chapter 2* and *Chapter 3* are static in nature, but the landscape of cybersecurity has been reformed dramatically by the recently emerging Advanced Persistent Threat (APT). It is uniquely featured by the stealthy, continuous, sophisticated, and well-funded attack process for long-term malicious gain, which renders the current defense mechanisms inapplicable. A novel design of defense strategy, continuously combating APT in a long time-span with imperfect/incomplete information on attacker's actions, is urgently needed. In *Chapter 4*, the stochastic evolutionary game model is utilized to simulate the dynamic adversary of cyber-attack-defense to solve this problem.

However, the evolutionary game model can only provide the selection of optimal controls for cyber defense remediation but can not provide a way to implement those control policies. Finally, in *Chapter 5*, we propose a Software-Defined Networking (SDN) enabled optimization scheme for the dynamical implementation of those optimal security controls. The scheme is an efficient and dynamic optimization model that determines a combination of optimal security services settings and QoS requirements of each class of SCADA communication service using elitist non-dominated sorting genetic algorithm (NSGA-II).

6.2 FUTURE RESEARCH

Cyber threats have increased extensively during the last decade, especially in EDS. Cybercriminals have become more sophisticated. Current security controls are not enough to defend networks from the number of highly skilled cybercriminals. Cybercriminals have learned how to evade the most sophisticated tools, such as Intrusion Detection and Prevention Systems ($IDPS$), and Advanced Persistent Threat (APT) is almost invisible to current tools. To defend against those advanced cyber threats, it is high time to apply Artificial Intelligence (AI) to increase the detection and protection rate of IDPS and the efficiency of cyber defense remediation. Machine Learning (ML) techniques can mine data to detect different attack stages of APT. However, the implementation of AI may bring other risks, and cybersecurity experts need to find a balance between risk and benefits.

BIBLIOGRAPHY

[1] "Mult-objective optimization using genetic algorithm," https://www.mathworks.com/help/gads/what-is-multiobjective-optimization.html, accessed: 2017-07-11.

[2] K. Stouffer, J. Falco, and K. Scarfone, "Guide to industrial control systems (ics) security," *NIST special publication*, vol. 800, no. 82, pp. 16–16, 2011.

[3] H. Abdo, M. Kaouk, J.-M. Flaus, and F. Masse, "A safety/security risk analysis approach of industrial control systems: A cyber bowtie–combining new version of attack tree with bowtie analysis," *Computers & Security*, vol. 72, pp. 175–195, 2018.

[4] S. Kriaa, L. Pietre-Cambacedes, M. Bouissou, and Y. Halgand, "A survey of approaches combining safety and security for industrial control systems," *Reliability engineering & system safety*, vol. 139, pp. 156–178, 2015.

[5] I. N. Fovino, M. Masera, and A. De Cian, "Integrating cyber attacks within fault trees," *Reliability Engineering & System Safety*, vol. 94, no. 9, pp. 1394–1402, 2009.

[6] G. G. Granadillo, A. Motzek, J. Garcia-Alfaro, and H. Debar, "Selection of mitigation actions based on financial and operational impact assessments," in *2016 11th International Conference on Availability, Reliability and Security (ARES)*. IEEE, 2016, pp. 137–146.

[7] A. Motzek and R. Möller, "Context-and bias-free probabilistic mission impact assessment," *computers & security*, vol. 65, pp. 166–186, 2017.

[8] M. A. R. Al Amin, S. Shetty, L. Njilla, D. K. Tosh, and C. Kamhoua, "Online cyber deception system using partially observable monte-carlo planning framework," in *International Conference on Security and Privacy in Communication Systems*. Springer, 2019, pp. 205–223.

[9] C. W. Anderson, J. R. Santos, and Y. Y. Haimes, "A risk-based input–output methodology for measuring the effects of the august 2003 northeast blackout," *Economic Systems Research*, vol. 19, no. 2, pp. 183–204, 2007.

[10] B. E. Strom, J. A. Battaglia, M. S. Kemmerer, W. Kupersanin, D. P. Miller, C. Wampler, S. M. Whitley, and R. D. Wolf, "Finding cyber threats with att&ck-based analytics," Technical Report MTR170202, MITRE, Tech. Rep., 2017.

[11] Q. Li, F. Zhang, and P. Huff, "Automated security patch and vulnerability remediation tool for electric utilities," Apr. 4 2019, uS Patent App. 16/150,042.

[12] H. Okhravi and D. Nicol, "Evaluation of patch management strategies," *International Journal of Computational Intelligence: Theory and Practice*, vol. 3, no. 2, pp. 109–117, 2008.

[13] B. Thornton and C. Li, "The applicability of network management systems in small businesses," in *IEEE Conference Anthology*. IEEE, 2013, pp. 1–7.

[14] H. Booth, D. Rike, and G. Witte, "The national vulnerability database (nvd): Overview," National Institute of Standards and Technology, Tech. Rep., 2013.

[15] S. Frei, M. May, U. Fiedler, and B. Plattner, "Large-scale vulnerability analysis," in *Proceedings of the 2006 SIGCOMM workshop on Large-scale attack defense*. ACM, 2006, pp. 131–138.

[16] M. Shahzad, M. Z. Shafiq, and A. X. Liu, "A large scale exploratory analysis of software vulnerability life cycles," in *2012 34th International Conference on Software Engineering (ICSE)*. IEEE, 2012, pp. 771–781.

[17] F. Li and V. Paxson, "A large-scale empirical study of security patches," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2017, pp. 2201–2215.

[18] Q. Li and F. Zhang, "Security vulnerability and patch management in electric utilities: A data-driven analysis," University of Arkansas, Tech. Rep., 2018.

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

[20] J. Kouns, "Open source vulnerability database," *The Open Source Business Resource*, p. 4, 2008.

[21] C. Strasburg, N. Stakhanova, S. Basu, and J. S. Wong, "A framework for cost sensitive assessment of intrusion response selection," in *2009 33rd Annual IEEE international computer software and applications conference*, vol. 1. IEEE, 2009, pp. 355–360.

[22] W. Kanoun, N. Cuppens-Boulahia, F. Cuppens, and S. Dubus, "Risk-aware framework for activating and deactivating policy-based response," in *2010 Fourth International Conference on Network and System Security*. IEEE, 2010, pp. 207–215.

[23] T. Toth and C. Kruegel, "Evaluating the impact of automated intrusion response mechanisms," in *18th Annual Computer Security Applications Conference, 2002. Proceedings.* IEEE, 2002, pp. 301–310.

[24] I. Balepin, S. Maltsev, J. Rowe, and K. Levitt, "Using specification-based intrusion detection for automated response," in *International Workshop on Recent Advances in Intrusion Detection*. Springer, 2003, pp. 136–154.

[25] N. Kheir, N. Cuppens-Boulahia, F. Cuppens, and H. Debar, "A service dependency model for cost-sensitive intrusion response," in *European Symposium on Research in Computer Security*. Springer, 2010, pp. 626–642.

[26] M. Jahnke, C. Thul, and P. Martini, "Graph based metrics for intrusion response measures in computer networks," in *32nd IEEE Conference on Local Computer Networks (LCN 2007)*. IEEE, 2007, pp. 1035–1042.

[27] K. Hasan, S. Shetty, and S. Ullah, "Artificial intelligence empowered cyber threat detection and protection for power utilities," in *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*. IEEE, 2019, pp. 354–359.

[28] K. Hasan, S. Shetty, J. Sokolowski, and D. K. Tosh, "Security game for cyber physical systems," in *Proceedings of the Communications and Networking Symposium*, 2018, pp. 1–12.

[29] C. T. Do, N. H. Tran, C. Hong, C. A. Kamhoua, K. A. Kwiat, E. Blasch, S. Ren, N. Pissinou, and S. S. Iyengar, "Game theory for cyber security and privacy," *ACM Computing Surveys (CSUR)*, vol. 50, no. 2, pp. 1–37, 2017.

[30] K. Merrick, M. Hardhienata, K. Shafi, and J. Hu, "A survey of game theoretic approaches to modelling decision-making in information warfare scenarios," *Future Internet*, vol. 8, no. 3, p. 34, 2016.

[31] J.-l. Tan, C. Lei, H.-q. Zhang, and Y.-q. Cheng, "Optimal strategy selection approach to moving target defense based on markov robust game," *Computers & Security*, vol. 8, no. 5, pp. 63–76, 2019.

[32] Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "A game-theoretic approach to fake-acknowledgment attack on cyber-physical systems," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 1, pp. 1–11, 2016.

[33] X. Li, C. Zhou, Y.-C. Tian, and Y. Qin, "A dynamic decision-making approach for intrusion response in industrial control systems," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 5, pp. 2544–2554, 2018.

[34] W. M. Czarnecki, G. Gidel, B. Tracey, K. Tuyls, S. Omidshafiei, D. Balduzzi, and M. Jaderberg, "Real world games look like spinning tops," *arXiv preprint arXiv:2004.09468*, 2020.

[35] K. Hasan, S. Shetty, A. Hassanzadeh, and S. Ullah, "Towards optimal cyber defense remediation in cyber physical systems by balancing operational resilience and strategic risk," in *MILCOM 2019-2019 IEEE Military Communications Conference (MILCOM)*. IEEE, 2019, pp. 1–8.

[36] K. Hasan, S. Shetty, S. Ullah, A. Hassanzadeh, and E. Hadar, "Towards optimal cyber defense remediation in energy delivery systems," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–7.

[37] K. Hasan, S. Shetty, A. Hassanzadeh, M. B. Salem *et al.*, "Self-healing cyber resilient framework for software defined networking-enabled energy delivery system," in *2018*

IEEE Conference on Control Technology and Applications (CCTA). IEEE, 2018, pp. 1692–1697.

[38] K. Hasan, S. Shetty, A. Hassanzadeh, M. B. Salem, and J. Chen, "Modeling cost of countermeasures in software defined networking-enabled energy delivery systems," in *2018 IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2018, pp. 1–9.

[39] X. Ou, S. Govindavajhala, and A. W. Appel, "Mulval: A logic-based network security analyzer." in *USENIX security symposium*, vol. 8. Baltimore, MD, 2005, pp. 113–128.

[40] X. Ou and A. Singhal, *Quantitative security risk assessment of enterprise networks*. Springer, 2011.

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

[42] M. Frigault, L. Wang, S. Jajodia, and A. Singhal, "Measuring the overall network security by combining cvss scores based on attack graphs and bayesian networks," in *Network Security Metrics*. Springer, 2017, pp. 1–23.

[43] D. Dolezilek, "Case study of a large transmission and distribution substation automation project," *Schweitzer Engineering Laboratories, Inc., Pullman, WA USA*, 1999.

[44] W. Leontief, *Input-output economics*. Oxford University Press, 1986.

[45] J. R. Santos and Y. Y. Haimes, "Modeling the demand reduction input-output (i-o) inoperability due to terrorism of interconnected infrastructures," *Risk Analysis: An International Journal*, vol. 24, no. 6, pp. 1437–1451, 2004.

[46] Y. Y. Haimes, *Risk modeling, assessment, and management*. John Wiley & Sons, 2015.

[47] A. Hassanzadeh and R. Burkett, "Samiit: Spiral attack model in iiot mapping security alerts to attack life cycle phases," in ics & scada cyber security research." in *5th International Symposium for ICS & SCADA Cyber Security Research 2018*, vol. 5. Hamburg, Germany, 2018, pp. 11–20.

[48] ——, "Samiit: Spiral attack model in iiot mapping security alerts to attack life cycle phases," in *ICS & SCADA Cyber Security Research,2018 5th International Symposium for*. BCS, 2018, pp. 11–20.

[49] T. G. Lewis, *Network science: Theory and applications*. John Wiley & Sons, 2011.

[50] M. Touhiduzzaman, A. Hahn, and A. Srivastava, "Arcades: analysis of risk from cyberattack against defensive strategies for the power grid," *IET Cyber-Physical Systems: Theory & Applications*, vol. 3, no. 3, pp. 119–128, 2018.

[51] S. Wang, "Optimal level and allocation of cybersecurity spending: Model and formula," *Available at SSRN 3010029*, 2017.

[52] Z. Zheng and A. Reddy, "Towards improving data validity of cyber-physical systems through path redundancy," in *Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security*. ACM, 2017, pp. 91–102.

[53] P. Pols and J. van den Berg, "The unified kill chain," *CSA Thesis, Hague*, pp. 1–104, 2017.

[54] O. Alexander, M. Belisle, and J. Steele, "Mitre att&ck® for industrial control systems: Design and philosophy," 2020.

[55] S. P. Anderson, J. K. Goeree, and C. A. Holt, "The logit equilibrium: A perspective on intuitive behavioral anomalies," *Southern Economic Journal*, pp. 21–47, 2002.

[56] T. G. Kurtz, "Solutions of ordinary differential equations as limits of pure jump markov processes," *Journal of applied Probability*, vol. 7, no. 1, pp. 49–58, 1970.

[57] J. Slowik, "Anatomy of an attack: Detecting and defeating crashoverride," *VB2018, October*, 2018.

[58] A. Shameli-Sendi, H. Louafi, W. He, and M. Cheriet, "Dynamic optimal countermeasure selection for intrusion response system," *IEEE Transactions on Dependable and Secure Computing*, vol. 15, no. 5, pp. 755–770, 2016.

[59] H. Hu, Y. Liu, H. Zhang, and R. Pan, "Optimal network defense strategy selection based on incomplete information evolutionary game," *IEEE Access*, vol. 6, pp. 29 806–29 821, 2018.

[60] H. Hu, Y. Liu, C. Chen, H. Zhang, and Y. Liu, "Optimal decision making approach for cyber security defense using evolutionary game," *IEEE Transactions on Network and Service Management*, 2020.

[61] C. Lei, H.-Q. Zhang, L.-M. Wan, L. Liu, and D.-h. Ma, "Incomplete information markov game theoretic approach to strategy generation for moving target defense," *Computer Communications*, vol. 116, pp. 184–199, 2018.

[62] J.-m. Zhu, B. Song, and Q.-f. Huang, "Evolution game model of offense-defense for network security based on system dynamics," *J. Commun.*, vol. 35, no. 1, pp. 54–61, 2014.

[63] A. A. A. Abass, L. Xiao, N. B. Mandayam, and Z. Gajic, "Evolutionary game theoretic analysis of advanced persistent threats against cloud storage," *IEEE Access*, vol. 5, pp. 8482–8491, 2017.

[64] Y. Hayel and Q. Zhu, "Epidemic protection over heterogeneous networks using evolutionary poisson games," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 8, pp. 1786–1800, 2017.

[65] K. Hasan, S. Shetty, A. Hassanzadeh, M. B. Salem, and J. Chen, "Software-defined networking for cyber resilience in industrial internet of things (iiot)," *Modeling and Design of Secure Internet of Things*, pp. 453–477, 2020.

[66] S. Luo and M. B. Salem, "Orchestration of software-defined security services," in *Communications Workshops (ICC), 2016 IEEE International Conference on.* IEEE, 2016, pp. 436–441.

[67] D. Wei, Y. Lu, M. Jafari, P. Skare, and K. Rohde, "An integrated security system of protecting smart grid against cyber attacks," in *Innovative Smart Grid Technologies (ISGT), 2010*. IEEE, 2010, pp. 1–7.

[68] "Ieee std 1815-2012 for electric power systems communications distributed network protocol (dnp3)," pp. 171–265, June, 2012.

[69] A. G. Association *et al.*, "Cryptographic protection of scada communications part 1: Background, policies and test plan," AGA Report, Tech. Rep., March, 2005.

[70] J. Kurose and W. Keith, "K. ross computer networking: a top down approach," 2007.

[71] "Ethernet frame," https://en.wikipedia.org/wiki/Ethernet_frame, accessed: 2017-07-01.

[72] K. Deb, S. Agrawal, A. Pratap, and T. Meyarivan, "A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii," in *International Conference on Parallel Problem Solving From Nature*. Springer, 2000, pp. 849–858.

[73] I. PES, "Ieee standard for scada and automation systems," *vol. IEEE Std C*, vol. 37, 2008.

[74] V. Khare, X. Yao, and K. Deb, "Performance scaling of multi-objective evolutionary algorithms," in *International Conference on Evolutionary Multi-Criterion Optimization*. Springer, 2003, pp. 376–390.

[75] N. Srinivas and K. Deb, "Muiltiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary computation*, vol. 2, no. 3, pp. 221–248, 1994.

[76] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: Nsga-ii," *IEEE transactions on evolutionary computation*, vol. 6, no. 2, pp. 182–197, 2002.

[77] D. A. Van Veldhuizen, "Multiobjective evolutionary algorithms: classifications, analyses, and new innovations," DTIC Document, Tech. Rep., 1999.

VITA

Kamrul Hasan
Department of Computational Modeling & Simulation Engineering
Old Dominion University
Norfolk, VA 23529

Education

- M.S. in Computer Information & Systems Engineering (CISE), Tennessee State University (TSU), Nashville, TN, 2016
- B.S. in Electrical & Electronics Engineering (EEE), Bangladesh University of Engineering (BUET) & Technology, Dhaka, Bangladesh, 2006

Publications

- Hasan, Kamrul, Sachin Shetty, Sharif Ullah, and Amin Hassanzadeh. "Towards Optimal Cyber Defense Remediation in Energy Delivery Systems." In 2019 IEEE Global Communications Conference (Globecom), pp.1-7. IEEE, 2019.
- Hasan, Kamrul, Sachin Shetty, Sharif Ullah, and Amin Hassanzadeh. "Towards Optimal Cyber Defense Remediation in Energy Delivery Systems." In 2019 IEEE Military Communications Conference (Milcom), pp.1-8. IEEE, 2019.
- Hasan, Kamrul, Sachin Shetty, Amin Hassanzadeh, Malek Ben Salem, and Jay Chen. "Modeling Cost of Countermeasures in Software Defined Networking-enabled Energy Delivery Systems." In 2018 IEEE Conference on Communications and Network Security (CNS), pp. 1-9. IEEE, 2018.
- Hasan, Kamrul, Sachin Shetty, Amin Hassanzadeh, and Malek Ben Salem. "Self-Healing Cyber Resilient Framework for Software Defined Networking-Enabled Energy Delivery System." In 2018 IEEE Conference on Control Technology and Applications (CCTA), pp. 1692-1697. IEEE, 2018.
- Hasan, Kamrul, Sachin Shetty, John Sokolowski, and Deepak K. Tosh. "Security game for cyber physical systems." In Proceedings of the Communications and Networking Symposium, p. 12. Society for Computer Simulation International, 2018.

Typeset using LaTeX.