Fall 2019

A Multi-Agent Systems Approach for Analysis of Stepping Stone Attacks

Marco Antonio Gamarra
Old Dominion University, gamarras.marco@hotmail.com

A MULTI-AGENT SYSTEMS APPROACH FOR ANALYSIS OF STEPPING STONE ATTACKS

by

Marco Antonio Gamarra
B.S. Physics and Mathematics 1997, Universidad Nacional San Antonio Abad Del Cusco, Perú
M.S. Mathematics 2011, Florida International University, USA

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

DOCTOR OF PHILOSOPHY

ELECTRICAL & COMPUTER ENGINEERING

OLD DOMINION UNIVERSITY
December 2019

Approved by:

Sachin Shetty (Director)

Chunsheng Xin (Member)

Dimitrie C. Popescu (Member)

Rao Chaganty (Member)

Laurent Njilla (Member)

ABSTRACT

A MULTI-AGENT SYSTEMS APPROACH FOR ANALYSIS OF STEPPING STONE ATTACKS

Marco Antonio Gamarra
Old Dominion University, 2019
Director: Dr. Sachin Shetty

Stepping stone attacks are one of the most sophisticated cyber-attacks, in which attackers make a chain of compromised hosts to reach a victim target. In this Dissertation, an analytic model with Multi-Agent systems approach has been proposed to analyze the propagation of stepping stones attacks in dynamic vulnerability graphs. Because the vulnerability configuration in a network is inherently dynamic, in this Dissertation a biased min-consensus technique for dynamic graphs with fixed and switching topology is proposed as a distributed technique to calculate the most vulnerable path for stepping stones attacks in dynamic vulnerability graphs. We use min-plus algebra to analyze and provide necessary and sufficient convergence conditions to the shortest path in the fixed topology case. A necessary condition for the switching topology case is provided.

Most cyber-attacks involve an attacker launching a multi-stage attack by exploiting a sequence of hosts. This multi-stage attack generates a chain of "stepping stones from the origin to target. The choice of stepping stones is a function of the degree of exploitability, the impact, attackers capability, masking origin location, and intent. In this Dissertation, we model and analyze scenarios wherein an attacker employs multiple strategies to choose stepping stones. The problem is modeled as an Adjacency Quadratic Shortest Path using dynamic vulnerability graphs with multi-agent dynamic system approach. With this approach, the shortest stepping stone path with maximum node degree and the shortest stepping stone path with maximum impact are modeled and analyzed.

Because embedded controllers are omnipresent in networks, in this Dissertation as a Risk Mitigation Strategy, a cyber-attack tolerant control strategy for embedded controllers is proposed. A dual redundant control architecture that combines two identical controllers that are switched periodically between active and restart modes is proposed. The strategy is addressed to mitigate the impact due to the corruption of the controller software by an adversary. We analyze the impact of the resetting and restarting the controller software and performance of the switching process. The minimum requirements in the control design, for effective mitigation of cyber-attacks to the control software, that implies a fast switching

period is provided. The simulation results demonstrate the effectiveness of the proposed strategy when the time to fully reset and restart the controller is faster than the time taken by an adversary to compromise the controller. The results also provide insights into the stability and safety regions and the factors that determine the effectiveness of the proposed strategy.

To the memory of my father Lucio,
my beloved mother Lucrecia,
my dear wife , Erika
and my sons Diego Marcelo & Edson Piero.

ACKNOWLEDGEMENTS

I would like to express my special appreciation and gratitude to my advisor Dr. Sachin Shetty, whose guidance, encouragement and support have played a central role in my personal development, in my research and in this dissertation. I appreciate all your contributions of time, ideas, and funding to make my scholarly experience productive and stimulating.

I want to thank my other committee members, Dr. Chunsheng Xin, Dr. Dimitrie C. Popescu, Dr. Rao Chaganty (Department of Mathematics & Statistics, Old Dominion University) and Dr. Laurent Njilla (Cyber Assurance Branch, U.S. Air Force Research Laboratory, Rome, NY) for their invaluable time and comments devoted to improving this dissertation.

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

A special thanks to all my family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 MOTIVATION

For the past 20 years, there has been an extensive use of the Information and Communication Technologies (ICT) in Critical Infrastructure (CI), as a collateral consequence there has been an increment in the frequency and severity of cyber attacks targeting CI. For example, the Stuxnet malware to the Iranian nuclear enrichment plant in 2010 [40], the Shamoon malware to the Saudi Arabian Oil Company Aramco in 2012 [14], and the BlackEnergy malware to the energy distribution companies of Ukraine in 2015 [41]. An effective Security Risk Assessment (SRA) is required for an optimal risk mitigation strategy.

SRA of critical infrastructure hinges on the ability to quantify the probability of lateral propagation of cyber-attacks [15, 29, 62, 74]. For example, attackers use spear phishing to identify a victim computer and then proceed to compromise vulnerable computers within a security enclave using Stepping Stone Attacks. In these cyber-attacks, an attacker gains access to a computer, from that position (stepping stone), attacks and gains access to some computer that he could not originally have access, from there repeating the strategy getting deep in the network, until some sensitive target is gained, in this way attackers make a chain of compromised machines to reach a victim (see Fig 1). The victim only sees the last host in the chain that makes challenging to find attackers from looking only the target.

The identification of the most vulnerable stepping stone attack paths is a crucial component for and effective SRA. A viable risk mitigation strategy should find an optimal balance between attack damage and response plan cost (mitigation actions), but also the selection of the response must be chosen without sacrificing the system functionalities (mission) [27, 38]. The most vulnerable stepping stone path encodes the number of stepping stones that should be considered in a response plan for optimal security resources allocation [9, 73].

The financial cost of a realistic response plan is constrained for a limited budget, and the resource allocation must be optimized, for this task the information encoded in the stepping stone paths provide to the defender an estimation of the financial cost required for a remediation plan, many cost models of budget allocation have been proposed in the literature; for example, if the response plan is to reduce the number of vulnerabilities and exploits, the financial cost of the resource budge for this task can be estimated with a *linear cost model* or an *exponential cost model* [42]. Also, the information encoded in the stepping stone path induces a resource budge required by the attacker to disrupt the system that can be estimated and anticipated by the defender.

Stepping stones are attractive to attackers mainly because it is easy to compromise a host on the internet; it is difficult to detect, allows attackers anonymity (the actual attacker is almost untraceable), and its scalability, even worse because Internet of Things (IoT) devices have been growing exponentially and is estimated by 2020 that the number of connected IoT devises worldwide will reach 20 Billion and that more than 65% of enterprises will adopt IoT products [35]. The massive deployment of IoT devices bring new security challenges, mainly because IoT devices have poor security or even none at all, typically do not run the full-version operative system, are resource-constrained making traditional risk mitigation strategies impractical. Provide security to the internet is already complicated, and adding billions of insecure devices make the task a mega challenge.

The choice of stepping stones is influenced by the attacker's goal, the increased difficulty of attribution and exertion of minimum effort. The identification of attacker stepping stones is relevant for security administrators to aid the mitigation process.

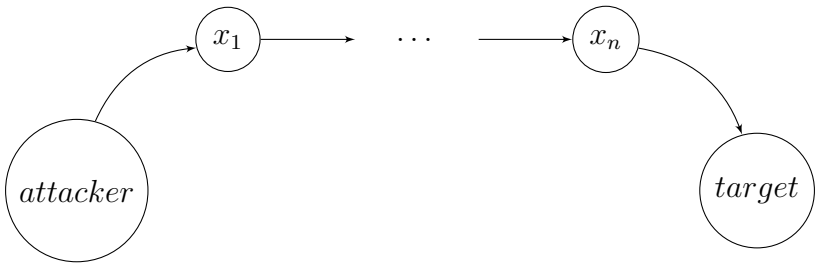

Figure 1. Attackers uses a chain $\{x_1, \ldots, x_n\}$ of compromised machines to reach a victim.

1.2 OBJECTIVE

The objective of this dissertation is twofold:

1. To develop formal mathematics models for *risk assessment* that quantify the propagation of stepping stone attacks in networks based on a multi-agent dynamical systems approach.

2. Because embedded controllers are ubiquitous in networks: To develop control systems strategies as a *risk mitigation* plan for embedded controllers to improve its tolerance to cyber-attacks.

1.3 RESEARCH CHALLENGES

1.3.1 ATTACK GRAPH COMPLEXITY

For more than two decades, researchers have been using attacks graphs in network security analysis, the idea of *attack graphs* appeared as early as 1996 [20, 44, 47, 52, 61]. An attack graph is a mathematical abstraction that models all the possible ways that an attacker can get access to a critical asset by exploiting a chain of vulnerabilities on the services running on the hosts. Even when attack graphs can be efficiently generated [36,60], the resulting *size and complexity* of the graphs makes its analysis and information processing a considerable challenge; it is virtually impossible to know which paths and vulnerabilities are the most important to the attackers success .

Depending on the node content, attack graphs can be divided into state-oriented attack graphs or attribute-oriented attack graph. In a state-oriented attack graph, the nodes typically represent the state of the system during an attack (attack node), which may encode how far the intruder has penetrated, and the attack's capabilities at that point. An edge from nodes s_i to another node s_j exits, if there is a vulnerability that can be exploited, transforming the system form the state s_i to the state s_j. The problem with state-oriented attack graph is that the number of states of the numbers of states will produce an "explosion" in the number of nodes increasing the graph complexity and the computational cost

for its analysis.

In this dissertation, *vulnerability graphs*, that are attributed-oriented attack graphs is used. In a vulnerability graph, typically nodes represent hosts, and each edge represents one way in which an attacker, who is on the source host, can gain access on the destination host through an exploit. If v_i and v_j are two nodes in a vulnerability graph, and edge from v_i to v_j exist, if there are an open port and a vulnerability in the host v_j, such that the system's rules allow an attacker that is resident in v_i to compromise v_j. From a visualization point of view, a vulnerability graph makes it easy to see how a chain of vulnerabilities can be compromised, from a graph complexity point of view, the correspondence node-host reduces the graph complexity. Even with this reduction in the complexity, the path analysis may be an NP-hard problem.

The construction of vulnerability graphs hinges on the following: availability of network connectivity; identification of software, operating systems, and network protocols; network and system access control rules; and vulnerability information. The National Vulnerability Database (NVD) and the Common Vulnerability Scoring System scores (CVSS) are typically used to quantify the exploitability of the vulnerabilities [48, 49].

Because attackers reach their objective by following attack paths, the analysis of attack paths is a widely studied attack graph-based security metrics. In this venue, the Shortest Path Metric, the Number of Path Metric, and the Mean of Path Lengths Metric are among other the most studied. The Shortest Path metric quantify the smallest attack path from an attacker's initial state to the attackers desired goal state [21, 37, 59, 61, 68]. The length of an attack path may be the number of exploits, the number of conditions or both of them, or any other complexity score associated with them. The shortest path metric assumes that the attacker is interested in using the least amount of effort to reach the goal target.

1.3.2 FROM HEURISTIC TO ANALYTIC MODEL

This dissertation has been inspired by [56], where the authors describe means of constructing models of networks, the access control mechanisms, and describe the software that they employ to approach the problem of finding which stepping stone paths are accessible for an attacker. The authors had used the shortest path metric in a vulnerability graph to model

and analyze stepping stone attacks with heuristic estimations of the shortest path. The basic idea of the shortest path metric comes from the attacker perspective, in the sense that given the option of different steps than the attacker can penetrate and disrupt any security policy, the attacker will choose the sequence of steps that require the least amount of effort. Effort exerted by an attacker has been represented by assigning an estimated amount of required time, resources to exploit vulnerabilities or complexity scores derived from them [37, 56, 68].

In [56], the authors describe a stepping stone attack as a path through a vulnerability graph, and according to the CVSS, a scoring function for every edge in the vulnerability graph has been constructed, called *exploit complexity score* that ranges between 0 and 10. This score is defined in [56] as

$$\epsilon = 10 - \varepsilon / A_v \tag{1}$$

where ε is the *exploitability sub score* and A_v is the *accessibility (vector) sub score* (provided by CVSS & NVD). This score (cost) quantifies the difficulty of compromise for each node. The smaller the score, the easier it is to exploit the vulnerability. In this approach, a **vulnerability graph** has been constructed such that given a set of n node hosts $\{h_1, \ldots, h_n\}$, a weighted edge exists from h_i to h_j if and only if there is a vulnerability that allows an attacker on h_i to compromise h_j. This weight ϵ_{ij} is the *exploit complexity score*, so $0 < \epsilon_{ij} < 10$. A *stepping stone path* is a path through a vulnerability graph, here the edges denote the vulnerabilities exploited by the attacker in the path that goes from the first node (attack source) to the last (attack target).

The **cost** of a stepping stone path is defined as the sum of the costs of the edges of the path. In this venue, the most vulnerable stepping stone path between a source node and a target node is the path with minimum cost, and if the vulnerability graph is static, can be calculated with the well known and efficient shortest-path algorithms. The problem of this approach is that according to the experience, there is no guarantee that the vulnerability scores (weights of the edges) will always remain the same during the attacker's lateral propagation, for example, due to defensive mechanisms that can result in modification to the firewall rules, patching of vulnerabilities, and application of security controls. Also, if the

attacker exploits a vulnerability in any host he gain experience, get passwords, certificates, etc., then he can exploit the same vulnerability in other hosts with more simplicity, in both cases, the vulnerability graph should change. Because the calculation of the shortest path in a graph with switching topology is NP-hard, the authors in [56] had used a heuristic approach with Montecarlo simulation to estimate the shortest path in a vulnerability graph with switching topology . An analytic model for this problem that explores the conditions in which is not NP-hard is required and has been developed partially in [23] and is expanded in this dissertation.

Example 1.3.1 (Illustrative example)**.** To illustrate the idea of vulnerability graph and stepping stone attacks representations, we are considering a network that is composed of three virtual servers and one physical server from [76] , the network topology is presented in Fig 2. The configuration and function information of the servers are shown in Table 1. In this example, it is assumed that an attacker takes the Root permissions in the target database PM1 as the final goal to obtain business data. In order to achieve this goal, an attacker can attack following many ways and means; see [76] for more details. For example, the attackers can find the SQL injection vulnerability CVE-2011-2688 on the web server VM21. Through this vulnerability, the attacker gets the user rights of the VM21 and establishes a connection with the database server PM1 on the VM21 with a legitimate identity. Then through the CVE-2012-2122 and CVE-2010-2693 vulnerabilities on the server PM1, the access mechanism is bypassed, getting the root permissions of the database server PM1. The associated vulnerability graph is presented in Fig 3.

In this context, the most vulnerable path that an attacker probably will follow is the one with minimum cost, for example, the stepping stone attack Host 0, VM21, PM1(User), PM1(Root) that has a $cost = 0 + 5.1 + 0.1 = 5.2$.

1.3.3 RISK MITIGATION STRATEGY

The dense deployments of IoT insecure devises have attracted the attention of attacker making IoT devises as a target for steeping stone attacks in enterprises, industry, critical infrastructure, etc., and because traditional risk mitigation strategies are impractical for IoT

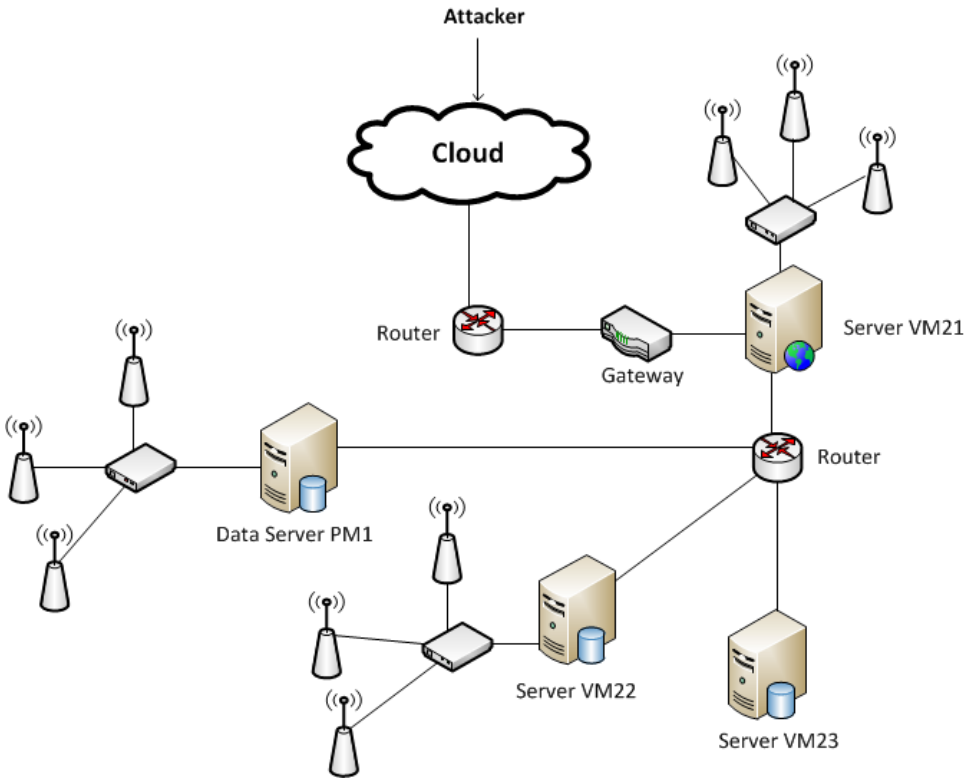

Figure 2. Network including 3 virtual servers and 1 physical server. The configuration and function information of the server are shown in Table 1.

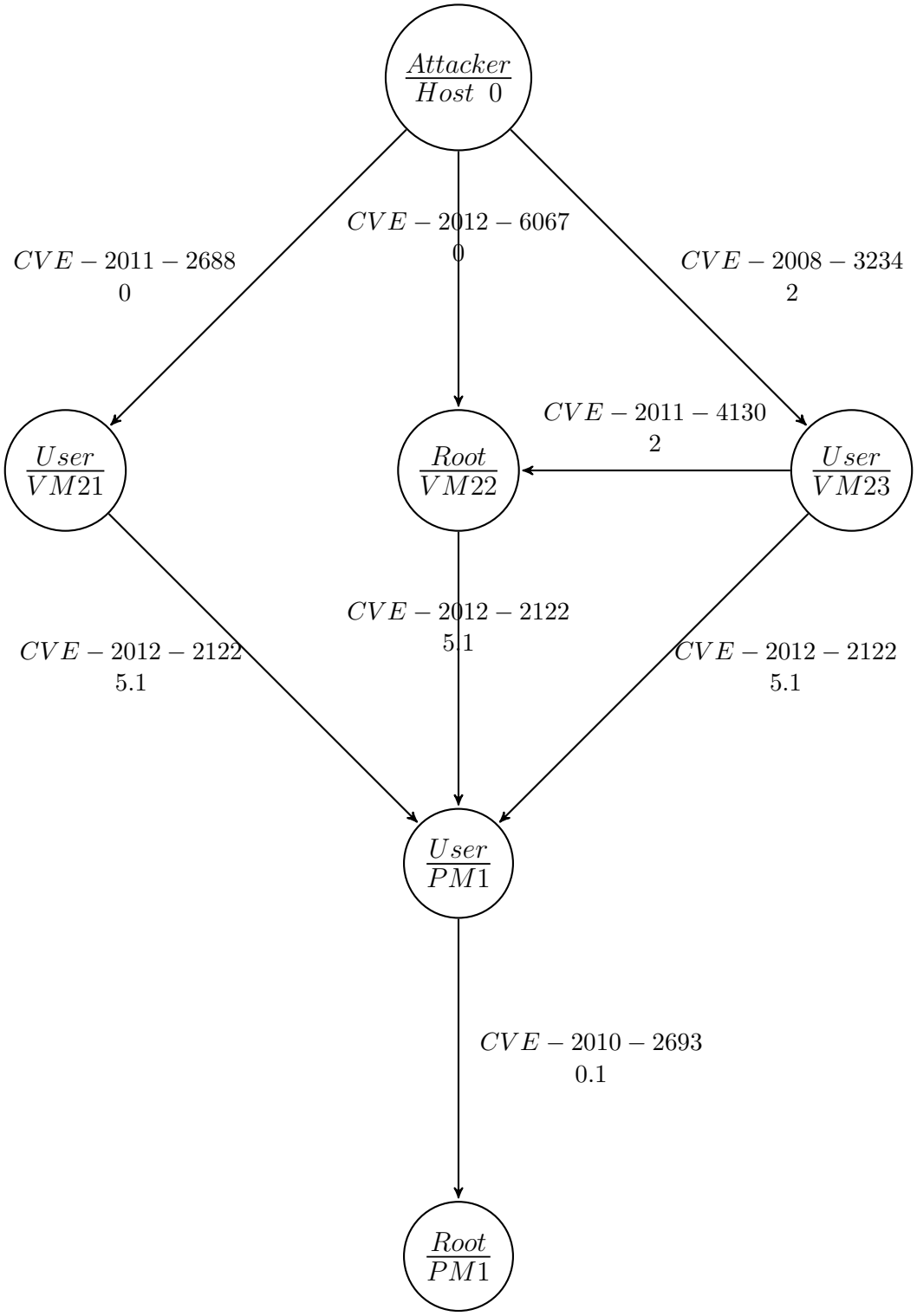

Figure 3. Vulnerability graph derived from the network presented in Fig 2. Every edge (a, b) is labeled with the CVE ID of the vulnerability b and its ϵ_{ab} score.

Host	OS	Function	Server	Vulnerability CVE-ID	Exploitability Sub score	A_v	Exploit Complexity score	NVD Last Modified
VM21	Redhat 5.4	Web Server	HTTP, SSH	CVE-2011-2688	10.0	1	0	08/28/2017
PM1	Redhat 5.4	Database Server	SSH	CVE-2012-2122, CVE-2010-2693	4.9 3.9	1 0.395	5.1 0.1	02/20/2014 07/14/2010
VM22	Redhat 5.4	File Server	FTP, SSH	CVE-2011-4130, CVE-2012-6067	8.0 10.0	1 1	2 0	12/08/2011 12/05/2012
VM23	Redhat 5.4	Host	SSH	CVE-2008-3234	8.0	1	2	09/28/2017

Table 1. Host Configuration, Function information and vulnerability scores information of the network presented in Fig 2.

devises, a new risk mitigation strategy is required for IoT ecosystems.

1.4 PROBLEM STATEMENT

1.4.1 PROBLEM 1: MATHEMATICAL MODEL FOR THE ANALYSIS OF STEPPING STONE ATTACKS IN NETWORKS WITH FIXED TOPOLOGY

To our best knowledge, the state of the art of the analysis of the risk assessment of the propagation of stepping stone attacks is limited to heuristic models and simulations. There is a need of formal mathematical models that faced this problem. In this dissertation, a Mathematical Model for the Analysis of stepping stone attacks in networks with fixed topology with multi-agent dynamic system approach is proposed [23].

1.4.2 PROBLEM 2: MATHEMATICAL MODEL FOR THE ANALYSIS OF STEPPING STONE ATTACKS IN NETWORKS WITH SWITCHING TOPOLOGY

According to the experience, a vulnerability graph is essentially dynamic; for example, if a new application is starting to run in a host, may introduce a set of new vulnerabilities that will modify the vulnerability graph topology in the number of nodes and edges. Also, there is no guarantee that the vulnerability scores (weight on the edges) will always remain the same during the attacker's lateral propagation due to defensive mechanisms that can result in a modification to the firewall rules, patching of vulnerabilities, and application of security controls. Moreover, if the attacker exploits a vulnerability in any host he gains experience,

gets passwords, certificates, etc., then he can exploit the same vulnerability in another host with more simplicity; in both cases, the vulnerability graph should change.

In [56], the authors propose an approach to calculate the most vulnerable paths corresponding to stepping stone attacks calculating the shortest path in a vulnerability graph with fixed topology, and a heuristic analysis using **Montecarlo** simulations has been proposed for the case of switching topology, assuming that when the attacker is at any host h and the graph topology changes, the path from the host h to the target is invariant or equivalently, as-yet unseen edge costs are invariant, then the standard shortest path algorithm can be used to **estimate** the minimum remaining cost. However, this assumption may induce a miscalculation of the remaining shortest path because there is no guarantee that the edge cost will always remain the same during the attackers lateral propagation due to defensive mechanisms that can result in modification to the firewall rules, patching of vulnerabilities, and application of security controls.

In this dissertation, the assumption of the invariant of as-yet unseen vulnerability is not considered, and a Mathematical Model for the Analysis and Simulation of stepping stone attacks switching topology with a dynamic multi-agent system approach is proposed [23]. In this approach, a dynamic vulnerability graph is introduced, which is a network of dynamic multi-agent systems with first order dynamics [58]. In this dissertation, Biased min-consensus protocol introduced in [80] is discretized for the calculation of the shortest stepping stone path between any two nodes when the vulnerability graph is static (fixed topology case), and when the vulnerability graph evolves in the time (switched topology case). The switching topology case is modeled as a switched dynamical system where the vulnerability graph changes according to a switching signal that is triggered by the network system defense. Min-plus Algebra [78] is used for the analysis of the convergence the shortest path in the fixed and switched topology cases. In the fixed topology case, a necessary and sufficient condition for the convergence to the shortest stepping-stone path is provided. In the switched topology case, a necessary condition for convergence to the shortest path in provided and the scenarios where the problem is not NP-hard is analyzed.

1.4.3 PROBLEM 3: STEPPING STONE PATH IN DYNAMIC VULNERABILITY GRAPH WITH CONSTRAINTS

Different models can be defined depending on the assumptions considered about the behavior of the attacker [37, 59]. In a more realistic scenario, the selection of attacker's next step is not limited to the minimize the difficulty of compromise a vulnerability, for example, an attacker may be interested also in minimizing the risk of detection as part of his strategy. To model this scenario, two scores (weights) can be defined for every edge, one that quantifies the difficulty of compromise a vulnerability and the other that quantifies the risk of detection in that task. By having multiple weights on each edge or multiple edges paths from every host, one can represent potentially-conflicting criteria in the selection of the next stepping stone (e.g.the attacker wishes to minimize both costs of difficulty and risk of detection). These kinds of problems include situations in which the choice of the shortest path is constrained by other metrics, or in general cases in which the objective function takes into account not only the cost of each selected edge but also the cost of the interactions among the edges in the solution. In the literature, these kinds of problems are called Quadratic Shortest Path Problem (QSPP). The QSPP is the problem of finding a path in a directed graph from the source vertex s to the target vertex t such that the sum of costs of the edges and the sum of interaction costs over all distinct pairs of edges on the path is minimized [65]. In this dissertation, the models developed for Problem 1 and Problem 2 are extended for the analysis of the stepping-stone stacks with constraints.

1.4.4 PROBLEM 4: RISK MITIGATION STRATEGY

Since traditional networks are well-protected with stable defensive mechanisms, attackers are now trying to intrude through the weak Internet of Things (IoT) devices to disrupt cyber-physical systems. Because IoT devices operate deep inside of networks, traditional perimeter defenses may be ineffective. Another problem in IoT ecosystems emerges because IoT devices typically do not run full-version operating systems, require low power consumption, and are resource constrained. Moreover, default passwords, unpatched bugs remain deployed long after vendors cease to produce or support them. Thus, traditional risk mitigation

mechanisms like antivirus, patches are impractical to expect in IoT ecosystems. Therefore, in a practical IoT and Industrial IoT (IIoT) ecosystems, it is almost impossible to patch up all the vulnerabilities existing in the devices within the network. To make the network more tolerant to cyber-attack, alternative risk mitigation strategies should be investigated. Unlike traditional IT ecosystems where host-based detection and prevention are prevalent, we believe that in a Cyber-Physical System (CPS), to mitigate the vulnerabilities introduced by IoT device, we need to leverage some intrinsic properties of the physical systems like the *inertia* of the physical system, to develop a strategy to tolerate cyber-attacks. In this dissertation, a *Dual Redundant Cyber-attack Tolerant Control System Strategy for Cyber-Physical Systems* is proposed as a risk mitigation strategy [24].

1.5 STATEMENT OF CONTRIBUTION

The contributions of this dissertation are:

1. A basic mathematical model for the analysis of the stepping stone attack as a network of multi-agent dynamical system in a vulnerability graph with fixed and switching topology is provided, where an interplay between biased min-consensus and min-plus algebra is used for modeling and analysis.

2. Necessary and sufficient conditions for a finite-time convergence of the shortest stepping-stone path in the fixed topology case is provided, and a necessary condition for the time interval between two consecutive switching signal that ensures the convergence of the shortest stepping-stone path in finite-time is also provided.

3. A mathematical model for the analysis of the stepping-stone attack with constraints as an Adjacent Quadratic Shortest Path Problem is proposed with multi-agent dynamical system approach. The conditions for a finite-time convergence to the shortest path are provided, and the cases where this problem is solved in polynomial time is analyzed.

4. To our best understanding, we are the first to propose a formal mathematical model for the analysis of stepping stone attacks with a multi-agent dynamical system approach.

5. A dual redundant control strategy that switches two identical controllers to mitigate the

impact of cyber-attacks that corrupt the integrity of the embedded-controller software in critical infrastructure has been proposed as a risk mitigation strategy.

1.6 RELATED WORK

Attack graphs in network security analysis have been investigated since 1996 [20, 44, 47, 52, 61]. Various forms of attack graphs have been proposed for analyzing the security of enterprise networks [3, 36, 57, 60, 61, 71].

In [56], vulnerability graphs were introduced with an heuristic model for the analysis and simulation of the stepping-stone attack with fixed topology and a combination of Montecarlo simulation with shortest path algorithms were used for the estimation of the stepping-stone shortest path in the case of switching topology.

The shortest path problem in a directed graph has been formulated as a linear equation in a min-plus algebra, which can be solved by the Bellman-Ford algorithm [22,78]. A variant of the Bellman-Ford algorithm for single-source shortest paths in graphs that optimize the algorithm, compared with the previously best variant by Yen [79], is reported in [5]. In all of these works, the graph topology is fixed. Average consensus in network with switching topologies is investigated in [58]. In [55], using max-plus algebra, max-consensus in graph with switching topology is investigated.

The Shortest Path Problem with Variance Constraint that is an special case of the shortest path with constraints, is studied in [72], and a general approach to the *Quadratic Shortest Path Problem* (QSPP) is studied [33, 34, 64, 65]. The *Adjacent QSPP* (AQSPP) is studied in [33, 64].

In [4], the authors propose a defense strategy for CPS that takes advantage of physical inertia to improve tolerance to cyber-attacks [4]. In [2], the authors propose an attack-tolerant design for embedded control systems, using a proactive reset of a Simplex architecture [70] with a proactively reset and switch between two redundant components.

1.7 DISSERTATION ORGANIZATION

This dissertation is organized as follows:

- In Chapter 2, the necessary background needed throughout the dissertation are presented.

- In Chapter 3, a mathematical model for the analysis of stepping stone attack in a vulnerability dynamic graph with fixed and switching topology is proposed.

- In Chapter 4, a mathematical model for the analysis of stepping stone attack in a vulnerability dynamic graph with constraints is proposed.

- Chapter 5 is devoted to proposing a control strategy for the risk mitigation in a Cyber-physical system, in this venue a *Dual Redundant Cyber-attack Tolerant Control System Strategy for Cyber-Physical Systems* is proposed.

- Chapter 6 presents conclusions and future research.

Chapter 2

BACKGROUND

In this chapter, the basic concepts of graph theory, vulnerability graphs, a network of multi-agent systems, consensus protocol, biased min-consensus, and other topics needed in this dissertation are presented.

2.1 GRAPHS

Given a finite set $V = \{v_1, ..., v_n\}$

Definition 2.1.1. A *directed Graph* over V is an ordered pair $G = (V, E)$, where E is a sub set of the Cartesian product $V \times V$. In this context V is called Vertex set and E is called Edge set. Every $v_i \in V$ is called **vertex** or **node** and every ordered pair (v_i, v_j) of E is called **directed edge**, where v_i is called the tail and v_j is called the head of the edge

Definition 2.1.2. A **directed weighted Graph** over V is an ordered triple $G = (V, E, w)$, where (V, E) is an directed graph and $w : E \longrightarrow \mathbb{R}$ is a function that associates a value to each edge.

Definition 2.1.3. Given a directed weighted graph $G = (V, E, w)$

1. A vertex v_j is said **adjacent** to v_i if and only if $(v_i, v_j) \in E$

2. For every vertex v_i is defined the set of all its **neighbors** as $N_i = \{v_j \in V / (v_i, v_j) \in E\}$

3. A **path** $\mathcal{C}$ of length m in G is a sequence of m+1 vertex $v_{i_1}, v_{i_2}, ..., v_{i_{m+1}}$ such that $(v_{i_k}, v_{i_{k+1}}) \in E$ for all $k = 1, ..., m$. If $v_{i_1} = v_{i_{m+1}}$, then $\mathcal{C}$ is called a **cycle** of length m. A **cycle** of length 1 is called a self-loop.

4. The weighted **adjacency** matrix associated to the weighted graph $G = (V, E, w)$ is defined as the square $n \times n$ matrix A such that $[A]_{ij} = w_{ij} > 0$ if $(v_i, v_j) \in E$ and $[A]_{ij} = 0$ in other case.

Definition 2.1.4. An *undirected Graph* over V is a directed graph $G = (V, E)$, such that for every directed edge (v_i, v_j) in E, there is a directed edge (v_j, v_i) in E, this two edges are denoted in a compact way as the unordered pair $\{v_i, v_j\}$ and is called undirected edge.

Definition 2.1.5. A directed graph is said to be strongly connected if there is a path for every two nodes, and it is called weakly connected if the graph obtained by adding an edge (v_j, v_i) for every existing edge (v_i, v_j) in the original graph is strongly connected. An undirected graph is connected if an only if is strongly connected.

2.2 VULNERABILITY GRAPH AND STEPPING STONES

Consider as network $\mathcal{N}$ with m host $\{h_1, \ldots, h_m\}$. Each host h_i has a set of applications, and each application has a set of well-known vulnerabilities (eventually empty), and an open port through an authorized, or an unauthorized user may gain access to h_i.

Definition 2.2.1. A **Vulnerability graph** $G = (V, E)$ associated with the network $\mathcal{N}$ is a directed graph that represents ways in which an adversary can exploit sequentially different vulnerabilities to disrupt the system. The set of nodes $V = \{v_1, \ldots, v_n\}$ represent all the vulnerabilities of the network $\mathcal{N}$ and $E \subseteq V \times V$ is the set of directed edges of G that represent the vulnerability relations.

If v_i and v_j are vulnerabilities of applications running in hosts h_k and h_l of $\mathcal{N}$ respectively, the directed edge (v_i, v_j) on G, means that the system rules allow accessing host h_l from host h_k trough the vulnerability v_j. In other words, the edge (v_i, v_j) on G will enable an attacker that is resident in h_k trough the vulnerability v_i, to reach h_l trough the vulnerability v_j.

Definition 2.2.2. The **edge cost** is a function ζ over the set of edges E that quantifies any property related to exploiting a vulnerability, that is

$$\zeta : E \longrightarrow A \subseteq \mathbb{R};$$
$$(v_i, v_j) \longmapsto \zeta[(v_i, v_j)]$$

In this dissertation, we will normalize all the edge costs to 10, for example, an edge cost function over a vulnerability graph G that is used in this dissertation, is the *exploit*

complexity score [56], defined as:

$$\begin{aligned} &\zeta_c : E \longrightarrow [0, 10]; \\ &(v_i, v_j) \longmapsto \zeta_c[(v_i, v_j) = \epsilon_{ij} = 10 - \frac{\varepsilon_{ij}}{A_v} \end{aligned} \tag{2}$$

that has a range between 0 and 10, where ε_{ij} is the *exploitability sub score* of the vulnerability v_j provided by CVSS and A_v is its *accessibility vector*. $\zeta_c[(v_i, v_j)]$ is denoted with ϵ_{ij} and quantifies the difficulty to exploit the vulnerability v_j. The smaller ϵ_{ij}, the easier it is to exploit the vulnerability. See [56] for more details on the construction of this score.

2.3 CONSENSUS PROTOCOL IN A DYNAMIC GRAPH

Let $G = (V, E)$ be a directed graph with a vertex set $V = \{v_1, \cdots, v_n\}$ and edge set E. Let $x_i \in \mathbb{R}$ denote the value of vertex v_i. Definition 2.3.1 and Definition 2.3.2 are introduced according to [58].

Definition 2.3.1. The pair (G, x) with $x = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$ is called an **algebraic graph** with value $x \in \mathbb{R}^n$ and topology (or information flow) G. The value x_i of a vertex v_i represents any physical quantities or any other attribute of the network.

Assume that every vertex-value x_i in the algebraic graph (G, x) is a dynamic agent with dynamics

$$\frac{dx}{dt} = \dot{x}_i(t) = f_i(x(t), u_i), \quad i \in I = \{1, \cdots, n\}$$

Definition 2.3.2. A **dynamic graph** is a dynamical system with a state (G,x) in which the value $x = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix}^T$ evolves according to the network dynamics

$$\dot{x}_i = f_i(x, u_i), \quad \forall i = 1, 2, \ldots, n \tag{3}$$

where

$$u_i = g_i(x), \tag{4}$$

is called a **distributed protocol** with topology G. In matrix form

$$\dot{x} = f(x, u), \tag{5}$$

where

$$\dot{x} = \begin{bmatrix} \dot{x}_1 \\ \vdots \\ \dot{x}_n \end{bmatrix},$$

$$f(x, u) = \begin{bmatrix} f_1(x, u_1) \\ \vdots \\ f_n(x, u_n) \end{bmatrix}, \quad \text{and}$$

$$u = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} = \begin{bmatrix} g_1(x) \\ \vdots \\ g_n(x) \end{bmatrix}.$$

An algebraic graph is called an **algebraic network**, and a dynamical graphs is called a **dynamic network**. In this dissertation a simple case of the dynamic network (3) when $f_i(x, u_i) = u_i$ is considered, that is,

$$\dot{x}_i(t) = u_i(t), \quad \forall i = 1, 2, \ldots, n \tag{6}$$

The next two examples are an illustration of a distributed calculation the linear function $\chi(x) = Ave(x(0))$ and of the nonlinear $\chi(x) = \min(x(0))$, using a dynamic graph (G, x), where G is assumed strongly connected and $N_i = \{x_j | (x_i, x_j)\ \textit{is an edge in}\ (G, x)\}$ denotes the set of all the neighbors of the vertex x_i

Example 2.3.1 (Average consensus)**.** The following distributed linear protocol

$$u_i = \sum_{j \in N_i} (x_j - x_i) \tag{7}$$

asymptotically solves the average consensus and is called the *average consensus protocol*,

that is, replacing (7) in (6) yields

$$\dot{x}_i = \sum_{j \in N_i} (x_j - x_i) \tag{8}$$

as is proved in [58],

$$\begin{aligned} \lim_{t \to \infty} x_i(t) &= \frac{1}{n} \sum_{i=1}^{n} x_i(0), \quad i = 1, \ldots, n \\ &= Ave(x(0)) \end{aligned}$$

Example 2.3.2 (Min-consensus)**.** The following nonlinear distribute protocol

$$u_i = \min_{j \in N_i} (x_j - x_i) \tag{9}$$

asymptotically solves the *min-consensus* problem. Replacing (9) in (6) yields

$$\begin{aligned} \dot{x}_i &= \min_{j \in N_i} (x_j - x_i) \\ &= -x_i + \min_{j \in N_i} (x_j) \end{aligned} \tag{10}$$

as is proved in [58],

$$\begin{aligned} \lim_{t \to \infty} x_i(t) &= \min\{x_1(0), \ldots, x_n(0)\}, \quad i = 1, \ldots, n \\ &= \min(x(0)) \end{aligned}$$

2.4 BIASED MIN-CONSENSUS

The following nonlinear distribute protocol

$$u_i = -x_i + \min_{j \in N_i} (x_j + w_{ij}) \tag{11}$$

where w_{ij} is the edge cost of (x_i, x_j), is called *biased min-consensus protocol* [80]. Then

$$\dot{x}_i = -x_i + \min_{j \in N_i}(x_j + w_{ij}) \tag{12}$$

asymptotically converges to the equilibrium point x_i^* [80], that is,

$$\lim_{t \to \infty} x_i(t) = x_i^*$$

which satisfies the following equation:

$$x_i^* = \min_{j \in N_i}(x_j^* + w_{ij}), \quad i = 1, \ldots, n$$

2.5 LEADER-FOLLOWER STRATEGY

In many applications of multi-agent systems, a leader-follower strategy is considered. In this approach, a subset of agents is called *leader set* N_l, and the remaining agents are called *follower set* N_f. In this context, the average consensus, the min-consensus, and the biased min-consensus are

$$\begin{cases} \dot{x}_i = \mu_i & \text{if } i \in N_l \\ \dot{x}_i = \sum_{j \in N_i}(x_j - x_i) & \text{if } i \in N_f \end{cases} \tag{13}$$

$$\begin{cases} \dot{x}_i = \mu_i & \text{if } i \in N_l \\ \dot{x}_i = -x_i + \min_{j \in N_i}(x_j) & \text{if } i \in N_f \end{cases} \tag{14}$$

$$\begin{cases} \dot{x}_i = \mu_i & \text{if } i \in N_l \\ \dot{x}_i = -x_i + \min_{j \in N_i}(x_j + w_{ij}) & \text{if } i \in N_f \end{cases} \tag{15}$$

respectively, where μ_i is an exogenous input. If $\mu_i = 0$, the systems are called *static leaders*.

The following theorem has been stated and proven in [80].

Theorem 2.5.1. *Let G be an undirected connected graph, and suppose that the dynamic network (G, x) evolves according to the protocol* (15) *with static leaders. Then the system asymptotically converges to the equilibrium point x^* of* (15)*, which satisfies the following equation* [80]*:*

$$\begin{cases} x_i^* = x_i(0) & \text{if } i \in N_l \\ x_i^* = \min_{j \in N_i}(x_j^* + w_{ij}) & \text{if } i \in N_f \end{cases} \tag{16}$$

2.5.1 BIASED MIN-CONSENSUS AND SHORTEST PATCH

The relationship between the biased min-consensus protocol in a dynamic network (G, x) and the shortest path in G has been developed in [80] as follows:

1. The "leader" agents are static, that is, $\dot{x}_i(t) = 0, \forall x_i \in N_l$, and are called *destination nodes*. The "follower" agents are called *source nodes*.

2. If there is an edge between x_i and x_j, the weight w_{ij} of this edge is the *length* between these agents.

3. The system evolves according to the protocol (15) with static leaders. Note that according to the optimality principle of Bellman's dynamic programming [8], the solution of the considered shortest path problem satisfies the following nonlinear equations:

$$\begin{cases} x_i^* = 0 & \text{if } i \in N_l \\ x_i^* = \min_{j \in N_i}(x_j^* + w_{ij}) & \text{if } i \in N_f \end{cases} \tag{17}$$

which are the equilibrium points of (15) with static leaders and $x_i(0) = 0, \quad \forall x_i \in N_l$.

The following theorem is stated and proven in [80].

Theorem 2.5.2. *If $x_i(0) = 0, \quad \forall x_i \in N_l$, then the equilibrium of the system* (15) *with statics leaders is given by* (17)*, which forms a solution to the corresponding shortest path problem* [80].

Remark 2.5.1.

1. When the estates have reached the equilibrium point (17), the shortest path can be found by recursively finding the parent nodes [80].

2. According Theorem 2.5.1 and Theorem 2.5.2, all the states values of the system globally converges to the lengths of the corresponding shortest path independently of the initial state values [80].

Chapter 3

MATHEMATICAL MODEL FOR THE ANALYSIS OF STEPPING-STONE ATTACKS IN VULNERABILITY GRAPHS

3.1 INTRODUCTION

In this chapter, a basic mathematical model that attempts to quantify the propagation of stepping stone attack is vulnerability graphs with fixes and switching topology is proposed [23] . The model is developed with a multi-agent systems approach where the biased min-consensus technique for dynamic graphs with fixed and switching topology is used as a distributed technique to determine the most vulnerable stepping stones path in dynamic vulnerability graphs. We use min-plus algebra to provide a necessary and sufficient condition for the convergence to the shortest path on a graph with fixed topology, and a necessary condition for the switching topology case.

3.2 ATTACKER'S MODEL

In this model, the following assumptions that are stated in [20] are used

1. A priory the attackers do not know the whole network topology (whole vulnerability graph). They only know the attacks that can be directly applied in a single step.

2. The attackers have a good memory. They remember all the sets of vulnerabilities they already exploited and remember all sets of privileges they already acquired during the intrusion process.

3. The attackers are sensible (intelligent). They will not attempt an attack that would give them privileges they already have; hence, the attacker never goes back to an already compromised state (monotonicity property).

3.3 STEPPING STONE DYNAMICS

3.3.1 FIXED TOPOLOGY CASE

Given a vulnerability graph G with n nodes $\{v_1, \ldots, v_n\}$, a dynamic graph (G, x) is assigned such that for every v_i there is a state $x_i \in \mathbb{R}$ and the weight of the edge (x_i, x_j) is its exploit complexity score ϵ_{ij} defined in equation (2). A leader-follower strategy is used where the state x_i evolves according to biased min-consensus dynamics (15) with static leaders; in this model, the leaders are called *valuable targets* and the followers are called *source nodes*. If x_l is a valuable target, the state of the source x_i is defined as the minimum stepping stone cost from x_i to x_l, that is, the minimum length of the path from x_i to x_l. Mathematically, the stepping stone dynamics can be written as

$$\begin{cases} \dot{x}_i(t) = 0, \quad x_i(0) = 0 & \text{if } i \in N_l \\ \dot{x}_i(t) = -x_i(t) + \min_{j \in N_i}(x_j(t) + \epsilon_{ij}) & \text{if } i \in N_f \end{cases} \tag{18}$$

then, according to Theorem 2.5.2, in the equilibrium $x_i^* \in N_f$ is the minimum stepping stone cost from x_i to any valuable target x_l, that is, the most vulnerable stepping stone path from v_i to the valuable target v_l.

As has claimed in [56], according to experience, as an attacker penetrates more deeply into a system, the *exploit complexity score* should change; as a consequence, the graph topology should change as well. A more realistic model where the *exploit complexity score* changes as the attack penetrates more deeply in the system is developed in the following subsection.

3.3.2 SWITCHING TOPOLOGY CASE

Let $\{G_1, \ldots, G_m\}$ be a finite collection of vulnerability multi-graphs with the same n nodes $\{v_1, \ldots, v_n\}$ and $s : \mathbb{R} \longrightarrow \{1, \ldots, m\}$ a switching signal. For every $s(t) = l$, a vulnerability multi-graph $G_l \in \{G_1, \ldots, G_m\}$ and its associated dynamic graph (G_l, x) are well defined, then the stepping stone dynamics *with switching topology* is equivalent to the

hybrid system

$$\begin{cases} \dot{x}_i(t) = 0, \quad x_i(0) = 0 & \text{if } i \in N_l \\ \quad l = s(t) & \\ \dot{x}_i(t) = -x_i(t) + \min_{j \in N_i}(x_j(t) + \epsilon_{ij}(l)) & \text{if } i \in N_f \end{cases} \tag{19}$$

where $\epsilon_{ij}(l)$ is the exploit vulnerability score of the edge (x_i, x_j) in the dynamic graph (G_l, x). Hence if the network has a vulnerability graph G_p and at any time $t > 0$:

1. An attack from h_i is detected in h_j, then the network's vulnerability graph switches to $G_{q=s(t)}$ such that $\epsilon_{ij}(q) > \epsilon_{lm}(p)$ for all $\epsilon_{lm}(p) = \epsilon_{ij}(p)$.

2. An attack from h_i compromises h_j, then the network's vulnerability graph switches to $G_{q=s(t)}$ such that $\epsilon_{ij}(q) < \epsilon_{lm}(p)$ for all $\epsilon_{lm}(p) = \epsilon_{ij}(p)$.

The progress of the stepping stone dynamics is monitored at δ time intervals; hence, the stepping stone dynamics at a discrete time for the fixed topology case is

$$\begin{cases} x_i[k+1] = x_i[k], \quad x_i[0] = 0 & \text{if } i \in N_l \\ x_i[k+1] = \min_{j \in N_i}(x_j[k] + \epsilon_{ij}) & \text{if } i \in N_f \end{cases} \tag{20}$$

where

$$x[k] \stackrel{\text{def}}{=} x(\delta k), \quad \delta > 0 \quad \text{and} \quad k \in \mathbb{Z}_0^+ \tag{21}$$

and for the switching-topology case is

$$\begin{cases} x_i[k+1] = x_i[k], \quad x_i[0] = 0 & \text{if } i \in N_l \\ \quad l = s[k] & \\ x_i[k+1] = \min_{j \in N_i}(x_j[k] + \epsilon_{ij}(l)) & \text{if } i \in N_f \end{cases} \tag{22}$$

3.3.3 MIN-PLUS ALGEBRA

Min-plus algebra consists of two binary operations, $\oplus$ and $\otimes$, on the set $\mathbb{R}_{min} = \mathbb{R} \cup \{+\infty\}$, defined as follows

$$a \oplus b = \min\{a, b\} \tag{23}$$

$$a \otimes b = a + b \tag{24}$$

The neutral element with respect of the min-plus addition $\oplus$ is $+\infty$, denoted as θ, and with respect to the min-plus multiplication $\otimes$ is 0, denoted as e. Both operations are associative and commutative, and the multiplication is distributive over the addition. Both operations are extended to matrices as follows. Given $A, B \in \mathbb{R}_{min}^{m \times n}$,

$$[A \oplus B]_{ij} = a_{ij} \oplus b_{ij}, \quad i = 1, \ldots, m \quad j = 1, \ldots, n$$

Given $A \in \mathbb{R}_{min}^{m \times q}$, $B \in \mathbb{R}_{min}^{q \times n}$,

$$\begin{aligned} [A \otimes B]_{ij} &= \bigoplus_{k=1}^{q} (a_{ik} \otimes b_{kj}), \quad i = 1, \ldots, m \quad j = 1, \ldots, n \\ &= \min_{k=1}^{q} \{a_{ik} + b_{kj}\}. \end{aligned}$$

The identity matrix of size n is a square matrix denoted by I_n and given by

$$[I_n]_{ij} = \begin{cases} e & \text{for } i = j \\ \theta & \text{for } i \neq j \end{cases}$$

If $A \in \mathbb{R}_{min}^{n \times n}$, for any integer $k \geq 1$,

$$A^k = \underbrace{A \otimes A \otimes \cdots \otimes A}_{k-1 \quad \text{multiplications}}$$

and $A^0 = I_n$. For more properties and application of min-plus algebra, see [22] and the references therein. If G is a vulnerability graph with n nodes $\{1, \ldots n\}$, a modified weighted

adjacency matrix $A \in \mathbb{R}_{min}^{n\times n}$ is associated to G defined as

$$A = \begin{cases} [A]_{ii} = e, & \text{if } i \text{ is a target node} \\ [A]_{ij} = \epsilon_{ij}, & \text{if } (i,j) \text{ is and edge} \\ [A]_{ij} = \theta, & \text{in other cases} \end{cases} \tag{25}$$

Notice that in this matrix, $[A]_{ij} \neq \theta$ means that there is one path of length 1 from i to j in G. Then in $A^2 = A \otimes A$, if $[A \otimes A]_{ij} \neq \theta$, means that there is a path of length 2 in G with a minimum cost from node i to node j; that is, there is a node l in G such that (i,l) and (l,j) are edges in G such that $\min_k\{a_{ik} \otimes a_{kj}\} = a_{il} \otimes a_{lj} = a_{il} + a_{lj}$, but also $a_{il} \otimes a_{lj} \neq \theta$ implies that $a_{il} \neq \theta$ and $a_{lj} \neq \theta$ simultaneously, and $a_{il} \otimes a_{lj} = \theta$ implies that $a_{il} = \theta$ or $a_{lj} = \theta$. Using the min-plus formalism, the stepping stone dynamics with fixed topology (20) can be rewritten as

$$\begin{cases} x_i[k+1] = x_i[k] = x_i[0] = 0 & \text{if } i \in N_l \\ x_i[k+1] = \bigoplus_{j\in N_i}(x_j[k] \otimes \epsilon_{ij}) & \text{if } i \in N_f \end{cases}$$

or in matrix form as

$$x[k+1] = A \otimes x[k] = A^{k+1} \otimes x[0] \tag{26}$$

and the stepping stone dynamics with switched topology (22) can be written as

$$\begin{cases} x_i[k+1] = x_i[k] = 0 & \text{if } i \in N_l \\ \quad l = s[k] & \\ x_i[k+1] = \bigoplus_{j\in N_i}(x_j[k] \otimes \epsilon_{ij}(l)) & \text{if } i \in N_f \end{cases}$$

or in matrix form

$$x[k+1] = A_l \otimes x[k], \quad l = s[k] \tag{27}$$

where A_l is the matrix associate with the vulnerability graph $G_l \in \{G_1, \ldots, G_m\}$.

Theorem 3.3.1. *A necessary and sufficient condition for which the stepping stone dynamics* (20) *converge to equilibrium* (16) *is that* $A^{k+1} = A^k$ *for any integer* $k \geq 1$.

Proof. **Sufficiency**

$$\begin{aligned} x[k+1] &= A \otimes x[k] \\ &= A^{k+1} \otimes x[0] \\ &= A^k \otimes x[0] \\ &= x[k] \end{aligned}$$

hence, $x[k+1] - x[k] = 0$, which implies (16).

Necessity: If the system is in equilibrium for any integer $k \geq 1$, implies that

$$x[k+1] - x[k] = 0$$

or equivalently

$$\begin{cases} x_i[k+1] = x_i[k] = x_i[0] = 0 & \text{if } i \in N_l \\ x_i[k+1] = x_i[k] = \min_{j \in N_i}(x_j[k] + \epsilon_{ij}) & \text{if } i \in N_f \end{cases}$$

hence,

$$[A^{k+1}]_{ii} = [A^k]_{ii} = 0 \quad \text{if} \quad i \in N_l$$

and

$$[A^{k+1}]_{ii} = [A^k]_{ii} = \theta \quad \text{if} \quad i \in N_f$$

because in the vulnerability graph there are no self-loops in the source nodes. If $i \in N_f$,

there is a path of length k from node i to any target node j with minimum cost

$$\begin{aligned} x_i[k] &= [A^k]_{ij} \otimes x_j[0] \\ &= [A^k]_{ij}, \end{aligned}$$

and there is a path of length $k+1$ from node i to the same target node j with minimum cost

$$\begin{aligned} x_i[k+1] &= [A^{k+1}]_{ij} \otimes x_j[0] \\ &= [A^{k+1}]_{ij} \\ &= x_i[k] \\ &= [A^k]_{ij} \end{aligned}$$

because $x_j[0] = 0$ is the cost for a target node, therefore

$$A^{k+1} = A^k$$

□

Corollary 3.3.1. *If $A \in \mathbb{R}_{min}^{n\times n}$ is the matrix associated with the vulnerability graph G and there is an integer $k \geq 1$ such that $A^{k+1} = A^k$, then $k \leq n-1$.*

Proof. $x_i[k+1] = [A^{k+1}]_{ij} = x_i[k] = [A^k]_{ij}$ implies that there is a path of length k from node i to node j with minimum cost, equivalently there is a simple path from i to j with length k. As the maximum length of a simple path in a graph with n nodes is $n-1$, then $k \leq n-1$. □

Remark 3.3.1. Corollary 3.3.1 implies that the stepping stone dynamics (20) converge to equilibrium in finite time $\tau = k\delta$ with $k \leq n-1$ instant communications.

Theorem 3.3.2. *The stepping stone dynamics with switching topology* (22) *converge to equilibrium if the time interval between two consecutive switching signals is slow enough, as $k = n-1$ instant communication.*

Proof. By Corollary 3.3.1, $n-1$ instant communications are the finite time period that guarantee the equilibrium for any fixed graph topology, then if the time interval between two switching signals is slow enough as n-1, then the system reach the equilibrium and the shortest path is calculated, which prove the theorem. □

Remark 3.3.2. In Theorem 3.3.2, notice that a convergence to equilibrium is possible for any time interval between switching signals in less than $n-1$ communication instants as shown in example 3.3.2, we will discuss this point in sections 5.

The model was developed for a directed graph with the condition that there exists a path from every source node to any target because, in general, a vulnerability graph is directed and acyclic. With the purpose of illustrating our approach, in examples 3.3.1 and 3.3.2, an idealized synthetic vulnerability graph that is strongly connected (undirected connected) is considered and assuming that an attacker has, in theory, the advantage of compromise any node in any layer of the vulnerability graph.

Example 3.3.1 (Fixed topology)**.** Consider the 10-node synthetic (undirected) vulnerability graph G presented in Fig 4, the *exploit complexity scores* are encoded in its weighted adjacency matrix M_ϵ is presented in equation (28) and its modified adjacency matrix A defined in (25) is presented in equation (29).

$$M_\epsilon = \begin{bmatrix} 0 & 3 & 0 & 0 & 2 & 6 & 0 & 0 & 0 & 0 \\ 3 & 0 & 8 & 0 & 0 & 4 & 2 & 0 & 0 & 0 \\ 0 & 8 & 0 & 4 & 0 & 2 & 4 & 3 & 0 & 0 \\ 0 & 0 & 4 & 0 & 3 & 1 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 3 & 0 & 2 & 0 & 0 & 0 & 0 \\ 6 & 4 & 2 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \end{bmatrix} \tag{28}$$

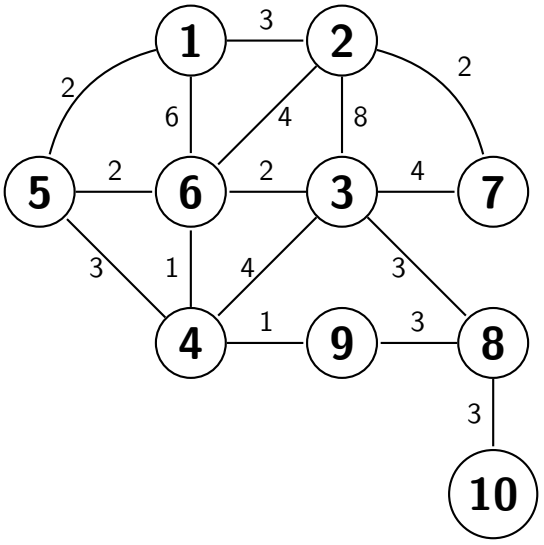

Figure 4. A 10-node vulnerability graph $G = G_p$ with node 1 as a target node.

$$A = \begin{bmatrix} 0 & 3 & +\infty & +\infty & 2 & 6 & +\infty & +\infty & +\infty & +\infty \\ 3 & +\infty & 8 & +\infty & +\infty & 4 & 2 & +\infty & +\infty & +\infty \\ +\infty & 8 & +\infty & 4 & +\infty & 2 & 4 & 3 & +\infty & +\infty \\ +\infty & +\infty & 4 & +\infty & 3 & 1 & +\infty & +\infty & 1 & +\infty \\ 2 & +\infty & +\infty & 3 & +\infty & 2 & +\infty & +\infty & +\infty & +\infty \\ 6 & 4 & 2 & 1 & 2 & +\infty & +\infty & +\infty & +\infty & +\infty \\ +\infty & 2 & 4 & +\infty & +\infty & +\infty & +\infty & +\infty & +\infty & +\infty \\ +\infty & +\infty & 3 & +\infty & +\infty & +\infty & +\infty & +\infty & 3 & 3 \\ +\infty & +\infty & +\infty & 1 & +\infty & +\infty & +\infty & 3 & +\infty & +\infty \\ +\infty & +\infty & +\infty & +\infty & +\infty & +\infty & +\infty & 3 & +\infty & +\infty \end{bmatrix} \tag{29}$$

If x_1 is a valuable target, then the stepping stone dynamics is

$$\begin{cases} x_1[k+1] = x_1[k] = x_1[0] = 0 \\ x_i[k+1] = \min_{j \in N_i}(x_j[k] + \epsilon_{ij}) \qquad \text{if } i \in \{2, \ldots, 10\} \end{cases} \tag{30}$$

The simulation is presented in Table 2 where the initial states are $x_i(0) = 1, \quad \forall i = 2, \ldots, 10,$

		stepping stone cost/iteration						
Source	x_2	3	3	3	3	3	3	3
	x_3	3	4	5	6	6	6	6
	x_4	2	3	4	5	5	5	5
	x_5	2	2	2	2	2	2	2
	x_6	2	3	4	4	4	4	4
	x_7	3	5	5	5	5	5	5
	x_8	4	5	6	7	8	9	9
	x_9	2	3	4	5	6	6	6
	x_{10}	4	7	8	9	10	11	12

Table 2. Stepping stone cost evolution from every source node to the target, where every numerical column shows the stepping stone cost from the respective source, for example, the row labeled with the agent x_8 shows that in the third iteration, its stepping stone cost is 6.

and the most vulnerable stepping stone paths from all the sources (shortest path) to the target have been reached after seven iterations. For example, the most vulnerable stepping stone path from node 10 to node target 1 has cost 12, and there are two paths: $10 \to 8 \to 3 \to 6 \to 5 \to 1$ and $10 \to 8 \to 9 \to 4 \to 6 \to 5 \to 1$. The most vulnerable stepping stone path from node 6 to node target 1 has cost 4 and is unique: $6 \to 5 \to 1$.

Example 3.3.2 (Switching topology)**.** Assume that at any time $t_1 > 0$ a 10-node vulnerability graph has topology $G = G_p$ presented in Fig 4, the *exploit complexity scores* are encoded in its weighted adjacency matrix $M_\epsilon(p) = M_\epsilon$ as presented in equation (28). Assume that at any time $t_2 > t_1$ an attack from host h_6 is detected in host h_5, then the network defense is activated with a switching signal $s(t_2) = q$ yielding a new network topology G_q presented in Fig 5, the new *exploit complexity scores* are encoded in its weighted adjacency matrix $M_\epsilon(q)$ as presented in equation (31), where $\epsilon_{ij}(q) = 9$ if $\epsilon_{ij}(p) = 2$ and $\epsilon_{ij}(q) = \epsilon_{ij}(p)$ if $\epsilon_{ij}(p) \neq 2$.

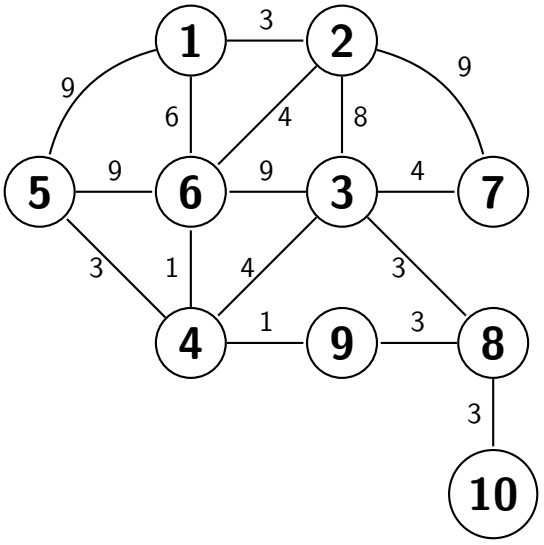

Figure 5. A 10-node vulnerability graph G_q derived from graph G_p in Fig 4 where $\epsilon_{ij}(q) = 9$ if $\epsilon_{ij}(p) = 2$ and $\epsilon_{ij}(q) = \epsilon_{ij}(p)$ if $\epsilon_{ij}(p) \neq 2$, with node 1 as a target node.

$$M_\epsilon(q) = \begin{bmatrix} 0 & 3 & 0 & 0 & 9 & 6 & 0 & 0 & 0 & 0 \\ 3 & 0 & 8 & 0 & 0 & 4 & 9 & 0 & 0 & 0 \\ 0 & 8 & 0 & 4 & 0 & 9 & 4 & 3 & 0 & 0 \\ 0 & 0 & 4 & 0 & 3 & 1 & 0 & 0 & 1 & 0 \\ 9 & 0 & 0 & 3 & 0 & 9 & 0 & 0 & 0 & 0 \\ 6 & 4 & 9 & 1 & 9 & 0 & 0 & 0 & 0 & 0 \\ 0 & 9 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 3 & 3 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \end{bmatrix} \tag{31}$$

The simulation is presented in Table 3 where the initial states are $x(0) = [0 \quad 4 \quad 7 \quad 4 \quad 9 \quad 8 \quad 7 \quad 8 \quad 3 \quad 9]^T$, the graph topology has changed from G_p to G_q in the fifth iteration, that is, for $k = 5$, $t_2 = \delta 5$, then $s[5] = q$. The most vulnerable stepping stone path from all the sources to the target (shortest path) has been reached after nine iterations. For example, the most vulnerable stepping stone path from node 10 to node target 1 has

	stepping stone cost/iteration								
x_2	3	3	3	3	3	3	3	3	3
x_3	44	8	6	6	9	9	11	11	11
x_4	31	5	5	5	5	7	7	7	7
x_5	2	2	2	2	8	8	9	9	9
x_6	6	4	4	4	6	6	6	6	6
x_7	42	5	5	5	10	12	12	12	12
x_8	33	44	11	9	9	9	9	11	11
x_9	41	32	6	6	6	6	8	8	8
x_{10}	83	36	47	14	12	12	12	12	14

Table 3. Stepping stone cost evolution from every source; notice that after the double vertical line, when $k = 5$, the graph topology has changed and the stepping stone cost has been recalculated with the new information.

cost 14 and is unique: $10 \to 8 \to 9 \to 4 \to 6 \to 1$. The most vulnerable stepping stone path from node 6 to node target 1 has cost 6 and is unique: $6 \to 1$.

3.3.4 DISCUSSION

If r is the minimum cost from node i to node j, and s is the minimum cost from node i to node l with $r < s$, equations (20) that are used in the calculation of the shortest path provide r, and the paths themselves are calculated according to remark 2.5.1; that means that the most probable stepping stone attack from the source node i is toward the target node j.

In the model, a path between every source and any target has been assumed, but also if there is a source without a path to any target, the equations (20) converge to the minimum path for all the other sources.

Theorem 3.3.2 provides a time interval $\tau = n - 1$ of instant communication between two consecutive switching signals that ensures the convergence of the system to the minimum path; notice that the convergence to the minimum path is by Corollary 3.3.1 $k \leq n - 1$ instant communications, so the convergence to the minimum path could be observed and detected with the equilibrium condition for $k \leq n - 1$. As was reported in [78] and [5], the Bellman-Ford algorithm can be optimized reducing the iteration in more that $n/2$, this analysis is out of the scope of this chapter and will be studied in a future research.

Chapter 4

MATHEMATICAL MODEL FOR THE ANALYSIS OF STEPPING STONE ATTACK WITH CONSTRAINTS

4.1 INTRODUCTION

In this chapter, a generalization of the model developed in Chapter 3 that attempts to quantify the most vulnerable stepping stone path when the attacker uses more that one criterion in the selection of the stepping stones is proposed. Most cyber attacks involve an attacker launching a multi-stage attack by exploiting a sequence of hosts. This multi-stage attack generates a chain of "stepping stones" from the origin to target. The choice of stepping stones is a function of the degree of exploitability, the impact, attacker's capability, masking origin location, and intent.

In this chapter, we model and analyze scenarios wherein an attacker employs multiple strategies to choose stepping stones. The problem is modeled as an Adjacency Quadratic Shortest Path using dynamic vulnerability graphs with multi-agent dynamic system approach. With this approach, the shortest stepping stone path with maximum node degree and the shortest stepping stone attack with maximum impact are modeled and analyzed. In [56] and [23], the models have assumed that the attackers exert the minimum effort in the selection of the stepping stone attack as a unique criteria, and the most vulnerable stepping stone path is calculated as the shortest path. However, according to the experience attackers are not limited to one criterion in the selection of the stepping stones, for mention a few examples of other criteria in the stepping stone selection we have:

- *Stepping stone path with maximum node degree:* To design an efficient method for searching a specific file in peer-to-peer networks, the so-called *maximum degree strategy* has been proposed [11, 12], this approach is based on the assumption that a node has information on its neighbors' degree. In the context of stepping stone attacks, an

attacker that is resident in a node searching a specific file may move whenever is possible to the neighbor node having the highest degree because the highest-degree nodes are connected to a significant fraction of all nodes in the network, and the attacker would need only a few steps to find a node that is a neighbor of the target. Then, the attacker wishes to minimize the attack cost but also moves to the highest-degree node.

- *Stepping stone path with maximum impact:* Attackers may cause several kinds of damages according to the knowledge they have on organizations configuration and of systems vulnerabilities. In a stepping stone attack, the attacker's damage in every step is according to the knowledge they have on the portion of the network configuration and vulnerabilities that they can see from his current position. Usually, damage evaluation activities are estimated in two ways. First, directly by searching the specific damages caused by the attack on the technological environment, which might be a complex task to perform due to destroyed by the attacker. Second, indirectly by comparing after-attack systems integrity to before-attack integrity; in this approach, a quantification of the damage is performed, focusing on estimating the integrity impacts which is a metric used by security scoring models. An attacker may wish to minimize the attack cost but also look to maximize the impact.

- *Stepping stone path with variance constraint:* The attackers are *risk-averse*, and they will not attack unless their perceived risk to be detected below some threshold [68], in this context, the attacker wishes to minimize the cost and probability of detection simultaneously.

The problems described above involve variants of the classic Shortest Path Problem (SPP) in which additional costs are considered with the presence of pairs of edges in the solution. In other words, the objective function takes into account not only the cost of each edge but also the cost of the interactions among the edges. In the literature, this kind of problem is called a Quadratic Shortest Path Problem (QSPP) [33,34,64,65], because it can be modeled by a quadratic objective function on binary variables associated with each edge [64]. The QSPP is the problem of finding a path in a directed graph from the source vertex a to the target vertex b such that the sum of costs of the edges and the sum of interaction costs overall

distinct pairs of edges on the path is minimized. In stepping stone attacks, an attacker that is resident in a host is *constrained* to exploit vulnerabilities of an adjacent host, that is that the vulnerabilities of the host that are not adjacent to the attackers' current position do not influence the selection of its next stepping stone. This is a variant of the QSPP called *Adjacent QSPP* (AQSPP) [33, 64]. In this chapter, we will develop mathematical models for the analysis and simulation of stepping stone attacks with a maximum degree and with maximum impact, respectively.

To represent *potentially-conflicting criteria* in the selection of the path in every step, multiple edge cost can be introduced, and the cost of the constraints is incorporated into the objective function, yielding an AQSPP problem [64,65]. In this venue, we will introduce more than one score to the same edge for modeling, and analysis of the most vulnerable stepping stone path. In correspondence with the related literature, we will call this problem as *Adjacent Quadratic Stepping Stone Attack*.

The main contribution of this chapter is twofold:

- A mathematical model for the propagation of stepping stone attacks in a dynamic vulnerability graph with multi-agent dynamic system approach when the attacker use more of one criterion in the selection of the stepping stones is proposed as an AQSPP, where an interplay between min-consensus and min-plus algebra is used for modeling and analysis.

- The AQSPP is solved as an SPP, providing necessary and sufficient conditions for finite-time convergence to the shortest path.

The rest of the chapter is organized as follows: sections 4.2 and 4.3 are devoted to developing the model of the adjacent Quadratic stepping stone attack with degree and impact constraints respectively, and Section 4.4 discusses the chapter's results.

4.2 STEPPING STONE PATH WITH MAXIMUM OUT-DEGREE NODE CONSTRAINT

As was mentioned before, according to the *maximum degree strategy* [11,12], the attacker may wish to minimize the attack cost but also move to the highest-degree node. We will model this scenario as follow:

Given the dynamic vulnerability graph (G, x), where $x = [x_1 \quad \cdots \quad x_n]^T$. For every edge (x_i, x_j) we introduce a *node degree complexity score* σ_{ij}^2 defined as

$$\sigma_{ij}^2 = 10 - \frac{10}{n}\delta_{out}(x_j) \tag{32}$$

where $\delta_{out}(x_j)$ is the number of outgoing edges from node x_j, called in the literature as the *out-degree* of the node. σ_{ij}^2 ranges between 0 and 10 and is classified as a node-centrality metric that is proportional to the out-degree of the node x_j, as closer to 0 is this score, as higher out-degree of x_j. Since an attacker that is resident in a node x_i can scan the information of the neighbors' nodes x_j ($j \in N_i$), and because of the maximum degree strategy assumption, he knows the out-degrees of the neighbors' nodes x_l of x_j and may select the one with maximum out-degree. Then in the context of our model, for an attacker that is in node x_i, the following edge score is well known:

$$y_j = \begin{cases} \min_{l \in N_j}(\sigma_{jl}) & \text{if there is} \quad l \in N_j \\ 0 & \text{if } j \text{ is a target} \\ 10 & \text{if there is not} \quad l \in N_j \end{cases} \tag{33}$$

and its interaction over the edge (x_i, x_j) is modeled by the score

$$y_{ij} = \sigma_{ij} y_j \tag{34}$$

Then by (20) the stepping stone dynamic with maximum out-degree constraint influenced by the neighbor's edges (AQSPP) is given by

$$\begin{cases} x_i[k+1] = x_i[k] = x_i[0] = 0 & \text{if } i \in N_l \\ x_i[k+1] = \min_{j \in N_i}(x_j[k] + \epsilon_{ij} + \sigma_{ij}^2 + y_{ij}) & \text{if } i \in N_f \end{cases} \tag{35}$$

Using the the min-plus formalism yields

$$\begin{cases} x_i[k+1] = x_i[k] = x_i[0] = 0 & \text{if } i \in N_l \\ x_i[k+1] = \bigoplus_{j \in N_i} (x_j[k] \otimes \gamma_{ij}) & \text{if } i \in N_f \end{cases} \tag{36}$$

where

$$\gamma_{ij} = \epsilon_{ij} + \sigma_{ij}^2 + y_{ij} \tag{37}$$

is a new complexity score for (x_i, x_j) that combines the properties described by ϵ_{ij} and σ_{ij}^2, but also the influence of the neighbors' paths described by y_{ij}. Let's denote with $\Gamma = [\gamma_{ij}]_{n \times n}$ the MWAM that encodes the complexity scores $\gamma_{i,j}$, then equation (36) in matrix form is

$$x[k+1] = \Gamma \otimes x[k] = \Gamma^{k+1} \otimes x[0] \tag{38}$$

From Theorem 3.3.1, Corollary 3.3.1, and Remark 2.5.1 equation (38) converges to the equilibrium if and only if $\Gamma^{k+1} = \Gamma^k$ for any $k \geq 1$, and the following theorem has been proved.

Theorem 4.2.1. *Equation* (38) *converges to the shortest path with maximum node-degree if and only if* $\Gamma^{k+1} = \Gamma^k$ *for any* $1 \leq k \leq n-1$.

An immediate consequence of Theorem 4.2.1 is stated in the following result.

Corollary 4.2.1. *The AQSPP defined by equation* (32)-(37) *can be solved as the SPP with* (38)

4.3 STEPPING STONE ATTACK WITH MAXIMUM IMPACT CONSTRAINT

For every edge (x_i, x_j) we introduce the *impact complexity score* $\hat{\sigma}_{ij}^2$ defined as

$$\hat{\sigma}_{ij}^2 = 10 - \psi(x_j) \tag{39}$$

where $\psi(x_j)$ is the impact subscore of x_j provided by CVSS, and assuming that every node has information about the vulnerabilities of the neighbors' nodes, then in the context of our model, if an attacker is a resident in a node x_i scanning a neighbors node x_j, because of our assumption, x_j has the vulnerabilities information about its neighbors' nodes x_l, then the scores $\hat{y}_l$ and $\hat{y}_{ij}$ for (x_i, x_j) defined by equations (33) and (34) respectively with $\hat{\sigma}_{ij}$ instead of σ_{ij} are well known for the attacker. Then the shortest path with maximum impact influenced by its adjacent edges is given by equation (36) with the new complexity score $\hat{\gamma}_{ij} = \epsilon_{ij} + \hat{\sigma}_{ij}^2 + \hat{y}_{ij}$ instead of γ_{ij}. Denoting with $\hat{\Gamma} = [\hat{\gamma}_{ij}]_{n\times n}$ the MWAM that encodes the complexity scores $\hat{\gamma}_{ij}$, then the following result can be stated.

Theorem 4.3.1. *Equation* (38) *with* $\hat{\Gamma}$ *instead of* Γ*, converges to the shortest path with maximum impact if and only if* $\hat{\Gamma}^{k+1} = \hat{\Gamma}^k$ *for any* $1 \leq k \leq n-1$.

An immediate consequence of Theorem 4.3.1 is a result identical to Corollary 4.2.1 , with $\hat{\gamma}_{ij}$ instead of γ_{ij}.

Remark 4.3.1. In general, the costs ϵ_{ij} and σ_{ij}^2 (or $\hat{\sigma}_{ij}^2$) can be multiplied by constants α and β respectively, to describe the influence of the constraint in the attackers' strategy, the estimation of this constants may be according to modelers' experience or historical data.

4.4 RESULTS AND DISCUSSION

For an illustration of our approach, a portion of a solar network integrated with an industrial plant control network like that one presented in [25] is considered as a strategic space for attackers potential propagation [75]. Its vulnerability graph is presented in Fig 6. In Table 4, the information of the nodes IDs and its vulnerabilities are presented, the source of the attack is node 12, and the target is node 1 (SCADA server). Table 5 summarizes the simulation using our approach, and are described in more detail as following:

First, using only the exploit complexity score ϵ_{ij} that encode a unique attackers' strategy, the most vulnerable stepping stone path is calculated with the equation (18) as the SPP from the source node 12 to the target, giving as results any path between both of them, because

Node ID	Devise	Vulnerability	ε	A_v	ϵ	ψ	NVD Last Modified
1	SCADA Server	CVE-2010-2772	3.4	0.395	1.4	10	08/16/2017
2	F& P Server	CVE-2008-0405	10	1	0	10	10/15/2018
3	CISCO ASA	CVE-2002-1278	10	1	0	6.4	09/10/2018
4	Active Directory	CVE-1999-0504	10	1	0	6.4	09/09/2008
5	Payment Gateway	CVE-2015-0075	3.9	0.395	0.1	10	10/12/2018
6	Mail Server	CVE-2002-1278	10	1	0	6.4	09/10/2008
7	LAN User	CVE-2017-11783	3.4	0.395	1.4	10	11/03/2017
8	LAN User	CVE-2013-0640	8.6	1	1.4	10	09/18/2017
9	Building Management System	CVE-2012-4701	8.6	1	1.4	10	02/15/2013
10	Solar Farm Inverter	CVE-2017-9859	3.9	1	6.1	5.9	08/21/2017
11	Solar Array Management Module	CVE-2017-9861	3.9	1	6.1	5.9	08/21/2017
12	Attacker	Source					

Table 4. Vulnerabilities of the devices corresponding to the IIOT network presented in Fig 6, where ε is the exploitability sub score, A_v is the accessibility sub score, and ψ is the impact sub score, provided by CVSS, and ϵ is the exploit complexity score.

edge score	Stepping stone path	cost	Impact cost	Out-degree cost
ϵ_{ij}	Any from from 12 to 1	8.89	48.7-55.9	6-10
γ_{ij}	$12 \to 11 \to 9 \to 8 \to 7 \to 2 \to 1$ $12 \to 10 \to 9 \to 8 \to 7 \to 2 \to 1$	104.91	55.9	10
$\hat{\gamma}_{ij}$	$12 \to 11 \to 9 \to 8 \to 7 \to 2 \to 1$ $12 \to 10 \to 9 \to 8 \to 7 \to 2 \to 1$	21.63	55.9	10

Table 5. Simulation results.

all have the same cost 8.89. Some of the shortest paths have a minimum impact cost of 48.7 and others have a maximum of 55.9, the out-degree cost for some of them is 6 (the minimum) and for others is 10 (the maximum). This result is a usual shortcoming of the use of only one edge cost with the shortest path metric; in general, the shortest path metric does not indicate the number of shortest paths that may exist in a network, and in consequence, a network administrator may arrive at an erroneous result.

Second, using the complexity scores γ_{ij}, which encodes two attackers strategies, equation

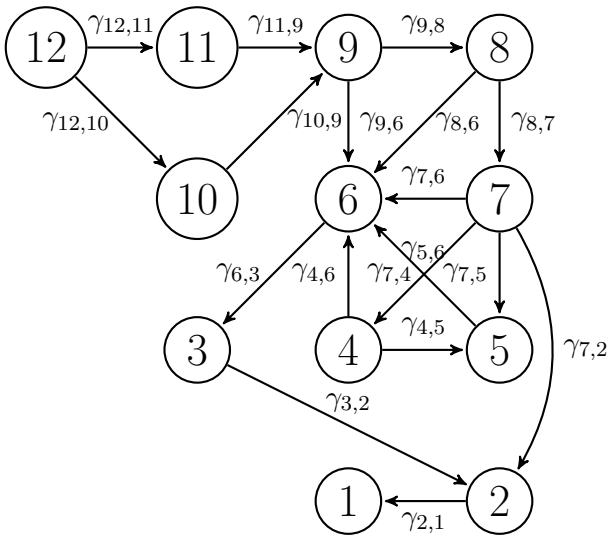

Figure 6. Vulnerability graph of a portion of an Industrial Internet of Things (IIOT) . The edges are labeled with the complexity scores γ_{ij}.

(38) gives us two shortest paths, then the most vulnerable stepping stone path with maximum out-degree node (shortest path with maximum out-degree node) from node 12 to the target are given by $12 \to 11$ (or 10) $\to 9 \to 8 \to 7 \to 2 \to 1$ with cost 104.91, both paths has the maximum out-degree cost of 10, and impact cost of 55.9.

Third, using the exploit complexity score $\hat{\gamma}_{ij}$ that encodes two attackers strategies, equation (38) with $\hat{\Gamma}$ instead of Γ provides two shortest paths, then the most vulnerable stepping stone path with maximum impact (shortest path with maximum impact) from node 12 to the target are $12 \to 11 \to 9 \to 8 \to 7 \to 2 \to 1$ with cost 21.63, out-degree cost of 10, and impact cost of 55.9.

The effort exerted by an attacker to exploit vulnerabilities has been represented by assigning the exploit complexity score ϵ_{ij}. The intuition underlying the shortest path is that from the perspective of the attacker; given the option of different stepping stones, the attacker will choose the series of stepping stones that require the least amount of effort. Resources of an attacker may include but are not limited to, tenacity, skills, and money. Our simulation results described in the second and third part, show that if an attacker is not constrained to the minimum effort (attacker with full resources), that is, if the attacker wants to use more the one criteria in the selection of the stepping stones, the most vulnerable paths may be

different from the paths with only one strategy. More elaborated scores can be introduced to describe more multiple criteria in the selection of the stepping stone by the attackers; our model provides one way to use one strategy and take any other as a constraint.

Chapter 5

A RISK MITIGATION STRATEGY: DUAL REDUNDANT CYBER-ATTACK TOLERANT CONTROL SYSTEMS STRATEGY FOR CYBER-PHYSICAL SYSTEMS

5.1 INTRODUCTION

In the last 20 years, remote cyber-attacks to Cyber-Physical Systems (CPS) have been reported, to mention a few of them:

- In 2010, a group of researchers showed that they could remotely inject messages into the Controller Area Network (CAN bus) of a vehicle and attack the physical components, like killing the engine and affecting braking [17].
- In 2015, the BlackEnergy malware attacked the energy distribution companies of Ukraine. Here the attackers used spear-fishing to gain access to the internal network utilizing a chain of compromised machines (stepping stones) to move laterally to the corporate network and gain access to the SCADA control system to trip breakers in substations [41].
- In 2010, the Stuxnet malware to the Iranian nuclear enrichment plant in 2010 [40], in this case the malware, executed by removable drives use a chain of compromised machines reaching and reprogramming the Programmable Logical devises (PLC).
- Many cyber-attacks vulnerabilities have been reported to Unmanned Aerial Vehicles (UAV) [39], and actual cyber-attacks to UAV have been reported. For example, in 2009, adversaries hacked a *predator* drone feeds in Iraq. In 2011, a US RQ-170 Sentinel UAV was captured in Iran. [28].

Also, since traditional networks are well-protected with stable defensive mechanisms, attackers are now trying to intrude through the weak Internet of Things (IoT) devices to disrupt CPS. In this chapter, embedded controllers that are a sensitive target for stepping stone attacks are considered, and a mitigation plan is proposed. Fig 7 present an example of a network attack scenario to an ICS like one provided in [75] where the PLC and RTU are sensitive targets; the attacks follow a chain of compromised machines.

In this chapter, a cyber-attack tolerant control strategy for embedded controllers in a CPS is presented. A dual redundant control architecture that combines two identical controllers that are switched periodically between active and restart modes is proposed. The strategy is addressed to mitigate the impact due to corruption of the controller software by an adversary. We analyze the impact of the resetting and restarting the controller software and performance of switching process. The minimum requirements in the control design, for effective mitigation of cyber-attacks to the control software, which implies a "fast" switching period, is provided. The simulation results demonstrate the effectiveness of the proposed strategy when the time to fully reset and restart the controller is faster than the time taken by adversary to compromise the controller. The results also provide insights into the stability and safety regions and the factors that determine the effectiveness of the proposed strategy.

CPS are comprised of integrated computational and physical components and processes to support a wide range of applications in military and commercial domains. The ubiquity of CPS systems has attracted attacks on the sensors, actuators, controller, and network components. The state-of-the-art CPS security defense mechanisms are either focused on protecting the physical component (sensors and actuators) or the cyber component (network protocol, control algorithm, communication software). These defenses approach only leverage the properties of physical components or employ IT security defenses to protect the cyber components. There is a need for CPS security defenses to leverage the unique properties of CPS.

Recently, a group of researchers proposed a defense strategy for CPS that takes advantage of physical inertia to improve tolerance to cyber-attacks [4]. The inertia here represents the ability of CPS to stay in motion or at rest for some time of period after partial or total loss of its control input and its resilience to bounded transient imperfections emerging from the

physical component.

The authors in [4] proposed a technique that frequently resets the controller software, thereby maintaining the safety of the CPS. For example, if the proposed system is deployed on a quadcopter, once a cyber-attack is detected, one can reset the controller software and leverage the quadcopter's physical inertia to continue the flight motion until the controller is back online. However, several research questions must be answered to practically deploy the combination of resetting a controller and leveraging the inertia of the physical system. In particular, frequent resets could destabilize the physical operation and cause fatal failures. In addition, by the time the controller is back online, the physical system could fail and cause catastrophic damage. Finally, we need to address the range of impacts a physical system would undergo in the presence of diverse cyber-attacks on the controller software.

In [2], the authors have proposed an attack-tolerant design for embedded control systems, using a proactive reset of a Simplex architecture [70]. In [2], the authors proactively reset and switch between two redundant components; One of them is a complex control component for high performance, possibly unverifiable, which usually generates safe output; the other one a trustworthy control component that has been verified and produces safe inputs all the time. But also, the authors have proposed a safe time window to restart the controllers in order to protect the safety of the physical system. However, the system proposed in [2] is a *switched* system, and the impact of resetting and restarting the controllers on the physical plant must be analyzed in this context. Many research questions under the switching control system approach have not been addressed; for example, the dwell time for the controllers that guarantee the physical plant stability, also, the switched system might become unstable for certain switching signals, even if all the individual subsystems are asymptotically stable [43]. There is a need to leverage the switched system theory to understand the impact of reset and restart controllers in the physical plant.

Inspired by the Stuxnet attack scenario, where the malware had propagated node to node thought the Operational Technology (OT) zones to sabotage the Iranian nuclear enrichment plant by reprogramming the Programmable Logic Controllers (PLC) [40], this chapter leverages the dual redundant fault-tolerant control system designed, the stability theory for switched systems, and the intrinsic inertial of physical components in a CPS in order to

design a cyber-attack tolerant control strategy. The case wherein an embedded controller of a cyber-physical system is hijacked trough a chain of compromised host, and a harmful control signal is injected into the control loop is considered. Fig 7 present an example of propagation of a malware in an Industrial Control System (ICS) [75], and Fig 8 presents an schematic of an attack scenario to an embedded system. In this chapter, a proactive reset of a two redundant controller is proposed as a cyber-attack tolerant control strategy. This chapter is based in [24] that was inspired by the work proposed in [2]; however, the main difference with [2] is that we reset, restart and switch the controllers proactively, and we also provide a formal analysis of the strategy under a switched control system point of view, providing the minimum requirements for the system's architecture that guarantee stability, and acceptable performance of a CPS under cyber-attack.

5.1.1 STATEMENT OF CONTRIBUTION

The chapter contribution is as follows;

- A dual redundant control strategy that proactively switches two identical high-performance controllers is proposed to mitigate the impact of cyber-attacks that corrupt the integrity of the controller software in CPS.
- The system is analyzed from a switched systems point of view.
- The minimum requirements for the control design that guarantee the success of the control strategy are provided.
- The requirements that guarantee stability for any "dwell time" that implies "fast switching" is discussed and presented as a function of the time required for fully resetting and restarting the controller platform.
- Simulations of the proposed strategy are presented that includes optimal calculation of the stability and safety regions.

Maintain the safety of physical components against possible *software faults* is widely studied in the fault-tolerant literature of CPS. Is worthy of mentioning that even with the

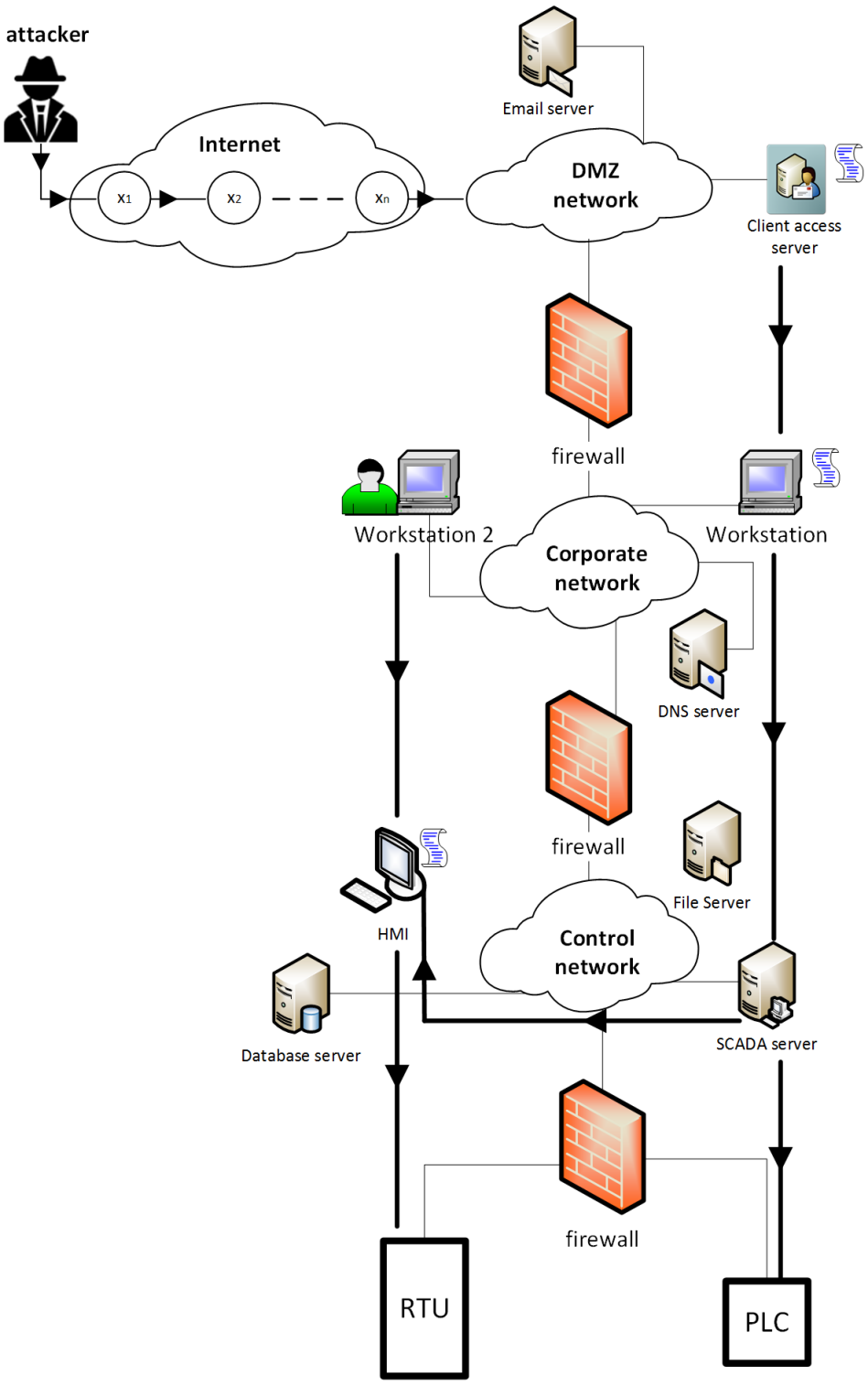

Figure 7. Stepping stone attack propagation in a ICS. The PLC or the RTU may be the sensitive target of the attack

similarities, there are fundamental differences between the approach to protecting a CPS from a fault and from cyber-attacks that we summarize as follow.

First

- In the scope of this dissertation a *fault* refers *software faults* that is a structural imperfection in the software implementation (bugs in the software implementation) that may cause the software to work poorly, produce incorrect results or crash, cascading equivalents result in the physical components of a CPS. The fault-tolerance to this kind of faults is achieved usually by systems based in redundancy like the Simplex architecture approach [10, 63, 69, 70].

- Adverse environments like lighting-charged environment and Hight Intensity Radiation Fields(HIRF) environments caused by Radio Frequency (RF) transmitters, radars, personal electronics devices and electromagnetic incompatibilities equipment installed in CPS may cause functional mode errors or *upsets* in embedded systems. This upset may degrade the performance of embedded systems in CPS [6, 7, 18, 77]. Embedded systems upset is permanent in the sense that it requires corrective action, such as resetting the system or reloading the software. Here the fault tolerance is achieved usually thought systems based in redundancy like the Scalable Processor-Independent Design for Electromagnetic Resilience (SPIDER) [26, 50, 66, 67].

- All the fault-tolerance design assumes that software faults are possible only within a subset of the software system. This assumption is not valid for systems under cyber-attacks. In cyber-attacks, attackers can disrupt all the software layers [2].

- The kind of fault that includes physical issues like broken sensors, actuators etc. is not in the scope of this dissertation.

Second

- A fault is a random process.

- A cyber-attacks is a software designed with a specific purpose to deliver an inaccurate result, degrade, destroy, etc. a particular process or component in a CPS.

Third

- In any fault-tolerant design, the *switching process*, which select and switch the redundant modules, is triggered by a *fault detection* subsystem, like a Decision Module (DM) and a Redundancy Management Units (RMUs) in designs based on the *Simplex* architecture [10, 63, 69, 70] and the *SPIDER* [26, 50, 66, 67] respectively.

- The main difference with our approach with a fault-tolerant system (simplex architecture) and with any other cyber-attack tolerant control system developed previously, is that in our approach, we are proposing a *switching process* that is ***independent*** of the detection of a cyber-attack to the controller software. Our supervisor module switches the redundant controllers *proactively*, *continuously* with the specifications and requirement to accomplish the control objectives, the physical constraint and the security of the physical components of a CPS.

The remaining part of the chapter is devoted discuss the challenges of our approach.

5.1.2 CHAPTER ORGANIZATION

This chapter is organized as follows. Section 5.2 presents the system and attack model. In Section 5.3 an attack-tolerant control strategy is given. In Section 5.4 an analysis and performance of the control strategy are presented, and Section 5.5 presents the validation of the strategy through simulations.

5.2 SYSTEM AND ATTACK MODELS

5.2.1 SYSTEM MODEL

In this chapter, an embedded control system that drives the physical plant in a CPS is considered. This kind of CPS uses an interface to interact with the user or other physical systems. This interfaces may be utilized for transmitting the plant states and sensors values or receiving new set points and operation plans. These processes must be executed in a bounded time, or the system may crash.

5.2.2 ATTACK MODEL

In this research, an adversary that uses a chain of compromised machines to exploit the control software of an embedded controller as a sensitive target in a CPS is considered. The scenario wherein the controller is hijacked and a harmful control signal is injected into the control loop is considered as well (see Fig 8). In this chapter, we are assuming the following attacker capabilities:

- The attacker does not have access to the sensor and actuator, which implies that the values reported by the sensor are trusted and commands sent to the actuators are executed accordingly.
- The attacker requires an external interface to launch an attack on the control software like serial ports or the network.
- The adversary has unlimited access to software resource.

We focus only on the safety and security of the software that provides control over the physical system.

The approach of this chapter does not guarantee the safety of the physical plant if the system is susceptible to sensor jamming attacks. Also, it does not mitigate network attacks such as man-in-the-middle or DoS attacks that restrict network access [1].

5.3 ATTACK-TOLERANT CONTROL STRATEGY

In this chapter a modification of a dual redundant fault-tolerant system, which consists of two identical controllers, called Controller 1 (C_1) and Controller 2 (C_2), which are *periodically switched* between *active mode* and *restart mode*, is used as a basic architecture. A scheme of this architecture is presented in Fig 9. In this paper, we understand that a controller is in *active mode* when it is receiving the sensor's data and generating the output to the actuator in the physical plant. A controller is in *restart mode*, if it has been disconnected from the sensors and actuators, and it is in a reboot process. Here we have considered a full controller reset, and control software reloads as a prophylactic mechanism that ensures that uncorrupted software is running in the controller immediately after finishing the restart

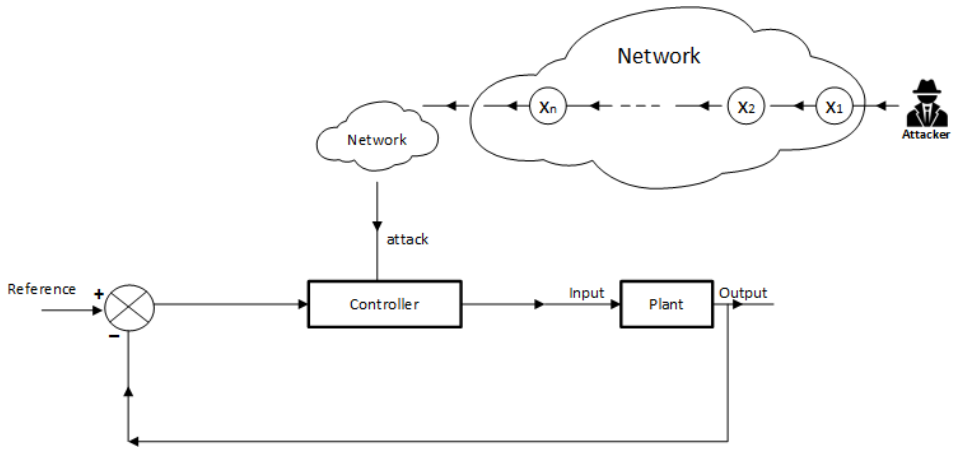

Figure 8. Attack scenario: An adversary that seeks to exploit the control software of an embedded controller in a CPS trough a chain of compromised machines is considered. The attacker does not have access to the sensors and actuators and requires an external interface like the network to lunch an attack to the controller software.

process. We understand that a controller is in a *hot-backup mode*; it is fully operative and ready to be connected to the control system. In the rest of this chapter, when we refer to controllers C_i and C_j we are summing that $i \neq j$.

5.3.1 BASIC STRATEGY

Like the fault-tolerant design, our cyber-attack tolerant control design is a control strategy; its architecture is presented in Fig 9. A description of our approach is given as follows:

1. At any time t, only one of the controller is in active mode and the other one is in restart mode or in hot-back up mode.

2. Every controller is isolated from the other one; it means that there is no intercommunication between both of them. In that way, if an adversary has corrupted the controller software in C_i, there is not chance to spread the corrupter software to C_j.

3. If C_i is in active mode, and C_j has finished its restarts process, that is that C_j is in hot-back up mode, then at time τ a switching signal $\sigma(\tau)$, switch the controllers, that means that C_i is disconnected and starts its reboot process and that C_j is is in active mode. The same strategy is repeated periodically at times $2\tau, 3\tau \cdots$, alternating

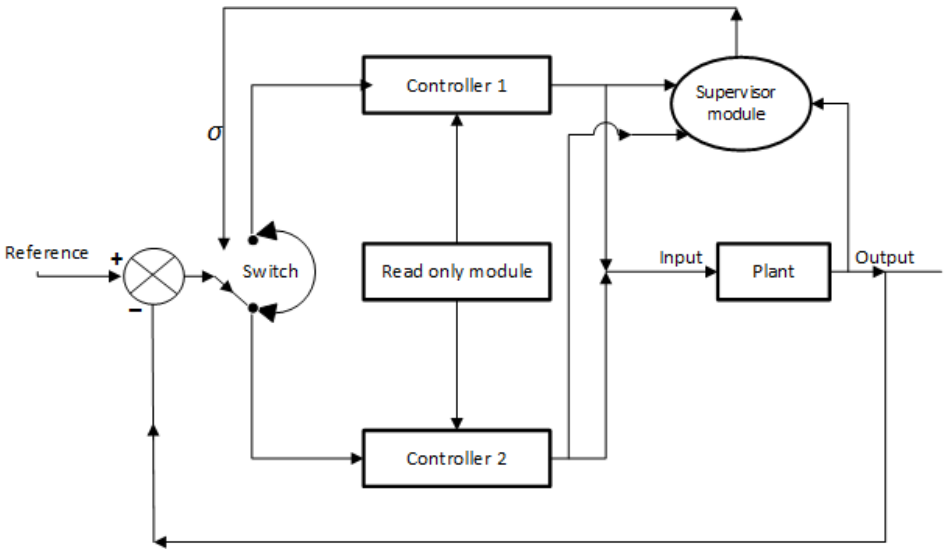

Figure 9. Scheme of a dual redundant cyber-attack tolerant control system. Two identical controllers are switched periodically between active and restart modes. The controller software is loaded to the controller that is in the restart process from a read-only module. An isolate supervisor module coordinates the switching signal .

the operational mode of C_1 and C_2. The time τ is called the *period* of the switching process.

4. The switching process is coordinated by a specifically designed *discrete logic* that is called a *supervisor* module [30, 31, 43] (see Fig 9). The supervisor uses the measurements to generate the switching signal. In our strategy, the supervisor generates a switching signal only if controller C_i is active (generating inputs to the plants), and controller C_j is in hot-back-up mode, and the measurements show that the physical plant is in steady-state. The construction of a logic that commands the switching between controllers is beyond the scope of this paper; for a reader interested in this topic see [30, 31, 43, 51] and the references therein.

According to the experience, only switching and restarting the controllers does not increase the difficulty for the attackers to penetrate the system again. Minimization of the switching period, the integrity of the switching signal, and the secure execution of the restart process are some of the key components of our design that are analyzed in the following subsection.

5.3.2 REQUIREMENTS OF THE BASIC STRATEGY

Let's denote with t_r the time required to restart a controller and with t_a the time necessary for an adversary to compromise the same controller.

- The switching process is only possible if the controller that is not in active mode, has finished its restart process and is in hot-back up mode. The supervisor module switches the controller only if $\tau \geq t_r$.

- If an adversary is attacking the control software of C_i, and C_i is switched to its restart process, then all the attacker's activity is cleaned by resetting C_i. The strategy is valid only if the switching period holds $\tau < t_a$. Then a condition for the cyber-attack tolerant strategy is given by

$$t_r \leq \tau < t_a \tag{40}$$

- Every controller that is in a restart process must be isolated, and the system software must be loaded into memory from a read-only storage unit protecting the system from intruders (see Fig 9), so the software in this process is uncorrupted.

- The switching signal is a critical variable to achieve the security goals of the proposed strategy and needs to be secure and incorruptible, then isolation of the supervisor module is required to protect manipulations of the switching process from adversaries.

The control architecture proposed in this paper is a *switched system*, and the stability and performance of the physical plant must be analyzed in this context. The next section is devoted to this analysis.

5.4 ANALYSIS OF STABILITY AND PERFORMANCE

5.4.1 SWITCHED SYSTEMS

A switched system can be described mathematically by a differential equation of the form

$$\dot{x}(t) = f_{\sigma(t)}(x(t)) \tag{41}$$

where $\sigma : [0, +\inf) \to \mathcal{P} \subset \mathbb{Z}^+$ is a piecewise constant function of time that is called *switching signal* and $\{f_m : m \in \mathcal{P}\}$ is a family of sufficiently regular functions from $\mathbb{R}^n$ to $\mathbb{R}^n$. A particular case where all the individual subsystems are linear is called a *switched linear system* and is given by the equation

$$\dot{x}(t) = A_{\sigma(t)}x(t) \tag{42}$$

The basic architecture proposed in this paper has switched signal

$$\sigma(t) = \begin{cases} 1, & \text{if} \quad n\tau \leq t < (n+1)\tau \quad \text{with } n \text{ even} \\ 2, & \text{if} \quad n\tau \leq t < (n+1)\tau \quad \text{with } n \text{ odd} \end{cases} \tag{43}$$

and can be represented as

$$\dot{x}(t) = \begin{cases} f_1(x(t)), & \text{if} \quad n\tau \leq t < (n+1)\tau \quad \text{with } n \text{ even} \\ f_2(x(t)), & \text{if} \quad n\tau \leq t < (n+1)\tau \quad \text{with } n \text{ odd} \end{cases} \tag{44}$$

In general, a switched system might become unstable for specific switched signals, even if every individual system is asymptotically stable [43]. Is possible to restrict the switching signals to a class of admissible inputs that guaranty stability [32, 43, 53]. According to our approach, we are interested in switch controllers as fast as possible without compromise the system stability. The *Dwell time* is a positive real number τ_D [51], that is defined as the minimum value of the time intervals between consecutive time samples in which switching occurs. It has been shown that sufficient large dwell time can guarantee the stability of the system provided that all individual plants are stable [32]. In this context, we can restrict the switching signals to a class of signals with the property that the interval between any two consecutive switching times is not smaller than τ_D. From those above, it is clear that

our basic strategy is effective if the individual systems f_1 and f_2 are stables and if

$$t_r \leq \tau_D \leq \tau < t_a \tag{45}$$

with τ_D "slow enough". The problem of a " slow enough" dwell time is that a slow switching system may facilitate the adversary objectives, which means that an adversary may have enough time to compromise the controller ($t_a \leq \tau_D \leq \tau$) and destabilize or damage the physical plant. Is clear that a "fast switching system" is required to guarantee inequality (45). Faster switching is discussed in the following subsection.

5.4.2 FAST SWITCHING

For the sequel, let's assume that the process to be controlled is modeled by a linear, time-invariant, stabilizable and detectable system of the form

$$\dot{x} = Ax + Bu, \qquad y = Cx \tag{46}$$

with

$$\text{state constraints:} \quad a_i^T x \leq 1, \quad i = 1, \cdots, p \tag{47}$$

$$\text{control constraints:} \quad b_j^T u \leq 1, \quad j = 1, \cdots, q \tag{48}$$

The state and control constraints represent the physical constraints of the physical system, and usually, they are called *hard constraints* [69, 70]. The set of all the states that holds (47) is called a set of *admissible states*, and the set of all inputs that holds (48) is called a set of *admissible inputs*. The safety of the physical system is concerned with the operation of the system without violating the physical constraints. In this venue, we take as a given a finite family of controllers with the property that, for each $\sigma(t) \in \mathcal{P}$, the feedback interconnection is asymptotically stable. Let then

$$\dot{x} = A_{\sigma(t)} x \tag{49}$$

denote the switched system that results from the $\sigma(t)$th such interconnection.
If the matrices $A_{\sigma(t)}$ commute pairwise, i.e., $A_k A_l = A_l A_k$ for $k, l \in \mathcal{P}$ the switched system (49) is asymptotically stable for any switched signal because every individual $A_\sigma(t)$ in (49) is stable [43].

In [54] and [51], an important observation was made in the sense that, the dwell time $\tau_D > 0$ can be *arbitrarily short*, without sacrificing stability for switching between *finite families* of *identically configured controllers*.

Since in our basic strategy, we are using two identically configured controllers; then in (49) we have to identical closed loop transition matrices $A_1 = A_2$, obviously $A_1 A_2 = A_2 A_1$, then the closed loop system (49) is asymptotically stable for any $\sigma(t) \in \mathcal{P} = \{1, 2\}$ because A_1 and A_2 are both stable [43]. For those above, our basic architecture for the system (46)-(49) is asymptotically stable no matter how small is the dwell time $\tau_D > 0$. In the context of our strategy, since the switching period τ is greater or equal to the dwell time τ_D, then the switching period that guarantees the physical plant stability, depends only on the time t_r required for fully restart the controller, and in consequence, depends on the controller platform, its operative system and reset strategies. The fully restart time for some platforms that range between $45ms$ and $2031ms$ has been reported in [1] and resetting strategies to minimize the fully restart time has been implemented in [4].

Remark 5.4.1. Since the dwell time is not a constraint for the stability of our switched system, in the rest of this paper, we will take the time required for fully restart the controller platform as its dwell time, that is $\tau_D = t_r$

Is well known that the transients caused by switching controllers may not be desired for some systems, to deal with this issue, in [32] the authors showed that this kind of transients could be avoided by suitable choice of the realizations for the controllers.

5.4.3 DIVERSIFICATION

A first assumption is that the application software that executes both controllers is identical; the main problem of this scenario is that if an adversary finds and attacks a vulnerability in the software of one of the controllers, he may exploit the same vulnerability on the other

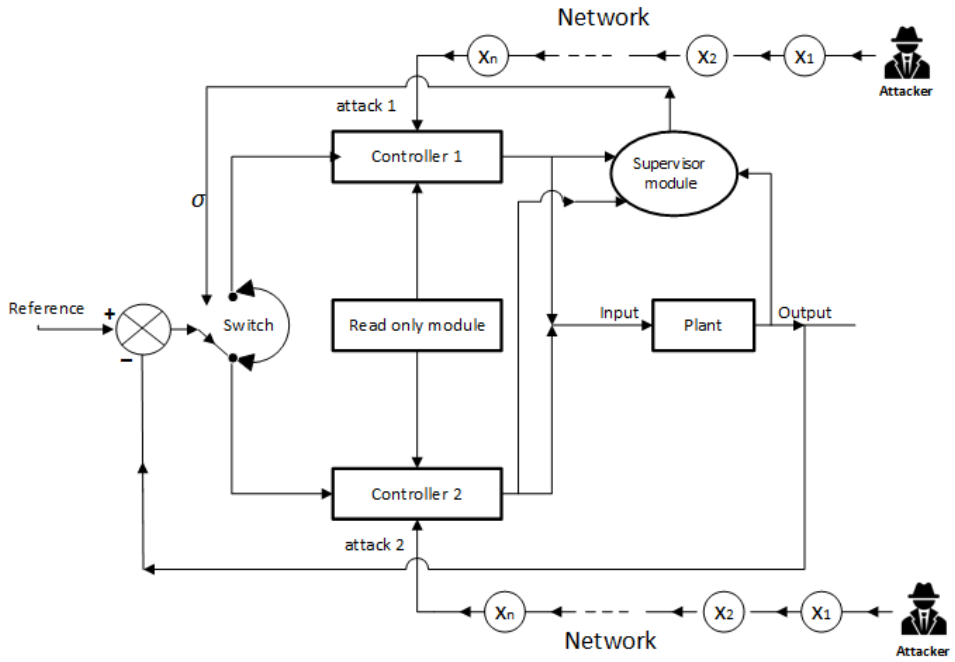

Figure 10. Software and hardware diversification on the controllers may increase the difficulty of compromising a controller by an adversary. The stepping stone cost of attack one may be different from the attack two at least in the last step.

controller faster making more vulnerable our strategy. To increase the tolerance of our switching system; software and hardware for every controller platform must be considered, with the only requirement that the final result is two identical controllers. From the attacker's perspective, the stepping stone path to attack the controllers may be different at least in the last step (see Fig 10). This diversification together with the switching process changes the cost of the stepping stone attack as faster as the switching process.

5.4.4 LIMITATIONS OF THE INERTIA

In the switching process, it is assumed that the loss of a few control cycles is manageable and does not have dramatic consequences to the controlled process [19]. This assumption is justified by the fact that, for some applications, values do not change significantly from one cycle to another, because the physical components have certain *intrinsic inertia*. However, it is necessary to have an upper bound for the time of period that the physical plant can be safe without any active control input. In this venue, let's consider the stabilization problem, with

feedback control in the form $u = -Kx$, where the gain K was designed by any methodology like Linear Quadratic Regulation (LQR) or pole placement, etc. Then the closed loop system is writing as

$$\dot{x} = A_{\sigma(t)}x, \quad \text{with constrains} \quad \alpha_l x \leq 1 \tag{50}$$

where $A_{\sigma(t)} = A - BK$ and $\alpha_l = a_i$, $l = 1, \cdots, p$ and $\alpha_l = -b_j K$, $l = p+1, \cdots, p+q$, and $j = 1, \cdots, q$. According to the Liapunov stability theory, the system (50) is asymptotically stable if and only if there is a positive definite matrix P such that $A^T_{\sigma(t)}P + PA_{\sigma(t)} < 0$, or $QA^T_{\sigma(t)} + A_{\sigma(t)}Q < 0$ with $Q = P^{-1}$. Is possible to show that the stability region of the system (50) is given by the state space set

$$S = \{x : (x - x_c)^T P(x - c_c) \leq 1\}, \tag{51}$$

that is an ellipsoid with a center in x_c, but also, if all estates of S satisfy the physical constraints, then all the trajectories of the systems will satisfy the physical constraints of the system and S is called *safety region* [69, 70]. If the system loses its controller at time $t = l$, then the system dynamic for $t \geq l$ is given by

$$\dot{\tilde{x}}(t) = Ax(t) + E, \tag{52}$$

where $E = B\delta_{el}$ is a constant matrix with δ_{el} denoting the last control input delivered by its active controller, then the evolution of the states is given by the solutions of equation (52), where $x_l = x(l)$ its initial condition, then the time in which the system states are in its safety region is given by the t-values that holds

$$(\tilde{x}(t) - x_c)^T P(\tilde{x}(t) - x_c) < 1 \tag{53}$$

If the control input is back online for some time in winch the left side of (53) is close to

1, it may be that no one command can save the system for its damage or destruction due to the inertia of the physical system. One way to deal with this issue and increase the attack tolerance of our strategy is to decrease the value of the right side of (51) to any value $0 < c < 1$, then a more conservative criterion is given for the t-values that holds

$$(\tilde{x}(t) - x_c)^T P(\tilde{x}(t) - x_c) < c \tag{54}$$

The region (54) is denoted with S_c. The selection of the c-value, depends basically on the intrinsic inertia of the physical plant, and the controller capabilities. If all the trajectories of S_c can be driven for any controller C_i inside of S_c, S_c is called *recoverable region.*

5.5 SIMULATIONS

For the simulation of our strategy, let's consider a constant speed approximation of a Cessna Citation 500 aircraft of the longitudinal linearized dynamics, when it is cruising and altitude of $5000m$ and speed of $128.2m/\sec$ presented in [45]. The elevator angle (rad) is the only input, and the pitch angle (rad), altitude (m), and altitude rate ($m/\sec$) are output. The systems is given by $\dot{x} = Ax + Bu$ with output $y = Cx + Du$, where

$$A = \begin{bmatrix} -1.2822 & 0 & 0.98 & 0 \\ 0 & 0 & 1 & 0 \\ -5.4293 & 0 & -1.8366 & 0 \\ -128.2 & 128.2 & 0 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} -0.3 \\ 0 \\ -17 \\ 0 \end{bmatrix},$$

$$C = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -128.2 & 128.2 & 0 & 0 \end{bmatrix}, \qquad D = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.$$

For the purpose of this example, we will pretend that this is an accurate representation of a UAV aircraft dynamics. The elevator angle is limited to $\pm 15^\circ$ (± 0.262 rad), and the elevator slew rate is limited to $\pm 30^\circ/\sec$ ($\pm 0.524\ rad/\sec$). These are limits imposed by the equipment design, and cannot be exceeded. The pitch angle is limited to $\pm 20^\circ$ (± 0.349 rad).

To implement our cyber-attack tolerant control strategy, two identical Linear Quadratic Regulator (LQR) controllers with gain $K = \begin{bmatrix} 0.9192 & -1.4028 & -0.1659 & -0.0058 \end{bmatrix}$ are used according to the architecture presented in Fig 9. We investigate the effectiveness of our strategy tracking the reference trajectory that is equal to the set point $r = \begin{bmatrix} 0 & 35 & 0 \end{bmatrix}^T$, the state feedback input is given by $u = -Kx$, and since $x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T$ we can write the constraints as

$$
\begin{aligned}
|u| &= |0.9192x_1 - 1.4028x_2 - 0.1659x_3 - 0.0058x_4 + 0.2030| \leq 0.262, \\
|x_2| &\leq 0.349, \\
|\dot{u}| &\leq 0.524.
\end{aligned}
$$

Normal operations: Fig 11 presents the normal operation plant simulated for 10 sec; in this simulation, only one LQR controller has been used.

System under attack: With the purpose of analyzing the sensitivity of the physical system when an adversary compromises the controller software and disrupt the control input, a synthetic attack is simulated. First, with a switching period of 5 sec. Fig 12 presents the scenario where an adversary has compromised the controller software of the active controller at a time $t = 4$ sec, such that the controller delivers a constant input $\delta_e = 10^\circ$ for one second, after that the switched system is activated at time $t = 5$ sec switching the controllers. As the figure shows, the system is stabilized, but in the process, the constraints have been violated, which means that the aircraft has been damaged or destroyed. Second, with a switching period of 4.5 sec. In Fig 13, the switching system is activated after 0.5 sec after the attack has started; in this case, after switching the controllers, the system is stabilized without violation of constraints. In both simulations, the controllers have been switched just once in order to observe the stabilization process after an attack.

Fast switching: Since in our architecture, the dwell time τ_D and consequently the switching period $\tau \geq \tau_D$ can be as faster as the time t_r required for fully restart the controller platform, and assuming that $t_r = 0.1$ sec, we have simulated our strategy for a switching period of $\tau = t_r = 0.1$ sec, the results are presented in Fig 14. In this simulation, the switching process is activated from the very beginning, and as the simulations show, the

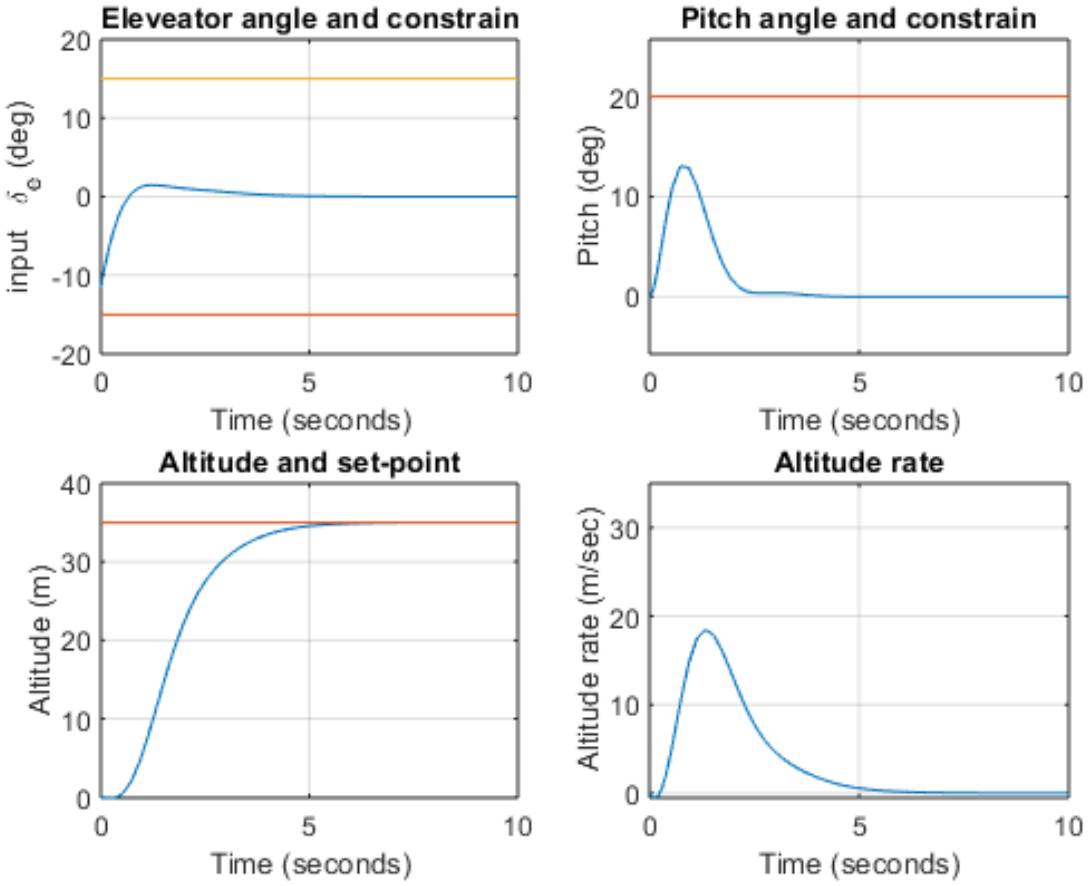

Figure 11. Cessna Citation 500: Altitude stabilization to $35m$ thought of an LQR controller with gain $K = [0.9192 \quad -1.4028 \quad -0.1659 \quad -0.0058]$, the elevator angle (rad) is the only input, and the pitch angle (rad), altitude (m), and altitude rate (m/s) are the outputs.

closed loop system is asymptotically stable as expected due to the controllers being identical, but also there are no transient disturbances due to the switching process, and its performance is comparable to the system performance under only one controller (Fig 11). This behavior is basically because LQR controllers are classified as *robust controllers*, but also because, in our simulation, we are assuming that all the state measurements are available. If not all the state measurements are available, and the estimation of some states are necessary, the transient induced by the switching process may be more notorious. If no one attacker can compromise the controller software in less than 0.1 sec, our system would be resilient to this kind of attack, but also if an attacker can compromise the controller software in less than

0.1 sec, this attack will not last more than 0.1 sec, and as we show in Fig 13, the system will tolerate this kind of attack if it lasts less than 0.5 sec.

Safety region: Since the LQR gain K is given, the closed loop matrix $A_\sigma(t) = A - BK$ is completely determined, and its safety region is given by the ellipsoid $\mathcal{R} = \{x : x^T P x \leq 1\}$ where $P > 0$ and $A_{\sigma(t)}^T P + P A_{\sigma(t)} < 0$ [13,69]. Since the matrices K and P are not uniques, the stability region is not unique, then in order to calculate the maximum time that the physical system is safe without control input, we will maximize the ellipsoid $\{x : x^T P x \leq 1\}$, and since the volume of the ellipsoid $\mathcal{R}$ is proportional to the determinant $det(P)^{-1/2}$, maximizing the volume of the ellipsoid $\mathcal{R}$ is equivalent to minimizing $\det(Q^{-1})$ where $Q = P^{-1}$ [13]. Then we can formulate the following LMI for the optimization;

$$\begin{aligned} minimize \quad & \log[\det(Q^{-1})] && (55)\\ subject \quad to \quad & Q > 0; && (56)\\ & Q A_{\sigma(t)}^T + A_{\sigma(t)} Q < 0; && (57)\\ & \alpha_k^T Q \alpha_k \leq 1, && (58) \end{aligned}$$

where

$$\alpha_1 = \frac{1}{0.349} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix},$$

$$\alpha_2 = -\alpha_1,$$

$$\alpha_3 = \frac{1}{0.524} \begin{bmatrix} 3.5749 \\ -5.3930 \\ -0.4291 \\ -0.0282 \end{bmatrix},$$

$$\alpha_4 = -\alpha_3,$$

$$\alpha_5 = \frac{1}{0.059}\begin{bmatrix} 0.9192 \\ -1.4028 \\ -0.1659 \\ -0.0058 \end{bmatrix},$$

$$\alpha_6 = -\frac{1}{0.465}\begin{bmatrix} 0.9192 \\ -1.4028 \\ -0.1659 \\ -0.0058 \end{bmatrix}$$

that give us $Q^{-1} = P$, with

$$P = \begin{bmatrix} 229.4178 & -262.7811 & -20.7218 & -1.6741 \\ -262.7811 & 312.8974 & 24.2148 & 1.9254 \\ -20.7218 & 24.2148 & 2.2846 & 0.1501 \\ -1.6741 & 1.9254 & 0.1501 & 0.0122 \end{bmatrix} \tag{59}$$

Limitations of inertia: The period that the physical system will be in its stability region holding its state constraints after disconnecting its controller is given by the t-values that hold the inequality (53) with center in $x_c = [0 \quad 0 \quad 0 \quad 35]^T$, and with the optimal matrix P given in equation (59), for example at $t = 4$ the system states are $x(4) = \begin{bmatrix} -0.0122 & 0.0017 & -0.0050 & 33.4933 \end{bmatrix}^T$, if the controller is disconnected and the actuator holds the last control input $u(t) = \delta_{el} = 0.2290°$ (0.0040 rad) for $t > 4$, then the system will be in its safety region for 3.8 sec, and if the actuators do not hold the last input, and by default it is $u(t) = \delta_{el} = 0$ for $t > 4$, then the system will be in its safety region for 7.4 sec.

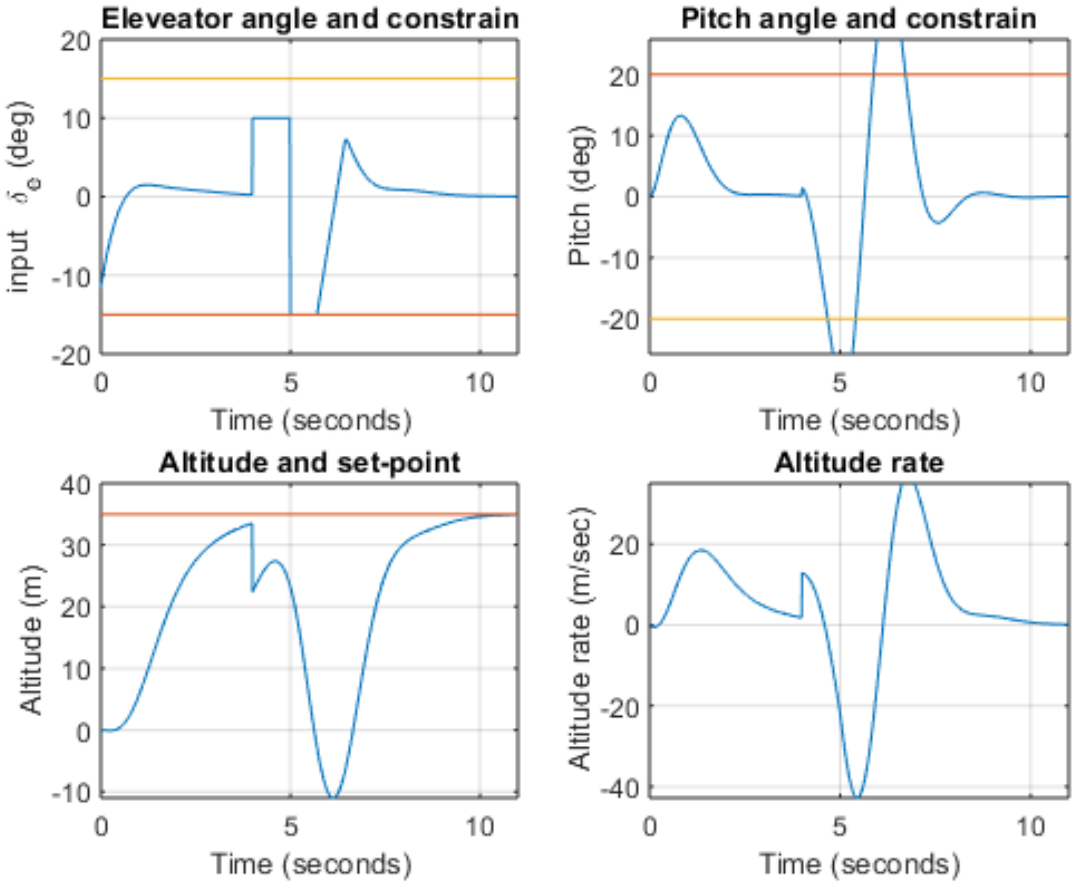

Figure 12. Cessna Citation 500 under attack for 1 sec: Switching period of 5 sec. At time $t = 4$ sec, an adversary that has compromised the control software of the active controller, introduce a constant input to the elevator $\delta_e = 10°$, after one second the controllers are switched. At time $t = 5$ sec, the new active controller delivers an input less than its constraint $-15°$. Since this constraint is a physical limitation of the elevator, it cannot be exceeded, then for a period the elevator angle has a constant value $\delta = -15°$, that means that is this period the system evolves as an open loop, after that the controller drives the system to its set point. The aircraft loses around $10m$ when the attack starts and loses around $40m$ in the next 2 sec. The pitch angle violates its constraints, during and after the attack.

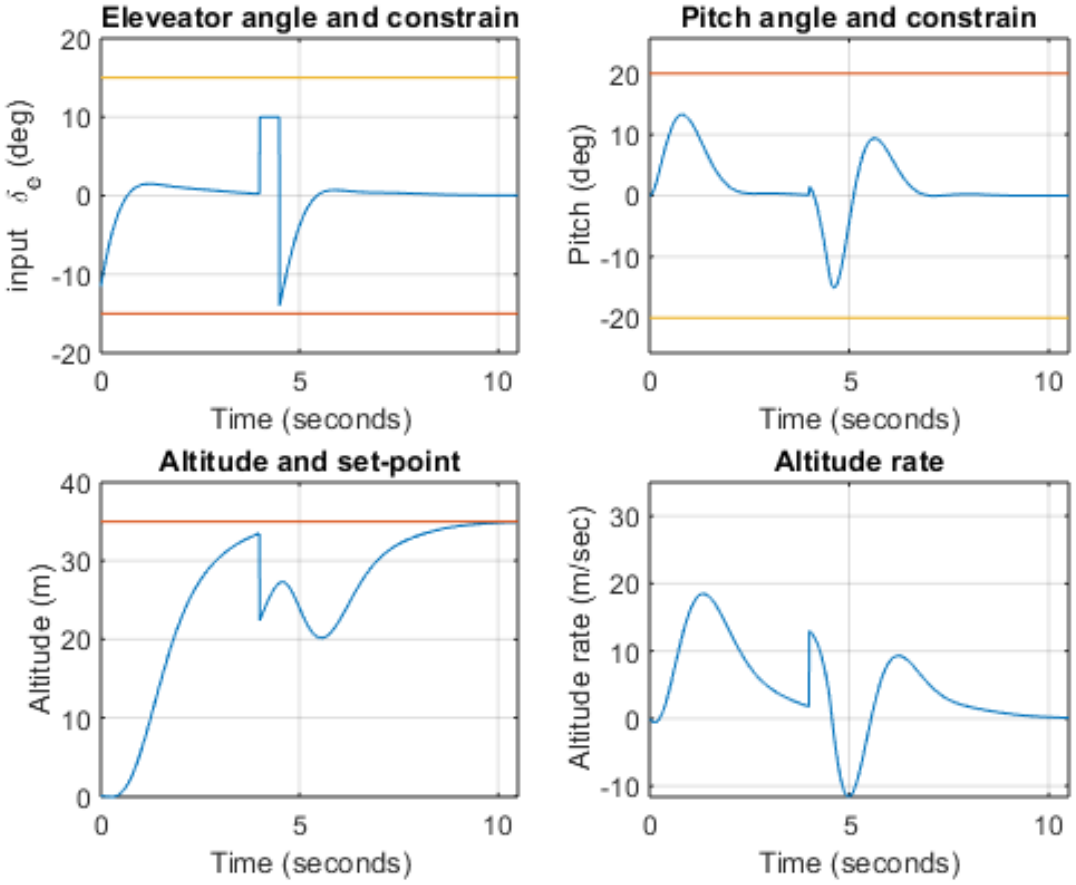

Figure 13. Cessna Citation 500 under attack for 0.5 sec: Switching period of 4.5 sec., at time $t = 4$ sec, an adversary that has compromised the control software of the active controller, introduce a constant input $\delta_e = 10°$, after 0.5 sec the controllers are switched, and the new active controller drives the aircraft to its set point. Even that the transient induced in the outputs by the attacks, do not violate the system constraints, we cannot say that the system has tolerated the attack, because the size and velocity of the transients may cause any permanent damage to the aircraft structure.

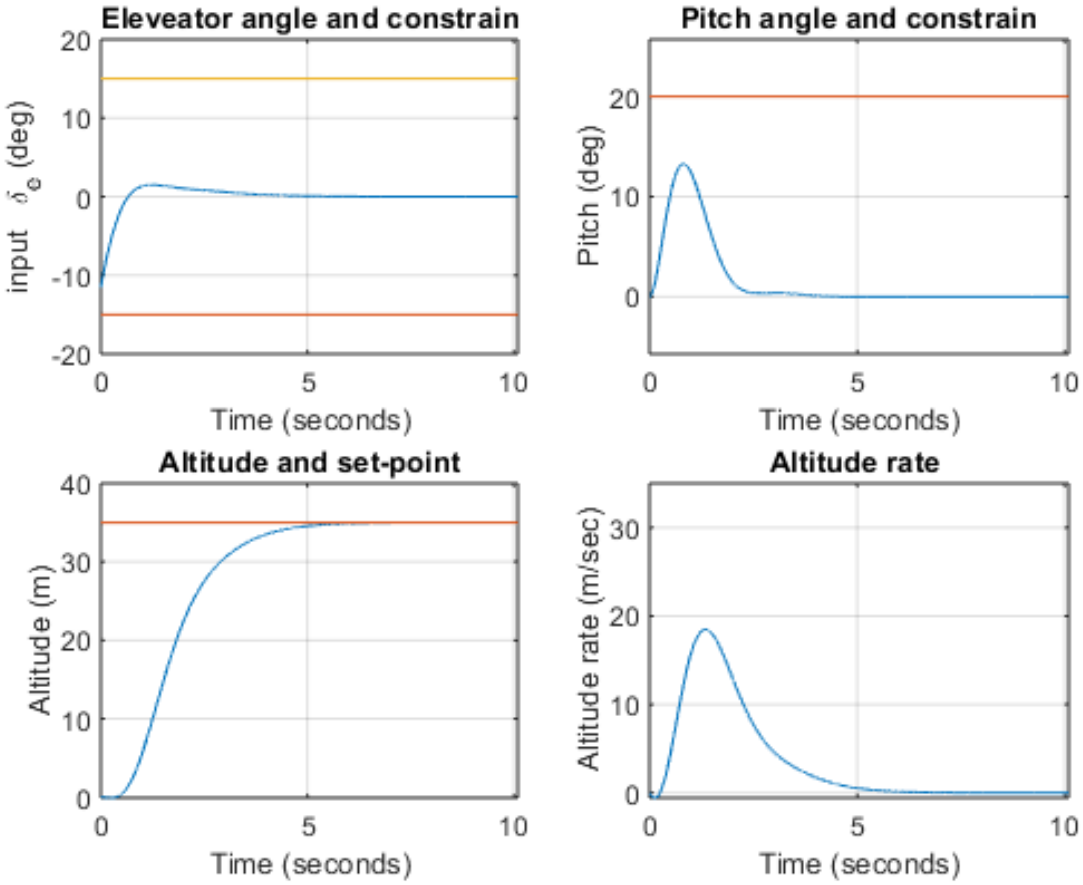

Figure 14. Elevator angle (input), pitch angle, altitude, and altitude rate driven by our dual redundant cyber-attack tolerant switched system with a switching period of 0.1 sec.

Chapter 6

CONCLUSIONS AND FUTURE REASERCH

6.1 CONCLUSIONS

In this dissertation, we have faced the problem of risk assessment and risk mitigation for stepping stone attacks. For the risk assessment problem, we have proposed a formal mathematical model as a multi-agent dynamical system that includes an interplay between graph theory, differential equations, and min-plus algebra. The result of the model is an equation in differences that provides the most vulnerable stepping stone path from an attack source to a sensitive target. Because embedded controllers are omnipresent in networks, as a part of the risk mitigation problem, we have proposed a controls system that is tolerant to cyber-attacks.

The problem of risk assessment of stepping stone attacks in networks with fixed and switching topology is addressed in Chapter 3. In this chapter, the stepping stone cost is modeled as a multi-agent dynamical system in a vulnerability graph with fixed and switching topology. A biased min-consensus protocol is used for distributed calculation of the shortest path, and since the network is monitored in discrete time, the model is discretized using min-plus algebra for modeling and analysis. Theorem 3.3.1 and Corollary 3.3.1 prove that the stepping stone dynamics in a vulnerability graph with fixed topology converge to the shortest path in a finite time given $k = n - 1$ instant communications. Theorem 3.3.2 provides a metric for the time interval between switching signals that guarantees convergence to the minimum path for the switching topology case.

The problem of risk assessment developed in Chapter 3 is expanded in Chapter 4, where we have presented a mathematical model for the analysis of the stepping stone attacks when attackers employ multiple strategies to choose stepping stones. The problem is formulated as an AQSPP in a dynamic vulnerability graph with a multi-agent system approach. This approach allows modeling the scenario when the attacker uses one strategy constrained by

another. As a result, the most vulnerable stepping stone path that satisfies both conditions is calculated, which can be interpreted as the stepping stone path most likely to succeed.

Theorem 4.2.1 and Theorem 4.3.1 provide a necessary and sufficient condition for a finite time convergence to the shortest path for the stepping stone attack models with a maximum degree and maximum impact, respectively, and show that can be solved as the SPP. The models can be expanded quickly for the case of vulnerability graphs with switching topology [23], using equation (22) with $\gamma_{ij}(l)$ or $\hat{\gamma}_{ij}(i)$ instead of $\epsilon_{ij}(l)$ for every vulnerability graph G_l.

The risk mitigation problem has been addressed showing that is possible to design control systems that are tolerant to cyber-attacks. As an example, in this dissertation, a dual redundant controls system strategy for embedded controllers that is tolerant to cyber-attacks has been proposed as a risk mitigation strategy.

In Chapter 5, a dual redundant switching control system has been proposed as a cyber-attack tolerant control system for a plant in a CPS that can be approximated by a linear time-invariant system with the property that is stabilized and detectable. The asymptotic stability of our switched system together with a frequently reset and restart process of the controllers for any positive switching period in our strategy is possible and depends only on time required for fully resetting and restarting the controller platform. The proposed strategy is effective if and only if the fully reset and restart time of the controller is faster than the time required for an adversary to compromise the controller. The simulation results demonstrate that the calculation of the stability and safety regions, the limitations of the inertia, diversification, and performance, depends on the physical plant and the choice of the controller's platforms.

6.2 FUTURE RESEARCH

Different research problems related to this dissertation can be proposed. To mention a few:

1. For the model developed in Chapter 3, Theorem 3.3.2 provides an lower bound for the time interval between switching signals that guarantees convergence to the shortest

path for the switching topology case. In this context, is there a minimum lower bound?, i.e., is there a minimum $k \leq n-1$ that guarantee the convergence to the shortest path?

2. The model develop in Chapter 4 solved the AQSPP, a following question is if this methodology can be expanded to attempt to solve the general QSPP.

3. The models presented in Chapters 3 and 4 will be modified and expanded to analyze problems related to mission impact analysis and moving target defense.

4. We plan to extend our strategy from a dual redundant control system (Chapter 5) to a multiple redundant control system, as long as all the controllers are identical. This may increase the robustness of our strategy, but also can decrease the switching period, for example if we have three controllers C_1, C_2 and C_3 we can organize the system such that when one is in active mode, the second one is hot back up mode and the last one is in restart mode, then the switching period can be less than the time required for fully resetting and restarting the controller platform. Randomly switched mode of operation [16, 46] will be studied in the context of our approach with multiple redundant strategies. We also plan to study the effectiveness of the strategy in the case of nonlinear plants.

5. The diversification in the controllers impact the dwell time and stability. In [51] the author has proved that if the plant is observable and if the transition matrices of the closed loop commute, the switching system is asymptotically stable for any positive dwell time, an equivalent result will be studied for the MIMO case.

BIBLIOGRAPHY

[1] F. Abdi, C. Chen, M. Hasan, S. Liu, S. Mohan, and M. Caccamo, "Restart-based security mechanisms for safety-critical embedded systems," *arXiv preprint arXiv:1705.01520*, 2017.

[2] ——, "Guaranteed physical security with restart-based design for cyber-physical systems," in *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*, ser. ICCPS '18. Piscataway, NJ, USA: IEEE Press, 2018, pp. 10–21. [Online]. Available: https://doi.org/10.1109/ICCPS.2018.00010

[3] P. Ammann, D. Wijesekera, and S. Kaushik, "Scalable, graph-based network vulnerability analysis," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*. ACM, 2002, pp. 217–224.

[4] M. Arroyo, H. Kobayashi, S. Sethumadhavan, and J. Yang, "Fired: Frequent inertial resets with diversification for emerging commodity cyber-physical systems," *arXiv preprint arXiv:1702.06595*, 2017.

[5] M. Bannister and D. Eppstein, "Randomized Speedup of the Bellman-Ford Algorithm," in *Proceedings of the Meeting on Analytic Algorithmics and Combinatorics*, ser. ANALCO '12. Society for Industrial and Applied Mathematics, 2012, pp. 41–47. [Online]. Available: http://dl.acm.org/citation.cfm?id=2790395.2790401

[6] C. M. Belcastro, "Digital system upset. the effects of simulated lightning-induced transients on a general-purpose microprocessor," 1983.

[7] ——, "Closed-loop hirf experiments performed on a fault tolerant flight control computer," in *16th DASC. AIAA/IEEE Digital Avionics Systems Conference. Reflections to the Future. Proceedings*, vol. 1. IEEE, 1997, pp. 4–1.

[8] R. Bellman, "On a routing problem," *Quart. Appl. Math.*, vol. 16, no. 1, pp. 87–90, 1958.

[9] M. Bloem, T. Alpcan, and T. Basar, "Intrusion response as a resource allocation problem," in *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, 2006, pp. 6283–6288.

[10] M. Bodson, J. Lehoczky, R. Rajkumar, L. Sha, and J. Stephan, "Analytic redundancy for software fault-tolerance in hard real-time systems," in *Foundations of Dependable Computing*. Springer, 1994, pp. 183–212.

[11] M. Boguná, R. Pastor-Satorras, A. Díaz-Guilera, and A. Arenas, "Models of social networks based on social distance attachment," *Physical review E*, vol. 70, no. 5, p. 056122, 2004.

[12] S. Bornholdt and H. G. Schuster, *Handbook of graphs and networks: from the genome to the internet*. John Wiley & Sons, 2006.

[13] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. Philadelphia, PA: SIAM, Jun. 1994, vol. 15.

[14] C. Bronk and E. Tikk-Ringas, "Hack or attack? shamoon and the evolution of cyber conflict," 2013.

[15] G. Brown, M. Carlyle, J. Salmerón, and K. Wood, "Defending critical infrastructure," *Interfaces*, vol. 36, no. 6, pp. 530–544, 2006.

[16] J. R. Chávez-Fuentes, O. R. González, and W. S. Gray, "Performance analysis of fault tolerant control systems with iid upsets," in *Proceedings of the 2010 American Control Conference*. IEEE, 2010, pp. 6197–6204.

[17] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, T. Kohno *et al.*, "Comprehensive experimental analyses of automotive attack surfaces." in *USENIX Security Symposium*, vol. 4. San Francisco, 2011, pp. 447–462.

[18] B. Clough, "Effects of electromagnetic interference on digital control systems," *Wright-Patterson AFB, Dayton, OH, Internal Report WL-TR-96-3122*, 1996.

[19] J. C. Cunha, R. Maia, M. Z. Rela, and J. G. Silva, "A study of failure models in feedback control systems," in *2001 International Conference on Dependable Systems and Networks*, July 2001, pp. 314–323.

[20] M. Dacier, Y. Deswarte, and M. Kaâniche, "Quantitative assessment of operational security: Models and tools," *Information Systems Security, ed. by SK Katsikas and D. Gritzalis, London, Chapman & Hall*, pp. 179–86, 1996.

[21] R. Dantu and P. Kolan, "Risk management using behavior based bayesian networks," in *International Conference on Intelligence and Security Informatics*. Springer, 2005, pp. 115–126.

[22] G. C. F.Baccelli, G. Olsder, and J. Quadrat, *Synchronization and Linearity: An algebra for discrete event systems*. http://www-rocq.inria.fr/metalau/cohen/DES/book-online.html: Wiley, Web edition., 2001.

[23] M. Gamarra, S. Shetty, D. M. Nicol, O. Gonazlez, C. A. Kamhoua, and L. Njilla, "Analysis of stepping stone attacks in dynamic vulnerability graphs," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, pp. 1–7.

[24] M. A. Gamarra, S. Shetty, O. R. Gonzalez, L. Njilla, M. Pendleton, and C. Kamhoua, "Dual redundant cyber-attack tolerant control systems strategy for cyber-physical systems," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, May 2019, pp. 1–7.

[25] G. George and S. M. Thampi, "A graph-based security framework for securing industrial iot networks from vulnerability exploitations," *IEEE Access*, vol. 6, pp. 43 586–43 601, 2018.

[26] A. Geser, P. S. Miner, V. Carreno, C. Munoz, and S. Tahar, "A formal correctness proof of the spider diagnosis protocol," in *NASA CONFERENCE PUBLICATION*. NASA; 1998, 2002, pp. 71–86.

[27] 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.

[28] K. Hartmann and K. Giles, "Uav exploitation: A new domain for cyber power," in *2016 8th International Conference on Cyber Conflict (CyCon)*. IEEE, 2016, pp. 205–221.

[29] 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.

[30] J. P. Hespanha, "Tutorial on Supervisory Control," in *Lecture Notes for the workshop Control using Logic and Switching for the 40th Conf. on Decision and Contr., Orlando, Florida*, 2001.

[31] ——, "Stabilization through hybrid control," in *Encyclopedia of Life Support Systems (EOLSS)*. Oxford, UK: Developed under the Auspices of the UNESCO, Eolss Publishers, 2004, vol. Control Systems, Robotics, and Automation.

[32] J. P. Hespanha and A. S. Morse, "Switching between stabilizing controllers," *Automatica*, vol. 38, no. 11, pp. 1905–1917, Nov. 2002. [Online]. Available: http://dx.doi.org/10.1016/S0005-1098(02)00139-5

[33] H. Hu and R. Sotirov, "On solving the quadratic shortest path problem," *arXiv preprint arXiv:1708.06580*, 2017.

[34] ——, "Special cases of the quadratic shortest path problem," *Journal of Combinatorial Optimization*, vol. 35, no. 3, pp. 754–777, 2018.

[35] M. Hung, "Leading the IoT, Gartner Insights on How to Lead in a Connected World," *Gartner Research*, pp. 1–29, 2017.

[36] K. Ingols, R. Lippmann, and K. Piwowarski, "Practical attack graph generation for network defense," in *2006 22nd Annual Computer Security Applications Conference (ACSAC'06)*. IEEE, 2006, pp. 121–130.

[37] E. Jonsson and T. Olovsson, "A quantitative model of the security intrusion process based on attacker behavior," *IEEE Transactions on Software Engineering*, vol. 23, no. 4, pp. 235–245, 1997.

[38] 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.

[39] A. Kim, B. Wampler, J. Goppert, I. Hwang, and H. Aldridge, "Cyber attack vulnerabilities analysis for unmanned aerial vehicles," in *Infotech@ Aerospace 2012*, 2012, p. 2438.

[40] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Security & Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[41] R. M. Lee, M. J. Assante, and T. Conway, "Analysis of the cyber attack on the ukrainian power grid: Defense use case," *SANS ICS*, 2016.

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

[43] D. Liberzon and A. S. Morse, "Basic problems in stability and design of switched systems," *IEEE Control Systems*, vol. 19, no. 5, pp. 59–70, Oct 1999.

[44] R. P. Lippmann and K. W. Ingols, "An annotated review of past papers on attack graphs," MASSACHUSETTS INST OF TECH LEXINGTON LINCOLN LAB, Tech. Rep., 2005.

[45] J. Maciejowski, *Predictive control: with constraints*. Pearson education, 2002.

[46] M. Mariton, *Jump linear systems in automatic control*. M. Dekker New York, 1990.

[47] C. Meadows, R. Wright, and P. Neumann, "A representation of protocol attacks for risk assessment," in *Proceedings of the DIMACS Workshop on Network Threats*, 1998, pp. 1–10.

[48] P. Mell, K. Scarfone, and S. Romanosky, "The common vulnerability scoring system (cvss) and its applicability to federal agency systems," National Institute of Standards and Technology, Tech. Rep., 2007.

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

[50] P. Miner, "Analysis of the spider fault-tolerance protocols," in *Proceedings of the 5th NASA Langley Formal Methods Workshop*, 2000.

[51] A. S. Morse, "Supervisory control of families of linear set-point controllers part 1: Exact matching," *IEEE Transactions on Automatic Control*, vol. 41, no. 10, pp. 1413–1431, Oct 1996.

[52] I. S. Moskowitz and M. H. Kang, "An insecurity flow model," in *NSPW*, vol. 97. Citeseer, 1997, pp. 61–74.

[53] E. Moulay, R. Bourdais, and W. Perruquetti, "Stabilization of nonlinear switched systems using control lyapunov functions," *Nonlinear Analysis: Hybrid Systems*, vol. 1, no. 4, pp. 482 – 490, 2007, proceedings of the International Conference on Hybrid Systems and Applications, Lafayette, LA, USA, May 2006: Part I. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S175157\0X06000264

[54] K. S. Narendra and J. Balakrishnan, "Improving transient response of adaptive control systems using multiple models and switching," *IEEE Transactions on Automatic Control*, vol. 39, no. 9, pp. 1861–1866, Sept 1994.

[55] B. M. Nejad, S. A. Attia, and J. Raisch, "Max-Consensus in a Max-Plus Algebraic Setting: The Case of Switching Communication Topologies," *IFAC Proceedings Volumes*, vol. 43, no. 12, pp. 173–180, 2010, 10th IFAC Workshop on Discrete

Event Systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S147466\7015324526

[56] D. M. Nicol and V. Mallapura, "Modeling and analysis of stepping stone attacks," in *Proceedings of the Winter Simulation Conference 2014*, Dec 2014, pp. 3036–3047.

[57] S. Noel, S. Jajodia, B. O'Berry, and M. Jacobs, "Efficient minimum-cost network hardening via exploit dependency graphs," in *19th Annual Computer Security Applications Conference, 2003. Proceedings.* IEEE, 2003, pp. 86–95.

[58] R. Olfati-Saber and R. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *Automatic Control, IEEE Transactions on*, vol. 49, no. 9, pp. 1520–1533, Sept 2004.

[59] R. Ortalo, Y. Deswarte, and M. Kaâniche, "Experimenting with quantitative evaluation tools for monitoring operational security," *IEEE Transactions on Software Engineering*, vol. 25, no. 5, pp. 633–650, 1999.

[60] X. Ou, W. F. Boyer, and M. A. McQueen, "A scalable approach to attack graph generation," in *Proceedings of the 13th ACM conference on Computer and communications security.* ACM, 2006, pp. 336–345.

[61] C. Phillips and L. P. Swiler, "A graph-based system for network-vulnerability analysis," in *Proceedings of the 1998 Workshop on New Security Paradigms*, ser. NSPW '98. New York, NY, USA: ACM, 1998, pp. 71–79. [Online]. Available: http://doi.acm.org/10.1145/310889.310919

[62] P. A. Ralston, J. H. Graham, and J. L. Hieb, "Cyber security risk assessment for scada and dcs networks," *ISA transactions*, vol. 46, no. 4, pp. 583–594, 2007.

[63] J. G. Rivera, A. A. Danylyszyn, C. B. Weinstock, L. R. Sha, and M. J. Gagliardi, "An architectural description of the simplex architecture." CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST, Tech. Rep., 1996.

[64] B. Rostami, A. Chassein, M. Hopf, D. Frey, C. Buchheim, F. Malucelli, and M. Goerigk, "The quadratic shortest path problem: complexity, approximability, and solution methods," *European Journal of Operational Research*, vol. 268, no. 2, pp. 473–485, 2018.

[65] B. Rostami, F. Malucelli, D. Frey, and C. Buchheim, "On the Quadratic Shortest Path Problem," in *14th International Symposium on Experimental Algorithms*, ser. 14th International Symposium on Experimental Algorithms, Paris, France, Jun. 2015. [Online]. Available: https://hal.inria.fr/hal-01251438

[66] J. Rushby, "Bus architectures for safety-critical embedded systems," in *International Workshop on Embedded Software*. Springer, 2001, pp. 306–323.

[67] ——, "A comparison of bus architecture for safety-critical embedded systems," *NASA/CR, Tech. Rep. NASA/CR-2003-212161*, 2003.

[68] G. Schudel and B. Wood, "Adversary work factor as a metric for information assurance," in *Proceedings of the 2000 workshop on New security paradigms*. ACM, 2001, pp. 23–30.

[69] D. Seto and L. Sha, "An Engineering Method for Safety Region Development," Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU/SEI-99-TR-018, 1999. [Online]. Available: http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=13483

[70] L. Sha, "Using simplicity to control complexity," *IEEE Software*, vol. 18, no. 4, pp. 20–28, Jul 2001.

[71] O. Sheyner, J. Haines, S. Jha, R. Lippmann, and J. M. Wing, "Automated generation and analysis of attack graphs," in *Proceedings 2002 IEEE Symposium on Security and Privacy*. IEEE, 2002, pp. 273–284.

[72] R. A. SIVAKUMAR and R. BATTA, "The variance-constrained shortest path problem," *Transportation Science*, vol. 28, no. 4, pp. 309–316, 1994. [Online]. Available: http://www.jstor.org/stable/25768653

[73] M. Tambe, *Security and game theory: algorithms, deployed systems, lessons learned*. Cambridge university press, 2011.

[74] C.-W. Ten, G. Manimaran, and C.-C. Liu, "Cybersecurity for critical infrastructures: Attack and defense modeling," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 40, no. 4, pp. 853–865, 2010.

[75] S. Ullah, S. Shetty, and A. Hassanzadeh, "Towards modeling attackers opportunity for improving cyber resilience in energy delivery systems," in *2018 Resilience Week (RWS)*. IEEE, 2018, pp. 100–107.

[76] H. Wang, Z. Chen, J. Zhao, X. Di, and D. Liu, "A vulnerability assessment method in industrial internet of things based on attack graph and maximum flow," *IEEE Access*, vol. 6, pp. 8599–8609, 2018.

[77] R. Wang, W. S. Gray, O. R. Gonzalez, and J. R. Chavez-Fuentes, "Tracking performance of distributed recoverable flight control systems subject to high intensity radiated fields," *IEEE Transactions on Aerospace and Electronic systems*, vol. 49, no. 1, pp. 521–542, 2013.

[78] S. Watanabe and Y. Watanabe, "Min-plus algebra and networks," in *Novel Development of Nonlinear Discrete Integrable Systems*. RIMS Kôkyûroku Bessatsu B 47, 2014.

[79] J. Y. Yen, "An Algorithm for Finding Shortest Routes from All Source Nodes to a Given Destination in General Networks," *Quart. Applied Math*, vol. 27, pp. 526–530, 1970.

[80] Y. Zhang and S. Li, "Distributed biased min-consensus with applications to shortest path planning," *IEEE Transactions on Automatic Control*, vol. 62, no. 10, pp. 5429–5436, Oct 2017.

VITA

Marco Antonio Gamarra
Department of Electrical & Computer Engineering
Old Dominion University
Norfolk, VA 23529

Education

- M.Sc. Mathematics, Florida International University, Miami, Fl, 2011.
- B.Sc. Physics and Mathematics, Universidad Nacional San Antonio Abad del Cusco, Cusco, Perú, 1997.

Publications

1. M. Gamarra, S. Shetty, O. Gonzalez and L. Njilla, "Modeling Stepping Stone Attacks with Constraints in Cyber Infrastructure," 2019 IEEE Global Communication Conference (GLOBECOM), December 2019.
2. M. Gamarra, S. Shetty, O. Gonzalez, D. M. Nicol, C. A. Kamhoua and L. Njilla, "Analysis of Stepping Stone Attacks in Internet of Things using Dynamic Vulnerability Graphs". In Modeling and Design of Secure Internet of Thinks, ed., by C.A. Kamhoua, L. Njilla, A. Kott, S. Shetty. Wiley-IEEE Press. Under review.
3. M. Gamarra, S. Shetty, O. Gonzalez, L. Njilla, M. Pendelton and C. A. Kamhoua "Dual redundant cyber-attack tolerant control system strategy for cyber-physical systems," 2019 IEEE International Conference on Communications (ICC) May 2019.
4. M. Gamarra, S. Shetty, D. M. Nicol, O. Gonzalez, C. A. Kamhoua and L. Njilla, "Analysis of Stepping Stone Attacks in Dynamic Vulnerability Graphs," 2018 IEEE International Conference on Communications (ICC), May 2018.

Typeset using LaTeX.