Cybersecurity for Urban Critical Infrastructure

by

Gregory J. Falco

B.S. Cornell University
M.S. Columbia University

Submitted to the Department of Urban Studies and Planning
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Cybersecurity,
Urban Science and
Infrastructure Management

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2018

© Massachusetts Institute of Technology 2018. All rights reserved.

Signature redacted

Author .................................................................

Department of Urban Studies and Planning

May 8, 2018

Signature redacted

Certified by .............................

Lawrence Susskind
Ford Professor of Urban and Environmental Planning
Dissertation Supervisor

Signature redacted

Accepted by .................................

Lawrence Vale
Chair, PhD Committee
Department of Urban Studies and Planning
Cybersecurity for Urban Critical Infrastructure

by

Gregory J. Falco

Submitted to the Department of Urban Studies and Planning
on May 8, 2018, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Cybersecurity,
Urban Science and
Infrastructure Management

Abstract

Our cities are under attack. Urban critical infrastructure which includes the electric grid, water networks, transportation systems and public health and safety services are constantly being targeted by cyberattacks. Urban critical infrastructure has been increasingly connected to the internet for the purpose of operational convenience and efficiency as part of the growing Industrial Internet of Things (IIoT). Unfortunately, when deciding to connect these systems, their cybersecurity was not taken seriously. A hacker can monitor, access and change these systems at their discretion because of the infrastructure's lack of security. This is not only a matter of potential inconvenience. Digital manipulation of these devices can have devastating physical consequences. This dissertation describes three steps cities should take to prepare for cyberattacks and defend themselves accordingly.

First, cities must understand how an attacker might compromise its critical infrastructure. In the first chapter, I describe and demonstrate a methodology for enumerating attack vectors across a city's CCTV security system. The attack methodology uses established cybersecurity typologies to develop an attack ruleset for an AI planner that was programmed to perform attack generation. With this, cities can automatically determine all possible approaches hackers can take to compromise their critical infrastructure.

Second, cities need to prioritize their cyber risks. There are hundreds of attack permutations for a given system and thousands for a city. In the second chapter, I develop a risk model for urban critical infrastructure. The model helps prioritize vulnerabilities that are frequently exploited for IIoT Supervisory Control and Data Acquisition (SCADA) systems.

Finally, cities need tools to defend themselves. In the third chapter, I present a non-technical approach to defending against attacks called cyber negotiation. Cyber negotiation is one of several non-technical cyberdefense tools I call Defensive Social Engineering, where victims can use social engineering against the hacker. Cyber negotiation involves using a negotiation framework to defend against attacks with steps urban critical infrastructure operators can take before, during and after an attack.

This study combines computer science and urban planning (Urban Science) to provide a starting point for cities to prepare for and protect themselves against cyberattacks.
Committee Member and Computer Science Research Advisor: Howard Shrobe
Title: Principal Research Scientist and Director of CyberSecurity@CSAIL

Committee Member: Stuart Madnick
Title: John Norris Maguire Professor of Information Technologies

Dissertation Supervisor: Lawrence Susskind
Title: Ford Professor of Urban and Environmental Planning
Acknowledgments

This dissertation, which will hopefully help to establish a new direction in computer science and urban planning research, is only possible because of the support from the below individuals for whom it is impossible for me to fully express my admiration and appreciation.

I would like to begin by thanking both my departmental advisor and committee chair in the Department of Urban Studies and Planning (DUSP), Larry Susskind and my research advisor and committee member at the Computer Science and Artificial Intelligence Laboratory (CSAIL), Howie Shrobe.

Larry, thank you for seeing the potential of my ideas and enthusiasm before I was capable of conveying these clearly to the broader department. Without you, I would not have been able carve out and draw attention to this important part of the planning discipline. Together we are building something very exciting. You have taught me all that I know about being a research scholar and I will always look to you as my mentor as I build my career.

Howie, thank you for taking a chance on me and seeing my potential to do interesting and important work that bridges industry and academic research in cybersecurity. Many were unable to see the value I could add to a research group considering my different background. Supporting me as an RA throughout my PhD and teaching me everything I know about cybersecurity has changed the course of my future and I’ll be forever grateful. Also, your selfless support of my startup has been instrumental to my development as an entrepreneur. I look forward to working alongside you in the years ahead.

Stuart Madnick and David Hsu have also been pivotal in reviewing and shaping my research agenda. Thank you both for all the guidance, pep talks, career conversations and extensive reviews for this important research. I really appreciate it.

Carlos Caldera and Alicia Noriega, I am extremely grateful for your research assistance for various components of this dissertation. Thank you so much for your efforts to help me think through this compendium of knowledge and work alongside me during the highs and lows of the research.

Mom, thank you for being up to doing absolutely anything with or for me, running the DC and for your help with my new family. Jonny, thanks for being a cocopod, contributing
to the world and thriving during my PhD.

Frida, you always believe in me and uprooted your life to join my peripatetic wild goose chase. Thank you for encouraging my never-ending pursuit of knowledge, sharing my love of learning and being there for me to come home to.

Milo, you’re my best friend. You have saved my sanity more times than one can count. You will always be my favorite little research assistant that reads all the books in Larry’s office or organizes the mugs and pillows in Howie’s after riding on the lab’s digger.
# Contents

1 Introduction ........................................... 11
   1.1 Industrial Control System Cybersecurity Threat Landscape .................. 13
   1.2 Urban Critical Infrastructure Risk Prioritization ............................. 14
   1.3 Cyberdefense Strategies ......................................................... 16
   1.4 Audience ................................................................. 17

2 A Master Attack Methodology for an AI-Based Automated Attack Planner for Smart Cities ........................................... 19
   2.1 Introduction ............................................................. 19
   2.2 Background ............................................................ 21
      2.2.1 Attack Tree Features ............................................... 21
      2.2.2 Benefits of Attack Trees ......................................... 23
      2.2.3 Challenges of Attack Trees ....................................... 24
      2.2.4 Current State ...................................................... 25
   2.3 Scalable, Cross-Sector Attack Tree Design .................................... 26
      2.3.1 Sequence of Phases for Waging Attacks ................................ 27
      2.3.2 Surface Area for Waging an Attack .................................. 29
      2.3.3 Actions Required for Waging an Attack ............................. 32
      2.3.4 Tools Needed for Waging an Attack .................................. 34
      2.3.5 Design Summary .................................................... 34
   2.4 Example Use Case ....................................................... 35
      2.4.1 Automated Attack Tree with Master Attack Methodology ............ 35
      2.4.2 Attack Rule Set ..................................................... 36
2.4.3 Manual Attack Tree ........................................ 38
2.4.4 Comparative Discussion ................................... 39
2.5 Future Work ....................................................... 40
2.6 Conclusion ........................................................ 41

3 IIoT Cybersecurity Risk Modeling for SCADA Systems 42
3.1 Introduction ....................................................... 42
  3.1.1 Problem Statement ......................................... 42
  3.1.2 SCADA IIoT Overview .................................... 43
  3.1.3 The National Policy & Regulatory Landscape .......... 44
  3.1.4 Vulnerability Identification and Classification ....... 44
3.2 Our Contribution ................................................ 47
  3.2.1 Experimental Findings ................................... 47
3.3 Methodology ..................................................... 49
  3.3.1 Data Collection ............................................ 49
  3.3.2 Analysis .................................................... 52
  3.3.3 Scoring ..................................................... 58
3.4 Research Implications ........................................ 61
  3.4.1 Operator Implications .................................... 61
  3.4.2 Technical Design Implications ......................... 62
3.5 Conclusion ........................................................ 63

4 Cyber Negotiation: Defending Urban Critical Infrastructure from Cyber-attacks by Negotiating with Cyberterrorists 65
4.1 Introduction ....................................................... 65
4.2 Background ........................................................ 67
  4.2.1 The Importance of Cyber Defense ...................... 67
  4.2.2 The Nature of Ransomware ............................... 70
  4.2.3 The Opportunity To Negotiate ......................... 74
4.3 Research Plan and Interview Protocol .................... 77
4.4 Interview Insights .............................................. 79
4.4.1 Pre-Attack Cyber Negotiation ........................................... 79  
4.4.2 The Attack ................................................................. 85  
4.4.3 Post-Attack ................................................................. 89  
4.5 Case Study Analysis .......................................................... 92  
4.5.1 Uber Technologies ....................................................... 92  
4.5.2 National Health Services ............................................... 95  
4.6 Future Work ................................................................. 99  
4.7 Conclusion ................................................................. 99  

5 Conclusion ........................................................................ 101  
5.1 Where Computer Science and Urban Planning Meet ................. 101  
5.1.1 Research in Data Science .............................................. 102  
5.1.2 Research on Computing Systems and Networks .................. 103  
5.1.3 Beyond Cybersecurity for Urban Science ............................ 104  

A Simulation screens and associated questions .......................... 106
# List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>Sample Fault Tree Hierarchy.</td>
<td>21</td>
</tr>
<tr>
<td>2-2</td>
<td>Example Attack Tree.</td>
<td>22</td>
</tr>
<tr>
<td>2-3</td>
<td>Attack Tree Framework Mapping.</td>
<td>28</td>
</tr>
<tr>
<td>2-4</td>
<td>MITRE’s Cyber Attack Lifecycle.</td>
<td>28</td>
</tr>
<tr>
<td>2-5</td>
<td>Example attack tree with kill chain for each branch [11].</td>
<td>30</td>
</tr>
<tr>
<td>2-6</td>
<td>Master Attack Methodology.</td>
<td>35</td>
</tr>
<tr>
<td>2-7</td>
<td>Example CCTV Network Topology [12].</td>
<td>36</td>
</tr>
<tr>
<td>2-8</td>
<td>Automatically Generated Attack Tree for Data Exfiltration.</td>
<td>38</td>
</tr>
<tr>
<td>2-9</td>
<td>Hand-Drawn Attack Tree for Data Exfiltration.</td>
<td>39</td>
</tr>
<tr>
<td>3-1</td>
<td>SCADA vs Non-SCADA Vulnerability Density</td>
<td>54</td>
</tr>
<tr>
<td>3-2</td>
<td>Prioritization Schema Steps</td>
<td>59</td>
</tr>
<tr>
<td>4-1</td>
<td>Lockheed Martin’s Cyber Kill Chain.</td>
<td>69</td>
</tr>
<tr>
<td>A-1</td>
<td>Port scanning screenshot.</td>
<td>107</td>
</tr>
<tr>
<td>A-2</td>
<td>Ransomware simulation screen.</td>
<td>107</td>
</tr>
<tr>
<td>A-3</td>
<td>Ransomware resolved screenshot.</td>
<td>108</td>
</tr>
</tbody>
</table>
# List of Tables

2.1 Industry Perspective On Cyber Resiliency – Lifecycle for Executives [64] 29
2.2 IoT Surface Areas and associated Category [78] 31
2.3 Mapping for ATT&CK Matrix to Cyber Kill Chain 33
2.4 Description of Pertinent Network Topology Features [12]. 37

3.1 ICS-CERT vs MITRE SCADA Vulnerabilities 50
3.2 Top SCADA CWEs by Density 52
3.3 Top SCADA CWEs by Exploit Density 53
3.4 Non-SCADA Exploits vs CWE Frequency, CVSS, Impact, and Exploitability Scores 55
3.5 SCADA Exploits vs CWE Frequency, Impact, and Exploitability Score 57
3.6 Top 10 SCADA CWEs by Impact and Exploitability Score 57
3.7 Top 5 Ranked CWEs per Category 59
3.8 Total Scores for Top-Ranked CWEs 60
Chapter 1

Introduction

Cyber Science is the study of the technical, social and policy levers available for defending digital systems from cyber threats. Critical Infrastructure Cybersecurity is a subfield of Cyber Science focused on securing national assets essential to the economic and political stability of a country. An increasingly looming threat is that either an adversarial nation-state or a sophisticated, malicious hacking group can interrupt the operations of critical infrastructure through virtual means. Analyzing critical infrastructure cybersecurity in the context of an urban setting can be the basis for helping cities defend themselves from cyberattack.

Every city has a human pulse, but they also have a digital heartbeat. Our built environments are becoming technologically "smarter" and increasingly dependent on highly specialized computers called industrial control systems (ICS). These computers are part of the emergent Industrial Internet of Things (IIoT). Unfortunately, these systems are easily hackable and difficult to secure. While security researchers, myself included, have spent considerable time analyzing the peculiarities of IIoT systems, a key stakeholder group — cities — are poorly informed about the cybersecurity risks facing urban infrastructure. State and local administrators throughout the US are not part of the most important conversations about cybersecurity, related national policy questions and the issue of possible military responses to cyberattacks.

Currently, there is no clear agreement on how best to prepare and respond to cybersecurity threats. This is especially true for urban critical infrastructure and our evolving smart cities [4]. We are relying more on smart grids, connected water networks and internet-
enabled transportation infrastructure. But, each step a city takes in this direction increases its reliance on connected devices and thus exposes it to even more serious cyber breaches. This threat is real and local as we have seen from the recent cyberattack on a Rye, New York damn where Iranian hackers took control over the system controlling flow rates for the water [95]. My overarching question is: how can we help cities understand their risk and prepare for cyberattacks accordingly so that they can be part of the growing dialogue around critical infrastructure cybersecurity? To answer this question will require a strong understanding of both computer science and urban planning. I hope that this research will allow me to define the intersection of these studies as a research field I call Urban Science.

For cities to take a strategic approach to managing their urban critical infrastructure’s cybersecurity, they will need to do three things. First, cities must evaluate their threat landscape and their surface area of potential cyberattack. Cities that have more connected infrastructure will have more threats. A city will likely be overwhelmed by the number of attack vectors that can be used to target and disable urban critical infrastructure. They will need a way to prioritize the risk of the different attacks. After prioritizing the risk of attacks, cities will need to figure out mechanisms to defend their urban critical infrastructure. This will require a mix of both technical and social defense mechanisms.

Based on this plan for city cybersecurity, I will cover the following themes throughout this dissertation:

- Industrial Control System Cybersecurity Threat Landscape,
- Urban Critical Infrastructure Risk Prioritization, and
- Cyberdefense Strategies.

Below, I provide some background for each theme. Also, I provide a policy recommendation that could accompany each component of this plan.
1.1 Industrial Control System Cybersecurity Threat Landscape

Critical infrastructure is broadly defined by the U.S. Department of Homeland Security (DHS) in terms of 16 sectors that are essential to the security of the nation. These include: chemical, commercial facilities, communications, critical manufacturing, dams, defense industrial base, emergency services, energy, financial services, food & agriculture, government facilities, healthcare & public health, information technology, nuclear reactors, materials & waste, transportation systems and water & wastewater systems. I will be concentrating on Critical Urban Infrastructure, that is, city-sustaining, physical infrastructure sectors. City-sustaining is defined in terms of meeting the basic everyday needs of citizens. Therefore, I include: water & wastewater systems, energy and transportation systems. Sectors such as food & agriculture, while important, are not physical, and therefore are not considered.

Urban Critical Infrastructure systems use cyber physical technology called industrial control systems (ICS). There are three common types of ICS: Programmable Logic Controllers (PLC), Distributed Control Systems (DCS) and Supervisory Control and Data Acquisition (SCADA) systems. ICS are used to control many types of critical services such as the electric grid, water networks and transportation systems. The cybersecurity of ICS offers its own unique set of security challenges. Some of these include ICS patching, proprietary code and associated programming languages as well as their communication protocols. Security patching for ICS is an issue because ICS systems operate in real-time and any downtime required to patch a system would result in down-time for the critical infrastructure. The proprietary code is an issue for ICS because only the vendor that created the systems can adjust settings or fix the code. This is especially a concern because many ICS systems have lifetimes beyond the lifetime of the ICS manufacturer resulting in systems that can no longer be properly serviced. The communication protocol for ICS systems is not encrypted and offers no security provisions.

The ICS domain has become particularly vulnerable to attack because of recent efforts to connect these systems. ICS devices and their associated sensors that are interconnected via the internet are part of the Industrial Internet of Things (IIoT). IIoT systems are increasingly
under attack because of their ease of access to hackers and their ability to manipulate critical infrastructure.

Because of the ubiquity of IIoT systems in cities ranging from Closed Circuit Television Cameras (CCTV) to smart meters, there are many opportunities for hackers to access and manipulate these devices. Cities will have challenges manually itemizing all digital assets that could be hacked and then determining how a hacker could proceed to compromise the system. Instead, cities should use an attack generator tool to automatically determine attack vectors for a given system. In this dissertation, I propose a method to enumerate all possible attack paths for any urban critical infrastructure system.

This research will build on the work of Shrobe for his pioneering work on using Artificial Intelligence (AI) planners to conduct automated attack graph generation [89]. I will also leverage the extensive body of knowledge on attack methodologies [36] and vulnerability classifications [62] to develop a comprehensive attack rule set that can mimic a sophisticated attacker.

Such a tool can be incredibly powerful as a security auditing software to determine vulnerabilities across digital assets in a city environment. It can equally be used by hackers to determine optimized pathways to break into urban critical infrastructure. Because of this, policy will be needed to both encourage the use of and regulate the tool. An example policy recommendation could be for cities to require that all new digital systems installed for urban critical infrastructure should be tested in a digitally and physically secure laboratory using this tool. System operators can benefit from knowing the potential pathways attackers can use to compromise the infrastructure. The tool should only be used in this digitally and physically secure environment so that it is difficult for a hacker to access the tool and learn about all of the city’s attack vectors.

1.2 Urban Critical Infrastructure Risk Prioritization

What cybersecurity risks matter most? This is the perpetual question asked by critical urban infrastructure operators and managers. It will be especially pertinent after operators are presented with the many attack vectors for urban systems using the automated attack
generator. It is impossible to account for and act on the innumerable vulnerabilities in the wild, all the attacks that have already been launched and the impacts these have caused. For example, does it make sense to employ a military-grade response to a hacker that uses a buffer error to wage a ransomware attack against a system administrator at a hospital, or an attacker that uses a ’man-in-the-middle attack’ to wage a state-estimation attack in the electric grid? It is not possible to answer these questions without taking into account a great deal of data. We need to be able to make a real-time assessment of the risks implied in each of these situations – taking account of the vulnerability inherent in the specific management systems under consideration. This risk model should make it possible for us to specify which variables are most important in determining the likelihood that a particular vulnerability can be exploited and then attacked. Further, this same risk model should be able to forecast the likely payback of a particular risk management investment.

This requires data science to determine which metrics and variables matter most when evaluating security prioritization. Specifically, I will search for indicators that can help to determine the likelihood that a vulnerability will be exploited. A model that can represent asset risk and an associated prioritization of concerns can be used to weigh the magnitude of importance of an attack. This, in turn, can serve as a metric to help determine a proportionate defensive or offensive move.

Work on this prioritization model builds on earlier theoretical work by several scholars. Rinaldi’s team showed how critical infrastructure are highly interdependent [83]. If one critical infrastructure system is compromised, there is high risk that others would be affected. Building on this work, Brown’s team showed a prioritization mechanism for critical infrastructure and modeled a defense strategy based on critical infrastructure interdependencies [10]. Ralston’s team developed an early risk assessment specific for ICS [80]. Grubesic’s team developed a typological framework for evaluating vulnerabilities across critical infrastructure [33]. Using a qualitative approach, Zhu’s team established a taxonomy of cyberattacks specific to SCADA systems [101]. This was later improved on by Stouffer’s team at the National Institute of Standards and Technology (NIST) [92]. These scholars have shown the unique attributes of critical infrastructure and established foundational principals for how to understand their risk of attack.
After urban critical infrastructure system vulnerabilities are prioritized using this model, policy will be needed to instruct system operators and managers how to act on the prioritized risks. One policy recommendation may include a mandate to constantly monitor all systems with the highest exploit risk vulnerabilities for anomalous activity. Another policy could include the requirement to establish a security patching program aimed at addressing the prioritized exploit risks so that over time the system risk is reduced.

1.3 Cyberdefense Strategies

After cities have taken stock of their systems, determined the various attack vectors attackers can use to subvert urban critical infrastructure and prioritized what vulnerabilities need to be addressed, it is time to defend against inevitable attacks. Technical defenses ranging from intrusion detection and prevention systems to firewalls are likely to be thought of first to secure these urban systems. Such technologies should definitely be considered and deployed where possible. In some cases, these technical defenses might be extremely expensive to comprehensively install across all digital assets and beyond the budget allocated to information technology departments of public agencies. For this reason, I propose that cities adopt a new class of non-technical cyberdefense strategies called Defensive Social Engineering (DSE) in addition to their technical defenses.

Social engineering is the act of a hacker tricking and manipulating a victim into providing information or access to a system. DSE relies on the premise that there is a person that is hacking a victim and this attacker is vulnerable to being tricked and manipulated. There are many non-technical tools that could be considered defensive social engineering. Some of these include: cyber negotiation, honeypots and decoy systems, obfuscation techniques, misinformation campaigns, employee education and awareness efforts, devaluing assets upon compromise and proactive defensive signaling. Of these tools, I believe cyber negotiation can be an effective defense strategy that cities could use if they are attacked.

To develop a cyber negotiation strategy for cities, I interview a series of urban critical infrastructure operators to develop an approach for negotiating with urban cyberterrorists. These interviews include a simulated ransomware attack where I ask operators how they
would respond and how they might advise others to negotiate the situation with the hacker. By analyzing results from the interviews and reviewing case studies of various ransomware attacks, I develop a negotiation playbook for urban critical infrastructure operators.

My cyber negotiation approach builds on previous scholarship on ransomware negotiation by DeMuro. He describes the differences between cyber piracy and cyber terrorism and how ransomware negotiation should be likened to negotiating with pirates [16]. I also draw on findings from Farringer who writes about ransomware in healthcare. She describes circumstances under which critical infrastructure organizations could be compelled to pay a ransom [22]. In addition to this scholarship on ransomware and cyber negotiation, I leverage negotiation strategies developed by Susskind which were developed for in-person negotiations to establish the cyber negotiation playbook [94].

If cities plan to use defensive social engineering tools such as cyber negotiation, policy should be established to guide the process of this defensive strategy. A policy recommendation for cyber negotiation is that all urban critical infrastructure organizations must establish a relationship with their local FBI office. This is because many organizations call the FBI to help with ransom negotiations, but the first time they ever interact with the FBI is after they are attacked. Mandating that urban critical infrastructure organizations establish a relationship prior to being attacked could help to align urban infrastructure operator priorities with the FBI, which could help to streamline the negotiation process.

1.4 Audience

Cybersecurity for urban critical infrastructure is inherently an interdisciplinary field. If computer science is given complete control over decisions about software and network vulnerabilities, experts in that field are likely to overlook key managerial and governance considerations in designing ways of responding to cyber threats. The same reasoning can be applied to giving managerial or social science experts complete control over the development of cybersecurity risks. They are likely to undervalue the software and hardware problems. Therefore, computer science, managerial and social science communities will all be considered as part of the audience for my dissertation. Each component of the city cybersecurity
strategy outlined above will speak to a different audience. However, the findings can be used in concert to develop a strong cybersecurity approach for urban critical infrastructure.
Chapter 2

A Master Attack Methodology for an AI-Based Automated Attack Planner for Smart Cities

2.1 Introduction

Critical infrastructure such as CCTV security networks, the electric grid, water networks and transportation systems operate using industrial control systems (ICS). Increasingly, as cities move to become "smart cities", ICS are networked together for ease of use and expense reduction. ICS devices and their associated sensors are interconnected via a network that now comprises the Industrial Internet of Things (IIoT).

While IIoT provides convenience, it comes at an associated cost. ICS that makes up the IIoT is constantly subject to cyberattack. Kaspersky Labs, a leading cybersecurity research and antivirus company, found that in 2016 one in every five industrial control systems are attacked each month [43]. Further, not all of these attacks used the internet to penetrate the IIoT. Some used other vectors including removable media (USB sticks) [43].

Public administrators need to understand cyber threats. However, given the number of critical infrastructure components in any municipality, and the vast variety of configurations involved, it would be too time consuming to enumerate every attack pathway adversaries
might take. The traditional approach to enumerating attack vectors involves creating attack
trees (also called attack graphs). Developing an attack tree for each critical infrastructure
would be tedious and require highly technical knowledge as well as associated knowledge
about mechanisms that might be used to attack each system. A public administrator or
his/her team is not likely to have the necessary expertise to do this. This leaves cities fully
exposed.

Today, artificial intelligence (AI) is being used in many industrial sectors and govern-
ment organizations to enhance efficiency and scale operations. The challenge of quickly and
easily enumerating critical infrastructure attack vectors can be addressed using AI. In this
paper, we describe an AI planning system design that can enumerate a set of multi-step
attack plans capable of penetrating and compromising systems across IP-networked devices.
Importantly, our proposed method is "industry sector agnostic" meaning that it is designed
to accommodate a wide range of organizations and computing systems. While automated
attack planners have been developed previously, they have not used standardized cyberse-
curity frameworks, leading to semantic deficiencies when describing particular attack plans.
Further, existing attack planners, because of the speed at which things are changing, do not
have rule sets built to accommodate modern IoT/IIoT systems. The contribution of this
research will be to develop a master attack planner's ontology. We call it "master" attack
methodology because our goal is to design an attack ontology that accommodates any IP
networked system in any industry sector. The example attack graph we develop provides
automatic identification of adversarial strategies that can be used to compromise a CCTV
network whose typology is similar to other IP-based networks. While an example of the au-
tomated methodology is provided, and compared with a manually generated tree, we do not
have sufficient system environment data available to test our new AI planning system across
more than one critical infrastructure system. That research will follow in future studies.
Therefore, this study is limited to developing, but not testing across multiple environments,
the efficacy of our automated alternative to existing attack planning systems.
2.2 Background

Attack trees are used to enumerate the threat pathways that attackers could use to penetrate a system. The first publication on attack trees was by Bruce Schneier in 1999 in Dr. Dobb’s Journal of Software Tools [88]. His article described an approach based on a well-documented and frequently used reliability analysis technique created in 1962 at Bell Telephone Laboratories called Fault Tree Analysis [34]. The intent of fault tree analysis was to evaluate system failure risks that could cause an inadvertent launch of an intercontinental ballistic missile.

As a general example illustrated in Fig. 2-1, the root of the fault tree structure is the failure, and the leaves are possible causes of the failure. Each leaf has an associated probability that the cause will occur. Causes may be dependently linked and categorized as ”and” logic gates. ”And” leaves must both occur for the failure to take place. This is distinct from ”or” leaves where two leaves may be present but only one of the two causes is needed to generate a failure. The tree proceeds downward from the root with some causes having subsequent levels of sub-causes which may include both the ”and” and ”or” logic gates [11].

![Sample Fault Tree Hierarchy](image)

Figure 2-1: Sample Fault Tree Hierarchy.

The fault tree is completed once there are no longer any traceable causes or sub-causes for a given failure that have not already been included.

2.2.1 Attack Tree Features

Attack trees are functionally similar to fault trees, however, they usually serve a different objective, rely on different risk quantification methods and call for different outcome inter-
pretation.

**Objective**

Where fault trees start with a specific failure of a system as the root of the tree, the root of an attack tree is the goal of the attacker. The goal might be stealing money from a safe, or stealing passwords from a secure online database [88]. The leaves of the attack tree, unlike a fault tree, are the discrete actions that must be taken to achieve the objective. An example of an attack tree appears in Fig. 2-2.

![Example Attack Tree](image)

**Quantification**

In addition to the root being different in attack trees, the quantification method varies as well. While some attack trees use probabilities to quantify risk, it is more common for attack trees to use qualitative (i.e. ordinal) measures to score each leaf [11]. Such qualitative measures make more sense because of the obstacles to assigning a probability to the likelihood of an attack vector being pursued. Ordinal measures might involve a rating of "difficulty to penetrate" and use a ranking of the leaf from 1 to 5. Or, they might involve ranking the level of knowledge needed to penetrate on a scale of 1–10. Another way of quantifying the leaves in an attack tree is to use economic indicators. An example might be that it costs $10 to pay
for a dictionary password cracker versus $100 to buy a listserv address so that a successful phishing attack can be waged. Still another quantification metric might be timing, where password cracking could be rated at 10 hours and a phishing attack could be rated at 24 hours.

**Outcome Interpretation**

Attack trees are most appropriately used to determine the easiest or optimal line of attack for a hacker. Some researchers will assess this by determining which pathways on the tree have the fewest leaves on them (suggesting the least complicated attack route) while others might use the combined quantification metrics as a guide to determine the most desirable attack vector (from an attacker’s standpoint).

**2.2.2 Benefits of Attack Trees**

The benefits of attack trees are manifold. Fundamentally, attack trees enable the user to identify the potential areas of intrusion based on a goal established by a putative attacker [11].

Attack trees provide a causal framework for thinking about possible disruptive events. They also help to structure the complex problem of defending against cyberattacks [80]. The sequential attack plans represented by each leaf force system “defenders” to think through all possible avenues of attack. Further, attack trees make it easier for defenders to enumerate possible defense mechanisms associated with each leaf of the tree [85].

The flexibility of attack trees is valuable from a usability perspective. A security researcher who cares more about managerial cyber security rather than technical cyber security can use the attack tree in a way best suited to his or her purpose. This is because attack trees allow researchers to conduct analyses at multiple levels of abstraction. Researchers can acknowledge an attack vector in their attack tree without deep knowledge of the sub-leaves of the pathway, and instead focus on the topics of greatest investigative interest [11]. For example, if a researcher wants to focus on possible social engineering attacks on a target, and wants to develop an attack tree, the researcher can take note of the top level leaves describing technical exploitation of the target, but spend most of their time focusing on the
nodes most vulnerable to social engineering forms of attack.

Attack trees can be used as predictive tools as well as reactive security tools. When designing a cyber-system, an attack tree can be used to evaluate the various security requirements needed to protect that system. Alternatively, an attack tree can be used to audit or evaluate the security of a legacy or existing system to determine how vulnerable it might be to attack. This could clarify the best investments for securing the system.

Finally, attack trees, if structured properly, can be scalable. Because the end goal of the hacker is not always specific to a particular model of a system, common attacker goal trees are reusable, and can help to anticipate more complex system attacks [89].

2.2.3 Challenges of Attack Trees

While fault tree analysis is considered the ”gold standard” for aeronautic reliability testing, attack trees have not yet achieved that standard in the cyber security arena [34]. The primary reason fault tree analysis assigns probabilities to each leaf is to quantify risk. This provides an historical baseline for analysis [34]. Unlike mechanical systems, cyber system reliability is not as readily calculable. Failure rates for mechanical systems can easily be determined in a lab by running the mechanical system constantly and subjecting it to all possible conditions [29]. Even if a cyber-system is constantly run and exposed to all known conditions, it is not possible to calculate precise failure probabilities because it is not possible to know how or at what frequency an intelligent attacker might attempt to exploit a vulnerability in a cyber system. The scope of intelligent attacker threats makes establishing accurate probabilities of attack difficult.

While the failure potential of a physical system is finite, this is not the case for cyber systems. For example, there are only so many ways a person can break into a safe. A cyber system, due to its complexity and interconnectivity, can be exploited in innumerable ways. The exploitability of a system largely depends on the other systems to which the device of interest is connected.

Perhaps the most significant challenge with attack trees is that they need to be prepared by an expert who has both full knowledge of the system and a comprehensive understanding of how best to attack the system. It is not always possible to secure the services of such
an expert. Developing comprehensive attack trees is time consuming. Further, manually creating attack trees always starts from ground zero, and thus is inconsistent across security experts. Semantic idiosyncrasies in the security researcher community introduce additional challenges when attempting to compare risks across different systems [81].

2.2.4 Current State

To address some of these challenges, attack trees can be made more accessible and readily available by using artificial intelligence planning logic. An early automated attack generator developed shortly after attack trees were first documented used classical planning to develop the trees [89]. Classical planning is a branch of artificial intelligence. Classical planning requires an initial state, a goal state and a series of operators. The goal is to sequence these operators to achieve the goal state, starting from the initial state. Because of the deterministic functions of computing systems and associated attacks, classical planning is an efficient means of developing attack trees in a scalable way.

There are two fundamental components of an attack planner:

1. an abstracted rule set describing various methods and techniques for attacking a system, which ideally should be broadly applicable across all systems;

2. a detailed system description of the environment for which the attack tree is needed.

The system environment describes the network topology, the system components and their subsystems, data access rights and locations, and associated dependency relationships.

For an organization seeking to understand the cyber risks they face, the automated generator removes the requirement of having a person develop the tree who is knowledgeable about all possible ways of attacking their system. The only input required is a system description of the environment. These tend to be generally available. A system that automatically generates attack trees, then enables organizations to spend more time establishing the correct metrics for evaluating risk, rather than focusing on the enumeration of attack vectors.

The attack tree generator used as the basis for our work was developed by Dr. Howard Shrobe at The Massachusetts Institute of Technology’s Computer Science and Artificial
Intelligence Laboratory (CSAIL). This planner was built to enumerate attacks against the CSAIL computing network as it was configured when the planner was designed in 2002. The planner incorporated a system model of the CSAIL computing environment as well as an attack method ruleset designed to defeat the security triad (i.e. confidentiality, integrity and availability) of the system environment. When the attack planner is given the goal of compromising a node of the system environment, the planner uses goal directed backward search to enumerate all possible pathways to achieve the attack goal. This is how the attack tree is ultimately generated.

While existing automated attack generators are more consistent than manually created attack trees, issues remain. Automated generators today do not incorporate standardized language from the cybersecurity community into the trees. This misses an opportunity to incorporate valuable cross-system data integration into tree construction. Such data could include MITRE’s Common Vulnerabilities and Exposures (CVEs)\(^1\) or First.org’s Common Vulnerability Scoring System (CVSS)\(^2\) Scores. Also, the attack rules in existing planners do not cover all modern systems – especially with the recent surge of IoT and IIoT systems. In order to develop an attack tree generator that is suitable for multiple systems, taking account of diverse attack goals, it is important to standardize the categorization of methodologies used by attacking systems to create a master attack rule set applicable across many system types and industry sectors.

### 2.3 Scalable, Cross-Sector Attack Tree Design

We propose a standardized approach to developing attack trees guided by a common sequence of methods that attackers use to penetrate a wide range of systems. There are a considerable number of tools and frameworks available to hackers and security researchers alike. Unfortunately, none of these tools or frameworks addresses the full lifecycle of an attack. To accomplish this, we developed an integrated methodology that combines a number of estab-

---

\(^1\)CVEs are documented vulnerabilities for computing systems which are submitted by security researchers to MITRE who maintains a running catalog of vulnerabilities in their database.

\(^2\)CVSS Scores are developed by First.org and aim to evaluate the extent of threat that a given vulnerability poses to an organization.
lished frameworks. This method will generate the information needed to develop attack trees that should be scalable across industries and computing systems. To do this, we formulated a master attack methodology using various established frameworks for vulnerability, threat and exploit analysis that represent the anatomy of an attack’s ”when”, ”where”, ”what” and ”how”. The phasing sequence of the attack, or what we call the ”when”, leverages Lockheed Martin’s cyber kill chain [36]. The surface area of where the attack could occur, or what we call the ”where”, references the Open Web Application Security Project’s (OWASP) attack surface areas [78]. The actions required to successfully accomplish the given phase of attack, or the ”what”, is represented by both MITRE’s Common Attack Pattern Enumeration and Classifications (CAPEC) [62] and MITRE’s Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) framework [63]. Finally, the tools used to execute the actions, or the ”how”, are represented by both Kali Linux tools [46] and known exploit tactics by MITRE’s ATT&CK Matrix [63]. Each framework occupies a level in the traditional attack tree format as seen in Fig. 2-3.

2.3.1 Sequence of Phases for Waging Attacks

Lockheed Martin originally developed the Cyber Kill Chain which lists the phases of a possible attack. The Cyber Kill Chain was initially published in 2011 and was developed to help security researchers map how attackers conduct advanced persistent threats (APTs), including sophisticated cyberattacks conducted by nation states. The Cyber Kill Chain was inspired by the U.S. military’s kill chain for traditional warfare which involves the steps required to ”target and engage an adversary to create desired effects” [36]. After being created by Lockheed Martin, MITRE rebranded these steps as the Cyber Attack Lifecycle [64]. The Cyber Attack Lifecycle can be found in Fig. 2-4 and its associated description can be found in Table 2.1. The Cyber Attack Lifecycle and Cyber Kill Chain are interchangeable for purposes of our study.

While the phases indicate the order in which an attack is waged, these phases are not always performed in sequence. A lot depends on the attacker’s goal. For example, it is possible to skip the Weaponize, Deliver and Exploit phases of an attack if during Recon, credentials are discovered which offers Control. Further, throughout an attack, it is likely
that an attacker iterates previous phases of the lifecycle to continue gathering information about their target and refining their attack.

While developing an attack tree, each phase belongs at Level 1 of the tree hierarchy underneath the goal as seen in Fig. 2-3. Depending on the goal, some phases will be needed, while others will not. All phases should contain AND gates indicating that each phase listed must happen and involve of its own branch of operators. The kill chain phases should
Table 2.1: Industry Perspective On Cyber Resiliency – Lifecycle for Executives [64]

<table>
<thead>
<tr>
<th>#</th>
<th>Phase</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reconnaissance</td>
<td>Adversary develops a target</td>
</tr>
<tr>
<td>2</td>
<td>Weaponize</td>
<td>Attack is put in a form to be executed on the victim’s computer/network</td>
</tr>
<tr>
<td>3</td>
<td>Deliver</td>
<td>Means by which the vulnerability is delivered to the target</td>
</tr>
<tr>
<td>4</td>
<td>Exploit</td>
<td>Initial attack on target is executed</td>
</tr>
<tr>
<td>5</td>
<td>Control</td>
<td>Mechanisms are employed to manage the initial victims</td>
</tr>
<tr>
<td>6</td>
<td>Execute</td>
<td>Leveraging numerous techniques, the adversary executes the plan</td>
</tr>
<tr>
<td>7</td>
<td>Maintain</td>
<td>Long-term access is achieved</td>
</tr>
</tbody>
</table>

consistently fall directly underneath the goal and be the top-level nodes for all attack trees. For example, if the goal of the tree was to delete data (which would fall at the top of the tree), immediately underneath should be the various phases of the kill chain.

Lockheed Martin’s cyber kill chain has been represented as part of an attack tree previously [86]. To date, attack trees that reference the kill chain move through phases of an attack for a given goal within a single branch as can be seen in Fig. 2-5. However, this inaccurately reflects the depth of each attack phase that an attacker might move through to reach their goal. Instead of having a single branch where each level in the hierarchy represents a new phase of attack and a subsequent phase is a leaf of its precedent, each phase of attack could be a separate branch connected by AND gates in the second layer of the attack hierarchy. This new representation would better illustrate the depth of complexity behind each phase of an attack by showing that there are many subroutines required to complete each phase. Also, this would lead to a more consistent approach to specifying an attack tree, making it easier to automate the generation of attack trees across systems. The details behind each phase of an attack cannot be fully described in a consistent and scalable manner when enumerated as nested leaves within a single branch.

2.3.2 Surface Area for Waging an Attack

Each phase of an attack must occur on a given surface area of a system environment. While an attack goal might need to move through all phases of the kill chain to be successful, it would be unlikely for any attack goal to involve a tree that covers all surface areas of a given
system. A limited surface area is more likely to be required to achieve a given kill chain phase relevant to a specific attack goal. For example, an attack goal of "exfiltrate server data from a system" probably does not require recon on every surface area of a given system. It only requires recon on relevant system components.

OWASP has developed a list of seventeen surface areas for IoT systems [78]. Over the course of writing this paper, OWASP added to its list of surface areas. The list will continue to evolve over time as more threats are discovered and documented by security researchers involved in the Open Web Application Security Project. In the interest of attempting to future-proof the proposed surface areas from further edits, we distilled them into four categories: software/hardware, architecture, network and organizational. Software/hardware relates to the physical or digital features and functions of a system, architecture refers to design decisions and system configuration, network includes anything involving communications, and organizational refers to how the system is managed and any security policies in effect. These surface areas and their component parts are listed in Table 2.2 below.

The column labeled "Vulnerability Examples" describes the types of vulnerabilities likely to be associated with a surface area category. Further, the vulnerability examples listed for a surface area describes the types of vulnerabilities likely to be taken advantage of along a
<table>
<thead>
<tr>
<th>Category</th>
<th>Attack Surface</th>
<th>Vulnerability Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizational</td>
<td>Ecosystem</td>
<td>Interoperability standards, Data governance, System wide failure, Individual stakeholder risks, Implicit trust between components, Enrollment security, Decommissioning system, Lost access procedures</td>
</tr>
<tr>
<td>Software/Hardware</td>
<td>Device Memory</td>
<td>Sensitive data, Cleartext usernames, Cleartext passwords, Third-party credentials, Encryption keys</td>
</tr>
<tr>
<td>Architecture</td>
<td>Device Physical Interfaces</td>
<td>Firmware extraction, User CLI, Admin CLI, Privilege escalation, Reset to insecure state, Removal of storage media, Tamper resistance, Debug port, Device ID/Serial number exposure</td>
</tr>
<tr>
<td>Architecture</td>
<td>Device Web Interface</td>
<td>Standard set of web application vulnerabilities, Credential management vulnerabilities</td>
</tr>
<tr>
<td>Software/Hardware</td>
<td>Device Firmware</td>
<td>Sensitive data exposure, Firmware version display and/or last update date, Vulnerable services (web, ssh, tftp, etc.), Security related function API exposure, Firmware downgrade possibility</td>
</tr>
<tr>
<td>Network</td>
<td>Device Network Services</td>
<td>Information disclosure, Injection, Denial of Service, Unencrypted Services, Poorly implemented encryption, Buffer Overflow, UPnP, Vulnerable UDP Services, DoS, Device Firmware OTA update block, Firmware loaded over insecure channel, Replay attack, Lack of message integrity check, Credential management vulnerabilities, Insecure password recovery mechanism</td>
</tr>
<tr>
<td>Architecture</td>
<td>Administrative Interface</td>
<td>Standard set of web application vulnerabilities, Credential management vulnerabilities, Security/encryption options,Logging options, Two-factor authentication, Check for insecure direct object references, Inability to wipe device</td>
</tr>
<tr>
<td>Organizational</td>
<td>Local Data Storage</td>
<td>Unencrypted data, Data encrypted with discovered keys, Lack of data integrity checks, Use of static same enc/dec key</td>
</tr>
<tr>
<td>Architecture</td>
<td>Cloud Web Interface</td>
<td>Standard set of web application vulnerabilities, Credential management vulnerabilities, Transport encryption, Two-factor authentication</td>
</tr>
<tr>
<td>Organizational</td>
<td>Third-party Backend APIs</td>
<td>Unencrypted PII sent, Encrypted PII sent, Device information leaked, Location leaked</td>
</tr>
<tr>
<td>Architecture</td>
<td>Update Mechanism</td>
<td>Update sent without encryption, Updates not signed, Update location writable, Update verification, Update authentication, Malicious update, Missing update mechanism, No manual update mechanism</td>
</tr>
<tr>
<td>Architecture</td>
<td>Mobile Application</td>
<td>Implicitly trusted by device or cloud, Username enumeration, Account lockout, Known default credentials, Weak passwords, Insecure data storage, Transport encryption, Insecure password recovery mechanism, Two-factor authentication</td>
</tr>
<tr>
<td>Organizational</td>
<td>Vendor Backend APIs</td>
<td>Inherent trust of cloud or mobile application, Weak authentication, Weak access controls, Injection attacks, Hidden services</td>
</tr>
<tr>
<td>Network</td>
<td>Ecosystem Communication</td>
<td>Health checks, Heartbeats, Ecosystem commands, Deprovisioning, Pushing updates</td>
</tr>
<tr>
<td>Network</td>
<td>Network Traffic</td>
<td>LAN, LAN to Internet, Short range, Nonstandard, Wireless (WiFi, Z-wave, XBee, Zigbee, Bluetooth, LoRA), Protocol fuzzing</td>
</tr>
<tr>
<td>Architecture</td>
<td>Authentication/Authorization</td>
<td>Authentication/Authorization related values (session key, token, cookie, etc.) disclosure, Reusing of session key, token, etc. Device to device authentication, Device to mobile Application authentication, Device to cloud system authentication, Web application to cloud system authentication, Lack of dynamic authentication</td>
</tr>
<tr>
<td>Organizational</td>
<td>Privacy</td>
<td>User data disclosure, User/device location disclosure, Differential privacy</td>
</tr>
<tr>
<td>Software/Hardware</td>
<td>Hardware (Sensors)</td>
<td>Sensing Environment Manipulation, Tampering (Physically), Damage (Physical)</td>
</tr>
</tbody>
</table>
given surface area. (The vulnerabilities listed here are not used as part of our attack tree; rather, they are for descriptive purposes only.)

Each attack surface category will make up a different nested branch on the attack tree. For the kill chain phases of recon, weaponize, deliver and exploit, any surface area may be relevant. However, as the tree is populated towards the latter half of the kill chain under the phases control, execute and maintain, the surface areas that an attacker is seeking to act on will be a subset of those from the previous phases. For example, if Recon is only conducted on Network and Software/Hardware surface areas, Maintain would not include the surface areas of Architecture or Organization.

2.3.3 Actions Required for Waging an Attack

Level 3 of the attack tree must represent ”what” actions need to be performed during each phase of the attack on the given surface area. For the pre-attack phases consisting of recon, weaponize, deliver and exploit, we primarily use MITRE’s Common Attack Pattern Enumeration and Classification (CAPECs) to populate the ”what” level of the tree hierarchy. Specifically, CAPECs are used for the recon and exploit phases of an attack. The mapping of the CAPEC mechanism of attack and the phases of attack can be seen in Table 2.3. The weaponize and deliver phases of attack do not match any given CAPEC. Instead, we use the Lockheed Martin kill chain recommendations for this hierarchy level to note ”what” is being weaponized and ”what” is being delivered. The Lockheed Martin kill chain was used instead of MITRE’s recently released Pre-ATT&CK Matrix because there are no details concerning ”what” is being weaponized or delivered.

For control, execute and maintain, MITRE has developed the ATT&CK matrix that describes the branch level of detail below these phases. We have mapped the ATT&CK matrix categories to specific phases of the kill chain as can be seen in Table 2.3. We use these categories to populate level 3. Note that Table 2.3 is only meant to reflect the top-level domain of the mapping, and does not represent the depth contained within each domain which is described later.
### Table 2.3: Mapping for ATT&CK Matrix to Cyber Kill Chain

<table>
<thead>
<tr>
<th>Attack Phases</th>
<th>Recon</th>
<th>Weaponize</th>
<th>Deliver</th>
<th>Exploit</th>
<th>Control</th>
<th>Execute</th>
<th>Maintain</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CAPEC</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Collect and Analyze Information</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Inject Unexpected Items</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Engage in Deceptive Interactions</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manipulate Timing and State</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Abuse Existing Functionality</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Employ Probabilistic Techniques</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Subvert Access Control</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manipulate Data Structures</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Manipulate System Resources</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Lockheed Martin</strong></td>
<td>Client applications</td>
<td>Email</td>
<td>Websites</td>
<td>Removable media</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>ATT&amp;CK Matrix</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Command and control</td>
<td>Execution</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Credential access</td>
<td>Collection</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Privilege escalation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Discovery</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lateral movement</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Defense evasion</td>
<td>Persistence</td>
</tr>
</tbody>
</table>
2.3.4 Tools Needed for Waging an Attack

The final layer of the attack tree, hierarchy level 4, represents "how" an attack is likely to be carried out. Here, the weakness or vulnerability of the system, represented as a Common Weakness Enumeration (CWE)\(^3\) or Common Vulnerabilities and Exposures (CVE) can be listed along with the tools and malware associated with each attack component. CWEs or CVEs may not always be available, but if included in the system model, they should be noted as leaves (since theoretically, each tool or malware takes advantage of a given CWE or CVE). To determine the tools and malware, for the pre-attack phases of recon and exploit, we assigned each CAPEC a tool from Kali Linux, a comprehensive penetration testing toolkit which contains most of the popular technical tools used by hackers to compromise systems. We applied a semi-automated procedure to match Kali tools to a given CAPEC using a document-distance algorithm with manual corrections. This mapping is used to populate Level 4 of the tree hierarchy describing how recon and exploit is ultimately carried out. The other pre-attack phases of deliver and exploit do not have Level 4 components due to the nature of the phase. For the attack phases control, execute and maintain, the Level 4 nodes of the tree consist of various malware that are described within the respective ATT&CK matrix categories.

2.3.5 Design Summary

To develop an attack tree generator that can scale and be equally effective across multiple critical infrastructure sectors, a comprehensive attacker methodology framework must populate the master attack rule set. Leveraging frameworks from respected cybersecurity authorities including MITRE, Lockheed Martin and Offensive Security (the Kali Linux creators), we have compiled a master ontology database that accounts for virtually all known attack vectors across all system types. An abstracted and complete version of this framework can be found in Fig. 2-6 with an example goal (exfiltrate data), which describes the various permutations possible for each constituent framework that makes up the master attack methodology. The attack goal of exfiltrating data as seen in Fig. 2-6 is only one example of

\(^3\)CWEs are categories of different types of vulnerabilities or CVEs that are created and maintained by MITRE.
goals for which the methodology is relevant.

2.4 Example Use Case

To demonstrate the core concepts in this methodology that we hope can be used for automatically generating an attack tree for numerous critical infrastructure sectors, we have created a test case for a generic CCTV urban surveillance system. After developing an automated tree using the master attack method, we create an attack tree for the same system by hand. The manually created attack tree is then compared to the automated attack tree using our proposed standardized methodology to demonstrate some perceived benefits of the proposed method.

2.4.1 Automated Attack Tree with Master Attack Methodology

System Model

First, we developed a system model of the CCTV network based on our interpretation of essential elements of the network typology found in Fig. 2-7. This CCTV network typology was
selected because it has representative components included in most IP networks (i.e. storage servers, analytics engines, peripheral devices/sensors, command modules, visualization components, etc.).

For purposes of abstraction and simplification, our system model focuses on several key areas of the CCTV system highlighted in yellow. We selected these specific areas because we believe them to be essential to the operation of the CCTV system. These included the IP surveillance camera, the storage device, the video surveillance manager (VSM), the console server (CM), the display server (DP), the video processing server (MD), the LAN and its associated router and switch. We developed the system description using assumptions about the interactions and functions of these system components. These assumptions about the system description are documented in Table 2.4.

2.4.2 Attack Rule Set

For purposes of this example, we selected an attacker goal of exfiltrating data from the IP camera. Based on the system environment and the master attack framework, attack rules were selected to develop the branches of the tree. The rules that were applied for the first branch of the tree (recon) for this attack goal are reflected in Fig. 2-8 and represented in a nested tree structure along with their respective AND/OR gates. The ellipses represent
Table 2.4: Description of Pertinent Network Topology Features [12].

<table>
<thead>
<tr>
<th>System Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IP Surveillance Camera</strong></td>
<td>• Multiple edge devices are networked together</td>
</tr>
<tr>
<td></td>
<td>• Centrally managed</td>
</tr>
<tr>
<td></td>
<td>• Edge devices are running an embedded Linux OS</td>
</tr>
<tr>
<td></td>
<td>• Have only admin privilege level</td>
</tr>
<tr>
<td><strong>Storage Server</strong></td>
<td>• Edge devices have limited data storage capacities</td>
</tr>
<tr>
<td></td>
<td>• After 8 hours of video recording data is sent from IP camera to storage server</td>
</tr>
<tr>
<td><strong>Video Surveillance Manager (VSM)</strong></td>
<td>• All edge devices are centrally managed by the video surveillance manager</td>
</tr>
<tr>
<td></td>
<td>• VSM has admin privileges across all IP cameras and storage device</td>
</tr>
<tr>
<td><strong>Console Server (CM)</strong></td>
<td>• Windows OS</td>
</tr>
<tr>
<td></td>
<td>• Supervisory control across all servers and the IP camera network</td>
</tr>
<tr>
<td><strong>Display Server (DP)</strong></td>
<td>• Renders data from storage server</td>
</tr>
<tr>
<td></td>
<td>• Transmits video data to third party display</td>
</tr>
<tr>
<td><strong>Video Processing Server (MD)</strong></td>
<td>• Processes data for interoperable analysis across different peripherals such as occupancy sensors</td>
</tr>
<tr>
<td></td>
<td>• Enables processing for eigenfaces so that facial recognition can be performed via computer vision as well as other needed computations that can interact with alarms</td>
</tr>
<tr>
<td><strong>LAN, Router &amp; Switch</strong></td>
<td>• Bus and hardware that enables communication across devices</td>
</tr>
</tbody>
</table>

Further branches and leaves of the tree which are omitted for purposes of brevity. It is important to note that based on the system model and selected goal, not all level 2 surface areas are applicable at every attack phase and not all CAPECs and ATT&CK level 3 "what" categories are applicable to every surface area. Further, if there is no CWE or CVE available for level 4, the generator will skip this and jump directly to the Tools/Malware.
**Goal: Exfiltrate data from IP camera**

If the goal is to Exfiltrate Data then you have to do

**AND**

Recon

If the goal is to do Recon then you have to do Recon on

**AND**

Network

**AND**

Device Network Services

If the goal is to do Recon on Device Network Services then you have to do

**OR**

Fingerprinting

**AND**

If the goal is to fingerprint network services then exploit "Information Exposure" weakness (CWE-200)

If the goal is to exploit the "Information Exposure" weakness then use nmap

Protocol Analysis...

Footprinting...

Ecosystem Communication...

Network Traffic...

Software/Hardware...

Architecture...

Organization...

Weaponize....

Deliver....

Exploit....

Control....

Execute....

Maintain....

---

Figure 2-8: Automatically Generated Attack Tree for Data Exfiltration.

### 2.4.3 Manual Attack Tree

For comparative purposes, we asked several security researchers hand-draw attack trees to demonstrate the differences for the same CCTV system. Fig. 2-9 shows a representative manually created tree. To successfully create this tree, the security researchers not only needed system knowledge of the CCTV network, but also an understanding of possible attack patterns relevant to the CCTV system.
Goal: Exfiltrate data from IP camera

If the goal is to Exfiltrate Data
then you have to

AND
Find the Data
If the goal is to find the data
then you have to

OR
Find the Local Camera
If the goal is to find the data on the local camera
then you need to

AND
Access the camera
OR
Steal Password
Exploit Vulnerability

Find the Hosting Server
If the goal is to find the data on the hosting server
then you need to

AND
Access the hosting server
OR
Steal Password
Exploit Vulnerability

Steal the Data

Figure 2-9: Hand-Drawn Attack Tree for Data Exfiltration.

2.4.4 Comparative Discussion

Perhaps the most important difference between the automated tree and the manual tree was the time required to create each one. The manual tree required time researching and mapping the system environment and possible attack vectors – this took an average of an hour for each security researcher to complete the manual tree, while the automated tree was completed in minutes by loading the model-based system description into the planner. The manual tree does not incorporate standardized language from various established cybersecurity frameworks. This is problematic because if two different security researchers tried to enumerate all attack pathways, there might be discrepancies between the trees. Of course, time is required to build a model-based system description of a computing environment, but our assumption is that system models may be readily available for use considering they are important for a variety of testing purposes. Standardization also makes it easier to train non-experts or semi-experts in generating attack trees without requiring detailed cyber security domain knowledge.
The automated master attack method of tree construction contains references to specific CAPECs which often connect back to specific CWEs and CVEs, while the manual tree lacks this information. This provides actionable insight to operators so they know which security patches are most important. The automated tree can scale for massive systems based on the details provided in the system description, while by hand it would not be possible to account for all vectors in an extremely complex environment without consuming considerable resources. An automated approach will also greatly reduce incomplete and incorrect attack trees caused by oversight and errors in enumerating the attack surface in complex systems.

Perhaps the most important difference between the two is that the automated tree has considerably more depth and information than the manual tree because it moves through each phase of the Cyber Kill Chain. This generates step-by-step insight into how the attack would probably be carried out compared to a less detailed hand-drawn tree. The comprehensive nature of the automated tree provides more insight to a defender on the risks of the system.

Traditionally, the domain knowledge or the thinking required to create unique attack vectors has resided with individual experts and has never been shared. Our automated approach enables us to capture domain knowledge from multiple experts for future reuse.

Automated attack tree generation using the proposed method shifts the burden from creating individual attack trees to system specification. Experts spend more time understanding and specifying system components, interfaces and interconnections. This is advantageous once system assumptions are explicitly code; it should result in more consistent and robust attack trees for a variety of industries and types of computing systems. With the manual approach, all system information and assumptions reside with the expert, making it difficult to validate the experts’ understanding of the system.

2.5 Future Work

To date, this research has involved designing a method for a scalable master attack method that can be used with automated attack generators. The proposed method has only been tested on a single critical infrastructure system environment – the example CCTV network shown above. Therefore, we cannot yet conclude that it can effectively scale across industries.
The next phase of our research will aim to demonstrate that the master attack method we have proposed can be applied across multiple critical infrastructure sectors involving different system models and computing systems. A persistent challenge, though, is getting access to data for system environments due to their often proprietary or sensitive nature. We are in active discussions with NASA’s Jet Propulsion Laboratory to collaborate on testing this method across some of their mission system environments. The goal will be to determine if the proposed design method can be broadly applied and scaled across systems and sectors. Other future work will involve refining and updating the categories in the frameworks maintained by multiple organizations.

2.6 Conclusion

As critical infrastructure continues to be subject to cyberattacks, defenders must remain vigilant and evaluate all possible attack vectors involving multiple systems. While attack trees can be used to provide guidance about possible attack vectors that defenders need to protect, there are operational challenges associated with developing such attack trees. Automating the attack tree generation process using AI planning can ease this operational burden.

The design of automated attack trees to date has not generated a comprehensive attack rule set that can be used across disparate critical infrastructure sectors. By combining attack frameworks from a number of respected authorities, we have developed a master attack method that can be used with classical planners to generate automated attack trees. We hope that this new approach, with further testing and refinement, will be useful across multiple critical infrastructure sectors, thereby easing the operational burden of cyber risk enumeration.
Chapter 3

IIoT Cybersecurity Risk Modeling for SCADA Systems

3.1 Introduction

3.1.1 Problem Statement

Cyberattacks can easily disable Industrial Internet of Things (IIoT) devices responsible for urban critical infrastructure. Urban critical infrastructure includes smart grids, water networks and transportation systems. In 2015, multiple power substations in Ukraine were compromised resulting in rolling power outages affecting 225,000 people [15]. Ukraine’s Supervisory Control and Data Acquisition (SCADA) system that is responsible for controlling the smart grid’s IIoT devices is vast and complicated such that it will be impossible to patch all vulnerabilities throughout the networks. While there are vulnerability taxonomies and cybersecurity frameworks that may help to mitigate risk, these tools do not provide data-driven guidance about SCADA security research priorities or a dynamic model to evaluate risk based on various operating parameters. This study provides a risk analysis of critical infrastructure SCADA vulnerabilities and exploits using statistical methods. Further, the study offers technical SCADA IIoT design recommendations to help mitigate future system exploit risk.

Evaluating IIoT exploit risk is challenging. The problem is accentuated by findings of
various security researchers that the Common Vulnerability Scoring System (CVSS) risk metrics created by First.org and used by the Department of Homeland Security (DHS) and the National Institute of Standards (NIST) are not effective at predicting exploits [2,69]. Further, NIST’s cybersecurity framework that intends to help organizations evaluate cyber risk for industrial control systems (ICS) faces adoption challenges and does not directly address exploit probability. Despite being labeled as best-in-class, reasons for slow adoption include the considerable time and expense required to implement the framework [17]. SCADA and critical infrastructure vulnerability taxonomies exist that could help to identify cyber risk [33,79,101]. While these taxonomies could be useful, the findings are not grounded in data-driven, empirical analysis which raises questions about their applicability to cyber risk in the field.

3.1.2 SCADA IIoT Overview

SCADA systems provide a supervisory control software layer across multiple programmable logic controllers (PLCs), which are a type of IIoT. SCADA systems are designed for use over long distances such as water or electric distribution. Because of these longer distances, there tends to be less control over the networks that use them. 80% of U.S. utilities run on SCADA systems [90]. SCADA operates using telephony communication or other third party networks, which reduces the speed, frequency and quality of communications [27]. For this reason, SCADA tends to be event driven meaning that data is only communicated from the devices to the software when there is a change in value [27]. Controlling other IIoT devices, SCADA systems require an operator console or human-machine interface (HMI) from which an engineer can view, command and control the devices connected to the system [98]. This HMI is also vulnerable to attack where an attacker could intercept the PLCs data and alter it on the HMI [92]. SCADA systems typically runs on a commercial off-the-shelf Windows PC which can expose the software to an array of operating system, Windows-based attacks [80]. A growing challenge is that there is an increased interest in connecting SCADA-based IIoT systems to IT networks. This can allow for hackers to access potentially vulnerable SCADA systems through backdoors using TCP/IP-based attacks.
3.1.3 The National Policy & Regulatory Landscape

In 2013, Executive Order (EO) 13636: Improving Critical Infrastructure Cybersecurity was published. The EO encourages the adoption of cybersecurity best practices and mandated that the National Institute of Standards and Technology (NIST) develop new ways of assessing cybersecurity risk [74]. The EO falls short, however, because it is entirely voluntary and contains no incentive structures. Also, it puts the burden for taking action only on the shoulders of critical infrastructure operators [56]. While NIST created a strong cybersecurity framework – which is hailed in industry as best-in-class – the financial burden of implementing NIST’s framework is a serious barrier to adoption [17]. A less time-intensive, expensive and streamlined alternative to NIST’s recommendations is needed for the SCADA community.

Industry organizations like the North American Electric Reliability Corporation (NERC), have tried to step in [100]. For example, in 2008, NERC proposed Critical Infrastructure Protection (CIP) Reliability Standards to the Federal Energy Regulatory Commission (FERC) to improve security for the electric grid [23]. FERC has adopted these recommendations, mandating U.S. electric companies comply with all voluntary cybersecurity regulation. Extensive survey results from NERC revealed that there are loopholes in the regulation. This enabled 75% of companies to opt-out of cybersecurity regulation while those companies that could not opt-out preferred to pay fines rather than update their system security [19].

3.1.4 Vulnerability Identification and Classification

Vulnerability frameworks are useful tools that draw attention to specific categories of threats. Several frameworks for vulnerabilities exist today. The MITRE Corporation, developed and maintains a database of Common Vulnerability and Exposures (CVEs) to keep track of known software vulnerabilities. Each CVE has an associated risk score created by First.org called the Common Vulnerability Scoring System (CVSS). The CVSS base score is calculated using a complex formula that is primarily a function of an Exploitability score and Impact score. NIST’s National Vulnerability Database (NVD) cites each score (CVSS, Impact and Exploitability) alongside each CVE. Findings by Allodi [2] and Nayak [69] indicate that
existing security research metrics such as CVSS, Exploitability and Impact scores for vulnerability are not an indication of exploit for software. Previous studies focused on software vulnerabilities without considering if there are certain subclasses of software where vulnerability risk metrics actually are effective at indicating exploitability. SCADA as a subclass of software should be investigated to understand the vulnerability metrics’ relationship with exploits.

Along with their database of CVEs, MITRE created a database of Common Weakness Enumeration (CWEs) [65]. CWEs classify CVEs by type of vulnerability resulting in a standardized and comprehensive list of cyber weakness classes. While CWEs provide a common language for how to define a vulnerability, it does not provide guidance for which CWEs are most relevant for certain classes of software like SCADA systems which would be relevant to urban critical infrastructure. From 2009 to 2011, MITRE and the SANS Institute created a prioritized list of CWEs called the CWE/SANS Top 25 Most Dangerous Software Errors. The list aimed to identify the greatest software vulnerability types, however it was nonspecific to a given class of software. The top 25 list used the Common Weakness Scoring System (CWSS) which evaluates vulnerabilities by assessing three metric groups: “Base Finding metric group (captures the inherent risk of the weakness, confidence in the accuracy of the finding, and strength of controls), Attack Surface metric group (assesses the barriers that an attacker must overcome in order to exploit the weakness) and the Environmental metric group (evaluates the characteristics of the weakness that are specific to a particular environment or operational context)” [57]. The principal weakness of the CWE/SANS prioritized list is that it fails to consider empirical evidence of exploits. A statistical prioritization would be more effective than a scoring prioritization such as CWE/SANS Top 25 because a data-driven study can account for the prevalence of exploits found in the wild.

Typologies and taxonomies of critical infrastructure attack and vulnerability exist [79]. Two previous studies on critical infrastructure vulnerabilities focus on different domains: Pak focuses on software attacks and Grubesic and Matisziw focuses on non-software vulnerabilities. These typologies are very useful to understand the broad critical infrastructure landscape, but fall short as insightful resources for security professionals and researchers because neither are specific enough to provide actionable insight to managers, administra-
tors or policy makers. Also, neither specifically analyze SCADA system security which is essential to city-sustaining systems.

Pak lists types of general attacks he believes are most relevant to CI such as distributed denial of service attacks, worms and Trojan horses. Pak also makes high-level organizational recommendations including strengthening information sharing practices among vulnerable CI sectors, publicly announcing vulnerabilities to ensure patching, and encouraging public/private collaboration to enhance security posture through training and education programs [79]. Further, he encourages continuous monitoring for open ports susceptible to attacks [79]. Pak’s recommendations lack specificity due to the breadth of cyber systems included in the standard critical infrastructure definition that includes industries as diverse as the financial and energy sectors. Therefore, security professionals are unable to leverage this research to further fortify their infrastructure.

Grubesic and Matisziw address critical infrastructure vulnerability but do not discuss software vulnerabilities. They propose the following variables are essential to understanding CI vulnerability: condition and decay, capacity and use, obsolescence, interdependencies, location and network topology, disruptive threats, policy and political environment, and safeguards [33]. While their vulnerability typology is applicable for CI SCADA systems, their omission of software vulnerabilities deprives OT security engineers of concrete and actionable recommendations.

A cyberattack taxonomy was developed by Zhu, Joseph and Sastry for SCADA systems. Zhu’s team provide recommendations for control engineers such as: beware of false data injection, man-in-the-middle, and denial of service attacks [101]. In addition to describing types of attacks control engineers should be cognizant of, Zhu’s team provides specific guidance in terms of hardware and software vulnerabilities for SCADA systems. The vulnerabilities they determined to be most critical for SCADA include: lack of privilege separation in embedded operating systems, buffer overflow and SQL injection [101]. While these are concrete vulnerabilities that control engineers can seek out to secure across SCADA systems, it is unclear from Zhu’s team’s analysis how they determined these attacks and vulnerabilities were most important for SCADA. The vulnerability list is supported by some examples of SCADA systems that have these vulnerabilities but there is no data-driven evidence that
these are the predominant risks for this class of industrial control system.

Based on existing literature, there is a need to understand the similarities and differences between SCADA and non-SCADA vulnerabilities and exploits. Also, the relationship between First.org’s vulnerability risk metrics and the prevalence of exploits for the software subclass of SCADA systems should be investigated. Further, a data-driven vulnerability prioritization schema for SCADA that is customizable based on an organization’s business parameters is needed to complement NIST’s complex ICS cybersecurity framework.

3.2 Our Contribution

In this paper, we reaffirm other scholarly findings that the CVSS risk metrics are not correlated with exploits for all software vulnerabilities; however, unlike our research colleagues we discover that CVSS risk metrics associated with the software subclass of SCADA systems are strongly correlated with exploit. We demonstrate that certain risk metrics are stronger indicators than others in evaluating the likelihood of exploits for SCADA systems. These metrics are used to generate a customizable prioritization schema for SCADA vulnerabilities. A schema can provide a focal point for security researchers to develop SCADA-specific solutions for the most critical vulnerabilities that extends beyond patching. Patching is not always feasible in the SCADA/IoT environment because these systems must be running at all times and there is little guidance from SCADA vendors on the effect a patch might have on a SCADA system [48, 87]. The vulnerability prioritization schema can also complement NIST’s cybersecurity framework for understanding ICS risk. Finally, by determining the prioritized exploit risk, we can make targeted SCADA IoT software development recommendations for mitigating the associated vulnerabilities.

3.2.1 Experimental Findings

To evaluate the landscape of vulnerabilities, a database was collated from the DHS’ Industrial Control System Computer Emergency Response Team (ICS-CERT) and the MITRE Corporation’s Common Vulnerability and Exposure (CVE) systems. 828 SCADA-relevant CVEs were found across the databases after accounting for duplicates and entries with in-
sufficient information. These CVEs were then classified by their categorical vulnerability type called Common Weakness Enumeration (CWE) which is published by MITRE. This categorization enabled the calculation of a SCADA CWE density which provides insight into the distribution of SCADA vulnerabilities across various CWEs. Risk metrics from NIST’s National Vulnerability Database (NVD) were collected for each CVE based on First.org’s rating methodology. The average risk score across all CVEs in a given CWE were then calculated, which provided average risk metrics for each vulnerability type. Exploits were then web-scraped from ExploitDB [14], CVEDetails [59] and the Metasploit [82] code database yielding 52 exploits across 44 SCADA-related CVEs. These exploits were then categorized by their associated CWE, which allowed for the calculation of an exploit density per vulnerability type (CWE).

A cosine similarity test was run on SCADA versus non-SCADA data to understand if there are differences in the distribution of vulnerabilities and exploits across the systems. The distribution of CWEs for SCADA and non-SCADA were found to be the same. However, the distribution of types of vulnerabilities exploited were shown to be different despite having similar vulnerability profiles. This indicates the importance of the exploit density metric for SCADA CWEs.

Multivariate regression models were then run to evaluate the relationship between various SCADA risk metrics and exploit density. An R2 value of 0.924 which is indicative of a strong correlation was found. The independent variables regressed against the dependent variable, exploit density included: CVE density (number of CVE’s per CWE), average impact score per CWE and average exploitability score per CWE.

These variables were then used to develop the SCADA prioritization schema. The top CWEs by vulnerability density, exploit density, exploitability score, and impact score were assessed and combined to generate the prioritization schema.

In summary, we make the following contributions in this paper:

- SCADA is a unique software subclass with unique attack targets. We statistically validate that exploits for SCADA systems focus on penetrating a specific set of vulnerabilities as compared to non-SCADA systems.
First.org’s CVSS risk metrics can be used to determine the risk of exploit for the software subclass of SCADA systems. Previously, studies concluded in blanket statements that First.org’s Exploitability and Impact scores were not indicative of exploit risk. This finding provides grounds for substantial further work to evaluate the correlation of exploit and CVSS scores for other software subclasses.

SCADA vulnerabilities can be prioritized by data-driven risk metrics in a customizable schema. This has two benefits. First, security researchers could use this schema to understand the greatest SCADA vulnerability risk and orient their research to addressing these vulnerabilities. Second, a customizable schema provides flexibility to organizations and IIoT operators to adjust the vulnerability prioritization based on business parameters. Additional variables can be incorporated to the schema or weights can be applied to tailor the prioritization to a given organization.

SCADA IIoT system developers can use the prioritization schema to easily identify the principal vulnerabilities based on exploit risk from our study and take measures to design systems without these vulnerabilities in the future. We offer technical design recommendations for SCADA IIoT system software developers to mitigate the primary exploit risks we identify. Inherently accounting for these vulnerabilities during SCADA system design will dramatically reduce the potential attack surface for IIoT urban critical infrastructure operations.

3.3 Methodology

3.3.1 Data Collection

Data was first captured on vulnerabilities specific to SCADA systems. Data was collected from publicly available sources including ICS-CERT, MITRE’s CVE and CWE database and NIST’s National Vulnerability Database (NVD). The intention was not only to collate the specific vulnerabilities for SCADA, but also metadata about these vulnerabilities. The types of information collected included: CVE name and number, associated CWE for each CVE, the CVSS base score for each CVE, the Impact score for each CVE and the Exploitability
score of each CVE. SCADA vulnerabilities were determined based on keywords in the description of each vulnerability across the databases. Keywords used included "SCADA" and "Supervisory Control and Data Acquisition". Other variations of these keywords were also used to capture potential misspellings.

There was an interesting discrepancy between ICS-CERT's SCADA vulnerabilities cited and MITRE’s SCADA-related CVEs. As represented in Table 3.1, ICS-CERT was missing 592 SCADA CVEs that were present in MITRE’s database where MITRE was missing 31 SCADA CVEs that were listed in ICS-CERT. This discrepancy could represent a lag between updating the two databases considering vulnerabilities are found more quickly than the database can be updated [58]. However, it could also represent the lack of integration between the two databases as they are independently curated. For purposes of this study, a master list of SCADA CVEs was created by combining the two databases and removing overlapping SCADA CVEs.

<table>
<thead>
<tr>
<th>Table 3.1: ICS-CERT vs MITRE SCADA Vulnerabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICS-CERT</td>
</tr>
<tr>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>Number of SCADA CVEs</td>
</tr>
<tr>
<td>Number of Missing SCADA CVEs</td>
</tr>
<tr>
<td>Total SCADA CVEs</td>
</tr>
</tbody>
</table>

Throughout the course of data collection, other data irregularities were also discovered. Some of the CVEs for SCADA in the MITRE database failed to have CWEs associated with them. This could be due to the CVE being a non-classified vulnerability type. As recently as CWE version 2.8 (as of May 2016 version 2.9 was released), man-in-the-middle vulnerabilities were not a classified CWE, yet 2.9 has been updated to include this CWE. The CWE list is an ongoing project and the absence of some CWEs are likely a function of this. For consistency of the dataset, all CVEs that lacked a CWE were not included in the analysis. While this could skew the results of the research and guide operators towards a specific CWE without accounting for non-CWE-classified vulnerabilities, there is an underlying assumption made that if a CWE does not exist for a class of CVEs, it is not a popular vulnerability. This assumption was further supported by only 57 out of the 885 SCADA vulnerabilities did not
have associated CWEs. Further manual analysis of the CVEs without CWEs confirmed that
the CVEs were not all typologically related thereby dismissing the possibility that a major
type of future CWE is missing.

After cleaning the data set and reconciling the discrepancies across the ICS-CERT and
MITRE vulnerability databases, the master list contained 828 SCADA-related vulnerabili-
ties.

After collecting all SCADA vulnerability data available, a similar process was conducted
on non-SCADA vulnerabilities. The intention of collecting non-SCADA data is to evaluate
the differences and similarities between SCADA prioritization schema and non-SCADA pri-
oritization schema. Considering the thousands of documented non-SCADA vulnerabilities, a
random sample was selected from the MITRE CVE database (excluding all SCADA-CVEs). The
random sample contained an equal number of vulnerabilities to those in the SCADA
master vulnerability list. Similar to the SCADA list, CVEs with missing metadata were
removed from the dataset to preserve consistency.

Once the master list of vulnerabilities was created, a similar list of exploits for the vulner-
abilities was developed. A web-scraper was developed to capture relevant exploits associated
with each vulnerability. The web-scraper pulled data from ExploitDB, CVEDetails and the
Metasploit code database. The intent of the collection was to search for all publicly avail-
able exploits that corresponded to the relevant CVEs on the master list (both SCADA and
non-SCADA). While some CVEs did not have any publicly available exploits associated with
them, others had multiple. In total, for the master CVE list, 44 SCADA CVEs were dis-
covered to have 52 associated exploits (some CVEs had more than 1 exploit) and 103 total
non-SCADA CVEs were found to have exploits.

It is important to note that an inherent limitation of the research is the availability of
publicly available information on both vulnerabilities and exploits. Similarly to how MITRE
contained vulnerabilities that ICS-CERT did not and vice versa, there are likely other sources
of vulnerabilities for SCADA systems that were not captured. The same is true of exploits,
the web-scraper only pulled from a finite source of exploits. Exploits that appear on forums
or on Github were not captured as part of this data collection process. Future work should
include expanding the search for available exploits relevant to SCADA CVEs.
3.3.2 Analysis

For purposes of this study, vulnerability analysis was rolled up to the CWE level. First, the vulnerability density of each CWE was calculated. This was done by dividing the total number of CVEs per CWE by the total number of vulnerabilities. For example, there were 202 CVEs in the CWE “Buffer Overflow”. This was divided by the total number of SCADA vulnerabilities, 828, to determine the CWE density of 24.40%. The density of SCADA CWEs are an indicator of how often these vulnerability types will be found in SCADA critical infrastructure and is important to establishing a prioritization schema. The top 5 CWEs by density are listed in table 3.2.

Table 3.2: Top SCADA CWEs by Density

<table>
<thead>
<tr>
<th>Rank</th>
<th>CWE</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>119: Buffer Overflow</td>
<td>0.244</td>
</tr>
<tr>
<td>2</td>
<td>200: Information Exposure</td>
<td>0.105</td>
</tr>
<tr>
<td>3</td>
<td>20: Improper Input Validation</td>
<td>0.100</td>
</tr>
<tr>
<td>4</td>
<td>79: Cross-Site Scripting</td>
<td>0.063</td>
</tr>
<tr>
<td>5</td>
<td>2: Path Traversal</td>
<td>0.062</td>
</tr>
</tbody>
</table>

While one class of CWE may have the highest density across a system type, it does not necessarily mean that there are exploits associated with these CWEs. Because of this, CWE density may not be what matters most to SCADA operators and security personnel. The density of CWE exploits could provide a better assessment of operational risk considering the exploits are readily available for use by attackers. The same formula was applied to the exploits per CWE. For example, there were 32 exploits associated with CVEs in the CWE “Out-of-Bounds Read”. This was divided by the total number of SCADA exploits, 52, to arrive at the exploit density for “Buffer Overflow” to be 61.54%. The top 5 CWEs for exploit density are listed in table 3.3.

An important observation is that CWE-200: Information Exposure is not listed under the top 5 CWEs for exploit density. This is likely because of the nature of the CWE. Information Exposure is the act of an operator providing credentials to an unauthorized actor. It is a managerial exploit rather than a technical one that can be found in a public database, hence
Table 3.3: Top SCADA CWEs by Exploit Density

<table>
<thead>
<tr>
<th>Rank</th>
<th>CWE</th>
<th>Density</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>119: Buffer Overflow</td>
<td>0.615</td>
</tr>
<tr>
<td>2</td>
<td>200: Path Traversal</td>
<td>0.115</td>
</tr>
<tr>
<td>3</td>
<td>20: Improper Input Validation</td>
<td>0.058</td>
</tr>
<tr>
<td>4</td>
<td>79: Permissions, Privileges, and Access Controls</td>
<td>0.039</td>
</tr>
<tr>
<td>5</td>
<td>22: Code Injection</td>
<td>0.039</td>
</tr>
</tbody>
</table>

The reason it is not covered under top CWEs for exploit density. Because of this, CWE-200 should still be considered a main concern for SCADA systems.

To provide insight for security professionals into SCADA-specific risks, a comparison was made to non-SCADA vulnerability types and their associated exploits. The intention is not to prove that SCADA is entirely different from IT system security, but to inform operators of nuances of SCADA systems.

Based on a side-by-side analysis of the density of CWEs, it is clear that SCADA security professionals should be looking for Buffer Overflow vulnerabilities, compared with non-SCADA which is dominated by Cross Site Scripting. Figure 3-1 illustrates these vulnerability density’s comparing SCADA and non-SCADA.

A comparison of SCADA versus non-SCADA CWE exploit density reveals that SCADA operators should be most concerned with Buffer Overflow vulnerabilities (as they have the greatest risk of having exploits associated with them). This can be compared to non-SCADA systems where it seems that the predominant CWE to have an exploit associated with it is SQL Injection.

The significance of these SCADA versus non-SCADA differences were evaluated by applying a cosine similarity test on the web-scraped data. Cosine similarity measures how similar two non-zero vectors are to each other. The closer the cosine similarity value is to 1 indicates a 0° separation between the two vectors (meaning the data sets are very similar). If the cosine similarity is closer to 0, it indicates that there is a 90° separation between the two vectors indicating the data sets are polarized. For purposes of this study, we will set a threshold of a cosine similarity of greater than 0.5 (indicating a vector angle of 45° or less)
is considered to be "similar" data sets and less than 0.5 as dissimilar data sets.

The cosine similarity of the vulnerability density of SCADA compared with non-SCADA was 0.860. This indicates that the overall distribution of the vulnerability types of SCADA versus non-SCADA are very similar and differences are not significant. However, the cosine similarity of the exploit density per CWE of SCADA compared with non-SCADA was 0.408. Considering the threshold set, we can affirm that the exploit landscape is different.
for SCADA versus non-SCADA in a significant way. This significance magnifies the importance of the CWE exploit density’s role in SCADA-specific prioritization. This shows that despite consistent vulnerability distributions across SCADA and non-SCADA systems, attackers choose to create exploits for distinctly different vulnerabilities for SCADA systems compared to the exploits they create for non-SCADA systems.

In addition to understanding the value of vulnerability and exploit density, the importance of CVSS, Impact, and Exploitability scores to evaluating risk was sought for SCADA systems considering Allodi [2] determined these scores were not strong indicators of exploit for IT systems. To do this, regression analyses were performed on these variables to determine the likelihood that an exploit exists for a given CWE.

Before investigating the SCADA relationship of exploit density and the First.org risk scores, Allodi’s [2] findings were verified by regressing the number of non-SCADA exploits with non-SCADA CWE frequency, CVSS, Exploitability, and Impact scores. Non-SCADA scores by First.org were indeed found to have no correlation with exploit density with an adjusted $R^2$ value of 0.098. The results of the test can be found in Table 3.4.

Table 3.4: Non-SCADA Exploits vs CWE Frequency, CVSS, Impact, and Exploitability Scores

|                      | Estimate | Std. Error | t-value | Pr(>|t|) |
|----------------------|----------|------------|---------|----------|
| Intercept            | -138.387 | 144.248    | -0.959  | 0.365    |
| Frequency            | 0.122    | 0.085      | 1.455   | 0.184    |
| CVSS Score           | -50.304  | 76.508     | -0.658  | 0.529    |
| Impact Score         | 37.497   | 54.048     | 0.694   | 0.507    |
| Exploitability Score | 28.329   | 36.668     | 0.773   | 0.462    |
| Adj. $R^2$           | 0.098    | p-value    | 0.340   |          |

Moving forward to understand SCADA’s relationship with these scores, a test was then performed to understand the relationship between number of SCADA exploits and the SCADA CVSS scores. The hypothesis was that the higher the average CVSS score was for a set of CVEs in a CWE, the more likely there would be exploits associated with the CWE. As a reminder, CVSS scores are metrics of risk evaluated based on factors including
Impact and Exploitability scores for a CVE. However, the CVSS score is not an average or sum of Impact and Exploitability scores. First.org provides the equations for calculating the seemingly complex CVSS scores on their website and it is replicated on NIST's National Vulnerability Database [25].

When conducting a linear regression of CVSS scores on exploits, it was surprising to find no correlation between CVSS scores and exploits with an adjusted $R^2$ value of $-0.074$. This indicated that in our SCADA prioritization schema, CVSS scores should not be a factor in determining which CWEs should be prioritized.

Next, a regression was run to determine if the number of vulnerabilities per CWE, the average Impact score for CVEs related to a respective CWE and the average Exploitability score for CVEs related to a respective CWE were correlated with a CWE having exploits. Similar to the assumption with the CVSS scores' relationship with the presence of exploits, the hypothesis was that a high number of vulnerabilities and high Impact and Exploitability scores were correlated with the existence of an exploit for a given CWE. In this case, the multiple regression model corroborated the hypothesis with an adjusted $R^2$ value of 0.924 showing a strong relationship between the presence of an exploit and the number of vulnerabilities for the given CWE, the average Impact score and Exploitability score. The results of the analysis can be found in Table 3.5. These results were surprising as they indicate that there is something unique about SCADA CWE frequency, Exploitability and Impact scores' relationship with exploit density that is not true of IT systems as found by Allodi [2]. Further, this indicates that in First.org's complex equation that converts Impact and Exploitability scores to CVSS scores, the correlation with the presence of an exploit for a given CWE is lost. This could suggest that the CVSS score is a flawed indicator of risk whereas the Exploitability and Impact scores are not (assuming risk can be accessed via the presence of an exploit as per the suggestion of this study).

To further validate the assertion that CVSS scores do not correlate with the presence of an exploit, other multiple regressions were run regressing exploits on variations of CVSS scores and other variables. All of these regressions consistently showed a weak relationship between exploits and CVSS scores, even when coupling CVSS scores with Exploitability and Impact scores.
Table 3.5: SCADA Exploits vs CWE Frequency, Impact, and Exploitability Score

|                    | Estimate | Std. Error | t-value | Pr(>|t|) |
|--------------------|----------|------------|---------|----------|
| Intercept          | -22.490  | 7.225      | -3.113  | 0.014    |
| Frequency          | 0.167    | 0.015      | 11.424  | 3.12E-6  |
| Impact Score       | 0.642    | 0.565      | 1.137   | 0.288    |
| Exploitability Score | 1.717   | 1.033      | 1.661   | 0.135    |
| Adj. R²            | 0.924    | p-value    | 2.27E-5 |

Based on this analysis, the magnitude of Exploitability and Impact scores for a given CWE are important. The top ten CWEs for Impact score and Exploitability score can be found in rank order in Table 3.6. It is interesting to note that while the top 10 CWEs for Impact and Exploitability are not the same rank, all top Impact score CWEs are also found in the top Exploitability score CWE list and vice versa.

Table 3.6: Top 10 SCADA CWEs by Impact and Exploitability Score

<table>
<thead>
<tr>
<th>Rank</th>
<th>CWE by Impact Score</th>
<th>CWE by Exploitability Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>119: Buffer Overflow</td>
<td>119: Buffer Overflow</td>
</tr>
<tr>
<td>2</td>
<td>20: Improper Input Validation</td>
<td>20: Improper Input Validation</td>
</tr>
<tr>
<td>3</td>
<td>264: Permissions, Privileges, and Access Controls</td>
<td>200: Information Exposure</td>
</tr>
<tr>
<td>4</td>
<td>200: Information Exposure</td>
<td>22: Path Traversal</td>
</tr>
<tr>
<td>5</td>
<td>22: Path Traversal</td>
<td>79: Cross-Site Scripting</td>
</tr>
<tr>
<td>6</td>
<td>255: Credentials Management</td>
<td>264: Permissions, Privileges, and Access Controls</td>
</tr>
<tr>
<td>7</td>
<td>399: Resource Management Errors</td>
<td>399: Resource Management Errors</td>
</tr>
<tr>
<td>8</td>
<td>287: Improper Authentication</td>
<td>255: Credentials Management</td>
</tr>
<tr>
<td>9</td>
<td>310: Cryptographic Issues</td>
<td>287: Improper Authentication</td>
</tr>
<tr>
<td>10</td>
<td>79: Cross-Site Scripting</td>
<td>310: Cryptographic Issues</td>
</tr>
</tbody>
</table>
3.3.3 Scoring

To develop a SCADA prioritization schema, the above analysis was used to evaluate which variables are most relevant to determining the SCADA IIoT risk. The variables of CWE density, CWE exploit density, Impact score and Exploitability score were ultimately used. Additional variables can be included for a prioritization schema if data is available and the data is found to correlate with exploit density. While there are many options to determine how to score each variable for the prioritization order, for purposes of this paper, a rudimentary system was selected intentionally for transparency. More sophisticated weight-based prioritization schemes can be created and customized for various organizations. The purpose of this study is not necessarily to generate the “correct” or ultimate prioritization order for SCADA system vulnerabilities, rather it is to establish a framework for how a data-driven study can be used to develop customized SCADA risk prioritization schemes. Future work is encouraged to address how to weight each variable for the prioritization schema.

Point values were assigned based on the ranked position of the CWE in each category. Each category (i.e. CWE density, CWE exploit density, etc.) were weighted equally. For purposes of this analysis, the top 5 CWEs from each category were ranked where the top ranked CWE receives a point value of 5 and the 5th CWE in the ranking receives a value of 1.

The top 5 ranked CWEs can be found for all 4 categories in Table 3.7 and the total allocated points per CWE can be found in Table 3.8. Figure 3-2 represents the steps required to generate the prioritization schema including the inputs and outputs of the model.

This prioritization schema for SCADA vulnerabilities logically makes sense based on the characteristics of SCADA operations. A closer look at the top three prioritized SCADA vulnerability types helps illustrate this. Buffer overflows are defined as a vulnerability where software can read or write to a memory location that is outside the intended boundary of the memory buffer. It is not surprising that buffer overflows warrant the highest priority for SCADA vulnerabilities as buffer overflows are inherent in older, low-level programing languages such as C which is common to SCADA. Further, SCADA devices are rarely rebooted due to their constant operating requirements. Systems that have not been rebooted for years
Table 3.7: Top 5 Ranked CWEs per Category

<table>
<thead>
<tr>
<th>Rank</th>
<th>CWE by Frequency</th>
<th>CWE by Exploit Density</th>
<th>CWE by Impact Score</th>
<th>CWE by Exploitability Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>119: Buffer Overflow</td>
<td>119: Buffer Overflow</td>
<td>119: Buffer Overflow</td>
<td>119: Buffer Overflow</td>
</tr>
<tr>
<td>2</td>
<td>200: Improper Input Validation</td>
<td>22: Path Traversal</td>
<td>20: Improper Input Validation</td>
<td>20: Improper Input Validation</td>
</tr>
<tr>
<td>3</td>
<td>20: Improper Input Validation</td>
<td>20: Improper Input Validation</td>
<td>264: Permissions, Privileges, and Access Controls</td>
<td>200: Information Exposure</td>
</tr>
<tr>
<td>4</td>
<td>79: Cross-Site Scripting</td>
<td>264: Permissions, Privileges, and Access Controls</td>
<td>200: Information Exposure</td>
<td>22: Path Traversal</td>
</tr>
</tbody>
</table>

Figure 3-2: Prioritization Schema Steps

* For the system of interest, determine the top 5 CWEs by frequency, CWEs with highest exploit density, CWEs with highest impact and CWEs with highest exploitability score.

* For each CWE category, assign points per CWE. The points are assigned based on their ranking and any weightings applied to the CWE categories.

* Add the total points for each CWE to determine which CWEs should be prioritized. Then determine which IIoT systems in use contain the prioritized CWEs to rank system risk.
Table 3.8: Total Scores for Top-Ranked CWEs

<table>
<thead>
<tr>
<th>CWE</th>
<th>Total Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>Buffer Overflow</td>
<td>20</td>
</tr>
<tr>
<td>Improper Input Validation</td>
<td>14</td>
</tr>
<tr>
<td>Information Exposure</td>
<td>9</td>
</tr>
<tr>
<td>Path Traversal</td>
<td>8</td>
</tr>
<tr>
<td>Permissions, Privileges, and Access Controls</td>
<td>5</td>
</tr>
<tr>
<td>Cross-Site Scripting</td>
<td>3</td>
</tr>
<tr>
<td>Code Injection</td>
<td>1</td>
</tr>
</tbody>
</table>

will accumulate memory fragmentation. This makes devices substantially more vulnerable to buffer overflow vulnerabilities [101]. Improper input validation is when software does not check input which enables an attacker to enter values that could cause control flow changes that are not expected by an operator. Considering one of the key differentiators of ICS versus IT systems is that industrial control systems are deterministic, this vulnerability is clearly a threat [27]. SCADA systems require low jitter and any disruption of the deterministic processes such as an attack exploiting the vulnerability class of improper input validation would severely impact operations. Finally, information exposure is the disclosure of information to an unauthorized person. This vulnerability type is also logical for SCADA considering the prevalence of default usernames and passwords used across systems [101]. Because default usernames and passwords are frequently used, attackers can easily obtain this information from an instruction manual or from a vendor discussion forum. Also, information exposure as a prioritized exploit is logical considering the prevalence of phishing attacks used to collect credentials from critical infrastructure operators. This was seen for the Ukrainian electric grid cyberattack and UglyGorilla’s cyber espionage program against 23 U.S. natural gas pipelines [15,50].

While information exposure is a borderline priority with path traversal, it is important to remember that information exposure lacked technical exploits publicly available in the databases searched because it is more of a managerial exploit than technical. Therefore, it was not appropriately captured in the exploit density data set, and indeed belongs at the
3.4 Research Implications

3.4.1 Operator Implications

This research, while niche to a subsector of IIoT, can have considerable impact for urban critical infrastructure security. Our findings indicate that there is a strong relationship between First.org risk metrics and exploit density, specifically for SCADA systems. There are three groups of critical urban infrastructure security experts that can benefit from this insight: Chief Information Security Officers (CISOs), Security Operations Center (SOC) Analysts and System Architects.

CISOs who oversee all security operations of an organization generally have the difficult responsibility to develop and manage programs to secure the organization at scale. Because of our findings, CISOs can streamline their programs for securing SCADA systems. Rather than establishing programs meant to help create metrics that can be used to assess the risk of various IIoT systems, CISOs could instead refer to First.org’s metrics of Exploitability and Impact to evaluate IIoT risk of exploit. There is no longer a need to start from scratch developing metrics considering we demonstrated that Exploitability and Impact metrics are valid predictors of exploit risk for SCADA systems.

SOC Analysts are another group of security experts that can benefit from our findings. SOC Analysts are often responsible for monitoring and fixing security risks as they occur. Instead of reactively seeking out security threats to address, our risk prioritization schema will help Analysts proactively seek out which IIoT systems are likely to be attacked. SOC Analysts can cross-check IIoT devices with CVEs and CWEs that we identified to be most exploited to arrive at their prioritized device list.

System Architects responsible for selecting components for urban critical infrastructure should use our findings to carefully select systems based on their vulnerability profile. While we acknowledge most urban critical infrastructure IIoT consists of legacy devices that are not often replaced, when new devices are procured, our risk prioritization schema can be used to
assess which SCADA systems should be installed. IIoT devices with the most vulnerabilities in the categories we discover to be of highest risk of exploit should be avoided.

3.4.2 Technical Design Implications

Future SCADA IIoT systems should be designed and developed with the intent to “design out” the prioritized vulnerabilities indicated in our study. Addressing the prioritized vulnerabilities in the design phase could help reduce the number of future attacks against this class of IIoT. Based on recommendations of the top three prioritized vulnerabilities of buffer overflows, improper input validation and information exposure, we can propose technical design strategies to help avoid these vulnerabilities.

Buffer overflows are prevalent in operating environments that are programmed in C. The language provides direct memory access, which can be used to help reduce the device’s energy consumption. Energy efficiency is important for the cost efficiency of SCADA systems especially considering their highly distributed nature in locations where resource availability might be limited. Further, C can be very memory efficient, which is also valuable for small devices required for urban critical infrastructure. Despite these benefits of C, the buffer overflow vulnerabilities that result from coding mistakes are a considerable downside. This prioritized vulnerability can be “designed out” by using a memory safe programming language when developing future SCADA systems. One memory safe language that is also memory efficient is Rust [52]. If future IIoT systems can be programmed in Rust, buffer overflows will no longer be an issue therefore removing this attack vector for IIoT SCADA systems.

SCADA design traditionally focuses on detecting and classifying control conditions that enables accurate monitoring in various states [40]. With focus on the functional operation of the SCADA system, proper input to the system is assumed and not accounted for in the design process. With increased skepticism of IIoT device inputs based on recent attacks, and the associated vulnerabilities involving improper input, SCADA designers must take measures to validate input. Design recommendations that could reduce the number of improper input validation vulnerabilities in systems include using an input validation framework such as Struts or OWASP ESAPI Validation API when creating the system or by identifying all possible areas where an attacker could input data and employ a whitelist strategy [60].
Frameworks like Struts help to guide software development so that there are few validation issues. A whitelisting strategy entails rejecting all inputs other than the few that are actually appropriate for the design specifications of the system’s purpose. The whitelist should account for all input properties ranging from length to syntax.

Information exposure may perhaps be the most challenging vulnerability to “design out” of a SCADA system. This is because many information exposure attacks happen as a function of the human element either by error or intentionally. A potentially effective mechanism to mitigate the damage caused by information exposure is to compartmentalize data systems [61]. Designing SCADA IIoT to be compartmentalized can limit the data leak or attack to only the compartment that was breached. If a centralized data store for SCADA IIoT is used, compromised access to the central hub will leave all data vulnerable.

These proposed SCADA IIoT technical design strategies may help to reduce the prevalence and risk of the top vulnerabilities identified in this study. Each SCADA designer will need to evaluate if these strategies can be used based on their specific technology requirements as not all design mitigation techniques will necessarily be appropriate for every IIoT system.

3.5 Conclusion

Unique contributions of this study are significant for security researchers investigating SCADA systems, SCADA IIoT designers and critical infrastructure operators working with IIoT. The research reveals that SCADA systems as a software subclass were found to have exploits that target a distinct set of vulnerabilities compared with non-SCADA systems. This indicates that the risk profile for SCADA systems varies compared with that of non-SCADA. The study also identifies highly correlated relationships between First.org vulnerability risk metrics and the density of SCADA exploits. These findings could encourage security researchers to reconsider their assertions that Exploitability and Impact scores are inaccurate predictors for the risk of exploit. Researchers should repeat these studies on risk metrics’ relationship with exploits specifically for subsets of software as was done for SCADA. Finally, findings suggest that security researchers, SCADA IIoT designers and SCADA operators should focus on a
core set of vulnerability types for SCADA systems. Considering the unique requirements of SCADA systems and the associated challenges with vulnerability patching, alternative security strategies concerning prioritized vulnerabilities should be investigated. The prioritization framework provided can be customized based on organizational requirements and parameters. Urban critical infrastructure operators can use the prioritization in parallel with NIST’s more comprehensive cybersecurity framework to understand their SCADA risk.

Because the SCADA prioritization schema is based on empirical, data-driven findings, it will need to be updated continuously as new exploits are published. If a series of new SCADA exploits are released that target a specific vulnerability class, the prioritization schema will be outdated. It is recommended that this prioritization is updated annually as was the CWE/SANS Top 25 list.

There are several future research opportunities related to this study. CVSS, Exploitability and Impact scores are being transitioned from version 2 to version 3 which entails new scores that are more specific. Once this new scoring methodology has been completed and vetted for accuracy, this study should be repeated with updated data so that the Exploitability and Impact scores can be normalized appropriately. Testing additional characteristics of vulnerabilities as variables to determine their association with the risk of exploit could be included in future work. As previously indicated, other sources of exploits can be compiled from repositories such as Github or sources that may reference managerial related exploits rather than technical ones to better capture the exploit potential of CWEs such as Information Exposure. Future research could also investigate the scoring mechanisms used for the prioritization schema, which can be further customized through weightings and new point allocation systems. Finally, further studies should investigate opportunities to incorporate this SCADA prioritization approach to the existing NIST framework to provide a data-driven approach to evaluating system risk. This should accompany IIoT security policy research intended to encourage a robust, quantitative approach for evaluating urban critical infrastructure risk.
Chapter 4

Cyber Negotiation: Defending Urban Critical Infrastructure from Cyberattacks by Negotiating with Cyberterrorists

4.1 Introduction

Cybersecurity is often portrayed as a "cat and mouse" game that tests each side's relative technical prowess; however, it can also be considered a battle of social wits. Humans are behind every hack, and attackers have human motivations. The social science that ought to inform every effort to deter and engage attackers can easily be overlooked when cybersecurity experts focus exclusively on technical exploits and ways to counter them (i.e. hackbacks). We call the social science tools and techniques available to defenders, Defensive Social Engineering (DSE). While the most commonly discussed DSE tool is education, we will focus on a tool that is rarely given much attention in the cyber community: negotiation.

While many cyberattacks leave most defenders with no idea about the attacker they are dealing with, or their whereabouts, ransomware attacks are a bit different. Ransomware generally involves solicitation of a payment in return for release of data that have been stolen or
restoration of services that have been rendered inoperable (by malware). Some ransomware attackers even provide a channel of communication for the defender to connect with them via untraceable means. While this is not always the case, an invitation to communicate with a hacker provides an opportunity to negotiate. Urban critical infrastructure and services, including electric grids, water networks, transportation systems and emergency services, including police departments and 911 assistance, are all vulnerable to attack because they are digitized with "smart" sensors and control systems connected to the internet [4]. Smart city infrastructure is a prime target for cyber terrorists eager to hold valuable assets for ransom. For example, 123 of 187 Closed Circuit Televisions (CCTVs) were compromised across Washington D.C. prior to the 2017 Presidential Inauguration [99]. CCTVs are video cameras strategically placed across the city and used for security surveillance. The ransomware attack that effectively disabled these CCTVs disrupted the police departments involved as well as the president’s security detail from retaining oversight of potential threats while their systems were down.

Even more recently, the City of Atlanta sustained a ransomware attack on March 22, 2018. The ransomware locked down city computers and their associated services for many city functions and demanded approximately $51,000 delivered in Bitcoin to return access [31]. Affected services ranged from the ability to accept payments for fines or fees to processing various requests online. The city jail had to resort to using pen and paper for operations [54]. The city did not pay the ransom and this resulted in some services being offline for over two weeks. As of May 5, 2018, the total cost to recover from the attack has surpassed $5 Million [75].

Negotiation in the cyber realm presents significantly different challenges from person-to-person negotiations in the boardroom or elsewhere. There is no chance to read the face of the attacker. Defenders may have limited opportunity to negotiate in real time and, so, will have little chance to ascertain the culture or values of the hacker. In such situations, preparation is the most important step a defender can take. Cyber defenders must prepare ahead of time for as many hacking scenarios as possible – from efforts to steal sensitive data to attempts to hold a whole city’s electric grid hostage. Negotiating with cyberterrorists requires pre-attack, mid-attack and post-attack preparation. Critical infrastructure oper-
ators need to work out strategies before an attack (to clarify their negotiation authority),
trigger a pre-prepared negotiation protocol during an attack and reflect, rebuild and modify
their negotiation strategy based on what they learn afterwards. These defensive techniques
require every bit as much attention as technical defenses.

To establish a playbook of cyber negotiation moves for critical infrastructure and service
managers, we conducted an extensive search and evaluation of past ransomware attacks. We
paid close attention to instances in which negotiation ensued or where there were missed
opportunities for cyber negotiation that could have helped mitigate the damage caused by
the ransomware attack. We also asked critical infrastructure and services operators to help
us understand the cyberattack defenses they have put in place by participating in simulated
ransomware attacks. The intent of these interviews was to understand how cyber specialists
in a range of critical urban infrastructure agencies actually implement their cyber defense
schemes. We used screen captures of actual ransomware attacks to build our hypothetical
scenario. The responses of our interviewees are the basis for this study as well as a future
negotiation playbook we are preparing.

4.2 Background

4.2.1 The Importance of Cyber Defense

The need for active cyber defense emerges, in part, from the lack of a clear definition of
boundaries in cyberspace. Cyberspace is a dynamic environment where new additions (and
removal) of networks, servers and embedded systems constantly reshape the landscape. Be-
cause of the relatively open nature of the internet and the lack of international security
standards, it is not always clear how cyberspace assets can be used and who should be al-
lowed access to them. This encourages hackers to push the boundaries of what might be
fair use of a network or system. In response, the creator/owner of each bit of cyberspace
must be proactive about defining their cyberspace boundary – resisting unwanted actors or
information. The constant push and pull of boundary definitions in cyberspace – between
hackers and cyberspace owners – requires organizations to adopt a strategy of "cyber per-
sistence” that defines the attack/defend cycle [26]. The complexity of virtual cyberspace, unlike a physical domain, creates a low barrier to entry. This results in many diverse actors constantly attempting to grab cyber territory by challenging the boundaries of ownership. These cyber attackers can range from an individual, to a terrorist organization to a nation-state.

Because of the disagreement about boundaries and norms, especially concerning when and how force can be used in cyberspace, urban critical infrastructure in the United States is constantly subject to cyberattacks. For instance, the United States and its allies subscribe to the United Nations Charter article 2(4) which asserts that ”operational restraint” should be the norm with regard to internationally imposed force. However, it is not clear if this norm should be consistently applied to cyberspace where ”force” is defined as a cyberattack [26]. The definition of force in United Nations Charter article 2(4) is the subject of debate. The three presiding interpretations include: force as armed violence, force as coercion, and force as interference [97]. Since the Cold War, the United States roughly interprets force as armed violence and has not taken a definitive stance on how cyberattacks should be classified. This ambiguity provides little reason for countries to demonstrate cyber restraint.

United States cyber policy under President Obama emphasized a strategy of deterrence. The U.S. sought to avoid the overt use of force against cyber adversaries [26]. While thinking about the best way to deter cyberattacks continues to evolve, private actors are basically free to respond to cyberattacks in whatever way they want. Indeed, there has been a sizeable growth in the market for cyber security technology and protection services [28]. U.S. private companies will likely continue looking to the market to help defend their cyberspace boundaries and remain cyber persistent until policy catches up to the current state of the attack/defend cycle.

While there are a great many technological security solutions available in the marketplace which have been adapted to urban critical infrastructure needs, the number of cyber breaches continues to grow. One reason why breaches persist, in our view, is that technology solutions only focus on protecting against late-stage attacks (i.e. where attackers are very close to achieving their objective). This is not the case with early-stage attacks. The full set of stages of attack is documented in Lockheed Martin’s Cyber Kill Chain [36].
Lockheed Martin created the Cyber Kill Chain to describe the "process flow" of a cyberattack. Each stage contains categories of steps that a hacker follows in order to achieve their goal [36]. The steps of the Cyber Kill Chain can be found in Figure 4-1 below.

![Cyber Kill Chain Diagram](image)

Figure 4-1: Lockheed Martin's Cyber Kill Chain.

An example of the Kill Chain applied to an attack follows. In order to achieve the goal of shutting down the electric grid, an attacker would start by conducting reconnaissance (stage 1 of the Kill Chain) of the grid in order to learn about its network topology and what connected devices could be attacked to achieve their objective. The attacker would then craft an exploit involving weaponizing a file (stage 2) and delivering it (stage 3) to a penetrable part of the grid. Such an exploit is used to gain access to a system (stage 4) and install malware (stage 5). After malware is installed, a hacker would probably tunnel through the network to access a component of the grid they want to compromise. They then would issue command and control mandates (stage 6) to fault the system, resulting in a power outage – their overarching objective (stage 7). Cybersecurity technologies generally focus on defending during the latter half of the kill chain. This leaves many early stages of the attack or the final stage exposed. This is especially true with regard to ransomware attacks.

To address the earlier phases of the kill chain, non-technological solutions such as education have been proposed [5]. Education refers to raising employee awareness about phishing campaigns that attackers can use to trick employees into disclosing confidential credentials.
Cybersecurity frameworks such as the National Institute of Standards and Technology’s (NIST) Framework for Improving Critical Infrastructure Cybersecurity barely mention non-technology defenses. In a section of the Framework called Awareness and Training [76] they offer the following description: "The organization’s personnel and partners are provided cybersecurity awareness education and are adequately trained to perform their information security-related duties and responsibilities consistent with related policies, procedures, and agreements."

They list five somewhat ambiguous steps aimed at meeting Awareness and Training objectives:

1. all users are informed and trained;
2. privileged users understand roles and responsibilities;
3. third-party stakeholders (e.g., suppliers, customers, partners) understand roles & responsibilities;
4. senior executives understand roles & responsibilities, and
5. physical and information security personnel understand roles & responsibilities.

This is the only non-technology defense proposed by NIST. We believe others can be used to help defend organizations against cyber threats. Considering the recent popularity of ransomware as a form of malware, which usually involves some form of human interaction as part of the attack, we think defensive social engineering (above and beyond education) is crucial to protecting critical urban infrastructure.

4.2.2 The Nature of Ransomware

Ransomware is a type of malware that takes a cyber asset hostage [51]. A cyber asset has been taken hostage when it has been made unusable by an attacker. From an Information Technology (IT) perspective, this usually involves encrypting an organization’s data. Once data are encrypted, a key is required to decrypt the files and enable access once again. Sometimes the data are not encrypted but actually destroyed or exfiltrated by an attacker [9].
Hackers make money by charging a fee for the key to decrypt the files they have secretly encrypted.

Ransomware only recently became a major threat, in part because of the emergence of Bitcoin, a pseudo-anonymous payment mechanism and Tor, a tool used to anonymize internet activity. Together, Bitcoin and Tor provide a means for attackers to extract and transfer ransom payments while maintaining their anonymity. Prior to Bitcoin and Tor, it was relatively easy to track down a hacker by following a ransom payment back to the hacker. If a hacker wanted to go unnoticed, they would have to generate a sophisticated money laundering scheme. Today, a hacker can create a Bitcoin wallet which is untraceable if accessed via Tor, request ransom payments be deposited into that wallet, and then close the wallet. This untraceable payment scheme, along with "ransomware-as-a-service" kits, make ransomware a profitable and simple attack mechanism. Ransomware kits are sold on various darkweb/black market sites by criminals, terrorist organizations and even nation-states. While there are many different kits available, they are all intended to make launching a ransomware attack "plug and play." There are even sophisticated economic models set up by would-be criminals to help fund and distribute ransomware [9]. One such economic model for funding ransomware distribution is cost-sharing. The purchaser of the ransomware kit might pay nothing for the kit (which could include the ransomware program and perhaps an exploit to help get the ransomware installed on a system). Instead, the ransomware technology would be set up to distribute payments by victims to both the creator of the kit and the ransomware distributor. This creates a very low barrier to entry for distributors of ransomware.

The simplest method of foiling a ransomware attack is to be certain that all of an organization’s data are continuously backed up and stored on a separate network or in a location that is remote and entirely disconnected or air-gapped. While it may seem obvious that organizations should backup their systems on a regular basis, almost 63% of organizations do not do this [41]. While many IT systems will not suffer substantially when a transition is made to a backup system, most urban infrastructure is classified as Operational Technology (OT), not IT. OT involves cyber-physical systems (i.e., digital devices that impact the physical world), which generally cannot be taken offline while transitioning to a backup
system without causing a substantial impact on operations. Taking OT offline can result in severe consequences such as a power outages or sewage leaks. An unwillingness to create and maintain back-up systems because of the difficulties of transitioning to them amplifies the risk that ransomware will be used in a cyberattack when OT systems are involved. If ransomware infects even a small part of an urban infrastructure network, the whole system may have to be taken offline to install a backup. This could have profound physical and financial implications [42]. So, while backing up systems is, in theory, a good way to forestall the adverse effects of ransomware attacks, critical urban infrastructure systems have special features that make this less than an ideal strategy. NIST’s guidance on disaster recovery for OT acknowledges that because it may be slow to restore software for critical systems or that something may go wrong, it might justify the need for running a parallel system [92].

An example of how long it can take to restore a critical urban infrastructure system is offered by a ransomware attack against a major transportation authority in 2016 [84]. The transportation authority was able to maintain system operations during the attack by resorting to handwritten bus route assignments, but it chose to take down fare payment systems from Friday, November 25 to Sunday, November 27th while restoring its software. This resulted in revenue losses of about $50,000 [84]. While the transportation authority refused to pay the 100 Bitcoin ($73,000) demanded by the hacker, many businesses that are ransomwared do pay. According to a 2016 IBM study, 70% of executives from (private) companies who have been ransomwared paid the hacker [37]. Another example of the challenges restoring embedded systems is how on February 10, 2018, traffic lights across New York City began flashing red and yellow [1]. The improper function of these lights was caused by a software patch for a security vulnerability that inadvertently introduced a software bug into the light system. Traffic light downtime could have caused motorists and pedestrians serious inconvenience or dangerous accidents. The NYC Transpiration Department had to engage contractors throughout the day to fix the disruption [1]. While this was not the result of a cyberattack, it further illustrates how OT systems can take time to repair.

For ransomware to encrypt data, an attacker must gain access to the critical infrastructure. To gain access, hackers use a number of techniques from various kinds of social engineering to exploiting technical vulnerabilities. A popular method of social engineering is
phishing. An attacker will disguise themselves as an employee of the organization they are seeking to attack and request access credentials from an operator. After gaining access, an attacker can cause harm in many ways. Beyond encrypting data files or corrupting firmware, an attacker can steal data [20].

In the case of ransomware attacks against urban infrastructure, it is not always easy to gauge the health of a cyber asset that has been or is being held hostage. This is largely because the defender has no way to determine what is happening to their data while it is inaccessible to them. While data are potentially recoverable once a decryption key is provided by the attacker (after ransom has been paid), an asset could be permanently compromised from a reputational and competitive advantage standpoint even if they data are returned intact. For example, if a hacker compromised an electric utility that was testing the valuable intellectual property of a third-party energy supplier, it is possible the hacker might steal documentation of this intellectual property and sell it on the black market. This would cause irreparable damage to both the energy supplier and the utility. The energy supplier could lose their market advantage (provided by the intellectual property that was stolen). The utility could also suffer reputation loss as a trustworthy partner, making it difficult to procure new market-leading intellectual property from other energy suppliers. Thus, damage caused by a cyberattack can extend well-beyond immediate damage caused by system downtime.

After an attacker targets a system and encrypts the data such that the asset is compromised, the operator usually has the option of paying a ransom or restoring functionality by deploying a backup of the system involved. There is no reason to believe, though, the hacker won’t be back. How can a defender of urban critical infrastructure know if their asset control has been fully reestablished? A recent ransomware trend indicates that once a ransom is paid, attackers lay dormant in the system they hacked and reappear, instigating the same kind of attack again [9]. Hackers often aim for persistence in the devices they infect. Persistence is when hackers stealthily remain on the victim’s network or device after an attack is supposedly over in order to re-launch additional attacks. There is no simple way to know if a hacker still has access to a system after an attack. This is why many organizations choose to re-flash the firmware of their devices after an attack and rebuild. This was the case in the CCTV hack in Washington DC [99]. This differs from a human hostage scenario because it
is unlikely that a hostage taker will have indefinite access to the human they initially took hostage. We will now explore additional parallels and differences between ransomware and human hostage-taking.

4.2.3 The Opportunity To Negotiate

Hostage negotiation usually focuses on situations in which people have been taken by physical force against their will. The obvious goal for a hostage negotiator is to recover the hostage in a healthy state. As hostage negotiators press for a hostage’s release they employ a series of tactics aimed at ensuring that the victim stays safe, such as requesting proof of life or demanding to speak directly with the hostage. In most hostage negotiations, while ascertaining the health of a human hostage is by no means easy, at least hostage health is more or less quantifiable. This is not the case when cyber assets are held hostage. As previously described, there is no way to determine what has been done to the data during the period it has been held (i.e. have the data been sold, shared, corrupted?). While we can draw parallels between piracy, hostage-taking, and cyber negotiations, they are distinct.

Bans on Ransom Negotiation

Federal statutes bar the payment of maritime piracy ransoms [45]. Should the official US stance on non-payment to maritime pirates apply to ransomware attacks as well? The basis of the ban rests on the theory that paying ransom encourages repeat and copycat behavior. Individual Somali pirate crew members were taking home an average of $50,000 each in 2011 [18]. They were engaged in a lucrative business with few incidents of successful prosecution or retribution. According to the World Bank, as much as 20% of ransom proceeds are put aside to fund future attacks [18]. Despite investing in deterrence such as navy escort patrols, rerouting shipping routes, and faster steam engines, pirate attacks continue.

The US and UK government decided that that the best way to deter would-be pirates from choosing piracy as a career ”and thereby protecting seafarers from continued threat of hijackings and hostage-takings” was to eliminate ransom payments. Of course, banning ransom payments to hostage-takers may have put innocent lives at risk. In the case of
ransomware attacks against hospitals in the United States, all patient data are frozen and hospital administrators responsible for life or death outcomes may be willing to pay to unencrypt systems that have been taken hostage [22]. While banning piracy ransoms may be good for the world community, it only works if everyone participate. Thus, the policy raises a collective action problem. Some states or individuals may prefer to pay ransom in the short-term. If they do, it undermines the effectiveness of the policy for everyone else [18].

Human Hostage vs. Digital Hostage

There is a substantial literature concerning hostage negotiation. Mostly, it urges negotiators to do what they can to move out of a state of crisis – ”high emotions and low rationality” – to a situation in which conversation can take place and accommodation can be achieved. The goal is for the hostage negotiators to position themselves as deal brokers and identify mutual interests with the hostage-taker [96]. Ransom negotiation almost always involves moving through a stage of communication and developing rapport with the attacker, buying time (i.e. pushing back whatever deadline has been set), defusing intense emotions, and gathering intelligence to ascertain the optimal negotiation or intervention strategy [44]. The Behavior Change Stairway Model (BCSM), developed by the FBI’s Crisis Negotiation Unit, explains how this kind of relationship-building works. Getting through five stages (i.e., active listening, empathy, rapport, influence, and behavioral change) hinges on establishing open lines of communication between attacker and negotiator [96].

Experienced hostage negotiators have a repertoire of tactics at their disposal. Again, success depends on establishing sufficient contact to influence some level of behavioral change. Possible negotiation tactics include laying the groundwork for an interest-based approach to negotiation (rather than positional bargaining), or what is traditionally referred to as ”principled negotiation.” This requires defining a problem in terms of both side”s interests in order to find a solution that satisfies both, with the ultimate goal of generating options that will mutually benefit both sides [32]. With regard to ransomware negotiation, this tactic requires understanding the motivation of the attacker.

While many of the ideas from human hostage negotiation can inform negotiation in ransomware situations, most of the tactics involved assume physical (vocal, or otherwise)
human contact between the hostage-taker and victim or the hostage-taker and the negotiator. It is this contact that is used to generate empathy that skilled negotiators use to build trust. It turns out that empathy and trust have been established in some ransomware “experiments” done by a team at F-Secure. They posed as a ransomware victim and got attacker-agents to extend payment deadlines, lower ransoms (3 out of 4 crypto-ransomware gangs negotiated leading to a 29% average discount), and provide step-by-step assistance regarding how to pay in bitcoin after feigning ignorance [21]. The hacker who targeted the previously mentioned transportation authority was himself hacked shortly after his ransomware attack against the transportation authority [42]. The hacked emails of the attacker revealed that other victims targeted along with the transportation authority successfully negotiated down their payments. Another victim, China Construction of America Inc. was ransomed for 40 Bitcoin, but ended up negotiating a payment of only 24 Bitcoin [42].

While there are very few studies detailing actual ransomware negotiations, there is considerable precedent for “working out a deal” in traditional hostage/ransom negotiations [16]. This is particularly true in the context of piracy or terrorism. The difference between the two is important. Piracy is a means of achieving a financial gain, while terrorism is usually aimed at promoting an ideology. In a study of ransomware that targeted healthcare systems, DeMuro argues that ransomware should be likened to piracy rather than terrorism since it is almost always a matter of doing business for hackers [16]. This is further supported by a detailed cost-benefit analysis conducted by Carbon Black, a leading cybersecurity firm, on the topic of ransomware kit sales sold on the black market. The Carbon Black study concluded that there has been a 2502% increase in ransomware kit sales between 2016 and 2017 and that the increase was the product of supply and demand [9]. Cybercriminals see ransomware as a quick and easy way to make money without much risk of being caught (by using bitcoin). These studies also indicate that when financial gain is the motive of a cyberattack, piracy negotiation strategies might be most appropriate; when financial gain is not the principal motive, terrorism negotiation strategies should be used.

Next, we will discuss how seven critical urban infrastructure operators responded to the ransomware scenario we posed. We wanted to see rather than hear about their strategy for negotiating with hackers. We then use what we learned from these interviews to analyze
actual situations in which critical urban infrastructure operators responded to actual ransomware attacks. Our analysis of the interviews and cases concludes with a set of suggested guidelines that form the basis of our emerging cyber negotiation playbook for critical urban infrastructure operators.

4.3 Research Plan and Interview Protocol

To understand how critical urban infrastructure operators are likely to handle a ransomware attack, we developed a hypothetical ransomware scenario and simulated an attack (with permission) using screen captures from past attacks. Before initiating the simulation, we asked the operators some questions about their organization’s cybersecurity posture.

Our pre-simulation questions included:

1. Do you have a cyberattack response plan in place for your system?
2. What, if any, technologies do you use to fend off hackers?
3. What, if any, non-technical strategies do you use to fend off hackers?

These questions set a baseline for our assessment of the critical urban infrastructure manager’s preparedness. To begin the scenario, we showed the operators an image of a port scan detected on an intrusion detection system. This was followed by a series of questions ascertaining how the operators would respond if they actually saw such an image. Next, we showed a ransomware screen indicating that files have been locked and that a payment in bitcoin is required to unlock them. At this point, we asked about the operator would respond, and whether (and how) they would engage the hacker to explore the possibility of negotiation. Finally, we showed the operator a screen indicating that the attack had been resolved. This was followed with questions about how the urban infrastructure organization might incorporate lessons from this hypothetical attack into their future cybersecurity efforts and whether damage control might be managed differently in the future. Our simulation screens and associated questions are included in Appendix A.

Finding critical urban operators willing to participate was difficult, hence the reason we only interviewed seven operators. Some declined because they did not want to draw attention
to themselves or their organization. Others appeared not to have a cybersecurity strategy in place, and did want that to become obvious. Some were unable to get permission from higher-ups to speak with us about cybersecurity strategy (although highly sensitive information was not requested). We approached more than 50 organizations by email. These were selected because they were either a) previously in the media because they had experienced a ransomware attack, or b) within close geographic proximity of members of our research team in Massachusetts and California. This gave us an opportunity to conduct our interviews in person. Ultimately, we found seven operators willing to be interviewed. We are not disclosing their names. They participated because we promised to preserve their anonymity. We are allowed to say that they represent police departments, electric utilities, government agencies, satellite operators (weather and GPS), emergency management services and transportation departments and were from the states of California, Connecticut, Massachusetts and Vermont. All interviewees answered questions individually, although in one instance three organizations met together to participate in the simulation. So, we requested that everyone who participated respond to our questions individually in writing so we could be sure they were not influenced by the other respondents. Each interview took approximately thirty minutes to complete. They were voice recorded for our review, and later destroyed. We also took written notes during each session. The individuals we interviewed held positions such as Chief Executive Officer, Chief Information Officer and Chief Information Security Officer. Everyone we talked to had an opportunity to review our study results and correct any possible misinterpretation of their inputs. Our research design was approved by the MIT’s Committee on the Use of Humans as Experimental Subjects.

Because of the small number of interviews, we augmented this research method with cross-case analysis of critical urban infrastructure ransomware attacks. The findings and insights may not be generalizable to all critical urban infrastructure organizations, but we believe it is a strong starting point for operators to prepare for, defend and recover from attacks.
4.4 Interview Insights

While responses to our questions varied considerably across the different infrastructure and service agencies, a clear pattern did emerge. The idea of negotiating with cyber terrorists does not tend to occur during an attack – at the point in time when they would be most likely. Instead, negotiation possibilities, if they are considered at all, tend to come up pre-attack and post-attack. Thus, we have organized each operator’s reactions and answers under these three phases of an attack. Note, it is our intent to develop this three-phase framework – pre-, during, and post-attack, as a way of organizing our findings. We are not yet certain that we will organize our cyber negotiation playbook in exactly this way.

4.4.1 Pre-Attack Cyber Negotiation

Pre-attack cyber negotiation is all about strategic positioning. To properly prepare an organization for the possibility of a cyberattack (and possible negotiations), we learned it is crucial to develop a cyber incident response plan, build organizational awareness of the plan and the potential for attack, deploy the proper technology to defend systems, formalize the internal and external lines of communication that will be activated during an attack, and establish relationships with selected external organizations (especially the FBI). This pre-attack phase of cyber negotiation is aimed at establishing points of internal leverage to use during an attack and subsequent negotiations. Without adequate leverage (e.g., authority) developed during this phase, few if any negotiation options are likely to emerge. We briefly discuss each pre-attack component (and the negotiation leverage involved).

Develop a Cyber Incident Response Plan

From our interviews we gathered that many critical infrastructure and services organizations already have general Incident response plans. These are the first line of defense in case of cyberattack. While potentially useful in ransomware situations, they are not specific to cyberattacks and are aimed at any kind of ”incident” that might occur. At present, incident response plans are the go-to guide for critical infrastructure operators if they are under attack. Unfortunately, most of incident response plans do not offer detailed instructions; instead,
they recommend turning to outside resources such as the FBI and Homeland Security for assistance. They do instruct managers under attack to convene a pre-named management team of internal authorities relevant to the incident type to make operational decisions in real time. A CIO from a state agency we interviewed commented that their ransomware-specific incident response plan was rather simple – wipe the system and restore from backups. While this might work in some ransomware scenarios, it might not be feasible in all situations. Incidents will have to evaluated on a case-by-case basis.

Incident response plans are widely available on the internet for organizations to adapt to their needs. It is generally the responsibility of the CIO to manage and maintain these plans on an annual basis. A popular resource is the University of California Berkeley’s Incident Response Planning Guidelines [6]. Their incident response plan template indicates: "listing the names, contact information and responsibilities of the local incident response team; referencing system details or the location of such information; procedures for reporting and handling a suspected incident; procedures for reporting security incidents to Internet Service Providers (ISP) and including the intake report so the ISP will assign a security analyst to coordinate follow-up incident response activities; and responding to the security incident in a timely fashion" [6]. It further defines detection, analysis, recovery and post-incident phases and mentions that organizations should determine a timeframe for how long each of these phases should take during an incident. These guidelines are quite general. Critical infrastructure CIOs will need to customize them to their own organization.

There is no specific playbook included with the incident response plan indicating how to respond to specific cyber incidents such as a ransomware attack. The actual response to each event varies considerably (based on our interview findings). Organizations typically look to their CIO for guidance on how to deal with specific incidents and whether (and when) to consult their ISP or the FBI, depending on the severity of the event. It appears that the CIO has decision-making power over what happens during an incident. We learned that some incident response plans, depending on the organization’s sophistication and its CIO’s knowledge of cyber response, may only include bureaucratic documentation needed for liability and disclosure purposes, instead of a robust guide to how best to respond to specific cyber incidents.
Build Awareness

All the critical infrastructure operators with whom we spoke indicated that they had some mandated employee training and awareness campaigns regarding cybersecurity in place. However, some organizations take this more seriously than others. A utility CEO emphasized how building awareness was not sufficient. Instead, he tries to foster a "culture of security". He aims to instill cybersecurity thinking across all aspects of his organization’s employee experience. This involved aligning employee priorities and success metrics with cybersecurity best practices. An example would be to report the number of successful phishing attacks against an employee as part of their annual job performance review. Also, he insisted that leadership must support a cybersecurity culture by not only encouraging it, but outwardly practicing good cybersecurity itself.

In contrast to this utility CEO, we spoke with a CISO who mentioned that their cybersecurity awareness program consisted of a computer-based training program that is required when employees complete their annual HR training. The training is 30 minutes long and was created by an outside consultant. It explains what a phishing attack is, and encourages employees not to leave their laptops unlocked. There is a test at the end which employees must pass to complete the training. However, employees can take the test as many times as they want. There is little accountability for knowing the material after the course is over. From our conversations it seems that trainings are seen more as a compliance exercise than as an actual educational tool.

Despite the varied approaches – most of our interviewees acknowledged the importance of building awareness in advance of possible cyber negotiations because employees are the first responders in any attack. Our interviewees all stressed that when an employee notices a ransomware attack or some other unusual cyber activity, they should immediately communicate the incident. A quick response can help limit damage to the defending organization since ransomware attacks often involve countdown timers. Hackers threaten to inflict increasing damage to the system if action is not taken by the mandated deadline.
Deploy Technical Defenses

We strongly believe that cyber negotiation should be used in tandem with technical defense strategies. Our interviewees described the myriad technical defenses used by their organizations. Importantly, they described how technical defenses can detect an intrusion on a network, but cannot resolve the problem – especially when a ransomware attack is underway. Technical defenses should be able to detect attacks early in the attack lifecycle. They should also provide insight into the attacker’s identity such as their IP address or the various attack vectors they used to penetrate the network. This is important background knowledge that can be used during a negotiation. It might also provide some insight into the motives of the attackers. Our interview with the CISO of a major satellite operator described how a system manager can never have enough information about a potential cyber negotiation adversary – especially when so much of the interaction is cloaked behind layers of anonymity. This is one reason our interviewees’ organizations spend millions of dollars a year to keep up with some of the subscriptions they rely on to enhance their technical security. Some popular technical security subscriptions include Palo Alto Networks that focus on intrusion detection and network security [70], Mozy which provides backup services [67], and McAfee that provides endpoint antivirus protection [53]. Technical defenses such as consistently backing up systems were cited by all our interviewees as absolutely critical to preparing for the possibility of cyberattack and negotiation. Having a backup system is, in fact, the ultimate leverage in a cyberattack negotiation. One may think that there is no need to negotiate if an urban infrastructure operator has a backup, but this is not the case. Installing backups requires taking a system offline and rebooting it. Because of the 24/7 operational nature of critical urban infrastructure systems, it might be better to negotiate return of system control than taking the system down to transfer to the backup. Also, just switching to the backup and cutting off communication with an attacker means that an organization might be vulnerable to another attack of the same kind. It might also mean that an organization has lost control of important confidential information. Thus, it would be better to get all that information back, even if that involves paying a ransom.
Formalize Communication Channels

The CIO and Emergency Services leads we spoke with from a state agency emphasized the importance of clear and formal communication pathways in the event of a cyberattack. Because a wide range of employees interact with computing systems that can remotely communicate with critical urban infrastructure, each one must know who to contact (and in what order) if they are the victims of ransomware. When speaking with the CIO of an urban transportation authority we learned that their network is directly connected to other urban infrastructure networks in the state. This is not uncommon. Networks are designed in this fashion to enhance ease of access for maintenance and third-party service providers. This can result, unfortunately, in a civil servant unintentionally infecting a system despite having no direct interaction with it. Therefore, every employee must know who to contact if there a breach, and what information they will be asked to provide. Pre-planned action can only be taken after an attack is discovered. Organizational leadership must have a clear understanding of who they are going to rely on under various sets of contingent circumstances. They must understand what they, themselves, are not equipped to handle on their own. While upward and downward internal communication plans are crucial, external plans need to be established as well. Cyberattacks often attract considerable media attention. If information about organizational chaos is leaked during an attack (as opposed to a clear and calm message about how the attack is being handled), the defending organization will be at a severe disadvantage during any cyber negotiations that follow.

A CEO we interviewed described a very strict external communication protocol their organization intends to follow during a breach. The script for each dialogue will be created in real time by the CIO or CISO and then shared with the organization’s legal team. General Counsel tends to take a heavy-handed approach to restructuring such communications. The public relations team is then supposed to approve the message for external use. Either the CEO or CIO is responsible for communicating the message to the press. The information that is ultimately conveyed is highly sanitized and likely to say very little about the actual incident. We learned that information about what actually happened is often leaked to various media outlets by employees after the press briefing. The external communication
process described by our interviewee sounded extremely cumbersome. We are not sure what will actually happen under pressure. Some of our interviewees did not know if a clear external communication protocol was in place for their organization. Indicating that they would just contact their public relations team.

**Establish External Organizational Relationships**

All critical urban infrastructure operators we interviewed indicated that they would be quick to call in external help if they were targeted by a ransomware attack. A utility CEO we spoke with described the importance of their relationship with federal authorities, such as the FBI, in dealing with ransomware. Others mentioned their relationships with national labs that do attack aftermath evaluations. Our utility interviewees indicated that Idaho National Labs has established a team to help with post-mortems for cyberattacks on industrial control systems. Some spoke of relationships with consultants who specialize in attack forensics and cyber negotiation. We learned that prior to a first cyberattack or cyber negotiation, some urban infrastructure operators establishing relationships with private entities who can assist in case of attack. When an attack is underway, however, is not the time to try to sort out lines of responsibility with private contractors. Urban infrastructure organizations must prepare for cyber negotiations by knowing exactly what they will handle and what external parties will do during an attack.

In summary, most of our respondents have made contact with the FBI and have an open line of communication to them in the event of a breach. However, the interviewees did not have a clear understanding about what they would do and what the FBI would do during and after an attack. Some indicated they would more or less turn over responsibility during an incident to an external consultant who would make decisions for them in conjunction with the FBI. Others thought that the FBI would assist as their own internal organization managed the attack. Interviewees generally agreed that their own organization is ultimately responsible for deciding when and how to resume operations. In all cases, interviewees believed their organization would have the right to decide how a cyber negotiation would unfold in terms of whether a ransom would be paid.
4.4.2 The Attack

During an attack, an urban infrastructure organization must act quickly hence the pre-attack negotiation preparation measures identified above. Insights about the prospect of engaging in cyber negotiations varied across our interviewees. Some strongly opposed the idea of negotiation. They are morally opposed to giving anything to criminals. Some were open to negotiating if a capable party (i.e. someone other than themselves) were going to lead the effort, although they simultaneously wanted to retain final decision-making authority. Others indicated an interest in cyber negotiations as a technique for buying time. A memorable point made by one utility executive is that critical urban infrastructure operators should never say never to negotiating. While not all operators agreed that negotiating is the right thing to do, all had ideas about how a cyber negotiation would probably unfold. They imagined the following steps: convene the incident command structure, evaluate the severity of the event, consult the legal team and bring in and brief outside experts to execute the negotiation. Each of these steps is described below based on our interviewees’ feedback.

Convene the Incident Command Structure

As part of the incident response plan, each critical urban infrastructure organization described a team of internal stakeholders and leaders they would convene in case of a cyber-attack. In each case, the incident command structure must decide how to proceed so that responsibility does not fall on the shoulders of one (unprepared) individual. The group is not responsible for problem-solving during the incident, however, they are the chief decision makers. For example, this group might ultimately decide a ransom should be paid. They would also be responsible for decisions such as when to call in the FBI and report the incident externally. For private companies, such groups functions autonomously. We learned that government affiliated organizations, however, are unable to refer an incident to a command structure during such cases. Instead, these organizations must wait for guidance from their governing body before responding to a ransomware incident. This means that they are not able to act until explicit instructions come from the most senior level of the agency.
Determine Event Severity

After the incident command structure is assembled, the team needs to evaluate the severity of the attack. Often, the incident command structure cannot do this on their own, so they consult operators closest to the system in jeopardy. Severity is determined by the number and criticality of the systems impacted. Criticality appears to be judged largely on the amount of downtime the infrastructure system can withstand. The technical advisors who might operate the Security Operations Center (SOC) of an urban infrastructure system or someone in a senior operations role would probably lead this discussion. If a system is not determined to be highly critical, our interviewees informed us, the "infected" system might be taken offline and rebooted using backups. If a system is deemed critical, and cannot be taken offline even for a short period without serious consequences (e.g. potential loss of life that would be caused by turning off a section of a grid during a below freezing night), negotiations might then begin in earnest. In a ransomware incident at the SFMTA, instead of paying a ransom, the group in charge chose to shut all payment systems down. This literally opened the gates of the rapid transit system, allowing customers to pass through fare-free as the agency tried to install a back-up and reboot. This is an instance in which system shutdown did not cause chaos or considerable disruption [91]. Smaller groups, such as the Melrose (MA) Police Department, do not have complete backups, cyber incident response teams, or alternate recovery options. So, when they were hit with a ransomware attack, the individuals responsible (in this case, it was their IT department) advised that they pay the ransom. So, that's what they did. The ransomware had infected their TriTech software responsible for dispatch and police records. Without this, they would have been unable to operate. That was not an option [13]. This might have been a good opportunity for cyber negotiation, but that was not tried.

Another interviewee whose agency had experienced a ransomware attack also had to determine the severity of the risk to their system at the time of incident. Rather than paying the ransom, the interviewee determined that the risk was limited to their financial systems and would not impact operations. The financial system was shut down until backups could be restored. Because all cyberattacks impact systems in different ways, it is exceedingly
challenging to calculate the extent of the risk of a cyberattack before the attack occurs. Nor is the method for calculating the risk at the time of the attack straightforward. In this interviewee’s case, because the ransomware was isolated in the financial system, the risk calculation was based on financial losses as opposed to a potential operational disruption which would involve public safety, public perception and political factors in the risk calculation. There are no agreed upon standards for calculating cyber risk. This is one reason why the cyber insurance business is still struggling to find adequate underwriting [77].

Consult the Legal Team

According to our interviewees, before any purposeful action can be taken, the General Counsel or several internal and external legal advisers need to be consulted. It is their job to calculate the organization’s liability under various scenarios. Lawyers specializing in cyber risk are asked to provide guidance on the organization’s options. Often, the General Counsel is a member of the incident command structure, but our interviewees made it clear that consideration of legal liability is a distinct step. In addition to briefing the incident command structure on potential liability, the legal team must advise on any cyber insurance coverage the organization might have, and whether it is appropriate to engage the insurer in real time. Legal specialists are extremely important in decision-making about cyber negotiation, especially in publicly-owned organizations which have local, state and federal obligations regarding hack disclosure.

This is not to say that private organizations don’t also rely on legal counsel. We spoke with two cyber lawyers who asked to remain anonymous about liability for cyberattacks. Each referenced the case of D-Link which is being sued by the Federal Trade Commission in January 2017 after D-Link routers and other devices were compromised by a botnet and subsequently attacked a major internet domain name server (DNS) provider, Dyn. The FTC lost the suit and D-Link was not held liable for the hacking of all their devices, despite numerous vulnerabilities in their routers [47]. Our lawyer interviewees expressed the view that this precedent makes cyber liability difficult to evaluate. Our understanding from these interviews is that assessing ransomware liability is no different from making similar assessments in other cyberattacks. All liability is determined on a case-by-case basis; hence,
the emphasis of our critical urban infrastructure operators on consulting with their General Counsel about how and whether to pursue ransomware negotiation.

**Negotiate**

None of the interviewees had much confidence in their own organization’s ability to proceed with a cyber negotiation. Everyone we interviewed assumed that negotiation experts would have to be called in. State agencies we spoke with indicated that the FBI would be called immediately. Private sector critical urban infrastructure operators also indicated that they would depend on the FBI, in addition to calling on help from private cybersecurity contractors specializing in ransom attacks and cyber negotiation. A comment from the lead of a state emergency management service was that when the FBI is contacted for a cyber security event involving a public agency they will immediately assume control of all systems. The infrastructure operator will no longer have any say about operations. Upon speaking with the FBI cyber crime division, we learned that this is not the case. This disparity points to a larger issue of the inaccurate expectations of infrastructure operators. The FBI claimed that their role is conducting forensic analysis and attempting to catch the hacker. This is one reason why it might be advisable for urban infrastructure organizations to call in a private contractor in addition to the FBI so that the private contractor can help with incident response and system recovery. The FBI’s official stance is that they will not negotiate with cyber terrorists (and that critical urban infrastructure operators should not either). This raises the utility CEO’s comment mentioned earlier – never say never. In such critical moments, all options ought to be on the table. Urban infrastructure organizations might want to put protocols in place pre-attack that gives them as much say as possible if negotiations do occur. Negotiations will hinge on whatever leverage the attack victim might be able to generate. Only they will know what this might be.

Across all our interviewees, none had considered negotiating with cyber attackers. Interestingly, several indicated that after our interview they might entertain the option. One reason none of our interviewees had entertained this idea is they had no idea how a cyber negotiation might unfold.
4.4.3 Post-Attack

After an attack is over, each cyber negotiation almost always continues. Our critical urban infrastructure operator interviewees revealed that whatever pre-attack measures they had put in place needed to be re-evaluated in light of what happened (or what has happened to others). Additionally, our interviewees identified other post-attack considerations, including putting in place a required post-mortem attack report, documenting and implementing whatever lessons were learned and initiating external damage control. Some organization take post-attack negotiation steps more seriously than others. For one of our interviewees, a ransomware attack motivated a full review of all their systems and security. The attack prompted them to launch security awareness training for all new employees. Post incident, their information security team prepared a review that has caused them to become more aggressive about ensuring that staff update their passwords. They have also reworked their firewall configurations and added additional scanning of their network.

Develop a Post-Mortem Attack Report

Extensively documenting and conducting a digital forensic investigation (i.e. steps leading to the attack, attack vector exploited, response protocol followed) of an attack was noted by several operators as an essential step in preparation for subsequent attacks. Without documenting what happened, there is no way to identify opportunities for improvement. A post-mortem can elucidate missed opportunities for gaining leverage in subsequent negotiations. All interviewees agreed a review is critical following an incident; however, two of the seven admitted they did not have time to do this following breaches they had experienced. This was not surprising because their priority was restoring operation. Once they were restored, all focus returned to day-to-day management. After-event review is deprioritized once the frenzy of an attack is over.

Share Information

While reconsidering what happened is very useful to an organization, most of our interviewees described the need to know more about what has happened to other victims of cyberattack.
Unfortunately, every victim’s priority is to protect confidential information. This makes it hard to aggregate information on attack patterns. The utility CEO we interviewed underscored the importance of information sharing. He posited that if a post-mortem takes place and stories are shared with the broader security community, feedback on how the attack was handled could be analyzed by both internal and external experts. Relationships can be built with external experts who will be in a better position to provide advice and assistance in the future.

Many critical infrastructure sectors such as utilities participate in Information Sharing Analysis Centers (ISACs). The National Council of ISACs indicates there are 24 such organizations developed expressly for the purpose of sharing threat and security information within industry sectors. The idea behind ISACs is that if one organization is breached and notifies others, a vulnerability that might be remediated before a hacker takes advantage elsewhere in the community. Our understanding from interviewees is that some ISACs have stronger participation of industry constituents than others. For example, the utility and financial ISACs have a very proactive membership base as opposed to the newly formed automotive ISAC.

Other external experts that could be called upon are organizations such as the previously mentioned Idaho National Labs (INL). INL offers a testbed for electric utilities to simulate attacks and defenses. They also have a strong incident response team that can be flown in to help with post-mortems. They are called the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT) [38].

**Document Lessons Learned**

Closely tied to the idea of post-mortems is the notion of preparing a summary of lessons learned from each attack. Urban operators we interviewed distinguish between creating a report that offers a clear analysis of how a particular organization’s cybersecurity failed. The CIO of one state agency expressed the need after an attack for a concrete action plan to incorporate lessons learned into existing programs. Some of our interviewees mentioned that lessons learned are often incorporated to incident response reports. Others indicated that lessons learned should not be theoretical. Instead; action implications should always be
drawn. For example, a state agency CIO we spoke with insisted that all lessons learned should be immediately addressed in a highly visible manner by the leadership of the organization. If staff in a certain department consistently clicked on phishing scams, that department should be targeted for additional phishing training, and should either be called out for their failure to protect cyber assets or commended if their resilience level improves. The CIO encouraged visible and transparent measures to fill vulnerable security holes in the organization. This could improve the organization’s future negotiation positioning because hackers may publicly see the efforts taken to ameliorate past issues and move on to a different target.

Conduct External Reputational Damage Control

External damage control was among the most discussed topics in our interviews. All of the urban infrastructure organizations we spoke with expect considerable blowback in the case of a breach. Both public and private organizations know they have to find out what kinds of information may have been compromised or stolen, and the criticality of that information. Decisions must also be made about the best way of handling communications and subsequent repair of reputational damage. Handling messaging post-attack is an extremely sensitive topic that only certain leaders are authorized to handle. Messages must be carefully crafted. None of the interviewees wanted to draw attention to their organization after an attack – even if their primary message is that they have since fortified their defenses. This type of outward messaging was presumed to invite further hacking against their organization. Finally, public agencies need to work with auditors to establish that they acted in good faith and were doing all they could to protect the taxpayers’ best interest during the breach. Our interviewees’ specifically noted that properly handling reputational damage control could be a form of leverage against future attacks. If a hackers’ goal is to undermine an urban infrastructure organization’s reputation, but the reputation is well-managed after a breach, this might discourage a future hacker from pursuing the same organization a second time.
4.5 Case Study Analysis

Now that we have an understanding of what needs to happen during the three stages of cyber negotiation, we will evaluate two actual cyberattacks in light of the pre, during and post-attack framework we have presented. By reviewing these cases after the fact, we will show how the failure to act appropriately contributed to suboptimal outcomes for the victim organizations. In the first case, the attack was a ransom cyberattack (without ransomware) in which the company chose to engage in a negotiation with the hackers. The second case involves a widespread ransomware attack in which there was a clear opportunity to negotiate with the hackers, but it appears that no one tried to pursue one-on-one negotiation as part of the attack response.

4.5.1 Uber Technologies

Modern transportation infrastructure in cities consists of mass transit in the form of busses and trains, personal vehicles and taxi services. Increasingly, the traditional model of hailing a cab to catch a ride has disappeared in favor of calling a cab on a digital platform such as Uber. The applications we rely on to get from point "A" to "B" have become integral to traversing the urban environments. In 2017, Uber had between 1-2 million active drivers in the United States [7]. While Uber considers its drivers contractors rather than employees, it collects and retains considerable amounts of data about all of these contractors – past and present. Not only does Uber store data about its drivers, but also its users. Uber's co-founder and former CEO Travis Kalanick mentioned in 2017 that Uber had 40 million active users each month [49]. This makes Uber a prime target for hackers seeking personally identifiable information about a diverse and large population (e.g. all present and past Uber users).

In 2016, Uber covered up a major data breach that exposed 57 million user accounts, including data from both drivers and users [30]. Most of the data stolen consisted of names, email addresses and phone numbers. Additionally, 600,000 U.S. driver license numbers were stolen [71]. This hack was achieved by two hackers who accessed a private Github account used by Uber engineers. They stole credentials embedded in code on the Github site from
an Amazon Web Services account where the data was stored [30]. Interestingly, the hackers did not encrypt or lock down the information. Nor did they agree to release it if a fine were paid. It is unclear if this was impossible given the constraints on the hackers, or it was not the preferred negotiation strategy of the hackers. Instead, the hackers reached out to Uber via email requesting funds to keep the breach quiet and not to disclose the data [71].

Details are not available about how the negotiation unfolded, but ultimately, Uber paid $100,000 to meet the hackers’ demands [71]. What we do know is that the hackers kept their side of the bargain after being paid. The only way the public found out about the hack was when the newly installed CEO, Dara Khosrowshahi, announced the breach after learning of the incident. The 2016 breach was hidden from the Federal Trade Commission (FTC) which, at the time, was investigating a separate data breach Uber experienced in 2014 [71].

While we cannot fully ascertain the details of what happened, it is clear that cyber negotiation was used to resolve this exploit and defend the digital platform. As we determined through our interviews with critical urban infrastructure operators, there were three discreet phases of the negotiation – pre-negotiation, negotiation and post-negotiation. Our interpretation of how these phases unfolded during the Uber attack are outlined below.

**Pre-Attack**

Based on publicly available information, our understanding is that there was no cyber incident response plan in place at Uber. If there was, it was not followed during this incident. Before the attack, Uber did not seem have taken appropriate precautionary measures to secure its credentials on the Github account. Further, formalized communication pathways for reporting the threat, either internally or externally, did not seem to be in place. Finally, no external connection was pre-emptively established during or after the hack. Uber put its energies into hiding the attack from investors, customers, drivers and the FTC. We assume that Uber had no help from the FBI or other cyber experts in pursuing the negotiations as they did.
The Attack

During the attack, hackers were able to collect data that were vital to Uber’s reputation. This was used as leverage in the request for a ransom payment. After the ransom was demanded, some form of an ad hoc incident command structure was assembled. We know that a deal was arranged by the CISO and agreed to by the CEO. Because of the eventual payout to the hackers, we assume that the CISO and CEO calculated that the data breach was a severe event. We also assume that legal counsel was consulted because the director of security and law enforcement was fired along with the CISO after the breach and the ransom payment were exposed. Further, the CISO was a trained lawyer who had practiced law previously and studied cyberlaw at University of Miami [39]. To initiate the negotiation, the hackers reached out directly to Uber via email. Because they did this, and did not use a third-party service to issue their ransom demand, we assume that the hackers were willing to engage in a conversation regarding their financial demands and what they would offer in return. The parties were able to arrive at a mutually agreeable number ($100,000). We suspect, though, that the hackers demanded more initially, given Uber’s $48 billion valuation [72].

Post-Attack

After the attack, the hackers maintained their side of the deal as the 57 million users’ and drivers’ data were never released (at least not publicly). It is unclear if a post-mortem attack report was ever made beyond the financial documentation indicating the payment made to the hackers. Uber chose not to engage authorities, share information or notify users of the breach. This is a missed opportunity for raising awareness of whatever vulnerability exposed Uber to the attack in the first place. It also made Uber continuously vulnerable to similar attacks until the attack was disclosed broadly. Further, as far as we can tell, no lessons learned were documented or acted upon. We wonder what precautionary measures Uber took in case the hackers defaulted on their agreement and released the user data. Finally, no external reputational damage control was done at the time. This ended up causing even worse damage for the company when word was finally released. In part because of its poor post-attack handling of the breach and its failure to disclose what happened to users, Uber
lost its license to operate in London [24]. Starting in May 2018, there will be consequential financial repercussions for failing to disclose data breaches under the European Union’s recently adopted General Data Protection Regulation (GDPR). Under GDPR, Uber’s post-negotiation failure would have cost them up to 4% of their annual revenues [55].

4.5.2 National Health Services

Healthcare institutions ranging from urgent care centers to hospitals are essential to everyday life in our urban centers. Such critical urban infrastructure collects and retains sensitive data about patients. Not only is sensitive data stored by healthcare institutions, but constant access to these data are required in real-time to prepare for and perform procedures and surgeries for patients. Typically, data are stored in electronic medical records (EMRs). Their paper equivalents have been discontinued in many healthcare systems. These data consist of highly personal information ranging from social security and insurance details to specific medical diagnoses, lab results and treatment plans. Lack of immediate access to EMRs could cause chaos in hospitals that rely on these data for most of their operations.

In 2017, healthcare facilities in over 150 countries were attacked by WannaCry ransomware. This attack targeted unpatched Windows 7 operating systems with a specific vulnerability that allowed the attack to spread automatically across many systems. England’s National Health Service (NHS) hospitals were among those particularly affected. Ransomware locked down the use of EMRs, and demanded a payment in bitcoin roughly equivalent to $300 per computer at the time. Reportedly, in England, 6,912 doctors’ appointments (including critical operations) had to be cancelled as a result, and approximately 19,000 appointments were affected in some way [73]. Five acute care hospitals had to divert emergency ambulances to other nearby hospitals [66].

---

1GDPR is a regulation that replaces the Data Protection Directive established in 1995. The Data Protection Directive set a minimum level of requirements concerning personal data privacy and in 2012, the directive was recommended for overhaul based on the modern digital age. The GDPR is a more robust mechanism to protect data privacy built for today’s pervasive technology environment. In addition to reinforcing previous data privacy rights, the GDPR provides the right to data portability, the right not be profiled using your data, and the right to be forgotten among others. GDPR also requires large scale private and public organizations to appoint a Data Protection Officer to ensure compliance with GDPR [93]. GDPR requires data protection for all EU citizens, regardless of where the data is stored or where the company is based. Perhaps the most impactful component of GDPR is that there will be fines and penalties levied for non-compliance.
Shortly after WannaCry was released causing major disruption across healthcare systems, McAfee, a leading provider of antivirus software, called the WannaCry ransomware "pseudo-ransomware" [3]. Pseudo-ransomware is malware used by hackers more interested in causing disruption than collecting money. This appears to be true for WannaCry because the ransomware only collected about $150,000 in ransom payments. This is an astonishingly low amount for an attack impacting so many and such critical systems. Similar ransomware like CryptoWall was used to collected $325M in ransoms [3].

McAfee’s hypothesis that WannaCry’s hackers were more interested in disruption than financial gain was validated in December 2017. The US and UK governments publicly accused North Korea of unleashing the WannaCry malware to "cause havoc and destruction" [68]. Because there was not a financial motive behind the hack, we believe that cyber negotiation might have been an excellent defensive strategy during the course of this attack. Below we outline an ideal pre, during and post negotiation strategy that might have been used in response to WannaCry.

**Pre-Attack**

Considering WannaCry only affected unpatched Windows 7 vulnerabilities, the ideal pre-attack strategy would have been to constantly patch EMR operating system using the latest security updates. The reality is that many system administrators in hospitals do not perform such updates and often fail to deploy obvious technical defenses. Before the attack, NHS Digital performed on-site cybersecurity assessments of 88 of the NHS trusts. None passed the cybersecurity standards inspection [35]. As part of a pre-attack cyber negotiation strategy, backups of all EMRs should be made in case systems were corrupted or disabled. Considering how few victims paid the ransom, we guess that backups of the EMRS were available. However, they were not readily accessible; hence, the disruption of hospital services was severe. Because of the large-scale nature of the attack against many NHS hospitals, it is unclear whether individual hospitals had cyber incident response plans in place or not. NHS had issued general guidance on how to handle such attacks, but it was not locally tested [8]. Considering the level of disruption, we assume that even if there were a plan in place for certain hospitals, it was poorly executed. We also surmise – based on how
widespread the ransomware attack was – that formal communication pathways were not followed to ensure that awareness of the attack quickly moved up the organizational ladder and definitive instructions came back down. Further, it is clear that that proper awareness was not built across the NHS employee base regarding possible phishing attacks because for the ransomware to be successfully installed, an employee had to click on the infected link.

The Attack

North Korea, the alleged WannaCry attackers, weaponized an exploit originally developed by the US National Security Agency (NSA) and released to the public by a cybercrime group called the Shadow Brokers. The ransomware attack was developed by attaching a system encryption mechanism to the NSA’s exploit so that when a victim clicked a phishing link, the exploit installed the encryption mechanism and locked the computer. North Korea released the attack and then hired a third-party managed services team to facilitate the ransomware "customer" (aka victim) support encrypted chat to help victims purchase bitcoin and understand how to get their data back [3]. One reason a third-party team was hired to manage the attack could be to obfuscate the fact North Korea launched it – providing a veil of anonymity.

When the attack was launched, it is unclear whether the hospitals involved convened an incident command team. Some likely did and determined that the attack posed very serious risks, considering hospital services were shut down entirely at numerous hospitals. Because of the high level of regulation surrounding national hospitals, legal teams were certainly convened to address the liability posed by the ransomware attack. While it is unclear if any hospitals negotiated directly with the hackers, there was an opportunity to do so. During the WannaCry attack, a McAfee security researcher asked the ransomware support operator via the customer support chat described above why the ransom was so low per machine. The operator responded that "those operating the ransomware had already been paid by someone to create and run the ransomware campaign to disrupt a competitor’s business" [3]. The defender could have used this information to their advantage considering it reveals the motive of the hacker (North Korea). Rather than paying the ransom, the defender might have argued that WannaCry had successfully disrupted their business. Thus, he attacker’s
objective was achieved. Depending on how much autonomy to negotiate the third party agency managing the ransomware has, such an argument could lead to a "Good for You, Great for Me" scenario. The premise being that a negotiating partner can appreciate the goals of their negotiating partner and offer an outcome that achieves a good result while allowing their negotiating partner to achieve their minimum goal [94]. In this case, the "Good for You, Great for Me" scenario would have entailed North Korea hearing that they had successfully disrupted the hospital’s operations, causing substantial damage, while the hospital received the decryption key it needed to unlock its EMRs and resume business without paying a ransom.

WannaCry illustrates the urgency of understanding the motives of an attacker in advance of any effort at cyber negotiation. If engagement with a hacker’s agent or emissary is possible, and the right questions are asked, it may be possible to achieve a satisfying outcome, avoiding considerable financial and operational losses.

Post Attack

After the WannaCry attack, unlike the Uber attack, the NHS carefully studied the impact the attack had on individual hospitals and NHS’ overall system. It developed a post-mortem attack report. From this analysis, documented in a National Audit Office report, several lessons were drawn and actions were recommended to improve future readiness and response to cyberattacks. These ranged from developing a response plan for the NHS to follow in the case of a future cyberattacks, especially to ensure that security updates and alerts are taken seriously and deployed immediately. The report, including lessons learned, was publicly shared for all to see. Further, NHS England and NHS Improvement are working with major trauma centers to allocate 21 million in capital funding to achieve cybersecurity enhancement [66]. This funding will be going to develop a security operations center (SOC) for the NHS at which all security-related incidents will be handled.

These actions should improve the NHS’ negotiation posture and help to foster a more resilient cybersecurity culture. Also, the widespread announcement of the lessons learned by the NHS shows the public and hackers that the attack was taken seriously and that the NHS will be better prepared in the future. This could be the best way to dissuade attackers from
targeting the NHS again, given that these measures have eliminated obvious weaknesses that other hackers might exploit.

4.6 Future Work

Our interviews provide a clear indication that it helps to think about cyber attacks and responses to them in terms of the three phase negotiation framework we have identified. By collecting additional interviews from critical urban infrastructure operators, we hope to refine the framework and our prescriptions further. Our ambition is to develop a comprehensive cyber negotiation playbook with the help of an international advisory group. The kind of person we hope to invite to serve on this panel is someone like Moti Crystal, an Israeli cyberterrorism expert who has been actively involved in critical urban infrastructure negotiations with cyberterrorists in many parts of the world. In addition, we hope that Chris Voss and others who helped to develop the FBI’s Hostage Negotiation Manual will agree to participate. Finally, we plan to tap senior members of the many Crisis Negotiation teams managed by big city police departments throughout the world.

4.7 Conclusion

Based on our interviews and case studies we are convinced that non-technical defense strategies (i.e. negotiation and other DSE tools) can be used to fend off cyber ransom attacks. While it is ideal if a defender can engage with their attacker, direct negotiation is not always possible. This does not mean, though, that negotiation is not an effective way of framing how a victim can survive a cyberattack. Pre-attack strategies such as developing a cyber incident response plan, attack strategies such as being ready to assess event severity and ensuring clear lines of communication, and post-attack strategies including conducting external damage control can be valuable without involving direct negotiations with an attacker. Proper preparation and clarity about how to proceed during an attack can reduce damage levels. And, if direct negotiation is possible, it is important to think about (1) sources of leverage, and (2) the basic rule of negotiation that a good outcome needs to meet the most important
objectives of both sides. While this may seem unpalatable to a group under attack, a good outcome may take the form of substantially reduced ransom or return of valuable encrypted information rather than no losses at all. We do not believe that every ransom attack should involve negotiation, but it should be considered a tool in the defensive social engineering toolbox. All critical infrastructure managers should invest as much time, energy and money in defensive social engineering as they do in technical defenses.
Chapter 5

Conclusion

Additional research, going well beyond what I have been able to present in this dissertation, will be needed to protect our urban critical infrastructure from cyberattacks. My work provides only a beginning. The way disciplines and fields are currently structured, the work I have in mind aligns most closely with what is called Cyber Science. Cyber Science is a field that was established by the United States Naval Academy. It involves the study of cybersecurity, encompassing the full range of technical, social, policy and managerial elements of cyber science. However, it has not yet addressed strategies for protecting critical urban infrastructure from attack. I believe a new research discipline that I would call Urban Science should be created. This would combine a number of elements from the fields of urban planning and computer science.

5.1 Where Computer Science and Urban Planning Meet

Most urban policy makers and city planners have very little familiarity with how a citys technical infrastructure operates. This will make it difficult for them to protect urban critical infrastructure from cyberattack. New interdisciplinary research in Urban Science will be required to achieve this goal. The research in Urban Science I have in mind will build on recent work dealing with smart cities, communication technology, big data, urban informatics and studies of urban infrastructure, but do a better job of integrating computer science and the applied social science aspects of urban planning. Some of the important
initial publications at the interface of urban planning and computer science include Battys The New Science of Cities, Foths Handbook of Research on Urban Informatics, Hamadas Critical Urban Infrastructure Handbook, Komninos Intelligent Cities, Lights From Warfare to Welfare, Townsends Smart Cities, Rattis City of Tomorrow, Zook et. al, Big Data and the City, and Zuboffs In The Age of The Smart Machine. But, there is new work needed that will require urban planners to learn a lot more about computer and data science. In particular, a new generation of urban scientists need to understand the algorithms and data science that can explain how humans interact with computer systems and techno-socio networks. They will also have to become completely familiar with the computing hardware and software that underlie the technology that enables cities to function.

5.1.1 Research in Data Science

The Urban Science research agenda I have in mind will require careful study of the algorithms by which city operations are managed. Connected IIoT technologies generate a vast amount of data. These data can explain the pattern of use of city systems which is crucial to building an internet-enabled method of evaluating security threats. The chapter of my dissertation entitled IIoT Risk Modeling for SCADA Systems, demonstrates how data science can be used to identify urban cybersecurity risks. Below I propose some questions that are now clearly defined. To understand urban critical infrastructure further, we need to provide a framework that cities can use to conduct technology asset inventories and assess the criticality of infrastructure systems. Doing so will help to identify which systems and insecurities should be given priority. One potential approach to prioritizing city assets includes evaluating how, when and why city inhabitants interact with each asset. We need to ask questions such as, What are uptime requirements for critical services before there is a measurable impact on citizens? Which demographic groups are most vulnerable if key services are unavailable? What are the relationships among infrastructure systems so that if one is compromised, others are impacted? To answer these questions, researchers will need access to all the data produced by infrastructure managers and the many sensors that are distributed around the urban environment. After acquiring these data, urban scientists can process this information using various machine learning and statistical data analysis techniques. Some of the most
basic machine learning algorithms such as those involving clustering could be especially effective in answering these questions.

Data science can further be used to predict human behavior as the field of computational sociology develops. This will help us understand how different segments of the population are likely to respond to various services being compromised by cyberattacks. City officials must know how citizens will react to city services being disabled. We can also use data science to begin parsing the many data packets pulsing through the networks that currently connect our urban critical infrastructure. This could help us to determine a baseline of internet communication activity for urban critical infrastructure computing systems. With this information in hand, we can develop network anomaly detection algorithms than can give us early warnings of attacks on critical infrastructure.

5.1.2 Research on Computing Systems and Networks

Beyond data analysis, research is needed on which computing systems are currently prioritized so that security deficiencies can be identified. Research questions for the next generation of urban scientists include: How can security weaknesses be remediated? Can we develop infrastructure systems that can sustain downtime while security patching is underway, without compromising critical functions? How do we design out security worries in the process of formulating new urban plans or redevelopment strategies? As a concrete example, planners should be able to identify and understand the operating systems behind classes of smart meters or other embedded devices that make up the Industrial Internet of Things (IIoT) that enable a networked city. At present, I do not think they do. Insights along these lines are crucial to identifying operating system vulnerabilities and patching opportunities, or at least the ability to monitor said vulnerabilities. A starting point for this type of research is provided in the chapter of my dissertation: A Master Attack Methodology for an AI-Based Automated Attack Planner for Critical Infrastructure. This is not something that most city planning departments understand. However, it is necessary so that design decisions regarding future cities can be informed by the security (or lack thereof) of proposed infrastructure computing systems.

For urban critical infrastructure IIoT, we also need to research and design security tech-
nologies or entire operating systems that will be secure against future attacks. This will likely entail developing various IIoT endpoint security software that can be installed to protect these systems. This needs to be developed in tandem with Defensive Social Engineering (DSE) tools to protect these urban critical infrastructure devices. Social strategies are essential to complement existing technical tools. The chapter of my dissertation: Cyber Negotiation: Defending Urban Critical Infrastructure from Cyberattacks by Negotiating with Cyberterrorists provides a basis for future work on social strategies to defend urban critical infrastructure from attack. Some DSE tools that I hope to continue studying include: honeypots (decoy systems that are meant to trick hackers into attacking), obfuscation techniques (scrambling data and source files to make deciphering data more difficult for attackers to use), misinformation campaigns (sending fake data out to confuse hackers), employee education and awareness efforts, devaluing digital assets upon compromise (open-sourcing data sets that were stolen so that they no longer have value on the black market) and proactive defensive signaling (indicating publically what an organizations policy is for dealing with potential attackers).

Protective software and DSE tools are not all that is needed to improve the cybersecurity posture of urban critical infrastructure. We will also need to work towards designing city services and infrastructure that are resilient to cyberattacks. Recent history has shown that regardless of how secure a computing system is thought to be, an attacker can almost always find a way to compromise the system. Therefore, in addition to building technical tools to protect infrastructure, we must look toward ways of designing and managing critical infrastructure so that it can recover quickly from attacks. This may involve backup programs that do not take several days to implement in the case of a ransomware attack. To develop such tools and techniques, knowledge of computing systems hardware and low-level software (code that interacts with the processor) for the urban critical infrastructure is required.

5.1.3 Beyond Cybersecurity for Urban Science

Establishing the new field of Urban Science will do more than enhance critical infrastructure cybersecurity. If urban scientists are properly equipped with knowledge of how to use advanced computer science methods for data processing and have a baseline understanding
of the operations of infrastructure systems, they can formulate more dynamic plans and management systems. Instead of creating static plans based on 3-dimensional spatial data, urban scientists can advise cities based on real-time 5-dimensional data sets which include both citizen-generated data and computer device data layered on top of traditional spatial data. Urban Scientists need to be able to design future-proof cities that can accommodate our society's increasing use of robotics and artificial intelligence algorithms.

Imagine a city that can conform to your daily life and adjust to new risks in real-time. The mayor would receive immediate notification of any change in the city's infrastructure risk level in the same way that Homeland Security provides advisories on terrorist threats. Danger hotspots would light up on the city's map so the mayor could deploy the necessary support accordingly. Predictive infrastructure risk analytics and management strategies can help reduce security risks and enhance resilience, enabling a safer community. Such a city using these sophisticated algorithms will be developed by urban scientists.
Appendix A

Simulation screens and associated questions

Question 1:

- Do you have a cyberattack response plan in place for your organization?
- Can you describe it?
- Is it different from your regular crisis response plan?

Question 2:

- What, if any, technologies do you employ to fend off hackers?

Question 3:

- What, if any, non-technical strategies do you use to fend off hackers?

Question 4:

- Have you checked to see if your systems are public on Shodan?

Simulation Screen 1: You see someone ran a port scan against your system (Figure A-1):

Question 5:

- Do you respond to this?
If so, how?

Question 6:

- What technical strategies might you deploy?

Question 7:

- What non-technical strategies might you prepare?

Simulation Screen 2: One of your operator consoles shortly after shows the following attack (Figure A-2):

Figure A-2: Ransomware simulation screen.

Question 8:
• How, if at all do you respond?

Question 9:
• If a hacker offered an opportunity to negotiate for the ransom, how would you advise others to proceed?

Question 10:
• Please describe your negotiation strategy.

Question 11:
• Would you advise others to handle these negotiations themselves?

• Or should they have an arrangement in place that would hand these negotiations over to someone else?

Question 12:
• Have you ever experienced a ransomware attack?

• What can you tell me about the experience?

Simulation Screen 3: The attack just ended (Figure A-3):

![Ransomware resolved screenshot.](image)

Figure A-3: Ransomware resolved screenshot.

Question 13:
• Who would the organization turn to create a post-mortem of the attack?

Question 14:

• What kind of "after-action" data gathering and discussion might occur?

Question 15:

• Who would have responsibility for deciding what the organization learned from the experience?

Question 16:

• Will your organization approach attack preparation differently now that you have been attacked?

Question 17:

• What type of damage control will need to be done?
Bibliography


[50] MANDIANT. APT1 exposing one of chinas cyber espionage units, 2013.


