



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**EXAMINING EMERGENT BEHAVIOR IN SUPPORT
OF CYBER OPERATIONAL PREPARATION
OF THE ENVIRONMENT**

by

David A. Beard

September 2023

Thesis Advisor:
Co-Advisor:

Kristin M. Giammarco
Logan O. Mailloux

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC, 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2023		3. REPORT TYPE AND DATES COVERED Master's thesis
4. TITLE AND SUBTITLE EXAMINING EMERGENT BEHAVIOR IN SUPPORT OF CYBER OPERATIONAL PREPARATION OF THE ENVIRONMENT			5. FUNDING NUMBERS	
6. AUTHOR(S) David A. Beard				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) The primary goal of Cyber Operational Preparation of the Environment (C-OPE) is to enhance commanders' decision-making capability by providing detailed situational awareness of the operational environment. However, current cyberspace methodologies and technologies often fail to realize this goal due to the complex nature of cyberspace operations and the need for advanced, scalable military capabilities. This research seeks to improve commanders' situational awareness of the cyber operational environment by evaluating the efficacy of Monterey Phoenix (MP) in performing C-OPE activities. Specifically, this study performs emergent behavior analysis to examine MP's potential to enhance commanders' understanding, decision-making, and situational awareness. The research reconstructs the cybersecurity operations process as detailed in the Information Assurance and Computer Defense "Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted Cyber Threat Information." The research is presented through the lens of systems engineering, enabling the results to be contextualized within the cyber operational environment. The results of this study provide detailed examples and recommendations to USCYBERCOM's planners, analysts, and decision-makers on the use of MP as a tool for C-OPE activities and its potential to enhance situational awareness of the operational environment.				
14. SUBJECT TERMS model-based systems engineering, Monterey Phoenix, cyber, operations, NIST Cybersecurity Framework			15. NUMBER OF PAGES 141	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**EXAMINING EMERGENT BEHAVIOR IN SUPPORT OF CYBER
OPERATIONAL PREPARATION OF THE ENVIRONMENT**

David A. Beard
Major, United States Marine Corps
BS, Roger Williams University, 2010

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING MANAGEMENT

from the

**NAVAL POSTGRADUATE SCHOOL
September 2023**

Approved by: Kristin M. Giammarco
Advisor

Logan O. Mailloux
Co-Advisor

Oleg A. Yakimenko
Chair, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The primary goal of Cyber Operational Preparation of the Environment (C-OPE) is to enhance commanders' decision-making capability by providing detailed situational awareness of the operational environment. However, current cyberspace methodologies and technologies often fail to realize this goal due to the complex nature of cyberspace operations and the need for advanced, scalable military capabilities. This research seeks to improve commanders' situational awareness of the cyber operational environment by evaluating the efficacy of Monterey Phoenix (MP) in performing C-OPE activities. Specifically, this study performs emergent behavior analysis to examine MP's potential to enhance commanders' understanding, decision-making, and situational awareness. The research reconstructs the cybersecurity operations process as detailed in the Integrated Adaptive Cyber Defense "Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted Cyber Threat Information." The research is presented through the lens of systems engineering, enabling the results to be contextualized within the cyber operational environment. The results of this study provide detailed examples and recommendations to USCYBERCOM's planners, analysts, and decision-makers on the use of MP as a tool for C-OPE activities and its potential to enhance situational awareness of the operational environment.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	MOTIVATION	1
B.	CYBER OPERATIONAL PREPARATION OF THE ENVIRONMENT ACTIVITIES	2
C.	RESEARCH OVERVIEW AND OBJECTIVES.....	4
D.	REPORT ORGANIZATION.....	5
II.	LITERATURE REVIEW	7
A.	THE CYBER OPERATIONAL ENVIRONMENT	7
1.	NIST Cybersecurity Framework.....	7
2.	The Federal Government Cybersecurity Incident and Vulnerability Response Playbooks	11
3.	Cyber Attack Life cycle Model and Cyber Kill Chain Methodology	14
4.	The Johns Hopkins Applied Physics Laboratory Information Assurance and Computer Defense Program	17
5.	Business Process Model and Notation.....	21
B.	ENHANCING A COMMANDER’S DECISION-MAKING CAPABILITY	23
1.	Subject Matter Experts	23
2.	Operations Research.....	23
3.	Data Visualization.....	25
4.	Advancements for Decision-Making.....	28
5.	Behavior Approach.....	28
III.	METHODOLOGY	33
A.	EMERGENT BEHAVIOR ANALYSIS	33
1.	Step 1 – Emergent Behavior Detection	34
2.	Step 2 – Emergent Behavior Prediction.....	36
3.	Step 3 – Emergent Behavior Classification	37
4.	Step 4 – Emergent Behavior Control	39
B.	OVERVIEW OF MONTEREY PHOENIX	41
C.	RESEARCH DESIGN AND APPROACH.....	46
1.	Stage One – Modeling System-of-Interest	47
2.	Stage Two – Event Trace Generation and Inspections.....	48
3.	Stage Three – Strength and Polarity Classifications	48
4.	Stage Four – Repetitive Investigation Cycles for Identifying Emergent Behaviors.....	49

5.	Stage Five – Simplification and Visual Representation.....	50
D.	LIMITATIONS	51
E.	DATA COLLECTION	51
IV.	RESULTS AND DISCUSSION	55
A.	STAGE ONE	55
1.	IACD Playbook	55
2.	MP Model	56
3.	Key Events in IACD Playbook and MP Model	59
B.	STAGE TWO	61
1.	Event Trace Analysis	61
2.	Human-Centered Analysis	63
C.	STAGE THREE	64
1.	Classification – Strength of Emergence	65
2.	Classification – Polarity of Emergent Behavior	66
3.	Analysis of Strength and Polarity in Classifications.....	67
D.	STAGE FOUR.....	67
E.	STAGE FIVE	70
1.	Mapping and Contextualization of the Cyber Environment.....	70
2.	Visual Representation of the Cyber Operational Environment.....	72
3.	Analysis of an Identified Emergent Behavior	75
V.	CONCLUSION	83
A.	SUMMARY OF KEY FINDINGS	83
B.	RECOMMENDATIONS.....	84
C.	FUTURE RESEARCH.....	85
	APPENDIX A. SCHEMA	87
	APPENDIX B. ALTERNATE MODELING TOOL	89
	APPENDIX C. SCOPE 2 TRACES	91
	LIST OF REFERENCES	113
	INITIAL DISTRIBUTION LIST	121

LIST OF FIGURES

Figure 1.	Vulnerability Response Phases. Source: CISA (2021).	12
Figure 2.	Incident Response Process. Source: CISA (2021).	14
Figure 3.	Sample Playbook. Source: IACD (2020).	21
Figure 4.	Detecting Malicious Activity	22
Figure 5.	Typography. Adapted from Wong (2010).	26
Figure 6.	Monterey Phoenix Root and Atomic Events	42
Figure 7.	Event Grammar Rules and Examples of Event Trace Instances. Source: Auguston (2020)	44
Figure 8.	IN and PRECEDES	45
Figure 9.	Coordinating Statements	45
Figure 10.	Code, Console, and Traces	46
Figure 11.	Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI. Source: IACD (2018)	53
Figure 12.	SCHEMA	57
Figure 13.	Event Trace Scope-2 Trace-2	58
Figure 14.	Event Trace Scope-2 Trace-9	59
Figure 15.	Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI. Adapted from IACD (2018)	60
Figure 16.	Event Trace Scope-2 Trace-42	62
Figure 17.	Decision Tree Classifier: Plot Tree	69
Figure 18.	Relationship to Governance or Regulatory Requirements. Adapted from IACD (2018).	71
Figure 19.	NIST Cybersecurity Framework Core Sunburst Chart	73
Figure 20.	NIST Cybersecurity Framework Core Sunburst Chart – Detect.	74
Figure 21.	NIST Cybersecurity Framework Core Sunburst Chart – Protect.	74

Figure 22.	NIST Cybersecurity Framework Core Sunburst Chart – Respond.	75
Figure 23.	Human Involvement and Branch Paths. Adapted from IACD (2018).	77
Figure 24.	Event Trace Scope-2 Trace-7.	78
Figure 25.	Top of Decision Tree Classifier.	79
Figure 26.	Event Trace Scope-2 Trace-3.	80
Figure 27.	Event Trace Scope-2 Traces 1–2.	91
Figure 28.	Event Trace Scope-2 Traces 3–4.	92
Figure 29.	Event Trace Scope-2 Traces 5.	93
Figure 30.	Event Trace Scope-2 Traces 6–7.	94
Figure 31.	Event Trace Scope-2 Traces 8–9.	95
Figure 32.	Event Trace Scope-2 Traces 10–11.	96
Figure 33.	Event Trace Scope-2 Traces 12–13.	97
Figure 34.	Event Trace Scope-2 Traces 14–15.	98
Figure 35.	Event Trace Scope-2 Traces 16–17.	99
Figure 36.	Event Trace Scope-2 Traces 18–19.	100
Figure 37.	Event Trace Scope-2 Traces 20–21.	101
Figure 38.	Event Trace Scope-2 Traces 22–23.	102
Figure 39.	Event Trace Scope-2 Traces 24–25.	103
Figure 40.	Event Trace Scope-2 Traces 26–27.	104
Figure 41.	Event Trace Scope-2 Traces 28–29.	105
Figure 42.	Event Trace Scope-2 Traces 30–31.	106
Figure 43.	Event Trace Scope-2 Traces 32–33.	107
Figure 44.	Event Trace Scope-2 Traces 34–35.	108
Figure 45.	Event Trace Scope-2 Traces 36–37.	109
Figure 46.	Event Trace Scop- 2 Traces 38–39.	110

Figure 47.	Event Trace Scope-2 Traces 40–41	111
Figure 48.	Event Trace Scope-2 Traces 42	112

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Cyber Attack Life cycle Model Stages. Adapted from MITRE (2018).....	15
Table 2.	The Lockheed Martin Cyber Kill Chain. Adapted from Lockheed Martin (2015).	16
Table 3.	Playbook Content Types. Source: Information Assurance and Computer Defense (2017).....	18
Table 4.	Classification of Behavior. Adapted from Giammarco (2022).....	49
Table 5.	Behaviors in the Event Traces	65
Table 6.	Strength of Emergence.....	66
Table 7.	Polarity of Emergence.....	66
Table 8.	Predicted Behaviors, Discovery, and Classification	76

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

AI	Artificial Intelligence
APL	Applied Physics Laboratory
BIDS	Behavior-based Intrusion Detection System
BPMN	Business Process Model and Notation
C-OPE	Cyber Operational Preparation of the Environment
CALM	Cyber Attack Life cycle Model
CDM	Continuous Diagnostics and Mitigation
CISA	Cybersecurity and Infrastructure Security Agency
CKC	Cyber Kill Chain
CODE	Collaborative Operations in Denied Environments
CSF	Cybersecurity Framework
CSC	Critical Security Controls
CTI	Cyber Threat Information
CVE	Common Vulnerabilities and Exposures
DARPA	Defense Advanced Research Projects Agency
DOD	Department of Defense
EBA	Emergent Behavior Analysis
FCEB	Federal Civilian Executive Branch
IACD	Information Assurance and Computer Defense
INCOSE	International Council on Systems Engineering
IOCs	Indicators of Compromise
MITRE	Massachusetts Institute of Technology Research & Engineering
MP	Monterey Phoenix
NASA	National Aeronautics and Space Administration
NIST	National Institute of Standards and Technology
NPS	United States Navy Naval Postgraduate School
PCS	Physical-Cyber-Social
SOC	Security Operations Center
SE	Systems Engineering
SoI	Systems-of-Interest

SoS	Systems of Systems
UAV	Unmanned Aerial Vehicle
USCYBERCOM	United States Cyber Command

ACKNOWLEDGMENTS

My profound gratitude goes to my cherished wife. Her unwavering support, especially amidst academia and our parenting journey, has been the cornerstone of this endeavor. Her strength as a new mother granted me both time and emotional respite. To her, I owe this achievement.

To our eagerly awaited “Peanut,” the joy of your impending arrival knows no bounds.

Our vibrant daughter, a constant source of joy and inspiration, fueled my drive. Even at a young age, her precocious wisdom and spirited nature underscored my dedication to this endeavor.

Lt Col Logan Mailloux, Ph.D., deserves special mention. His dedication, constructive insights, and holistic guidance have been instrumental, making him more than an advisor—a life, military, and academic mentor.

I sincerely thank Dr. Kristin Giammarco for introducing me to the MP community. Her academic excellence has enriched the field, and her impactful contributions to the USMC stand out.

Kudos to Dr. Wally Owen, whose invaluable insights and tireless efforts in advocating for active-duty military students have seamlessly woven the fabric of my military commitment with academic aspirations.

An ode to my cohort, a constellation of brilliant minds and kindred spirits. Their encouragement, wisdom, and unwavering support transformed challenges into a fulfilling journey. To them, I owe many cherished memories.

Finally, I deeply appreciated my military leaders, whose faith and insights spurred me to exceed my boundaries. Their steadfast backing of this thesis and the NPS program guaranteed its requisite attention and prominence.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

The primary goal of Cyber Operational Preparation of the Environment (C-OPE) is to enhance the commander's decision-making capability by providing detailed situational awareness. In 2010, the Department of Defense codified "Cyber Operational Preparation of the Environment (C-OPE)" in the *Joint Terminology for Cyberspace Operations*:

Non-intelligence enabling functions within cyberspace conducted to plan and prepare for potential follow-on military operations. C-OPE includes but is not limited to identifying data, system/network configurations, or physical structures connected to or associated with the network or system (to include software, ports, and assigned network address ranges or other identifiers) for the purposes of determining system vulnerabilities; and actions taken to assure future access and/or control of the system, network, or data during anticipated hostilities. (Cartwright 2010)

In contrast to traditional Operational Preparation of the Environment (OPE), which focuses on physical battlefield conditions such as communication lines, supply chains, and fortifications, C-OPE concentrates on cyber threats rather than conventional ones. Moreover, cyber operations differ from conventional military operations in that they often require access to or work within adversary networks during the planning and preparation phase.

A. MOTIVATION

The main aim of C-OPE is to help commanders make better decisions by clearly understanding the operational environment. Unfortunately, current cyberspace technologies often fall short of achieving this objective effectively. One reason is that they require more advanced and scalable capabilities to provide accurate and timely situational awareness. Another reason is the complexity of cyberspace operations, which involve multiple layers of technology, networks, and people. Additionally, the lack of coordination and integration among these capabilities and tools can cause information overload, confusion, and prone to error which can negatively impact the decision-making process of commanders.

To achieve maximum effectiveness while maintaining mission security and confidentiality, cyber operators must craft tailored accesses and payloads for offensive operations (Buchanan and Cunningham 2023, 63). Gaining and sustaining access to a target environment can be difficult and resource-intensive, presenting distinct challenges (Borghard and Lonergan 2019). Success in a cyber operation, the operation's longevity, and its capacity to minimize unintended consequences depend on the attacker's preparation and comprehension of the target's environment and potential defensive reactions during the attack (Lin 2012).

Analyzing a historical example like the Stuxnet attack is helpful to fully comprehend the significance of C-OPE in advanced offensive cyber operations. The Stuxnet code's development involved extensive reconnaissance conducted for months, or even years, to gather information on Iranian targets (Hammond 2012). Kaspersky, a cybersecurity company, found evidence that the attackers applied this preparatory work to achieve a sophisticated, multi-phased assault (Kaspersky n.d.), demonstrating the importance of careful planning and preparation in cyber operations.

To ensure success in cyberspace, it is important to take all necessary steps in advance to guarantee the successful execution of the mission. C-OPE is a valuable tool that enables organizations to tailor their plans to meet their objectives and safeguard confidential information. The Stuxnet attack is a prime example of attackers who conducted intensive surveillance to become intimately familiar with their target environment and achieve their objectives. Conducting thorough reconnaissance is essential to gather target information and develop effective military plans.

B. CYBER OPERATIONAL PREPARATION OF THE ENVIRONMENT ACTIVITIES

Approved and incorporated into the DOD Dictionary, Joint Publication 3-05 (2020) defines OPE as "The conduct of activities in likely or potential operational areas to set conditions for mission execution" (Joint Chiefs of Staff 2020). The methodologies and tools used in C-OPE can vary depending on an organization's specific requirements and capabilities.

In C-OPE, network mapping and penetration testing are the most significant activities for enabling cyber operators to identify potential weaknesses in the target system's network (Joint Chiefs of Staff 2020). These techniques and tools offer insights into designing a plan to achieve objectives while ensuring the confidentiality of information. Furthermore, properly understanding the operating environment gives organizations an idea of the scope and scale of necessary resources for execution.

An organization's C-OPE may involve clandestine activities as a crucial element. In these activities, the organization conducts secret operations to acquire intelligence or other classified information, which may involve espionage, sabotage, and counter-intelligence operations physically, virtually, or through deception operations (DeVine 2022).

Clandestine cyber activities involve using computer networks to gather information while remaining undetected. Cyber clandestine activities typically involve infiltrating a network, gathering intelligence and data, and taking measures to ensure that adversaries do not discover the action (DeVine 2022). An example of a social engineering attack is a phishing attack, which involves sending malicious emails to unsuspecting victims to gain access to sensitive information or resources. These emails often contain malicious attachments or links disguised as legitimate content, enabling access to user accounts and sensitive data.

Collecting information through clandestine physical activities helps to understand what may be happening or who may be involved, providing insights into the adversary's tactics, techniques, and procedures that decision makers can use to devise strategies accordingly (Miller 2012). If physical evidence uncovers an adversary's use of a particular system, organizations can strategically allocate the necessary resources for execution.

Deception operations in clandestine activities misdirect adversaries from intended targets (Miller 2012). These techniques can effectively disrupt a range of cyberattacks while obtaining valuable information on the operating environment. For example, one deception technique in C-OPE involves setting up honeypots, decoy virtual machines, or servers designed to appear vulnerable to attackers (Miller 2012). Honeypots can monitor a

target organization's network for malicious activities, distract adversaries from more valuable targets, and provide intelligence about the attacker's tactics and techniques (Miller 2012). In 2018, a team of researchers tracked an ongoing campaign by a hacker group targeting web applications by setting up several honeypots (Zhang et al. 2018). After studying the data, the researchers discovered that the hacker group diverted two-thirds of attacks from their intended targets (Zhang et al. 2018). This analysis also provided insight into the attackers' identity and techniques.

C. RESEARCH OVERVIEW AND OBJECTIVES

This research focuses on improving situational awareness of the cyber operational environment and examines how Monterey Phoenix (MP) can improve decision-making for commanders. Specifically, the research spotlights the suitability of MP in carrying out C-OPE activities. The goal is to provide examples and recommendations to USCYBERCOM's planners, analysts, and decision-makers on using MP for C-OPE.

This study integrates a modeling and simulation research method into its design for conducting emergent behavior analysis. It employs a multi-staged exploration process to investigate the cyber operational environment, leveraging MP across all stages. The initial phase focuses on modeling the Systems-of-Interest (SoI). Second, event traces are generated and exhaustively analyzed for any fallacies, such as incorrect concepts or biased assumptions made during the initial modeling stage. The third phase centers around categorizing emergent behaviors based on their strength (weak or strong) and polarity (positive or negative). In the fourth phase, the process is subjected to repeated investigation cycles to identify emergent behaviors. The final phase concentrates on integrating research findings within an engineered system context, thereby presenting the information in a manner that enables commanders to access it both intuitively and efficiently.

Several key assumptions underpin this study. It postulates that the models under evaluation should be able to identify emergent behaviors likely to appear in a cyber operational environment. It further assumes that current C-OPE capabilities lack the ability to detect emergent behaviors, a deficiency that could be addressed to enhance situational awareness. Finally, the study asserts that increasing situational awareness within the cyber

operational environment directly bolsters commanders' abilities to make informed decisions.

The benefits of this research extend to several fronts. It offers valuable insights and advice to USCYBERCOM's planners, analysts, and decision-makers on employing MP to enhance commanders' decision-making. Furthermore, the study's findings contribute to understanding behaviors in cybersecurity operations processes. Hence, this research aims to fill gaps in understanding and theory, offering new insights to enhance commanders' situational awareness within the cyber operational environment.

D. REPORT ORGANIZATION

Chapter I establishes the foundation of the report by emphasizing the research motivation and exploring the intricacies of C-OPE. It also outlines the research's primary objectives.

Chapter II explores existing literature, academic publications, and pertinent resources. It introduces the cyber operational environment, highlighting key frameworks like the NIST Cybersecurity Framework and The Johns Hopkins Applied Physics Laboratory's (APL) Information Assurance and Computer Defense (IACD). The chapter further analyzes methodologies to bolster a commander's decision-making, emphasizing subject matter experts, operations research, and modern decision-making advancements.

Chapter III delineates the research's methodological approach. It delves into the emergent behavior analysis, detailing essential steps. The chapter presents an exhaustive overview of the MP, followed by an in-depth exploration of the research design, encompassing five distinct stages designed to identify specific behaviors and patterns.

Chapter IV presents the research findings. It revisits each methodology stage, discussing corresponding results. The chapter covers the Systems-of-Interest (SoI), event trace analysis, and classifications based on emergent behavior strengths and polarities. It concludes by contextualizing the cyber environment and offering a streamlined visual representation of the SoI within the broader system.

Chapter V, the concluding chapter, synthesizes the research's main findings. It explores the study's implications and contributions. This chapter ends with recommendations for future research, highlighting potential areas for further exploration.

II. LITERATURE REVIEW

This chapter presents a thorough literature review using relevant sources such as journals, textbooks, and current research. The review encompasses subjects related to the research problem: Cyber Operational Preparation of the Environment (C-OPE) and enhancing a commander's decision-making ability. This literature review is foundational for providing useful insights into vital academic principles and facilitating a more rigorous evaluation of the research problem.

A. THE CYBER OPERATIONAL ENVIRONMENT

The cyber operational environment is a complex and ever-changing space, and its representation can vary depending on the specific purpose and perspective. For instance, the most suitable representation depends on the specific use case and intended audience. Cybersecurity frameworks and playbooks can serve as excellent resources for understanding the environment in which operations occur.

1. NIST Cybersecurity Framework

According to Calder's guide on the subject (2018), the NIST Cybersecurity Framework (CSF) offers an organized approach with its guidelines and processes to provide a comprehensive perspective on the cyber environment. This framework is characterized by five primary functions: Identify, Protect, Detect, Respond, and Recover. Each of these functions presents specific guidelines and processes tailored to manage cyber risks effectively.

a. Identify

The NIST CSF Identification function assesses how attackers could exploit vulnerabilities in a system or network. This function produces a system risk profile, the foundation for an organization's risk assessment (Calder 2018).

Organizations must inventory their assets to identify and assess their system's risk profile. This inventory identifies system components such as network architecture, devices,

services, applications, and data stores (NIST 2018). Moreover, organizations must prioritize based on the value of their assets and data while comprehending the negative impact of successful attacks (Calder 2018).

According to Calder (2018), the Identify function also encompasses processes for identifying threats and vulnerabilities which could adversely affect organizational resources. For example, organizations should evaluate external and internal threats such as phishing schemes, viruses, malware, and ransomware, among other malicious software. In addition, organizations should consider their susceptibility to physical security risks, including data breaches and natural disasters. By comprehensively cataloging the possible dangers, organizations can determine weaknesses in their security posture and formulate ways of addressing them.

In addition to identifying threats and vulnerabilities, organizations must regularly evaluate their security controls' effectiveness. This evaluation includes assessing compliance with industry standards and organization-specific security policies or procedures. Organizations must use risk assessment tools to identify vulnerabilities or gaps in their security process (Calder 2018). Through regular assessments, organizations ensure they stay up to date and maintain their security protocols.

b. Protect

The Protect function aims to safeguard organizations from potential cyber threats and vulnerabilities by deploying appropriate security controls (Calder 2018). This function involves developing policies, procedures, and technical measures to mitigate risks or damages caused by an attack or incident. The Protect function's controls, such as access control, authentication protocols, encryption techniques, firewalls, and antivirus software, ensure that an organization's systems and data maintain confidentiality, integrity, and availability. Additionally, the function includes procedures designed to identify malicious activities or suspicious behavior within the network (Calder 2018).

The Protect function also includes protocols to address cybersecurity threats or incidents. An essential aspect of this function is an organization's contingency plans for

recovering any compromised data or systems in case of an attack or breach of sensitive information (Calder 2018).

c. Detect

The Detect function identifies cybersecurity events by continuously monitoring the network for unusual or suspicious activity (Calder 2018). This function allows organizations to catch potential threats and prevent them from causing damage.

The Detect function includes various processes for monitoring systems and implementing alerting mechanisms such as intrusion detection systems, antivirus software, and firewalls that can detect known threats and anomalous activities. To be successful organizations must ensure visibility into all network activity, including logs of user actions, machine logs, system configuration changes, and files accessed on the web (Calder 2018).

The function requires organizations to review their security posture regularly. They should update their alerting mechanisms whenever new threats emerge to remain vigilant of any potentially malicious activity on the network (Calder 2018)

The Detect function furthermore involves developing incident response strategies. It includes outlining the steps to take when an attack is detected or suspected and how to notify relevant individuals within the organization of these findings (Calder 2018).

d. Respond

The fourth function of the NIST CSF, Respond, encompasses several critical processes for mitigating the effects of cybersecurity incidents (Calder 2018). These processes involve implementing strategies to contain malicious activity as soon as detected to prevent it from spreading further throughout an organization's systems and networks.

According to Bykowski (2022), organizations should implement automated processes whenever possible to reduce or limit the disruption caused by incidents. Likewise, automation should streamline response efforts by automatically triggering methods such as isolating affected systems, deactivating user accounts, and alerting internal personnel about the incident whenever malicious activity is detected.

Furthermore, the Respond function has procedures for notifying critical stakeholders about a cybersecurity incident (Calder 2018). These procedures involve identifying the necessary individuals within the organization responsible for preventing further damage. Additionally, they should involve notifying external parties affected, such as customers or partners, whom attackers may have compromised during the attack.

Finally, the Respond function requires organizations to develop plans to recover from any incidents on their network (Calder 2018). Organizations should be able to quickly restore services following an attack so that normal operations can resume without significant disruption or loss of data. The Respond function also includes procedures to ensure that organizations securely restore systems, preventing future attacks from succeeding.

e. Recover

The fifth and final function of the NIST CSF is to recover, which is critical to ensuring that organizations promptly and securely reinstate their systems following a cybersecurity incident (Calder 2018). The Recover function involves implementing technical controls and procedures that aim to remediate all malicious activity on the network and minimize any disruption of operations.

The Recover function includes a reliable backup process to facilitate recovery by restoring data or applications if necessary (Calder 2018). Restoring normal operations quickly after a security incident involves safely bringing all systems back online and remediating any malicious activity.

The Recover function also encompasses the organization's processes for conducting post-incident investigations (Calder 2018). This process involves analyzing logs, studying traffic patterns, and evaluating user accounts for suspicious activity. Post-incident activities should incorporate mechanisms to monitor systems accurately, such as software safelists or application control policies. These mechanisms help identify any remaining malicious activity on the network and alert personnel if further threats are detected.

2. The Federal Government Cybersecurity Incident and Vulnerability Response Playbooks

The *Cybersecurity Incident and Vulnerability Response Playbooks*, published by Cybersecurity and Infrastructure Security Agency (CISA), are a resource for understanding the cybersecurity environment and providing standardized procedures for Federal Civilian Executive Branch (FCEB) Information Systems (2021). “These playbooks provide FCEB agencies with a standard set of procedures to identify, coordinate, remediate, recover, and track successful mitigations from incidents and vulnerabilities affecting FCEB systems, data, and networks” (Cybersecurity and Infrastructure Security Agency [CISA] 2021, 3).

The Vulnerability Response Playbook has the top priority of protecting systems from compromise, according to CISA’s 2021 report. This standardized process helps agencies and organizations understand critical and dangerous vulnerabilities in their cyber environment. The Playbook facilitates communication and coordination between the incident and vulnerability response activities and provides common definitions for key cybersecurity terms (CISA 2021, 3). It addresses vulnerabilities recognized by the affected agency, CISA, industry partners, and others in the related mission space. These vulnerabilities are typically associated with common vulnerabilities and exposures (CVE) descriptors, but organizations may encounter new vulnerabilities without a CVE or vulnerabilities due to misconfigurations.

CISA follows a standard vulnerability management plan with various stages. Figure 1, provided by CISA, illustrates the process starting with proactive identification. This stage involves monitoring threat feeds, information sources, such as CISA resources, external threat or vulnerability feeds, and internal SOC monitoring and incident response (CISA 2021). The plan prescribes a vulnerability assessment to determine its severity, susceptible software versions, and other investigation steps, including indicators of compromise (IOCs) that can identify exploitation.

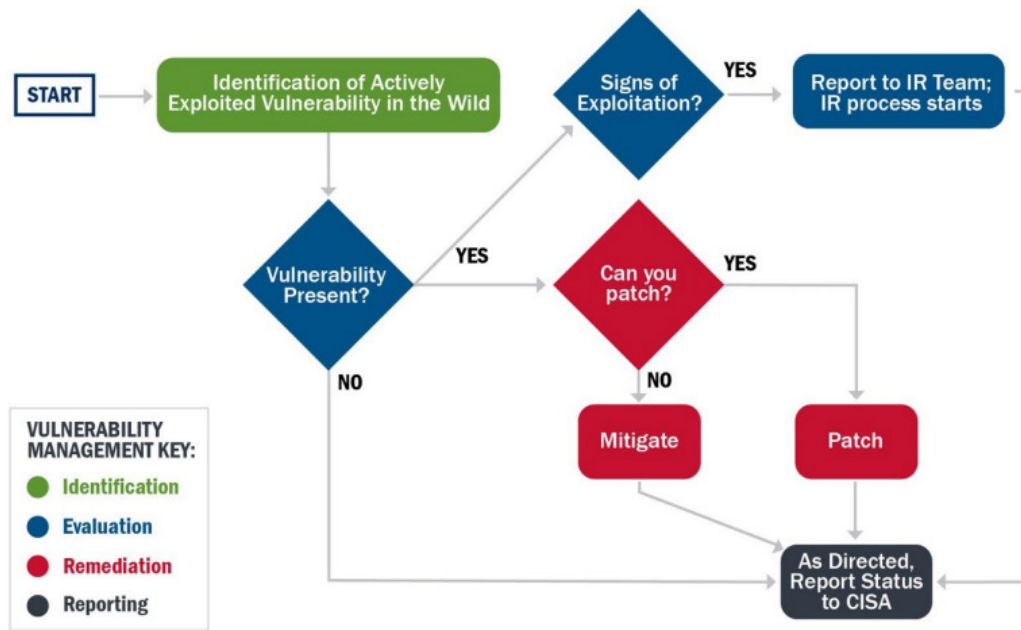


Figure 1. Vulnerability Response Phases. Source: CISA (2021).

During the evaluation stage, stakeholders assess the presence and severity of a vulnerability in the environment, employing methodologies like Stakeholder-Specific Vulnerability Categorization (SSVC) to gauge the criticality of affected software or hardware (CISA 2021). Should exploitation occur, stakeholders investigate the severity and take corresponding measures. They utilize best practices to identify signs of exploitation and irregular activities tied to vulnerable systems. Additionally, they search for known IOCs associated with the exploitation, comply with detection processes mandated in CISA directives, and engage third-party incident responders if needed. If exploitation transpires, stakeholders promptly launch activities as delineated in the Incident Response Playbook.

The remediation phase follows the evaluation stage. In this phase, stakeholders remediate the vulnerability by applying appropriate patches, workarounds, or other mitigations. Verifying the effectiveness of the remediation process is necessary to ensure that the vulnerability no longer exists in the environment (CISA 2021). Stakeholders ensure the security of systems by reporting vulnerabilities to relevant parties, including system

owners, incident response teams, and CISA. They also inform external partners and stakeholders, such as other federal agencies, industry partners, and the public.

To respond to vulnerabilities effectively, agencies must enact robust vulnerability management practices, necessitating creating and maintaining a comprehensive asset management system (CISA 2021). Such a system includes precisely tracking all systems and networks operated by the agency and those managed collaboratively with other organizations, contractors, and service providers, encompassing cloud-based systems (CISA 2021). Furthermore, agencies must monitor operating systems and applications across all systems to gauge the pertinence of vulnerabilities within the environment (CISA 2021). The Vulnerability Response Playbook targets specific vulnerabilities that adversaries may exploit to access computing resources without authorization. It is designed to supplement existing vulnerability management initiatives, not to supersede them.

The Vulnerability Response Playbook takes a comprehensive approach to address and remediate security vulnerabilities, encompassing all stages of the vulnerability management process from proactive identification to reporting. Additionally, the Playbook offers guidance on pinpointing indicators of compromise and orchestrating incident responses. Understanding an organization's security stance and defense strategies enables precise risk assessment, detection of weaknesses, and development of informed countermeasures if needed.

The Incident Response Playbook, shown in Figure 2, outlines a step-by-step process for responding to cybersecurity incidents, adhering to NIST Special Publication 800-61 Rev. 2 guidelines. This process confirms malicious cyber activity, regardless of whether officials have declared it a major incident or have not yet excluded it. Such incidents might encompass lateral movement, credential access, data exfiltration, network intrusions affecting multiple users or systems, and compromised administrator accounts.

These playbooks standardize processes and procedures, facilitating coordination and effective responses, tracking cross-organizational actions, cataloging incidents for future management, and guiding analysis and discovery. By understanding these processes and procedures, agencies can make more informed and customized decisions for their

specific environment, enhancing situational awareness and enabling them to make decisions in the face of contingencies during a cyber operation.

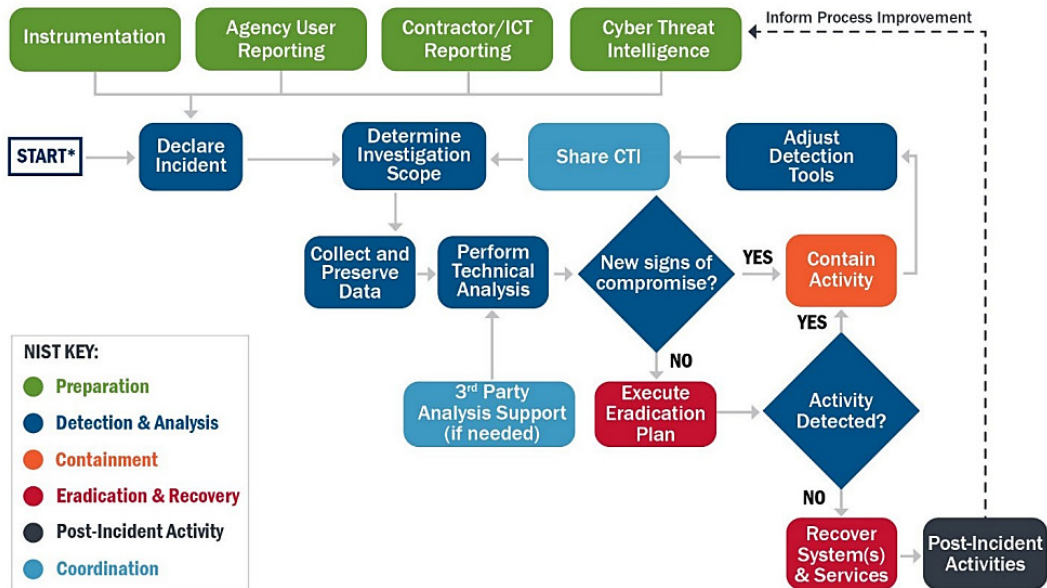


Figure 2. Incident Response Process. Source: CISA (2021).

3. Cyber Attack Life cycle Model and Cyber Kill Chain Methodology

The Cyber Attack Life cycle Model (CALM) and The Cyber Kill Chain methodology are two prominent frameworks for describing the cybersecurity environment during a cyber-attack.

The Cyber Attack Life cycle Model (CALM) from MITRE Corporation helps organizations to understand and mitigate cyber threats by providing a structured framework for identifying, analyzing, and responding to potential attacks throughout their life cycle (MITRE 2018). Systematically dismantling an attack into phases can be a successful strategy to enable rapid detection of vulnerabilities and effectively allocate resources to mitigate them. CALM consists of seven stages described in Table 1: Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control, and Actions on Objectives (MITRE 2018).

Table 1. Cyber Attack Life cycle Model Stages. Adapted from MITRE (2018).

Kill Chain: The 7 Stages of a Cyberattack	
Phases	Description
Reconnaissance	Attackers gather information about their target to identify weaknesses in the system that they can exploit in subsequent stages.
Weaponization	Attackers create malicious code, such as a virus or a malicious website, which can be used to infiltrate the target's computer and execute nefarious activities.
Delivery	Attackers deliver the attack vector to the target using various methods, such as email or the internet, to ensure that it reaches its intended target.
Exploitation	Attackers using the weaknesses they discovered during the Reconnaissance phase to gain unauthorized access to the target's computer or system, enabling them to conduct their activities.
Installation	Attackers place malicious software on the target's computer, which can be accomplished through several means, such as sending a virus via email or accessing a website.
Command and Control	Attackers control the target's computer using the malicious software they installed, enabling them to carry out unauthorized activities.
Actions on Objectives	Attackers achieve their goals, such as stealing data. To prevent attackers from reaching this final stage, it is crucial to detect and stop them in the earlier stages of CALM.

Response teams use CALM to quickly assess the situation and prevent further damage when dealing with cyber-attacks. By accurately understanding the stage of the attack, the team can focus their efforts on the most effective measures, reducing the chances of success for the attackers. This approach also helps identify vulnerabilities and misconfigurations, allowing organizations to take preventative action before potential threats become a reality.

According to Lockheed Martin, the Cyber Kill Chain (CKC) methodology is a more advanced approach to gaining situational awareness in cybersecurity (Lockheed Martin, n.d.). While CALM examines attacks holistically, covering all aspects of malicious activity, the CKC provides detailed insights that enable organizations to closely monitor each step within the attack cycle (Lockheed Martin, n.d.). The CKC methodology identifies

seven steps to understand the progression of cyber threats, described in Table 2, each step emphasizes the roles of offense and defense.

Table 2. The Lockheed Martin Cyber Kill Chain. Adapted from Lockheed Martin (2015).

LOCKHEED MARTIN CYBER KILL CHAIN: Seven Steps Overview	
Steps	Description
Reconnaissance	In this step, adversaries collect information on potential targets, such as email addresses, employee profiles on social media, and internet-facing servers. Defenders aim to detect these activities, uncover adversaries' intentions, and prioritize their defenses accordingly.
Weaponization	Adversaries produce and automate malicious tools, integrating them into deliverable payloads. Although defenders cannot detect this process in real time, they analyze malware artifacts to understand the weaponization phase and strengthen their defenses.
Delivery	Adversaries transfer malware to the target using direct web server attacks, malicious emails, or compromised websites. This phase presents defenders with an opportunity to intercept and halt the cyber-attack by understanding the delivery methods and targeted assets.
Exploitation	Adversaries exploit software, hardware, or human factors vulnerabilities to gain access. "Zero day" refers to the unique exploit code used during this phase. Defenders counter these threats by hardening systems, training users, and conducting regular vulnerability assessments.
Installation	Adversaries ensure their presence on a victim's system, often by installing backdoors or other malware. Tactics may include manipulating file timestamps to hide malicious activities. Defenders deploy endpoint tools to detect these activities and identify unusual system behaviors.
Command and Control	In this step, the malware establishes a communication channel with the adversaries, typically over the web, DNS, or email protocols. Defenders aim to sever this communication to prevent adversaries from controlling the compromised system.
Actions on Objectives	Adversaries execute their mission upon gaining access, varying from data collection and exfiltration to system destruction. The longer the access persists, the greater the potential impact. Defenders must promptly detect and respond, using forensic tools for assessment and applying rapid response protocols.

In summary, utilizing behavioral models like the Cyber Attack Life cycle Model and Cyber Kill Chain is useful for enhancing awareness of the cyber operational environment. Through these methods, organizations can comprehend the dependencies between their systems and identify potential flaws and risks. Organizations can make more well-informed cyber operations decisions by gaining deeper insights into system dynamics.

4. The Johns Hopkins Applied Physics Laboratory Information Assurance and Computer Defense Program

The Information Assurance and Computer Defense (IACD) program at the Johns Hopkins Applied Physics Laboratory (APL) provides organizations with a comprehensive framework for implementing security processes that integrate technology, policy, and operations (Integrated Adaptive Cyber Defense [IACD] 2018). The framework provides 14 example playbooks, 10 example workflows, and one local instance. These are categorized according to the NIST Cybersecurity Framework's Five Functions: Identify, Protect, Detect, Respond, and Recover. This section discusses the IACD framework and how the framework serves as a structured reference to comprehend an organization's cybersecurity environment.

To support risk and readiness assessments, it is important to update playbooks regularly. Playbooks are documents used to define policies, procedures, best practices, standards, and other essential information related to security. These must be traced to the policy and reflect any operational requirements changes. The IACD Playbook approach includes five essential types of content: Initiating Condition, Process Steps, End State, Best Practices & Local Policies, and the Relationship to Governance or Regulatory Requirements. Table 3 outlines the five types of content in the IACD playbook.

Table 3. Playbook Content Types. Source: Information Assurance and Computer Defense (2017).

Playbook Content	
Types	Description
Initiating Condition	The pre-defined event or condition that the documented security process is designed to address.
Process Steps	The individual steps in the process being documented – can be performed manually or automatically.
Best Practices & Local Policies	Located in a box below Process Steps, this section contains a variety of options and candidate actions that can be tailored to fit the policies and practices of the organization, allowing it to evolve with added capabilities.
Relationship to Governance or Regulatory Requirements	Maps the playbook’s captured processes to appropriate governance or regulatory requirements to enable tracking, auditing, and assessment. Each playbook is mapped to a subset of controls associated with governance or regulation policy.
End State	Clearly details the desired outcome after implementing the activities detailed in the playbook, based on the initial stimulus.

By delineating processes in a playbook, organizations can pinpoint the initiating condition, enumerate the necessary steps, define the targeted end state, and outline optimal practices or alternatives for executing the steps. Additionally, the content types chart the connections between these processes and pertinent governance or regulatory mandates.

To further automate security processes, IACD creates workflows that are machine-understandable codifications of playbooks (Johns Hopkins University Applied Physics Laboratory [JHU APL] 2020). Workflows are codifications of playbooks that machines can execute more quickly, machine-to-machine. Organizations can adopt these workflows in different environments, making them shareable (JHU APL 2020). IACD also creates local instances as part of the automation effort. These are tailored versions of a workflow designed for a specific environment. The local instances are repeatable and auditable. They can be customized based on the environment’s needs and capabilities while remaining consistent with local policies, procedures, thresholds, decisions, and processes (JHU APL 2020). The local instances incorporate local technologies, products, and assets deployed in the environment.

The IACD framework is a resource for gaining insight into an organization's cyber environment. Playbooks, workflows, and instances that document management policies, procedures, and strategies a comprehensive understanding of the entire digital ecosystem can be achieved.

Diagrams featuring graphical components can also represent the framework, helping comprehend policies, activities, and processes. Figure 3 is a sample playbook which provides a visual representation to support the description of the IACD framework and its approach (IACD 2017). The sample playbook is an illustrative example of the five types of content typically found within an IACD playbook. These content types, namely Initiating Condition, Process Steps, End State, Best Practices & Local Policies, and Relationship to Governance or Regulatory Requirements, play vital roles in developing a comprehensive playbook that addresses cybersecurity challenges effectively.

The Initiating Condition, shown in the upper left-hand section of the playbook, identifies the specific event or condition that triggers the documented process (IACD 2017). It is a starting point, enabling organizations to respond to potential threats or incidents proactively. By clearly defining the initiating condition, organizations can better understand the context and purpose of the playbook, ensuring that the subsequent process steps align with the identified trigger.

The Process Steps provide instructions on executing the process. Depending on the organization's needs and capabilities, these steps can be performed manually or via automation. Each step involves a specific action or task that needs completion to attain the desired security level. This structured approach aids organizations in enhancing their operations and responding more efficiently.

To further enhance the effectiveness of the playbook, the Best Practices & Local Policies section, shown in a box at the bottom of the figure, provides a set of options and candidate actions (IACD 2017). This section recognizes that different organizations may have individual policies and practices, and therefore, it offers flexibility to tailor the playbook accordingly. By incorporating best practices and aligning with local policies, organizations can ensure that the playbook adapts to their specific needs and evolves with the addition of new capabilities.

The playbook's Relationship to Governance or Regulatory Requirements section, as shown in Figure 3, crucially links documented processes to corresponding governance and regulatory frameworks (IACD 2017). These frameworks include the Continuous Diagnostics and Mitigation (CDM) program, the National Institute of Standards and Technology (NIST) Cybersecurity Framework, NIST Special Publication 800-53, and the Center for Internet Security (CIS) Critical Security Controls (CSC). Incorporating these frameworks into the playbook enables organizations to prove compliance during audits and consistently monitor their cybersecurity posture.

The playbook strives to reach a well-defined End State or a desired outcome (IACD 2017). This outcome stems from the initial stimulus or triggering event. By clearly stating the end goal, organizations can align their efforts and evaluate the effectiveness of their cybersecurity processes. Figure 3 is a practical representation of how the framework is not merely a theoretical concept, but a tangible tool organization can utilize to define and implement cybersecurity processes. It highlights the practical application of the playbook in real-world scenarios, emphasizing its value in strengthening an organization's cybersecurity posture.

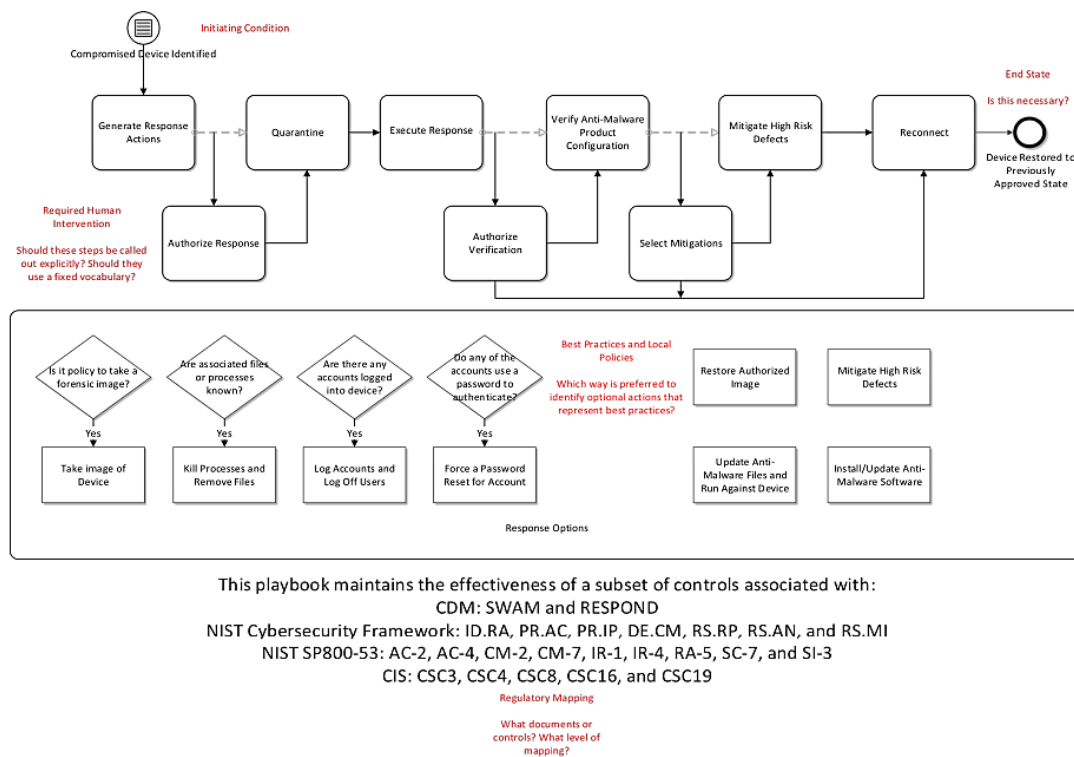


Figure 3. Sample Playbook. Source: IACD (2020).

5. Business Process Model and Notation

Business Process Model and Notation (BPMN) offers a clear understanding of the complete system while also helping to identify redundancies or inefficiencies (Object Management Group [OMG] 2011). The standardized graphical representation of business process activities and flows allows organizations to effectively document, analyze, optimize, and simulate their workflow operations. Using nodes and arc symbols, users can visually depict entire business processes from start to finish—including tasks, events, gateways, data objects, messages, and exceptions.

The use of BPMN in cyber operations is on the rise as it provides an enhanced view of how IT processes are executed (Graja, Kallel, and Guermouche 2019). By creating models that accurately depict digital systems' workflow, organizations can better understand their cyber environments. This enhanced understanding of cyber environments

makes it easier to develop more efficient cybersecurity policies and procedures that can help reduce cyber threats.

Figure 4 presents a BPMN diagram, a straightforward illustration organization can leverage to structure their malicious activity detection process.

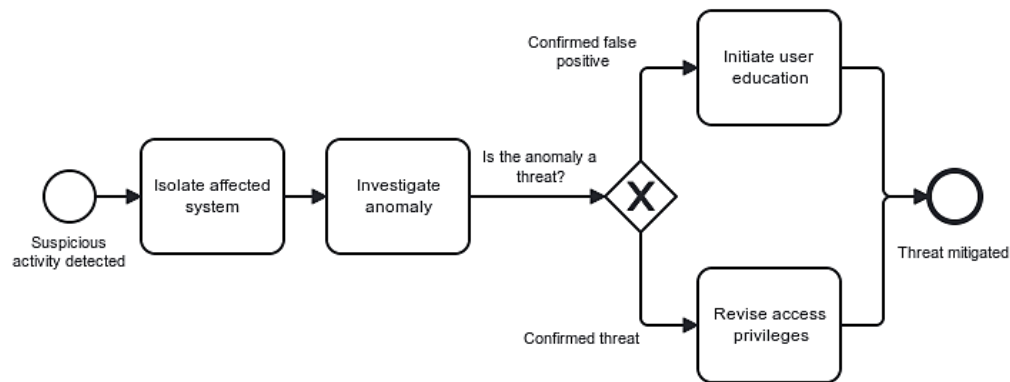


Figure 4. Detecting Malicious Activity

By modeling these systems with BPMN, organizations can better understand potential threats and weaknesses in their network design (Marshall et al. 2017). This understanding can then inform the adjustment of settings or the development of additional strategies for more effectively mitigating threats (Hua et al. 2015).

Moreover, BPMN models can help identify opportunities for automating certain processes within a security practice. Automating tedious manual tasks, such as log analysis or incident response, frees practitioners' time to focus on more important activities like developing new strategies and quickly responding to incidents (Marshall et al. 2017). Automation also helps ensure security controls remain current with industry trends and standards and reduces response time when incidents happen (Hua et al. 2015; Vatsavai et al. 2019).

Overall, BPMN offers multiple advantages for organizations aiming to improve their cyber defense posture including gaining a better understanding of their cyber environment. Detailed visual representations of workflows allow practitioners to gain enhanced visibility into the system while reducing associated risks with operations.

Additionally, BPMN aids in the coordination of responses during incident management processes, which helps reduce downtime due to remediation tasks (Chen et al. 2017; Marshall et al. 2017; Hua et al. 2015; Vatsavai et al. 2019). These benefits make BPMN a valuable asset for any organization wanting to deepen its knowledge of its cyber environment.

B. ENHANCING A COMMANDER'S DECISION-MAKING CAPABILITY

In military operations, commanders' success hinges on making informed decisions swiftly in dynamic environments. The ensuing sections delineate how gathering critical information through specialized teams, Operations Research, and data visualization techniques can augment decision-making abilities, leading to superior outcomes.

1. Subject Matter Experts

Specialized teams of experts, knowledgeable in areas relevant to the mission, enhance a commander's decision-making process. As an invaluable resource, these teams contribute to military strategy, cultural nuances, and tactical maneuvers. These professionals assist commanders in understanding geopolitical situations and offer valuable insight into local cultures and customs critical to military operations (Antúnez 2021). Their contributions enable commanders to anticipate enemy actions and make informed decisions based on the team's collective wisdom (Finney 2008).

For example, in 2014, a specialized team of cultural specialists, tactical observers, and intelligence analysts brought critical insight into local cultures that impacted military operations in the middle east (Antúnez 2021). This crew offered unique perspectives on diverse components of the dispute, such as religious and tribal elements, which enabled commanders to make judicious decisions considering these factors. The combination of cultural aptitude and tactical experience helped the commander make well-informed choices, resulting in successful outcomes for U.S. forces (McAfee and Brynjolfsson 2017).

2. Operations Research

Operations Research is a discipline leveraging mathematical and analytical methods to optimize decision-making and complex systems (Taha 2017). It applies

quantitative techniques to scrutinize and solve problems across diverse fields, including business, engineering, logistics, and military operations.

According to Taha (2017), systems optimization seeks the best arrangement of system components to achieve specific goals. It focuses on enhancing performance, efficiency, and effectiveness while considering constraints, objectives, and trade-offs. In Operations Research, this optimization often employs mathematical modeling and specific techniques. The goal is to pinpoint the optimal decision variables that maximize or minimize an objective function within the system's limitations or requirements. Here, the objective function symbolizes the aspect to be optimized, while constraints outline the system's boundaries.

Analysts can examine the interaction of diverse system components by applying system optimization to cyber operations, yielding essential insights into environmental complexities. This method allows analysts to model these interactions, pinpointing strengths and vulnerabilities. They secure a competitive edge by equipping commanders with information to make judicious decisions on resource allocation and strategy. For example, they may discover network configurations resilient to specific cyber-attacks or detect frailties in security measures. Such insights can prompt proactive steps like bypassing robust systems or adjusting vulnerable devices.

Operations Research techniques can generate information to assist commanders in making more well-informed decisions, such as:

- **Risk Assessment:** Operations Research models aid in assessing and quantifying risks tied to cyber threats and vulnerabilities. Commanders can prioritize resources and efforts by analyzing the likelihood and impact of different cyber-attacks. Consequently, they can either mitigate the gravest risks or seize the most valuable opportunities in alignment with their objectives.
- **Resource Allocation:** Operations Research techniques can streamline the distribution of limited resources, like budgets, personnel, and technology, to enhance operational effectiveness. By weighing constraints and

objectives, models can suggest ideal resource allocation strategies that harmonize cost, risk, and performance.

- **Incident Response:** Operations Research can model diverse scenarios and rigorously analyze response strategies. Using these simulations, commanders can evaluate potential outcomes and the effectiveness of different approaches during a cyber incident. Consequently, Operations Research empowers commanders to devise empirically grounded plans for judicious resource distribution, reducing cyber-attack impact. Additionally, it offers a robust framework for assessing potential responses within a targeted environment.

Understanding the interaction of system components is vital for grasping the operational environment, enabling commanders to make informed decisions. They can act proactively by identifying robust network configurations and frail security links. Such actions might include bypassing a resilient system or modifying a susceptible device, affording a strategic edge. Thus, Operations Research is pivotal in deepening our understanding of the operational environment, a key factor in the success of military missions.

3. Data Visualization

Modern technology allows commanders to quickly access extensive data sets and intuitively visualize intricate information, simplifying the navigation through disparate data points (Bremer et al. 2022). By integrating interactive dashboards with potent analytics engines that draw from various data sources simultaneously, commanders can achieve real-time visibility into operational activities and more effectively anticipate changes (Schubert et al. 2018).

In this era of data-driven decision-making, crafting clear and engaging charts is vital. Charts serve as formidable tools for commanders, enabling them to communicate and interpret information efficiently, thus streamlining informed decision-making processes.

However, creating compelling charts necessitates adherence to best practices that guarantee legibility and simplicity (Wong 2010).

In chart design, Dona M. Wong's book *The Wall Street Journal Guide to Information Graphics* emphasizes the crucial role of typography in accurately conveying information. The key factors involve choosing the type, size, and style while ensuring legibility. It is crucial to keep typography simple and easy to read. Avoid using small or condensed types. Use bold or italic styles sparingly to emphasize specific points without overwhelming the chart. The objective is to describe the data effectively through typography, maintaining its significance without allowing it to distract. Figure 5 illustrates the “Dos” and “Don’ts” of typography in charts.

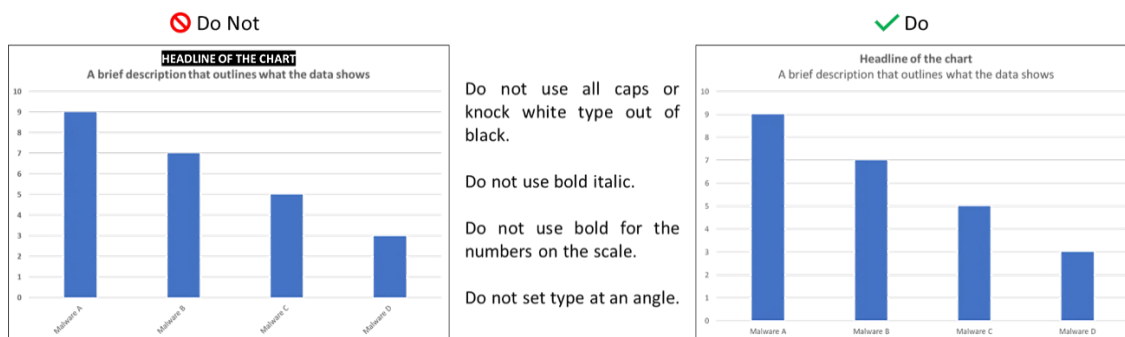


Figure 5. Typography. Adapted from Wong (2010).

When crafting charts for commanders, simplifying and minimizing complexity is vital to ensure clear, precise visuals. Eliminate extraneous graphical elements, letting the underlying data take center stage, and maintain consistent font styles while thoughtfully arranging chart components for improved clarity and interpretation. Color selection is another crucial factor; strategic use of colors enhances visualization and comprehension. Designing a harmonious color palette with basic colors and varying shades promotes clarity and uniformity. Additionally, consider the needs of color-blind readers, opting for schemes that maintain readability (Wong 2010).

In modern command and control settings, dashboards play a vital role. They allow commanders to access and visualize complex information intuitively and efficiently using

comprehensive data sets. In a 2006 *Information Dashboard Design* book, Stephen Few uses the analogy of a car's dashboard to illustrate the purpose of a dashboard. He suggests that a dashboard provides essential information at a glance, thus enabling efficient operation. It aims to display the appropriate amount of information necessary to achieve specific objectives on a single computer screen for easy monitoring. The design concentrates on visualizing these objectives, allowing for instant and effective viewing.

Integrating interactive dashboards with robust analytics engines, drawing from various data sources simultaneously, lets commanders gain real-time visibility into operations and foresee changes. In 2019, the U.S. Army leveraged this technology during an exercise to monitor troop movements and enemy strategies. This design afforded commanders a user-friendly interface, empowering them to make informed decisions and act decisively based on the collected intelligence (Schubert et al. 2018).

In command-and-control settings, interactive dashboards and analytics engines offer a comprehensive view of operations, facilitating the analysis of complex data sets to detect patterns, trends, and irregularities. Their real-time data integration heightens situational awareness and enables swift reactions to unfolding situations. Additionally, these dashboards act as communication instruments, fostering easy sharing and collaboration through clear, succinct information.

However, it is important to note that designing effective command and control dashboards requires careful consideration of the specific operational context and user requirements. Concentrating on the essential information is best to prevent overwhelming the user (Few 2006). As command-and-control environments evolve, refining dashboard design principles becomes crucial for optimizing their potential in enhancing decision-making, promoting effective communication, and supporting operational success.

Commanders can bolster operational awareness and decision-making efficiency in real time by utilizing robust data analytics and interactive dashboards. Adherence to design principles emphasizing clear, simple, and effective information communication is equally crucial. The primary goal is to craft clear, comprehensive, and intuitive representations of complex data. As command-and-control environments evolve, so must the tools within

them. Persistent refinement and adaptation ensure that these tools effectively facilitate informed, timely, and decisive responses to diverse situations. Commitment to this iterative process allows military and strategic organizations to maintain a critical advantage in the ever-changing landscape of modern warfare.

4. Advancements for Decision-Making

Commanders can enhance their decision-making abilities using Artificial Intelligence (AI). As defined by Geetha and Sendhilkumar (2023, 16), Machine Learning, a type of AI, is the science of detecting patterns and making predictions from vast, diverse, and rapidly changing data sets, often outperforming traditional methods in accuracy, scale, and speed. Predictive analytics enabled by AI can give commanders detailed insights and intelligence, giving them an edge in making informed decisions. AI can uncover patterns, trends, and potential outcomes that typically go unnoticed by humans, resulting in faster and more effective responses to changing scenarios (McAfee and Brynjolfsson 2017). Ultimately, incorporating AI can lead to better outcomes and more successful operations.

For example, in 2017 DARPA launched the “Collaborative Operations in Denied Environments” (CODE) program (Targeted News Service 2018). This program involved the deployment of AI-powered drones designed to gather and share information among Unmanned Aerial Vehicles (UAVs). The drones were equipped with AI algorithms to detect trends and activities in their surroundings, assessing potential dangers for U.S. Forces (Barbaro et al. 2018). The CODE program also enabled communication between drones from different units, allowing for unprecedented collaboration. For example, when one drone detects unusual ground activity, it can relay this information to another drone. Utilizing AI’s capability to process extensive data sets, the second drone can determine whether to engage or avoid the area based on the collaborative data set.

5. Behavior Approach

In the late 1970s, J.C. Willems introduced an innovative behavioral approach to systems and control theories. The approach addressed inconsistencies in classical methods, including state space, transfer function, and convolution representations. It focused on the

need for a comprehensive analysis and control framework that remains true to its physical foundations.

The revolutionary aspect of this behavioral approach rests on its balanced prioritization of input and output variables. This approach lays the foundation for system theory and control within a robust framework. Integrating existing procedures and introducing findings regarding systems' controllability and interconnection control (Willems 1997). Moreover, it provides insights into system identification (Kaneko 2008) ensuring all signals pass through this comprehensive framework.

To enhance commanders' decision-making in the cyber domain, they must achieve comprehensive situational awareness grounded in a deep understanding of system behaviors. Rather than relying solely on known threat "signatures," this approach assesses typical user and system behaviors, flagging deviations as potential threats (Ragan and Strasser 2020). Given the rapid evolution of cyber threats that often target human vulnerabilities, this proactive stance is indispensable for safeguarding data and countering threats. The two behavioral approaches most relevant to this research are the Behavior-Based Intrusion Detection System (BIDS) and Physical-Cyber-Social (PCS).

Malek and Trivedi (2018) introduced the BIDS as an AI-driven approach that harnesses machine learning algorithms and data mining techniques to identify and protect against anomalies in user behavior and malicious activities. As an anomaly-based detection system, BIDS identifies deviations from normal behavior, examining user actions such as data access or access attempts to detect abnormal patterns. This strategy helps BIDS identify previously unknown attacks that may have slipped past other security measures, making it effective at detecting and preventing abnormal network traffic and behavior-related cyber threats.

In their scholarly article, Malek and Trivedi (2018) explore various detection methods employed by BIDS. The first method detects potential malicious traffic or anomalous user actions by identifying network activity patterns. The second method leverages signature-based identification and statistical analysis to recognize known intrusion attempts and subtle deviations from normal behavior. BIDS also monitors user

accounts for signs of compromise or unusual activities, such as login attempts from unfamiliar devices or locations. Lastly, the system includes a response mechanism that alerts administrators to suspicious activity, enabling them to act appropriately and promptly.

This advanced system offers organizations a robust solution for the swift identification and response to potential threats, limiting the likelihood of extensive damage. The distinguishing feature of BIDS compared to traditional intrusion detection systems lies in its ability to detect subtle anomalies. The automation of its response mechanism further streamlines the containment process, significantly reducing downtime associated with manual response procedures (Malek and Trivedi 2018).

Despite its many advantages, BIDS has certain limitations. For instance, the system's reliance on AI algorithms may limit its ability to recognize specific attacks if attackers modify their tactics slightly (Chinyane et al. 2019). Furthermore, BIDS's pattern recognition techniques may trigger false alarms when regular activity suddenly deviates from expected norms (Fawcett et al. 2004). Nevertheless, most experts agree that BIDS provides organizations with a powerful tool to protect against malicious actors (Tong et al. 2013; Chinyane et al. 2019).

Sheth, Anantharam, and Henson (2013) proposed the Physical-Cyber-Social (PCS) computing framework as an integrative data, information, and knowledge approach. This framework interprets, integrates, and correlates information, providing human agents with contextually relevant abstractions. Consequently, it affords a holistic view of human-environment interaction.

Regarding security, data analytics is pivotal in monitoring user activity across PCS interaction realms to detect suspicious activities before they escalate into serious threats (Tran et al. 2016). Through predictive analytics based on past user behavior, PCS analysis can anticipate changes in security threats before they occur and enable organizations to stay ahead of malicious actors, protecting themselves against potential attacks (Mao et al. 2016).

PCS computing also provides organizations with valuable insights into user preferences and interests by analyzing their interactions with physical objects and digital content (Kanakia et al. 2017). For example, retail stores or media outlets can offer customers personalized experiences that cater to their specific needs, leading to better engagement with products or services (Yoo et al. 2016).

The capacity of PCS computing to generate personalized solutions has cemented its role as a foundational element in informed decision-making processes. Hence, at the dawn of the 21st century, various stakeholders have increasingly recognized PCS analysis. This recognition derives from its ability to offer tailored solutions, strengthen security measures, and enhance the quality of decision-making procedures.

THIS PAGE INTENTIONALLY LEFT BLANK

III. METHODOLOGY

This chapter offers an in-depth assessment of relevant background knowledge and scholarly principles that shape the research approach. The investigation utilizes emergent behavior analysis to examine the cyber operational environment and employs Monterey Phoenix (MP) to create scenarios as event trajectories. These scenarios lay the groundwork for a cyclical analysis process to uncover emergent behaviors. The study aims to identify specific behaviors within the MP models to enhance situational awareness in the operational environment.

A. EMERGENT BEHAVIOR ANALYSIS

The concept of emergence has drawn significant attention in recent years due to its future applications in domains such as software engineering and robotics (Nielsen 2021). Emergent behavior, as defined in the International Council on Systems Engineering (INCOSE) *Systems Engineering Handbook* (2015), “is a behavior of the system that cannot be understood exclusively in terms of the behavior of the individual system elements.” This definition emphasizes the intricacy of systems, indicating that collective behavior may manifest unique traits that are not inferable from the behaviors of the singular elements.

Researchers have studied emergent behavior in various contexts, such as bird flocks, ant colonies, reaction-diffusion networks in chemistry, human societies, and computer networks. In these situations, the collective interactions between individual components lead to emergent behavior. Conway’s Game of Life, a 2D cellular automaton, exemplifies this phenomenon. Individual cells on a grid evolve based on four interaction rules, resulting in complex global patterns emerging from simple initial configurations (Johnston and Greene 2022).

Investigators have proposed various strategies to model or simulate emergent behavior in software systems, as Seidel et al. (2010) discussed. The study of emergent behavior can be traced back to Arthur Koestler’s early work that introduced the term holarchy, who explores how higher-level behaviors emerge from lower-level components (Koestler 1967). Other noteworthy contributions include Stuart Kauffman’s research into

self-organizing dynamical systems (Kauffman 1993), John Holland’s genetic algorithms (Holland 1992), Ray Kurzweil’s pattern recognition theory (Kurzweil 1990), Alan Turing’s cellular automata models (Turing 1952), and Stephen Wolfram’s computational mechanics’ approach (Wolfram 2002).

Scholars have expanded the application of emergent principles beyond physical and biological systems, incorporating larger-scale economic and social phenomena. For instance, they have examined emergent behavior within the contexts of financial bubbles and fads (Lux and Marchesi 1999) and political movements like populism or nativism (Aslanidis 2018). The aim is to understand these complex patterns through network analysis and collective dynamics, focusing on the nonlinear interactions between individual entities within these broader structures.

Modern programming languages are heavily rooted in emergent behavior and functional concepts, as researched by Hunt (2018). Likewise, ongoing progress in artificial intelligence aims to create smarter algorithms by utilizing emergent techniques, as Sutton and Barto (1998) noted.

In her 2022 chapter, “Exposing and Controlling Emergent Behaviors Using Models with Human Reasoning,” Kristin Giammarco explores identifying, understanding, and managing unpredictable behavior in emergent systems. This approach combines human reasoning and machine tasks to identify trends in emergent behaviors. The Emergent Behavior Analysis (EBA) Methodology, as outlined by Giammarco (2022), consists of four steps to uncover emergent behaviors:

1. Step 1 – Emergent Behavior Detection

The first step in the EBA Methodology is to render emergent behaviors visible within system models for detection. This foundational step is pivotal for the subsequent stages of comprehending and managing the identified emergent behaviors. It stresses the importance of creating a behavior superset, encompassing both desired and excess behaviors.

To initiate the detection of emergent behaviors, practitioners model the system behaviors they already know (or can predict). An “event” is a fundamental building block for behavior, representing various activities, actions, processes, and conditions. This abstract definition allows events to unify concepts, creating simple models that give rise to complex behaviors.

Two fundamental relations define each event: precedence, which sequences events, and inclusion, which positions them within a hierarchy. These relations allow practitioners to create algorithms describing behavior, including alternate paths, optional actions, and iterative loops. This method streamlines the modeling of system objects’ activities, processes, and behaviors.

It is crucial to model the system’s behaviors and environment to uncover emergent behaviors. In the context of large complex systems, this entails modeling the behaviors of individual systems within the systems of systems (SoS) and the behaviors of the SoS environment. These environmental behaviors trigger actions and responses within the system and may include alternative, optional, and iterative events. Such modeling offers a thorough understanding of possible behaviors and reveals otherwise overlooked dependencies.

In structuring alternative branches, the model must incorporate high-level events and general scenarios that foster intriguing emergent behaviors. Instead of merely concentrating on isolated events, it is more fruitful to delineate the system’s architecture or underlying principles. Practitioners can craft diverse event mixtures that provoke forecasts of potential future conditions by weaving in alternative, optional, and recurrent events at every refinement tier.

Strategically increasing the decision points in the model from a limited few to a broader array empowers analysts to manage the model’s scope and avert unwarranted combinations. During this growth phase, it is crucial to carefully identify and understand undesirable combinations rather than dismissing them prematurely, as they may conceal welcome emergent behaviors. This targeted strategy emphasizes relevant event blends that might reveal unforeseen behaviors. By probing these event sequences and scrutinizing both

the SoS and the behaviors of its environment, practitioners markedly enhance their ability to detect emergent behaviors that could remain hidden.

The first step of the EBA Methodology is pivotal, illuminating emergent behaviors through methodical modeling of observed and predicted actions. Practitioners can pinpoint expected emergent phenomena in this initial phase by employing abstract event notions, meticulously portraying the system's actions and surroundings, and thoroughly examining event combinations. This approach amplifies the identification of emergent behaviors and lays a robust groundwork for ensuing phases.

2. Step 2 – Emergent Behavior Prediction

The second step of the EBA Methodology, delineated by Giammarco (2022), thoroughly examines all event traces for potential fallacies. Here, fallacies refer to mistaken assumptions or biases that the model's author or other involved parties initially harbored. This scrutiny aims to detect and confront these fallacies, unearthing possible inconsistencies or deficits within the model. This process highlights areas for improvement and ensures a more accurate and reliable understanding of the system's behavior.

Giammarco (2022) provides an example: consider a fundamental authentication scenario where analysts have accepted certain assumptions, such as 'credentials are always presented before verification,' or 'a user will not repeatedly input credentials after denial.' Though the model's creators regarded these assumptions as factual during its inception, they critically evaluated them in this methodology phase. It is vital to recognize that the model's creators may deem these assumptions as "common sense" and incorporate them into the model without formally defining them as requirements. Thus, they embed these implicit expectations. During the second step, the creators carefully assess these assumptions. This stage fosters critical thinking and challenges established beliefs. By rigorously inspecting the least restrictive event sequences, analysts can uncover potential inaccuracies and biases they might have overlooked previously.

Concurrently, analysts, individually or in teams, brainstormed to devise conditions or event sequences that might challenge existing beliefs. During this session, analysts must employ creativity and critical storytelling to craft scenarios refuting the recognized

fallacies. Crucially, this brainstorming process should remain free of judgment, bias, or preconceived notions regarding the system or environment to ensure an authentic exploration of potential scenarios. Analysts must consciously refrain from “fixing” the model at this stage, as such alterations could unintentionally stifle the emergence of other vital behaviors. Giammarco (2022) further underscores the effectiveness of including diversity in these brainstorming sessions. Participants from varied backgrounds contribute diverse perspectives, thereby simplifying the identification of potential fallacies and the creation of possible counterexamples.

At the end of the brainstorming phase, analysts document the derived ideas and choose to present them in various formats, such as figures or tables. These summarized ideas form the basis of the predicted behaviors, essential for detecting potential emergent behaviors that analysts might have overlooked. These creative predictions, in turn, inform the following steps of the methodology.

The second step of the methodology amplifies the analyst’s ability to challenge assumptions, create inventive counterexamples, and document possible future emergent behavior scenarios. To complete this step, analysts must sustain an open and probing mindset. Ideally, this method equips them with the capability to perform an exhaustive and impartial analysis of the entire behavior set, ensuring a robust understanding of potential outcomes and dynamics.

3. Step 3 – Emergent Behavior Classification

The third step in the EBA Methodology involves classifying emergent behaviors. This step is typically performed concurrently or immediately after the prediction step. The purpose of classification is to identify and categorize the types of emergent behaviors detected and predicted during the previous stages of the analysis process.

In the classification step, analysts analyze emergent behaviors, strength (weak or strong), and polarity (positive or negative). This analysis depends on the characteristics of the behaviors and their alignment with the system’s anticipated objectives and outcomes. Weak emergence pertains to expected behaviors that comply with the system structure’s intentions or limitations, including those that analysts predict during the design and

development phase. Conversely, strong emergence describes unforeseen behaviors typically identified once the system undergoes simulation, testing or encounters an unanticipated situation during operation. Analysts did not predict these behaviors in the system's design and development phase (Adcock 2023, 153).

Additionally, emergent behaviors are also classified based on their polarity, distinguishing between positive and negative outcomes. Positive emergence refers to behaviors that fulfill the purpose of the SoS and contribute to the optimum performance of its constituent systems within their desired ranges. These behaviors are considered desirable and acceptable. Negative emergence, on the other hand, represents behaviors that do not align with the SoS's objectives and manifest undesired or unacceptable behaviors (Mittal and Rainey 2015; Zeigler 2016; Quartuccio 2018, 2020).

The classification of emergent behaviors is subjective and depends on the context. Various perspectives and stakeholders might interpret the same behavior differently according to their roles and priorities. For example, Giammarco (2022) illustrates a scenario where a safety analyst might consider a behavior unexpected, while a system engineer, with their in-depth knowledge, might anticipate it. Similarly, an end-user might perceive a behavior as negative or undesirable, whereas a regulatory body could regard it as compliant or acceptable.

Subject matter experts who have experience in their relevant domain but have yet to participate in the modeling effort directly validate the classification labels. In contrast, the model authors and individuals deeply involved in the system's modeling and design act as the primary references for classification. Their profound understanding of the system and its complexities is essential for determining the initial classification. If a behavior surprises them, it becomes especially valuable, heightening awareness among those who need it most. Furthermore, when a prediction surprises an entire organization, the value increases, offering top experts' new insights into the system.

While the perception of emergent behaviors as strong emergence might be temporary, lasting only as long as the surprise itself, it is crucial to document these surprising moments to preserve the learning that transpires (Giammarco 2022).

Acknowledging that hindsight bias can shape the perception of emergent behavior is essential. What might seem obvious in retrospect might not have been clear at the time of realization—without hindsight, identification and addressing unexpected behaviors in real-time presents a challenge. For instance, critics attacked the Mars Climate Orbiter incident for a simple math error that, in hindsight, seemed avoidable (Iannotta 2022). Yet, the need to consider retrospective viewpoints and the evidence of consequences often has to play a larger role in real-time decision-making. An unexpected behavior unearthed through modeling activities might initially face a similar bias. However, if the design process does not include awareness of possible behavior, and the modeling effort brings that awareness, it can have substantial value.

The detection step in the EBA Methodology is vital as it brings example behavior instances to the forefront, enabling a critical consideration of mis-combinations of events (Giammarco 2022). The prediction step facilitates a thorough exploration of the underlying causes and helps answer questions such as “How could that happen?” Interestingly, even a simple model, without explicitly addressing specific factors like unit conversion, has the potential to identify mis-combinations of events and trigger further investigation (Giammarco 2022). The involvement of knowledgeable reviewers or teams in this process can lead to insightful predictions and assessments.

In summary, the third step of the EBA Methodology involves classifying emergent behaviors by strength and polarity, a process shaped by stakeholder perspectives and alignment with system objectives. While subject matter experts may validate these classifications, the primary references are the model authors and individuals deeply involved in the system’s design. Surprising behaviors, particularly those that highlight unanticipated aspects of the system to experts with profound system knowledge, hold significant value as they enhance stakeholder awareness.

4. Step 4 – Emergent Behavior Control

The fourth step of the EBA Methodology focuses on controlling detected, predicted, and classified emergent behaviors. It aims to manage adverse behaviors and transition strong emergent behaviors to weak ones. The methodology seeks to retain only

the model's expected (weak) and positive or neutral emergent behaviors (Giammarco 2022). Analysts move from strong to weak emergence, eliminating unexpected and unwanted behaviors. As a result, the EBA Methodology produces a defined set of expected and acceptable behaviors for the SoS.

When unexpected emergent behaviors arise, analysts adhere to a structured procedure. First, they document the behavior, saving a model copy and annotating the detailed events and predictions that led to its realization to ensure retention, even if subsequent model changes might obscure it. Next, they meticulously examine the behavior model to identify any missing constraints or logic errors that may have caused the unexpected emergence. If they discover such issues, they modify and run a copy of the model, assessing the effect on the event traces.

Subsequently, analysts modify the model, adding necessary circumstantial events inspired by unexpected behavior. This alteration introduces alternatives, turning the unexpected behavior into an expected one. With a clear comprehension of the emergence mechanism, analysts reclassify the behavior from strong to weak, marking it as unexpected but not unwanted. However, they retain the discovery documentation in the original model, illustrating the learning achieved through the methodology.

In cases of expected but unwanted emergence, analysts treat them as recognized risks, refining the behaviors to convert them from unwanted to acceptable. They can explore failure modes and potential recovery procedures by maintaining some weak negative emergence in the model. Additionally, they diligently document any new or overlooked requirements during this control step, sharing them with project members. By recording requirements as constraints, they ensure that the unacceptable behavior identified in the model does not manifest in the actual system, thereby emphasizing the knowledge gained through the methodology.

To guide the model's actions, analysts logically adjust behavior, adding new alternative events or modifying existing constraints to define the system's preferred behavior pattern. Depending on the use and anticipated outcomes, they employ specific constraints such as logic, design, simplification, and definition. These control strategies

alter the behavior model to achieve the desired patterns. For example, analysts introduced additional events like multi-factor authentication and account locking in the authentication case study and documented previously unstated requirements (Giammarco 2022).

The following section explores the tools and approaches required for emergent behavior analysis. It outlines the essential model structure and automates the computationally demanding steps of the process to ensure efficiency. A profound understanding of model logic is crucial for predicting, identifying, and managing emergent behavior. The modeling language of MP serves this purpose, enabling comprehensive event traces that represent various system outcomes. Moreover, MP's graphical event trace representation simplifies the communication of findings.

B. OVERVIEW OF MONTEREY PHOENIX

The Navy developed Monterey Phoenix (MP), a powerful language, approach, and tool for modeling and reasoning systems and process behaviors (Auguston 2009). By employing MP, organizations may identify emergent behavior within complex systems and forecast potential outcomes. This section delineates the origins and core components of MP to prepare the reader for the analytical findings presented in Chapter IV.

1. Origins

Dr. Mikhail Auguston, a professor in the Department of Computer Science at the U.S. Navy Naval Postgraduate School (NPS), is credited with creating the MP language and approach (Auguston 2009a; Giammarco 2017a). Dr. Auguston led research efforts to develop the MP tool and verify its capabilities. Recognizing the need for a tool to manage and analyze behavior execution traces, Dr. Auguston and his team developed MP, an instrument with an accessible interface that enabled users to construct and operate behavior models effectively.

In their seminal 2006 paper, “Environmental Behavior Models for Automation of Testing Assessment of System Safety,” Dr. Auguston and Dr. Michael Shing spotlighted three groundbreaking case studies. This paper marked the genesis of the event grammar that would ultimately evolve into MP. It delved into automatic scenario generation from

environmental behavior models for testing real-time reactive systems using attributed event grammar. The authors demonstrated how the application of attributed event grammar unveils numerous opportunities for optimizing workflows. Additionally, they suggested its potential extension to diverse projects, ranging from software system architecture and business process specification to workflow design studies.

In 2009, “How to Make Software Architecture Executable” emerged, featuring the early draft of MP (Auguston 2009). This position paper proposed a novel approach to the formal specification of software system architecture emphasizing behavior models that articulate system behavior as an assembly of events as event traces. Of significance, the paper introduced the schema framework, an element MP actively integrates today. Its significant contribution to the field of software architecture resides in its innovative perspective on constructing architectural descriptions that facilitate refinement toward more detailed design specifications.

2. Scope, Event Traces, Grammar, and Syntax

MP provides a user-friendly method for individuals to examine the relationships between distinct entities. MP allows users to construct interconnected models that result in event trace outputs. Figure 6 depicts a simple example of a trace generated using MP.



Figure 6. Monterey Phoenix Root and Atomic Events

In Figure 6, the green boxes represent root events, constituting top-level system activities that interact with other events described in the model without possessing a parent. Conversely, the blue boxes in Figure 6 signify atomic events, containing intricate information that dissects a process into individual steps. Root events serve as parents to

composite or atomic events, and arrows delineate the sequence of structured precedence and inclusion among these components, visually illustrating their relationship.

An event trace reflects a particular implementation of the system or a specific instance. MP produces these traces following the rules set by the event grammar, and subsequently refines or sifts them according to the composition operations and schema constraints. This method strengthens the idea of an “executable architecture model” (Auguston, 2020, 5).

In MP, “scope” refers to a preset boundary that dictates how many valid event traces MP is directed to generate from a given schema. Typically, the scope constrains the number of iterations in grammar rules, with a modest limit of three iterations generally meeting most practical needs. Creating and scrutinizing event traces for the schema shares commonalities with the conventional software testing process. With the capability to produce all event traces within a given range for an MP model, comprehensive testing becomes feasible—a feature seldom found in conventional software testing (Auguston, 2020, 8). The thorough creation of event traces within a set boundary can establish a comprehensive benchmark for traditional “black box” testing methodologies employed in software systems, presenting a level of abstraction that is not only higher but also preserves significant interpretation.

Based on the Small Scope Hypothesis (Jackson 2006), most errors can be suitably depicted through relatively modest examples. A thorough examination of the produced event traces may assist in identifying unexpected behavior patterns within cyber operations. The traceability of the MP model streamlines the evaluation of a specific behavioral occurrence, such as an individual event trace, as opposed to a generalized representation of all behavioral tendencies.

Figure 7 presents Auguston’s documentation on the System and Software Behavior Modeling Language, emphasizing that schemas portray models of a system. These illustrations capture interactions between components and any associated actions, whether internal or external, to the system. Furthermore, event grammar provides insight into the structure of potential event sequences. In formulating a grammar rule, it is essential to

define an event type concerning its IN and PRECEDES relations (Auguston, 2020). Writing “A: pattern_list;” where A represents the name of an event type, and pattern_list is composed of individual events to specify the construction.

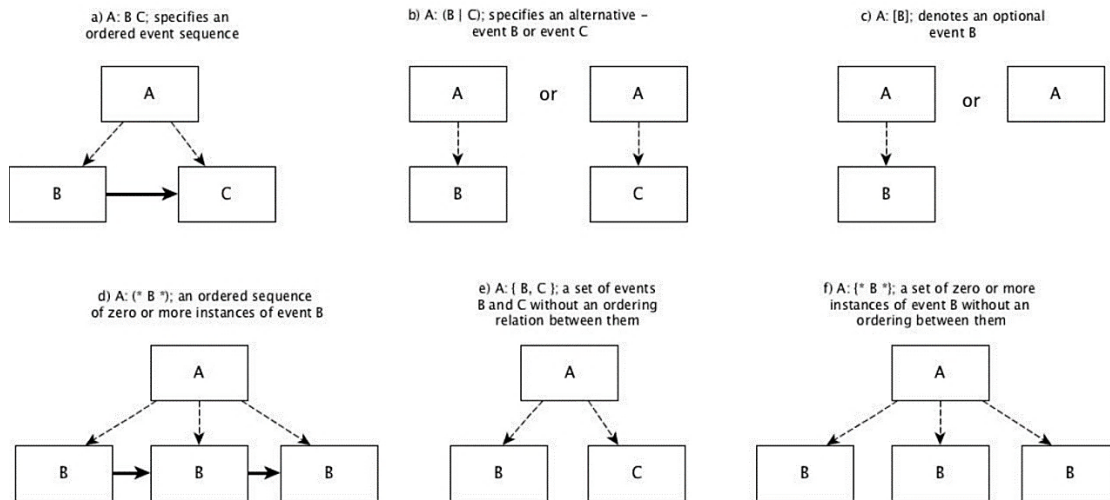


Figure 7. Event Grammar Rules and Examples of Event Trace Instances.
Source: Auguston (2020)

In the upper left of Figure 7, the grammar rule “A: B C” specifies that events falling under type A contain ordered occurrences of events B and C, respectively, expressed through the PRECEDES relation (Auguston 2020). Moreover, this rule requires that relations B IN A, C IN A, and B PRECEDES C all hold to be considered valid. Figure 8 represents this relationship using boxes representing individual event occurrences and dashed arrows indicating they are IN relations. Solid lines indicate their respective PRECEDES relations.

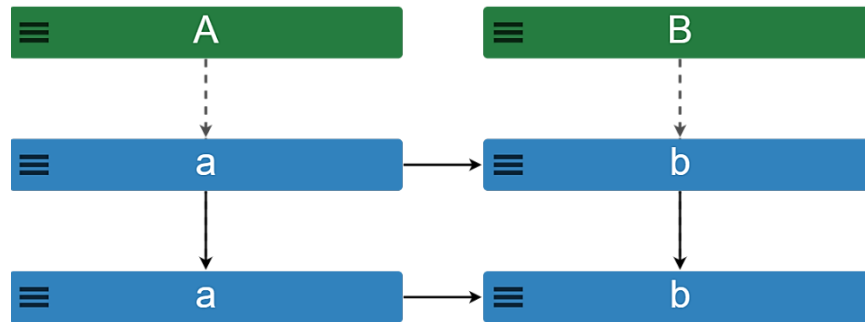


Figure 8. IN and PRECEDES

The forcing function for precedence between and among events in different roots is provided by coordinate statements. To illustrate this, Figure 9 presents the code of a coordinate statement. The syntax of each coordinate statement is the same: it first announces a specific atomic event from its parent root as “\$x” then declares the other atomic event that relates precedence from its parent root as “\$y” and finally, a “DO” statement loops over instances of \$x and \$y in each trace and pairs them as \$x PRECEDES \$y.

```
COORDINATE  $x: a FROM A,
             $y: b FROM B
             DO ADD $x PRECEDES $y; OD;
```

Figure 9. Coordinating Statements

Executed at a scope of 2, the code displayed on the left-hand side of Figure 10 generated the event traces observed on the right. These visual aids are pivotal in enhancing our understanding of system behavior. They articulate discrete events through MP’s proprietary syntax and illuminate intricate interactions. The connection between the specified scope and the ensuing event traces highlights the complexity and interconnectedness of these systems, providing a thorough view of the interaction between the code and its resulting output.

The SCHEMA outlines the overarching behavior and interaction. Two ROOTs, A and B, are defined as actors in this model, each with processes coded as atomic events. In A's and B's cases, the order of these atomic events sets a built-in precedence graphed using arrows. Each actor also has their respective atomic event, a and b, embraced in a sequence of zero or more instances of said event.

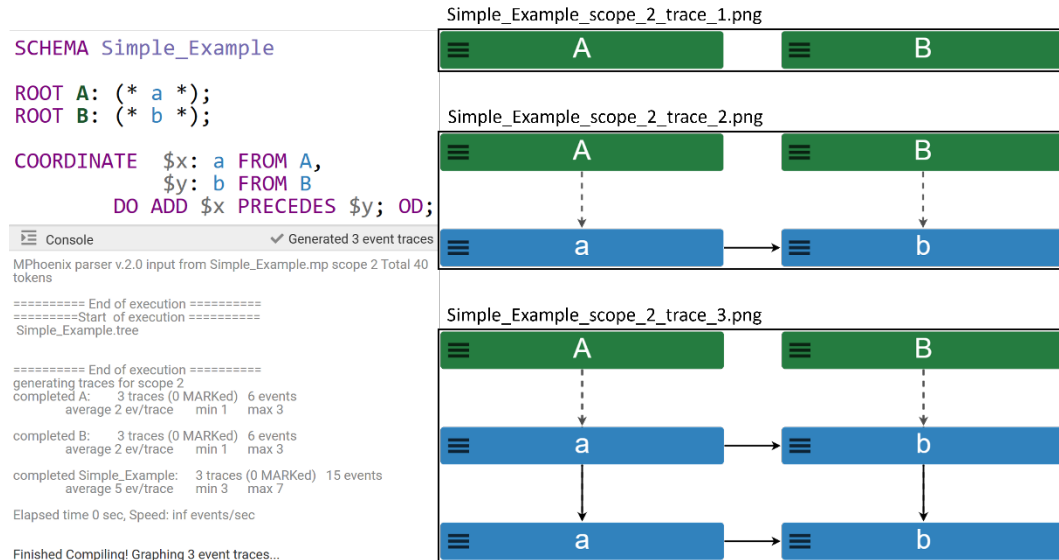


Figure 10. Code, Console, and Traces

C. RESEARCH DESIGN AND APPROACH

This thesis employs the EBA Methodology as detailed by Giammarco (2022), adopting a multi-stage modeling and simulation research approach to investigate emergent behaviors within the cyber operational environment by leveraging MP throughout each stage. In the initial stage, the methodology incorporates MP to model the Systems-of-Interest (SoI). The second stage utilizes MP to generate event traces, rigorously examining each trace to identify fallacies, such as incorrect concepts or biased assumptions made during the primary stage. The third stage classifies emergent behaviors, and the fourth stage engages in iterative investigation cycles to deepen the understanding of emergent behavior, encouraging continuous refinement of interpretations to uncover novel insights. Finally, the fifth stage visually represents the SoI, aligning with the latter portion of EBA

Methodology's Step Four. This visualization promotes a comprehensive understanding of the SoI within the cyber operational environment, reflecting the purpose of EBA Methodology's Step Four to articulate findings within the context of an engineered system. By integrating key aspects of the EBA Methodology's steps, this research aims to build upon prior research to enhance situational awareness of the cyber operational environment.

1. Stage One – Modeling System-of-Interest

This research seeks to gauge how effectively MP can probe the cyber operational environment and reveal any behavioral patterns or sequences that previous studies may have overlooked. The methodology adheres to a systems approach, using Adcock (2023, 169) description of a SoI, an open, interactive subsystem that reacts and engages with other systems within its environment.

This study characterizes the cyber operational environment as the Wider Systems-of-Interest (WSOI). This WSOI is composed of various SoIs working in harmony. The WSOI establishes a logical boundary that encapsulates all necessary components for comprehending the cyber operational environment. It is important to highlight that this study acknowledges the WSOI as not operating in isolation. Rather, it functions in an expansive context encompassing natural, social, and engineered environments (Adcock 2023, 222). The study aims to enhance commanders' situational awareness of the operational environment (i.e., the WSOI).

The research draws on the principles of systems thinking. It applies the concept of separation of concerns (Adcock 2023, 169), which aims to balance examining parts of the SoI and maintaining an overview of the system (WSOI). It will employ abstraction to focus on specific areas of concern while ensuring that the complete system remains under consideration. The views—internal, external, static, and dynamic—provide insight into the structure, behavior, and interactions within the WSOI. This integrative view is critical for enhancing situational awareness of the cyber operational environment.

2. Stage Two – Event Trace Generation and Inspections

During the second stage, MP utilizes the models created in the first stage to produce Event Traces. The research thoroughly scrutinizes each Event Trace to identify any fallacies, including incorrect concepts or biased assumptions made during stage one.

The research design acknowledges the role of human cognition in this stage, aiming to gain a deeper understanding of the SoI's behavior. The reliance is not solely on the software tool (MP) for data analysis; instead, there is substantial involvement of human judgment and interpretation. This approach emphasizes the importance of the researcher's role in extracting meaningful insights from the data.

Underpinning the stage's design is a critical approach that prompts constant questioning of underlying assumptions about the cyber operational environment. Such an approach characterizes the research as reflective, requiring the researcher to consistently remain aware of assumptions and biases and challenge them throughout the process. Lastly, this stage aims to uncover behavior patterns that had previously gone unnoticed. To support this discovery, an output of this stage is a series of open-ended questions to encourage open-minded thinking.

3. Stage Three – Strength and Polarity Classifications

In the third stage, the research employs a narrative classification technique to organize behaviors observed in the MP Event Traces. This particular phase of the research leans towards an interpretative research design. While the second stage focuses on understanding the context and significance of the behaviors observed within the event traces, this stage classifies the behaviors into strong or weak and positive or negative. This classification process serves an evaluative function, providing a means to assess the potential impacts of behaviors.

The researcher acknowledges that the classification of emergent behavior is subjective and depends on the context, perspectives, and stakeholders involved. Different interpretations may arise based on the roles, priorities, and knowledge of the system. The classification serves as a framework for categorizing emergent behaviors, but the specific behaviors and their classification may vary depending on the system and the analysts

involved. Table 4 summarizes the behavior classification, where emergence is classified based on its strength and polarity.

Table 4. Classification of Behavior. Adapted from Giammarco (2022).

Classification of Behavior		
	Classification	Description
Strength of Emergence	Weak Emergence	The research expects an emergence corresponding to the system's structure in the initial two stages. This emergence stems from the combination of the system's element properties and their interrelationships. Being a predictable behavior, this weak emergence is considered beneficial and desirable.
	Strong Emergence	Strong Emergence, usually noticeable beyond the second stage, refers to phenomena unexpectedly appearing during the initial stages or when a system encounters unexpected circumstances. Although these phenomena occur early on, their impact and significance do not tend to become evident until Stage Three.
Polarity of Emergent Behavior	Positive Emergence	Behaviors that meet the objectives of the System of Systems (SoS) and enhance the performance of its component systems within their desired parameters are considered beneficial. Being behaviors that are desirable and acceptable, and being behaviors that align with the system's goals, contribute positively to system functionality.
	Negative Emergence	Behaviors that do not align with the objectives of the System of Systems (SoS) are deemed undesirable as they manifest undesired or unacceptable actions. These deviations from the intended system behavior can lead to failures, disruptions, or adverse effects. As such, they are viewed as unfavorable and require appropriate measures for mitigation or resolution.

4. Stage Four – Repetitive Investigation Cycles for Identifying Emergent Behaviors.

Stage Four of the research actively engages in refining the understanding of the SoI by repeatedly cycling through scenarios depicted as event traces. This method aligns well

with Constructivism, an approach to learning to assert that individuals actively construct knowledge and that the learner's experiences determine reality (Elliott, Kratochwill, and Littlefield 2000, 256).

Stage Four focuses on continuously exploring the SoI behavior. This phase is crucial for fostering a nuanced comprehension of emergent behaviors through iterative cycles of scenario examination. The research reconstructed its understanding of the SoI's capabilities and limitations by revisiting the MP Event Traces to reanalyze every scenario.

Furthermore, the MP model against a reference model is a critical component of Stage Four. Drawing inspiration from Choi and Kang's formal modeling phase, the research constructed a model using an alternate modeling tool (2013, chap 2). This process facilitated a systematic analysis of the MP model compared to a reference, allowing for the clarification and examination of complex behaviors (2013, chap. 2). Appendix B offers details concerning the alternative modeling tool employed.

Stage Four constitutes a critical phase in the research, marked by an active and iterative examination of the SoI. This repeated cycle of investigation enhances the understanding of the SoI and, through further examination, clarifies complex behaviors.

5. Stage Five – Simplification and Visual Representation.

In the fifth research stage, the study implements the modeled SoI in an engineered system context. The objective of this stage is to give commanders a comprehensive and simple summary of the SoI within the WSoI. Importantly, the research presents a comprehensive overview of the emergent behaviors that were classified in Stage Three.

The research endeavors to map the multifaceted cyber environment through the lens of Systems Engineering. Within this context, the SoI corresponds to categories and subcategories in the five functions defined by the NIST Cybersecurity Framework (CSF): Identity, Protect, Detect, Respond, and Recover. These functions provide a standardized understanding of the cyber environment. Furthermore, the framework facilitates the transformation of the cyber environment into coherent, manageable formats, including components, subsystems, systems, and systems-of-systems. Complying with the core

principle of the systems approach—considering the complete engineered system context rather than limiting it to a singular system (Forsberg et al. 2015)—this research furnishes commanders with an all-encompassing, multifaceted, and adaptable understanding of the cyber operational environment.

D. LIMITATIONS

This analysis, especially in its application of methodologies, presents several inherent limitations that must be acknowledged. Firstly, the complex nature of cyberspace operations may sometimes correspond differently with the research approach, posing challenges to the study’s applicability.

Secondly, certain assumptions underlying the research may impact its applicability. The assumption that identifying emergent behaviors would improve situational awareness may have influenced the applicability of its findings. Similarly, the assumption that current C-OPE activities cannot detect emergent behaviors may also affect the research’s relevance.

Third, translating theoretical modeling into tangible, actionable strategies frequently presents challenges, as the research explicitly indicates. The perpetual evolution of cyber threats contributes to this complexity, requiring adaptable and resilient tools. The methodologies this study introduces may fall short of addressing these specific needs.

Lastly, implementing emergent behavior analysis requires comprehensive understanding and expertise. Variations in a model may lead to diverse interpretations or insights, adding another layer of complexity to the application of the research.

E. DATA COLLECTION

The Johns Hopkins Applied Physics Laboratory (APL) offers several example playbooks for analysis through its Information Assurance and Computer Defense (IACD) program. These playbooks facilitate the description of the cyber operational environment. IACD categorizes each playbook according to the NIST Cybersecurity Framework’s Five Functions and delineates the relationship each maintains with governance or regulatory requirements. The comprehensive design of these playbooks rendered them suitable data

sources for the research's SoI, necessitating an in-depth characterization of the cyber operational environment. IACD furnishes 14 example playbooks across all five functions of the NIST Cybersecurity Framework, accessible at www.iacdautomate.org/playbook-and-workflow-examples. Moreover, IACD emphasizes real-world application, ensuring that the playbooks capture the managed, repeatable cybersecurity and network operations processes (IACD 2020). This focus on practicality enhances the reliability and relevance of the research.

The “Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI” outlines a comprehensive framework of procedures and techniques for identifying, analyzing, and investigating Cyber Threat Information (IACD 2018). The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) defined Cyber Threat Information (CTI) in Special Publication (SP) 800–150 as “any information that can help an organization identify, assess, monitor, and respond to cyber threats” (Johnson, Badger, and Waltermire 2016, ii). While sharing cyber threat information can enhance an organization's security posture and even improve the postures of other organizations (Johnson, Badger, and Waltermire 2016, ii), challenges exist in evaluating the quality of the information received (Johnson, Badger, and Waltermire 2016, 5). Before acting on threat information, an organization must ensure its accuracy and assess the relevance of the threat (Johnson, Badger, and Waltermire 2016, 5). As a vital layer in the cyber operational environment, the playbook aims to assist organizations and Information Brokers in processing new CTI submissions. The decision to designate this cybersecurity operations process as the research's SoI reflects the valuable insights and potential benefits it may offer USCYBERCOM's planners, analysts, and decision-makers.

IACD advocates the creation of playbooks as diagrams illustrating the process steps—a practice evident in their provided examples (IACD 2020). Furnished by IACD, these visual aids assist participants in effectively implementing the associated strategy and framework. The pictorial presentation of the playbooks delivers a comprehensive overview, enhancing comprehension of their functionality. As an integral component of

the IACD strategy and framework, Figure 11 showcases the specific IACD playbook that the research designated as its SoI.

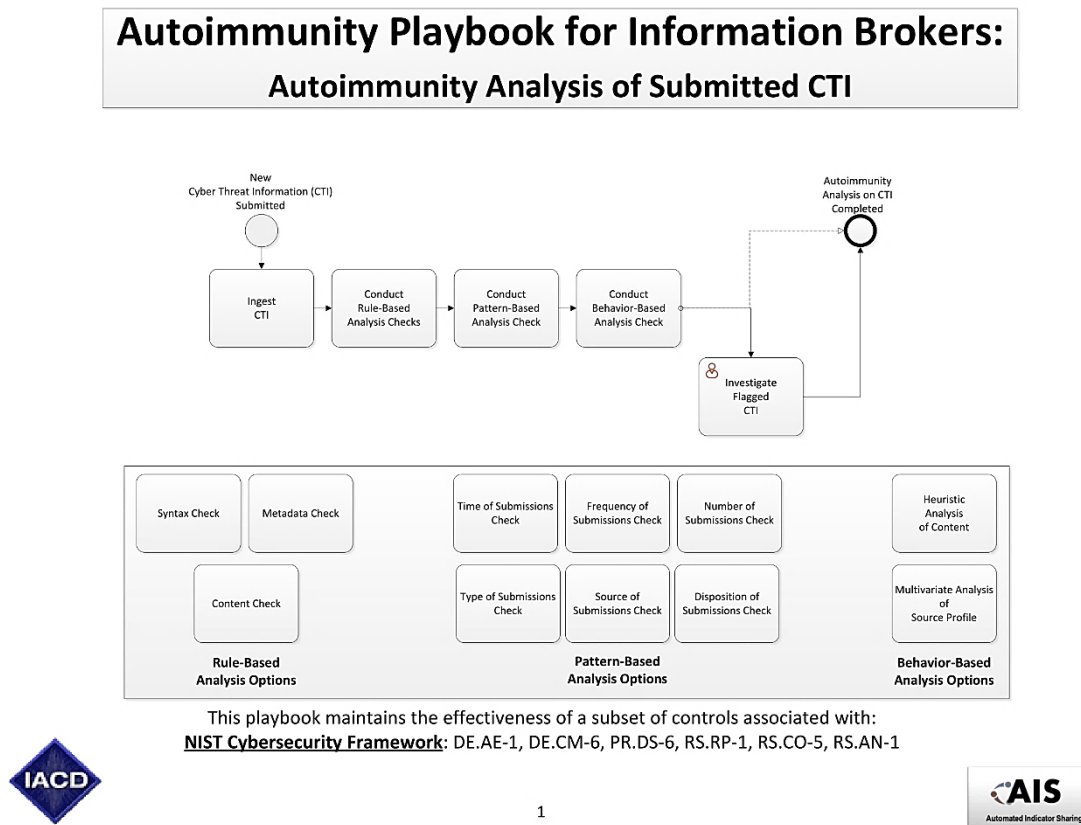


Figure 11. Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI. Source: IACD (2018)

The playbook integrates complex interactions and layered analysis procedures, thereby facilitating a comprehensive exploration of behaviors between SoIs. Modeling these interactions in MP may permit commanders to acquire new insights, thereby enhancing situational awareness in the cyber operational environment. As a system of systems, the playbook is consistent with the research methodology. Moreover, it aligns with governance and regulatory requirements, standardizing the context for analysis.

As a SoI, the “Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI” fosters academic depth in the cybersecurity operations process. The deliberate integration of the playbook into the research’s design offers significant potential to enhance commanders’ situational awareness within the cyber operational environment.

This research design blends a robust combination of qualitative, interpretative, constructivist, critical, and exploratory methodologies. The study conducts a comprehensive investigation into the cyber operational environment using MP, complemented by iterative scrutiny, integration of human cognition, and continual assessment of underlying assumptions. The objective is to uncover novel insights to enhance commanders’ situational awareness of the cyber operational environment.

IV. RESULTS AND DISCUSSION

The chapter presents and interprets the research's complex findings across five stages. Each stage is crucial in exploring the SoI within the cyber operational environment. The chapter begins with Stage One, where the research models and characterizes the SoI, and concludes with Stage Five, focusing on visually representing and simplifying the explored SoI. The following sections detail each stage's methods, results, and implications, deepening understanding of the SoI to enhance situational awareness of the operational environment.

A. STAGE ONE

Stage One of the research employs MP to model the SoI with the cyber operational environment depicted as the Wider System-of-Interest (WSOI). The stage commences with a description of the SoI in a narrative format. This choice of format aims to distill the complexity of the SoI, provide contextual insight, and ensure logical coherence (Bell-Watson 2021). By aligning with human cognition and increasing accessibility, a narrative format approach facilitates a deeper understanding of emergent behaviors within the system.

1. IACD Playbook

The SoI for this study is the IACD playbook, titled "Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted Cyber Threat Information." This playbook constitutes a comprehensive system (i.e., a process) designed to analyze new Cyber Threat Information (CTI), focusing on identifying and detecting malicious CTI submitted to an information broker. The multifaceted SoI provides a detailed, standardized, and adaptable methodology for examining CTIs. Integrating various components forms an inclusive system that enhances cybersecurity and situational awareness within information brokering.

Understanding the entirety of the SoI is crucial in the analysis process. This understanding requires capturing the complexities and nuances of the IACD playbook,

which the analysis reconstructs in MP. Representing the IACD playbook in a narrative format allows the analysis to capture these intricate details. The description in the narrative format is as follows:

Upon receiving a new CTI, the SoI initiates the analysis process according to a well-defined sequence. The process commences with ingesting the CTI and conducting three distinct types of analyses: rule-based, pattern-based, and behavior-based. These analyses constitute the foundation of the process. If a CTI is determined to be uncertain or exhibits suspicious characteristics, it gets flagged. This flagging is a result of its alignment or mismatch with predefined rules (rule-based analysis), its patterns that may suggest malicious activities (pattern-based analysis), or potential behaviors resembling known malicious actions (behavior-based analysis). When flagged, the procedure demands a more detailed examination, triggering an in-depth analysis and designating the task for human intervention to ensure the nuances and complexities are effectively addressed. After the analysis of a specific CTI, the particular execution concludes. The main audience for the SoI includes information brokers accountable for collecting, analyzing, and disseminating CTIs. Furthermore, the SoI aligns with the standards and regulations outlined in the NIST Cybersecurity Framework.

2. MP Model

Central to exploring the SoI is the language tailored specifically for investigating system and process behaviors. MP facilitates detailed examination of the SoI through customized syntax and relationship semantics. This innovative approach yields a nuanced understanding of subsystem interactions, contributing to a more comprehensive understanding of the SoI and unveiling previously hidden patterns.

Figure 12 presents the SoI in MP's language. The flexibility of MP, which allows for the modeling of states for an event, stands out as one of its key strengths. This is evident in line 4, where the *CTI_report* manifests as a composite event, either *CTI_is_self_referential* or *CTI_is_NOT_self_referential*. The CTI report, represented by the root event *CTI_Source* in line 7, initiates the playbook. Coordinating this root event

with the *Security_System* root event in line 23 ensures that the submission of a CTI always precedes the security system's ingestion of it.

```

1 SCHEMA Advanced_Autoimmunity_Analysis_Playbook
2
3 /* Composite event that represents a CTI report. This event can either be self-referential or not. */
4 CTI_report: ( CTI_is_self_referential | CTI_is_NOT_self_referential ) ;
5
6 /* Root named CTI_Source. This root starts with the initiating event. */
7 ROOT CTI_Source: {+ New_CTI_Submitted CTI_report +} ;
8
9 /* Composite event that represents the process of conducting analysis checks.
10  If CTI is not flagged then the playbook has met the end state. */
11 Conduct_Analysis_Checks: Conduct_Rule_Based_Analysis_Checks
12                        Conduct_Pattern_Based_Analysis_Check
13                        Conduct_Behavior_Based_Analysis_Check
14                        ( CTI_is_NOT_flagged Autoimmunity_Analysis_on_CTI_Completed | CTI_is_flagged ) ;
15
16 /* Root named Security_System. This root can include any number of
17  Ingest_CTI and Conduct_Analysis_Checks events. */
18 ROOT Security_System: {* ( Ingest_CTI Conduct_Analysis_Checks ) *};
19
20 /* Coordinate the New_CTI_Submitted event from CTI_Source with the
21  Ingest_CTI event from Security_System, such that New_CTI_Submitted
22  always precedes Ingest_CTI. */
23 COORDINATE $x: New_CTI_Submitted FROM CTI_Source,
24            $y: Ingest_CTI FROM Security_System
25            DO ADD $x PRECEDES $y; OD;
26
27 /* Composite event that represents the process of investigating a flagged CTI.
28  The completion of a investigating a flagged CTI meets the end state. */
29 Investigate_Flagged_CTI: Analysis_Options
30                        ( confirms | refute )
31                        Autoimmunity_Analysis_on_CTI_Completed ;
32
33 /* Root named Information_Broker. This root can include any number of
34  CTI_is_flagged and Investigate_Flagged_CTI events. */
35 ROOT Information_Broker: {* ( CTI_is_flagged Investigate_Flagged_CTI ) *};
36
37 /* Make the CTI_is_flagged event shared between Security_System and Information_Broker.
38  Investigating a flagged CTI will only occur if CTI is flagged. */
39 Security_System, Information_Broker SHARE ALL CTI_is_flagged;
40
41
42 /* Define a root named Playbook. This root represents a sequence of
43  steps involved in handling a CTI. */
44 ROOT Playbook: ({* New_CTI_Submitted
45                I_Initiating_Condition
46                II_Process_Steps
47                III_Best_Practices_and_Local_Policies
48                IV_End_State
49                V_Relationship_to_Governance_or_Regulatory_Requirements
50                *});
51
52 /* Make the New_CTI_Submitted event shared between Playbook and CTI_Source.
53  This adds all five playbook components to the trace for each initiating condition */
54 Playbook, CTI_Source SHARE ALL New_CTI_Submitted;

```

Figure 12. SCHEMA

As shown in Figure 12, the *Security_System* root event performs various analyses on the CTI report through the composite event *Conduct_Analysis_Checks* on line 11, including rule-based, pattern-based, and behavior-based checks. The process is complete if the system does not flag the CTI report. However, the event *CTI_is_flagged* on line 14 prompts further actions by the root event *Information_Broker* on line 35. Handling of conditional events and the depiction of different pathways through the system showcase MP's flexibility.

The *Information_Broker* examines the flagged CTI report, as represented by the *Investigate_Flagged_CTI* composite event on line 29. Upon completion of the investigation, the process concludes.

Figure 13 displays the result of the *Security_System* and *Information_Broker* sharing the *CTI_is_flagged* event. This shared event ensures that an investigation of a flagged CTI only proceeds when the CTI is indeed flagged. Event sharing, a crucial feature of MP, is evident in this model. Such a representation emphasizes the precision of MP in modeling dependencies and coordinating between distinct system components.

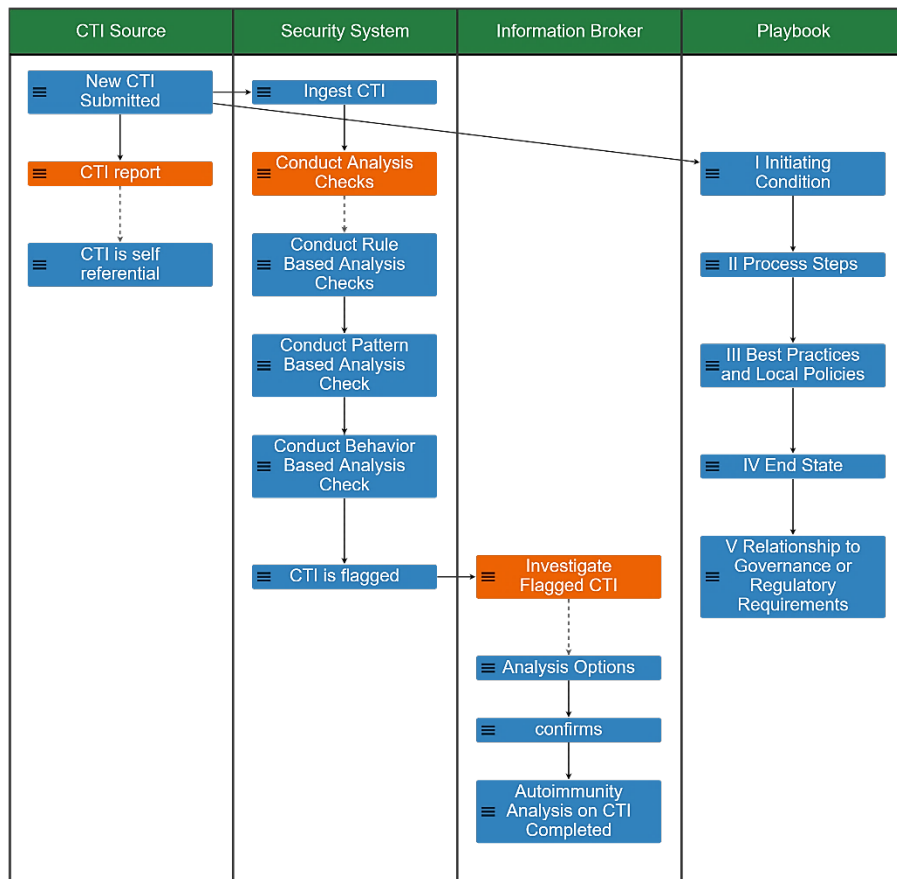


Figure 13. Event Trace Scope-2 Trace-2

Shown in Figure 12 on line 44, the Playbook root event symbolizes the IACD's playbook, as reconstructed in MP. The *Playbook* and *CTI_Source* root events share the

New_CTI_Submitted event. As depicted in Figure 14, this arrangement ensures that the system traces a new instance of the *Playbook* for each initiating condition. In this specific scenario, two initiating conditions, each represented by *New_CTI_Submitted*, result in two distinct executions of the *Playbook*. These two instances of the *Playbook* are observed on the right side of Figure 14.

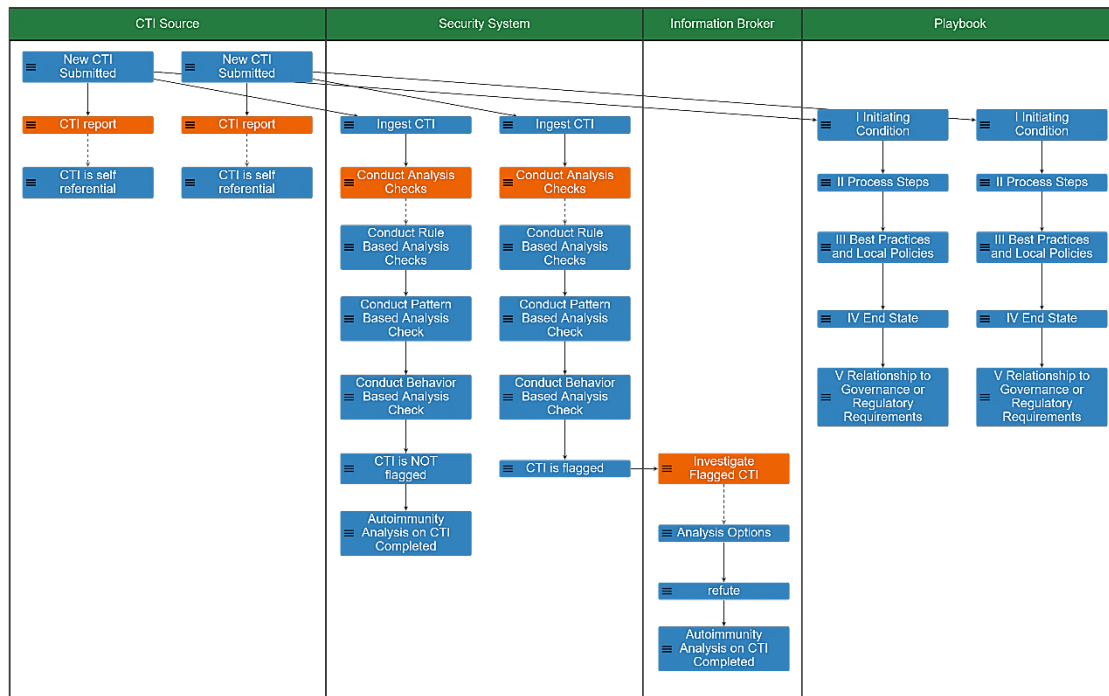


Figure 14. Event Trace Scope-2 Trace-9

3. Key Events in IACD Playbook and MP Model

According to Auguston (2020, 5), simple models are often more comprehensible to human observers. The visual inspection of event traces played a crucial role in comparing the MP model with the IACD playbook. Figure 15 illustrates this comparison and confirms the precision and effectiveness of the MP model to the IACD playbook. The figure also demonstrates that each key event is present and maintains the same order of precedence.

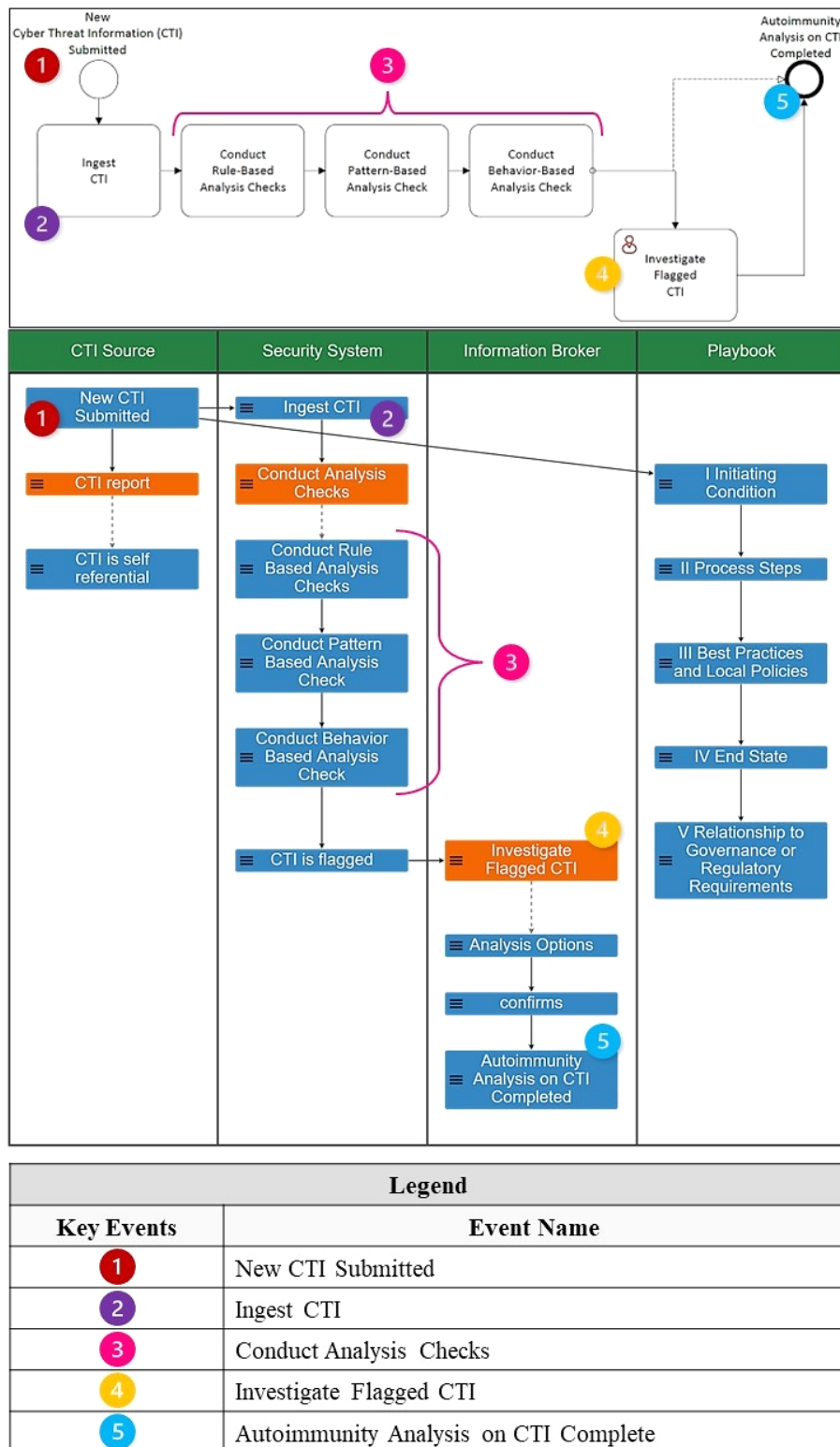


Figure 15. Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI. Adapted from IACD (2018).

Submitting a CTI in the IACD playbook involves several key events, each of which plays a critical role. The process begins with the “New CTI Submitted” event, which is essential for assessing and responding to threats. Next, the “Ingest CTI” event signifies the system’s acknowledgment and initiation of processing the submitted CTI, a step necessary for further analysis. The “Conduct Analysis Checks” event follows, performing multiple analysis checks—rule-based, pattern-based, and behavior-based—to determine the CTI’s nature and severity. If the checks flag the CTI, the “Investigate Flagged CTI” event initiates a deeper investigation into that CTI to address nuances and complexities related to the CTI effectively. Lastly, the “Autoimmunity Analysis on CTI Completed” event marks the end of the CTI analysis process, indicating that thorough analysis has taken place, thus fulfilling the desired result of the playbook. Together, these events form the fundamental steps in a comprehensive process that effectively identifies, analyzes and addresses the submission of a new CTI.

B. STAGE TWO

Stage Two of the analysis focused on MP event trace generation and inspections. Guided by human and automated analysis, the stage involves carefully examining event traces to understand the SoI’s behavior. In addition, the stage formulates open-ended questions to investigate the SoI’s behavior further.

1. Event Trace Analysis

The MP event traces serve as the system’s behavioral roadmap; each “event trace represents an instance of a particular execution of the system” (Auguston 2020, 7). These event traces offer detailed insights into the system’s interactions and potential states. Within the SoI, the event traces present a comprehensive perspective of the CTI analysis process and its complex dynamics.

Figure 16 illustrates one of the 42 traces generated for the SoI. The MP model, functioning as an executable architecture model, successfully acquired all potential scenarios the system might encounter when the grammar rules were limited (scope) to two iterations (Auguston 2020, 7). Within the context of the SoI, a scope of two sufficed to obtain comprehensive results, reflecting both straightforward, linear processes—where the

system does not flag CTIs—and more complex scenarios where flagged CTIs lead the Information Broker to initiate further investigations. Some traces, such as the one shown in Figure 16, represent these more intricate scenarios. Appendix C contains all 42 event traces from the MP model when executed with a scope of two iterations.

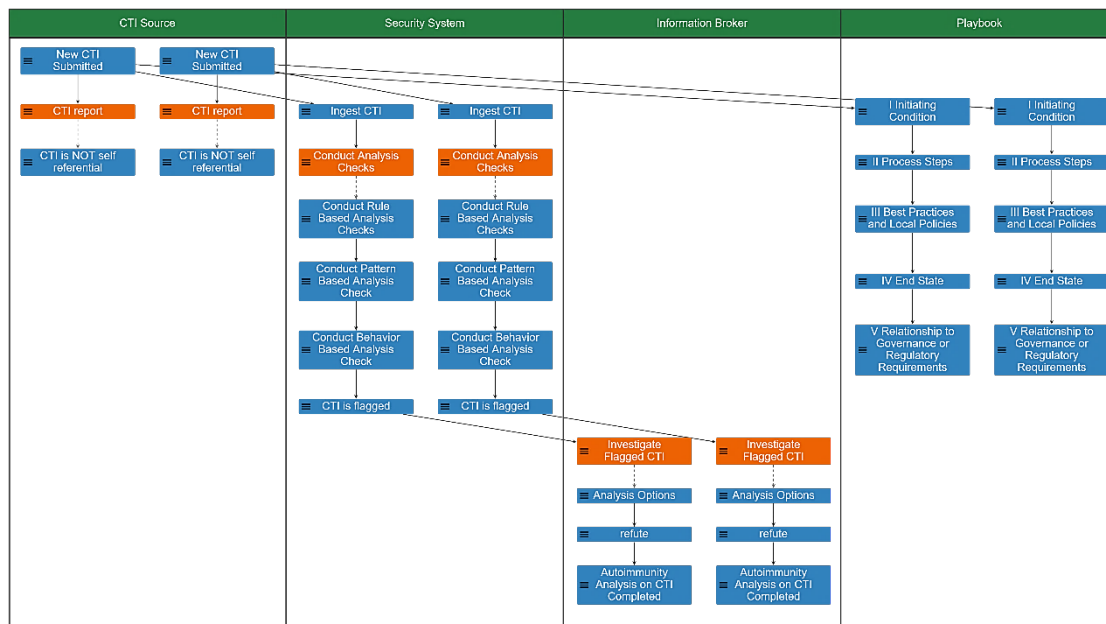


Figure 16. Event Trace Scope-2 Trace-42

A thorough analysis of all 42 event traces resulted in several findings:

1. Identification of Key Interactions: The strong connections between the *CTI_Source*, *Security_System*, and *Information_Broker* events became evident. For example, the shared *CTI_is_flagged* events between the *Security_System* and *Information_Broker* underscores the Sol's collaborative nature, ensuring a thorough investigation of every flagged CTI.
2. Emphasis on Conditional Behaviors: The traces clarified conditions that trigger specific events, such as launching an investigation when flagging a CTI, highlighting the system's adaptability.

3. **Discovery of Potential Vulnerabilities:** Some traces hinted at possible system vulnerabilities or bottlenecks. For example, where the Information Broker's investigation took precedence, this could potentially lead to delays or inefficiencies.
4. **Foundation for Scenario Testing:** Researchers can quickly simulate all 42 scenarios to easily see the event traces and test the system's reactions. This process helps confirm the model's behavior aligns with real-world expectations and allows for adjustments if any discrepancies are identified.
5. **Towards System Assurance:** The event traces of the SoI shed light on both the intended system behavior and the accuracy of its modeling within the MP framework. If inconsistencies appear in a trace, they may indicate design issues or modeling oversights, necessitating a reassessment of the MP schema to enhance precision.

2. Human-Centered Analysis

MP facilitates the modeling of intricate systems, but understanding the contexts of the produced event traces and comprehending the wider implications of the system's behaviors require human analysis. Behaviors with potential substantial ramifications may only reveal themselves in nuanced interactions. Acknowledging that MP functions as an extension of human analysis contributes to a more profound understanding of the system and its environment.

As the analysis investigated the SoI further, the researcher, following the guidance of the EBA Methodology (Giammarco 2022), formulated open-ended questions based on specific challenges and considerations relevant to the cyber operational environment. These questions, crafted by the researcher, are designed to challenge assumptions and stimulate innovative thinking, ensuring alignment with the research objectives:

1. "How would the SoI handle a rapid surge in CTI report submissions?"
2. "Where might bottlenecks occur in the SoI's analysis checks?"

3. “Under which circumstances could the Information Broker’s investigation contradict the security system’s initial CTI flagging?”
4. “How does the SoI coordinate timely interactions among the *CTI_Source*, *Security_System*, and *Information_Broker*?”
5. “What external factors or threats might jeopardize the SoI?”

These questions pinpointed areas for further investigation, including handling rapid surges in CTI submissions, identifying potential bottlenecks in the SoI’s analysis checks, examining contradictions in Information Broker investigations, coordinating timely interactions, and assessing external threats to the SoI. This focus prompted the researcher to look beyond the MP event traces, encouraging the exploration of edge cases and unconventional scenarios.

C. STAGE THREE

Stage Three of the analysis involves categorizing emergent behaviors based on their strength and polarity, in accordance with the EBA Methodology (Giammarco 2022). This classification functions evaluatively, appraising the potential impacts of the behaviors identified. The stage recognizes that the specific behaviors and their categorization might differ based on the system in question and the analysts involved, underscoring the subjective nature of the classification of emergent behaviors (Giammarco 2022, 19).

Table 5 details the potential outcomes of the SoI’s event traces, capturing all possible behaviors. This comprehensive identification contributes to a deeper understanding and informs the classification effort. The classification aligns with the EBA Methodology, categorizing behaviors as either ‘weak’ (expected) or ‘strong’ (unexpected) and as ‘positive’ (desired or acceptable) or ‘negative’ (unwanted or unacceptable) (Giammarco 2022, 20).

Table 5. Behaviors in the Event Traces

Behaviors in the Event Traces		
Condition present: The CTI can either be self-referential or not.	Conduct Analysis Check by the Security System flags the CTI or does not.	Information Broker only investigates a flagged CTI. Investigation into a flagged CTI confirms or refutes the flag.
CTI is self-referential	CTI is flagged	Investigation confirms
		Investigation refutes
CTI is self-referential	CTI is not flagged	Information Broker only investigates a flagged CTI
CTI is not self-referential	CTI is flagged	Investigation confirms
		Investigation refutes
CTI is not self-referential	CTI is not flagged	Information Broker only investigates a flagged CTI

1. Classification – Strength of Emergence

Understanding emergence is crucial for detecting unexpected and novel behaviors in the SoI. Giammarco’s EBA Methodology classifies the “Strength of Emergence” as either weak or strong. The identified behaviors from Table 5 are classified in Table 6.

Weak Emergence: Scenario 1 and Scenario 4 were classified under weak emergence. These categories represent anticipated behaviors of the SoI. Both behaviors conform to the SoI’s intended purpose, illustrating the phenomenon within the expected and predefined parameters of the system.

Strong Emergence: Scenario 2 and Scenario 3 indicate strong emergence, representing unexpected system behaviors within the SoI. Such behaviors highlight potential vulnerabilities or inefficiencies in the SoI, revealing areas that may require further analysis to enhance system performance and security.

Table 6. Strength of Emergence

Strength of Emergence		
	Classification	Scenario and Scenario Description
Strength of Emergence	Weak Emergence	Scenario 1: CTI is self-referential, and CTI is flagged Scenario 4: CTI is not self-referential, and CTI is not flagged
	Strong Emergence	Scenario 2: CTI is self-referential, and CTI is not flagged Scenario 3: CTI is not self-referential, and CTI is flagged

2. Classification – Polarity of Emergent Behavior

Complex systems often exhibit emergent behaviors with properties requiring in-depth analysis. The EBA Methodology categorizes the polarity of emergent behavior as positive or negative based on its alignment with the system's goals and objectives. Positive emergent behavior reinforces the system's intended functions, whereas negative emergent behavior diverges from or undermines the system's objectives. The SoI behaviors from Table 5 are further classified in Table 7.

Positive Emergence: Scenario 1 and Scenario 4 correspond to positive emergent behaviors within the SoI. These behaviors indicate that the SoI functions correctly in specific scenarios, effectively identifying and analyzing a CTI.

Negative Emergence: Scenario 2 and Scenario 3 denote negative emergent behaviors, signifying deviations from the SoI's desired operations. Such deviations can introduce security risks and spotlight areas in the SoI needing refinement or adjustment.

Table 7. Polarity of Emergence

Polarity of Emergence		
	Classification	Description
Polarity of Emergence	Positive Emergence	Scenario 1: CTI is self-referential, and CTI is flagged Scenario 4: CTI is not self-referential, and CTI is not flagged
	Negative Emergence	Scenario 2: CTI is self-referential, and CTI is not flagged Scenario 3: CTI is not self-referential, and CTI is flagged

3. Analysis of Strength and Polarity in Classifications

In Scenarios 1 and 4, weak emergence shows that the system functions within its defined parameters, reflecting a stable design that meets the expected behavior. This stability is vital in cyber operations, ensuring the system adheres to its intended purpose. Yet, the classification of weak emergence raises concerns about the SoI adaptability and resilience to unexpected challenges. If the system functions solely within preset boundaries, it may need more flexibility to react to new or evolving threats. Despite these concerns, the alignment of Scenarios 1 and 4 with positive emergence confirms that the SoI can effectively identify and analyze a CTI.

Strong emergence in Scenarios 2 and 3 signals potential vulnerabilities and inefficiencies, warranting investigation into the SoI's ability to adapt to unforeseen stimuli. In particular, the classification of strong and negative emergence in Scenarios 2 and 3 may indicate underlying problems, such as a mismatch between the system's goals and the actual requirements of the operational environment.

Classifying emergent behaviors within the SoI provides valuable insights into areas needing more analysis, including adaptability, resilience, and alignment with the SoI's goals. Following Giammarco's EBA Methodology (2022), these classifications do more than merely describe the behaviors; they enhance situational awareness by delivering actionable information.

D. STAGE FOUR

Stage Four signifies a vital phase in the analysis by focusing on the iterative examination of the SoI. An ongoing series of investigations characterizes this stage, as the analysis actively explores and refines its understanding of the SoI's behavior. Stage Four fosters a more nuanced comprehension of emergent behaviors by repeatedly cycling through scenarios. This approach encourages the researcher to construct, examine, reconsider, and reconstruct knowledge by persistently engaging with the MP language and the MP event traces.

During Stage Four iterative examination, the SoI's capabilities and limitations became more obvious. For example, the model underscored the essential coordination

between the CTI Source and Security System. This coordination ensures that the security system quickly processes every new CTI submission. However, the analysis identified potential bottlenecks, particularly when receiving multiple CTIs or when the Information Broker's findings challenged the security system's alerts.

Engaging continuously with the MP language and examining the event traces refined an understanding of the SoI's behaviors. Each review added detail and uncovered subtle nuances in the playbook's function which fostered key questions like: How effectively does the system handle a surge in CTIs? And do specific CTI patterns consistently lead to false negatives or positives?

The iterative approach of Stage Four prompted the researcher to test various scenarios and consider several "what-if" situations. This proactive approach enhanced the analysis, improving understanding of the SoI. Examining the relationship between the Information Broker and Security System, through the event traces, yielded critical insights. It highlighted the necessity for immediate and precise communication, guaranteeing swift and accurate investigations of flagged CTIs.

During Stage Four, the MP model underwent further analysis against a reference model using an alternative modeling tool. This additional analysis within the iterative stage aimed to clarify and examine a particularly complex behavior. It highlights the comprehensive nature of the analysis, focusing on exploration and refinement.

The alternative modeling tool in Stage Four utilizes Decision Tree Classifier (Pedregosa et al. 2011). As shown in Figure 17, this strategic inclusion used the Decision Tree Classifier's visual and structural capabilities.

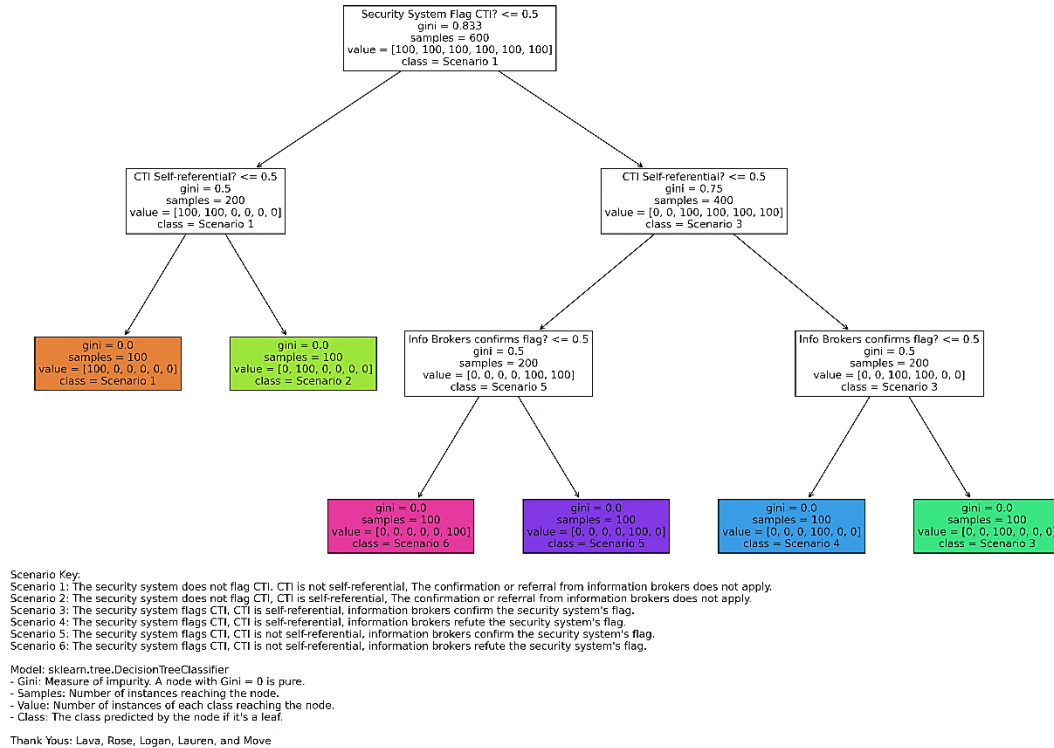


Figure 17. Decision Tree Classifier: Plot Tree

The Decision Tree Classifier segments datasets into tree-like decision structures. This method provides an intuitive means of depicting complex relationships (Pedregosa et al. 2011). Given this research's design, the decision tree's cascading structure aligned seamlessly, promoting a systematic analysis of scenarios.

As shown in Figure 17, the decision tree visualizes the MP event grammar and highlights potential points of contention. This visual representation imparted clarity and structure when aligned with the MP's event traces, simplifying the understanding of the Sol's MP event traces.

A key feature of the Decision Tree Classifier was its ability to segment data based on specific criteria quickly, thus enabling prompt identification of areas requiring attention. For instance, at the top of Figure 17, where the system's response to a surge in CTIs is analyzed, the decision tree provides a clear, hierarchical view, simplifying the analytical challenge. Appendix B contains the detailed notes on the Decision Tree Classifier.

Incorporating the Decision Tree Classifier into the research methodology fostered synergy rather than merely substituting one method for another, thereby subjecting the findings to a dual level of examination. This assessment ensured that inconsistencies between the models would prompt further investigation, reinforcing the integrity and resilience of the conclusions. This alternative tool did not diminish the role of the MP model nor replace it but rather enriched the overall analysis. The study secured in-depth and actionable insights by embedding supplementary analysis into the iterative stage while stressing a holistic approach to exploration and refinement.

E. STAGE FIVE

Stage Five constitutes a critical juncture in the analysis, refining the deliverable to impart heightened clarity for commanders. This phase utilizes methods to simplify, illustrate, and map the SoI within the broader system context (the WSoI). By visually portraying the SoI in the context of the five functions of the NIST Cybersecurity Framework (CSF), it situates the SoI within a standardized frame of reference. The NIST CSF's focus on an exhaustive and multi-faceted comprehension of the environment affords commanders a lucid and amplified perspective on the emergent behaviors and the interaction of the SoI within the WSoI. Most importantly, this stage identifies an emergent behavior and supplies a succinct synopsis of examining that behavior.

1. Mapping and Contextualization of the Cyber Environment

The International Association for Cyber Defense (IACD) program delineated the intersections between each playbook and specific controls within the NIST CSF. Figure 18 illustrates these intersections at the bottom for the SoI.

The relationship enables the research to connect its analysis to the cyber environment, empowering commanders. It also allows commanders to base their decisions on recognized cybersecurity norms. The NIST CSF subcategories that the “Autoimmunity Playbook” addresses:

DE.AE-1 (Anomalies and Events): The playbook sets a baseline for network operations, especially when a new CTI surfaces.

PR.DS-6 (Data Security): The playbook prioritizes rule-based, pattern-based, and behavior-based analysis, ensuring the integrity and security of software, firmware, and information.

RS.RP-1 (Response Planning): The playbook outlines a clear response plan for incidents, guaranteeing a unified and effective response to cybersecurity threats.

RS.CO-5 (Communications): By promoting voluntary information exchange with external stakeholders, the playbook strengthens cybersecurity situational awareness, drawing invaluable insights from various sources.

RS.AN-1 (Analysis): The playbook emphasizes analyzing alerts from detection systems, ensuring comprehensive scrutiny of threats and vulnerabilities.

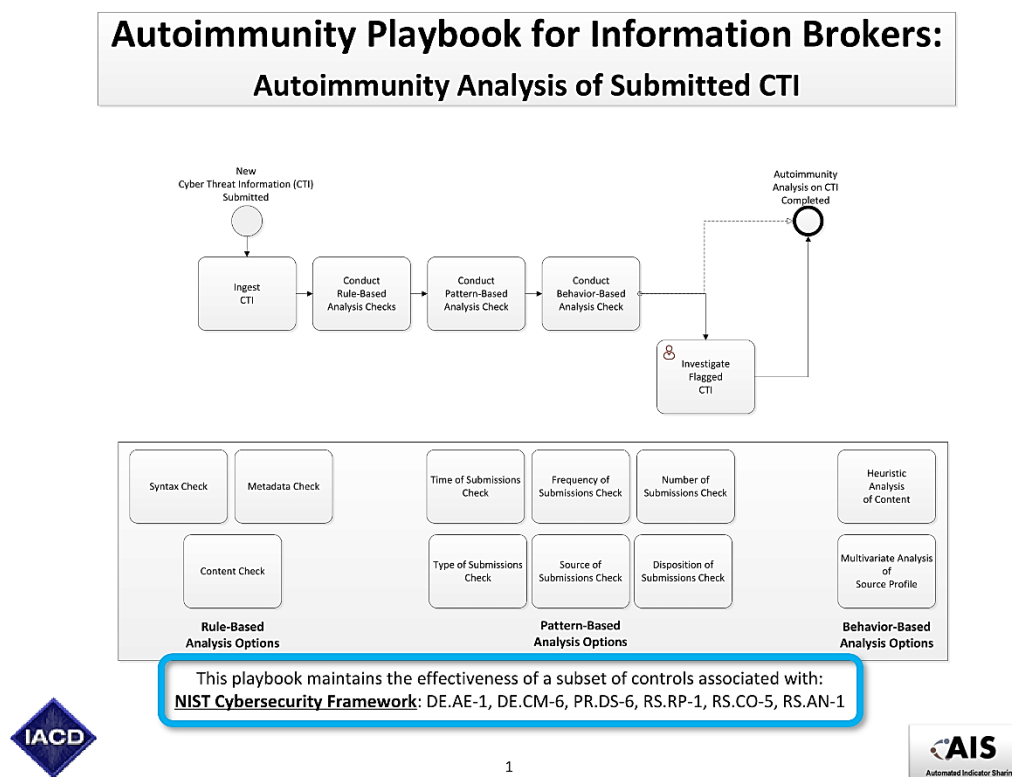
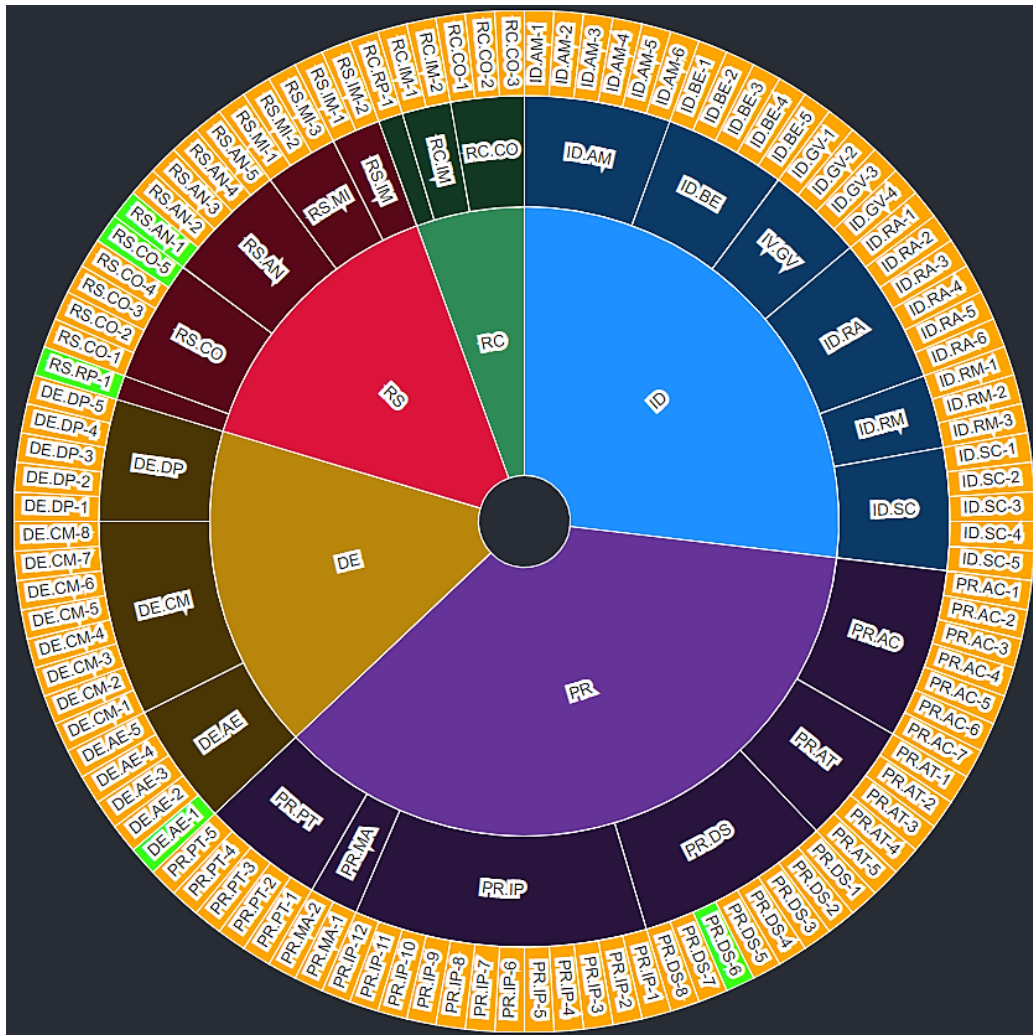


Figure 18. Relationship to Governance or Regulatory Requirements. Adapted from IACD (2018).

2. Visual Representation of the Cyber Operational Environment

Identifying the five NIST CSF subcategories allows for a comprehensive yet straightforward summary, providing commanders with a clear understanding of the SoI in the context of the WSoI.

Figure 19 presents a sunburst chart of the NIST CSF, providing a simplified view of the SoI within the WSoI. This chart offers a comprehensive overview of the NIST CSF, encompassing the Five Functions, 23 Categories, and 108 Subcategories. Specifically, it emphasizes, in bright green, the five subcategories with which the SoI aligns, thus identifying the study's key focus areas within the extensive framework. Such a visualization assists commanders in understanding the structure of the cyber operational environment.



Figures 20–22 further clarifies the NIST CSF into three principal sections, each concentrating on specific functions Detect (emphasizing the subcategory DE.AE-1 Anomalies and Events), Protect (PR.DS-6 Data Security), and Respond (RS.RP-1 Response Planning, RS.CO-5 Communications, and RS.AN-1 Analysis).

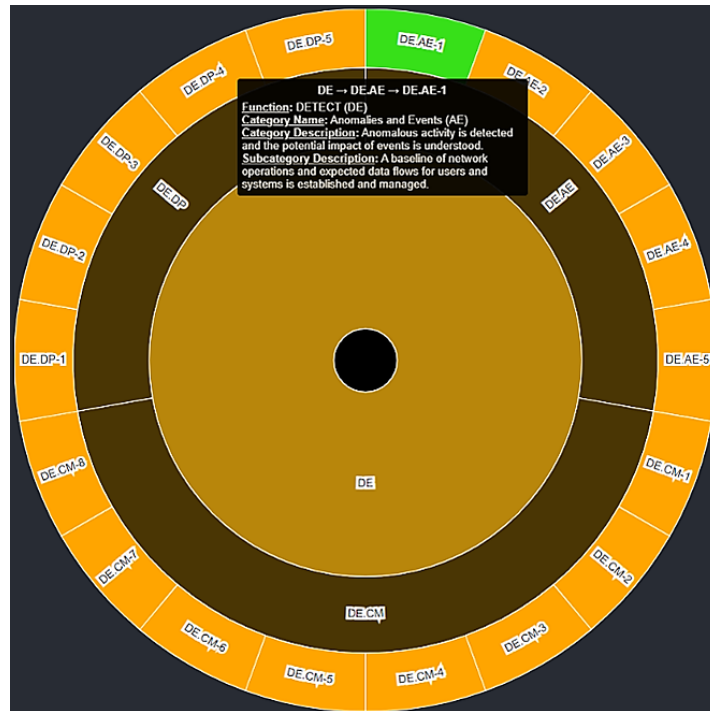


Figure 20. NIST Cybersecurity Framework Core Sunburst Chart – Detect.

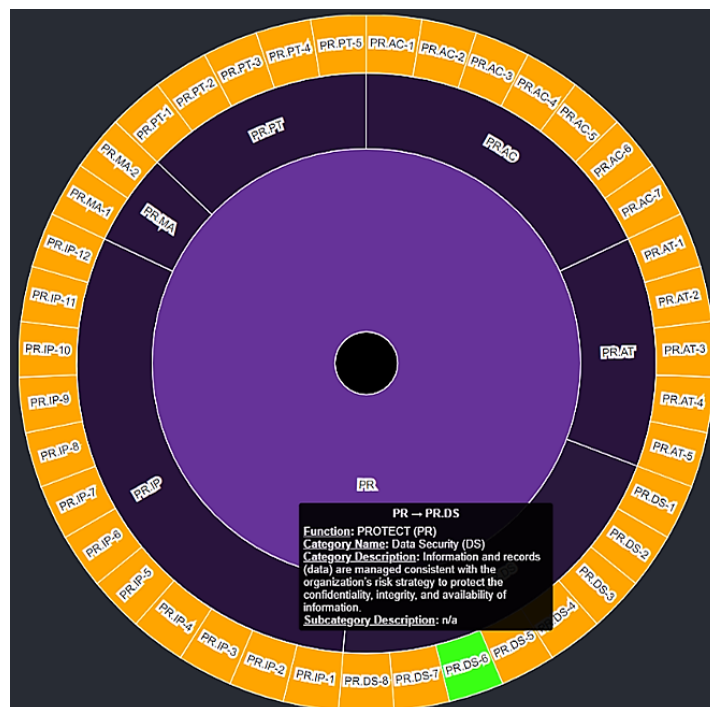


Figure 21. NIST Cybersecurity Framework Core Sunburst Chart – Protect.

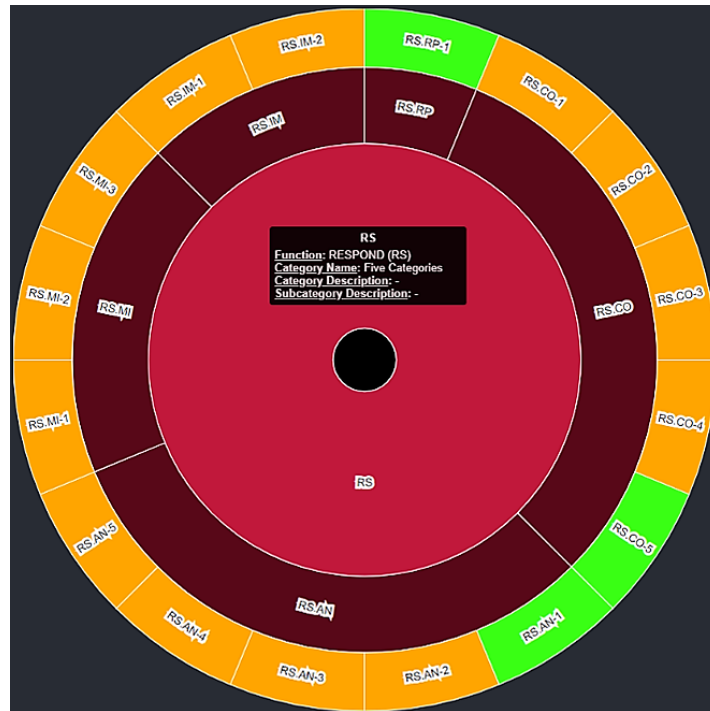


Figure 22. NIST Cybersecurity Framework Core Sunburst Chart – Respond.

Figures 19–22 serve as essential visual aids, providing nuanced insights into the WSoI within a standardized cyber operational environment. They elucidate specific alignments of the SoI within the NIST CSF, affording commanders a more accurate understanding of the cyber operational environment. This enhanced clarity supports more well-informed decision-making.

3. Analysis of an Identified Emergent Behavior

Stage Three involved classifying and recording several instances of ‘Strong Negative’ emergence. Table 8 encapsulates the researcher’s predicted behavior prior to Stage Three and the discovery made during the subsequent analysis.

Table 8. Predicted Behaviors, Discovery, and Classification

Predicted Behaviors, Discovery, and Classification		
Predicted Behaviors	Discovery	Classification
The state of a CTI is the most significant distinguishing factor in the ‘Investigate Flagged CTI’ task. The human responsibility of investigating a flagged CTI depends on particular conditions or states of the CTI, coupled with the execution of specific analysis checks.	The ‘Investigate Flagged CTI’ event is exclusively an alert-driven task, with no other factors driving the event.	Strong Negative

As illustrated in Figure 23, the IACD playbook, ‘Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI,’ employs a small symbol in the upper left corner to differentiate the “Investigate Flagged CTI” task and to indicate human involvement. The IACD program outlines the use of Human Involvement in its specification under PS-02, stating, “Process steps shall reflect the potential for administrator or analyst involvement (e.g., decision-making oversight to authorize or approve)” (IACD 2017, 7). Additionally, the IACD playbook integrates a branching path before “Investigate Flagged CTI,” and each branch has a condition that determines whether or not it will take that path. Under PS-04, the IACD program specifies using Branching Paths, stating, “Process steps shall document potential branching paths, without including decision logic” (IACD 2017, 7). If an “Analysis Checks” flags a CTI for investigation, the branch meets the condition, and the task sequence includes “Investigate Flagged CTI.” Conversely, if “Analysis Checks” do not flag a CTI, the alternate condition applies, and the task sequence does not require “Investigate Flagged CTI.”

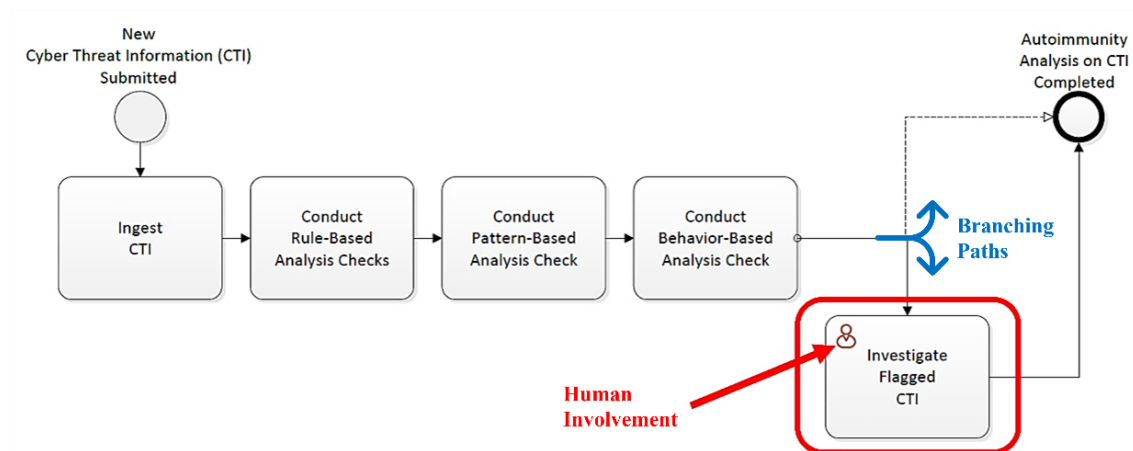


Figure 23. Human Involvement and Branch Paths. Adapted from IACD (2018).

Alert-driven task management enables organizations to respond to significant incidents based on critical alerts informing teams of potential risks and threats (SendQuick 2022). The use of alert-driven task management contrasts with the researcher’s predicted behavior depicted in Table 8, where the event ‘Investigate Flagged CTI’ relies on specific conditions or states of the CTI. However, when the event is alert-driven, the event ‘Investigate Flagged CTI’ solely depends on the alert itself rather than any additional underlying factors related to the CTI.

As Figure 24 illustrates, CTI Source exhibits two events where “CTI is self-referential,” and the Security System exhibits two events where “CTI is NOT flagged.” Without the Security System flagging a CTI, the “Investigate Flagged CTI” event does not occur.

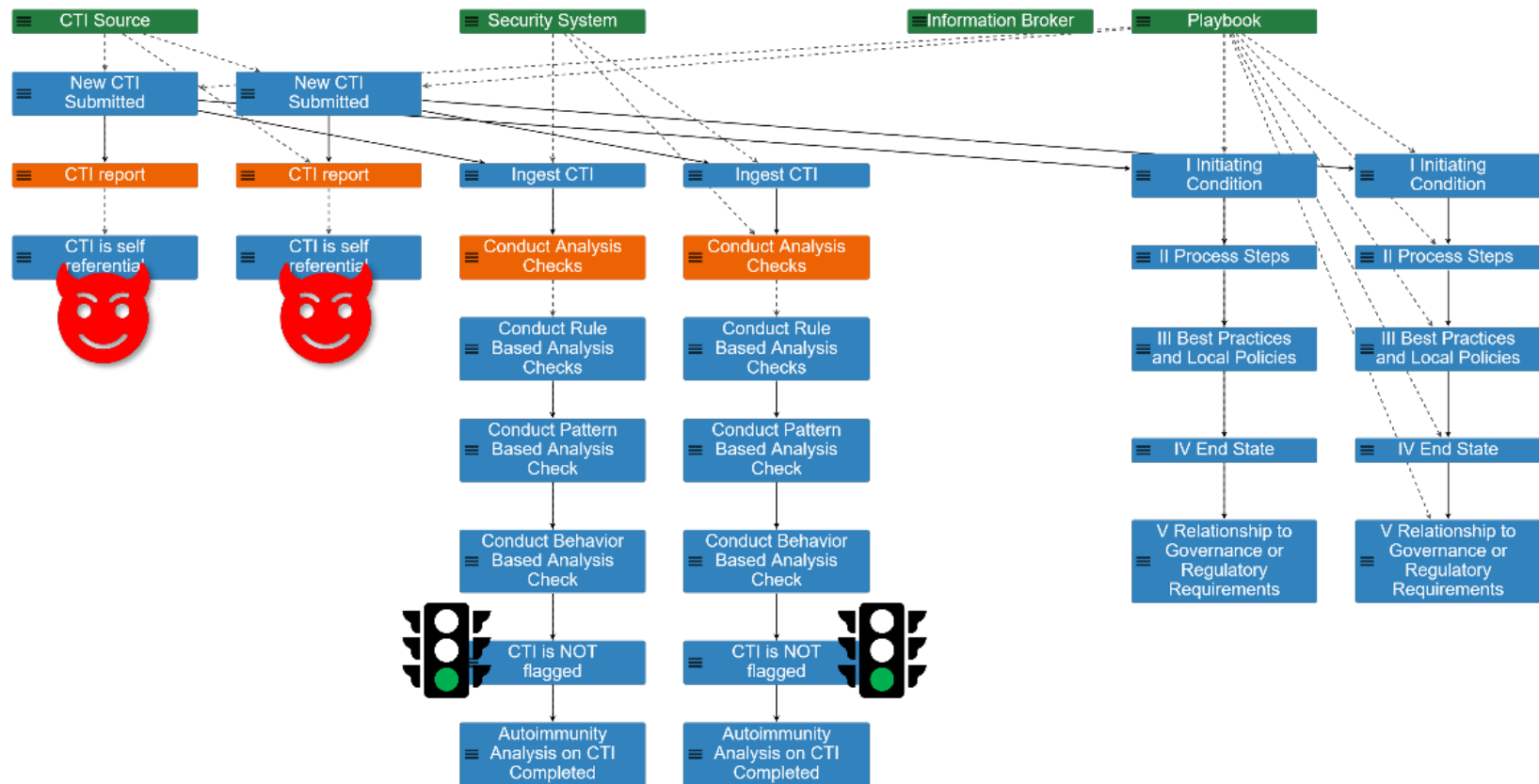


Figure 24. Event Trace Scope-2 Trace-7

The Security System’s flagging of a CTI is paramount in differentiating between various MP event traces. The occurrence of the event “CTI is flagged” is a necessary precondition for the subsequent event “Investigate Flagged CTI.” This relationship delineates a logical sequence whereby the investigation can only proceed following the flagging of a CTI. In Stage Four, the alternate modeling tool used in the analysis confirmed this emergent behavior, as shown in Figure 25. “Security System Flag CTI?” occupies the top position of the tree. According to the resulting Gini impurity metric used in this modeling tool (0.833), the event ‘Security System Flag CTI?’ emerges as the most discriminative event. This means splitting the data based on the values of “Security System Flag CTI?” provides the most information about which scenario might be occurring in the model. Additional information on the alternate modeling tool and the Gini impurity metric is found in Appendix B.

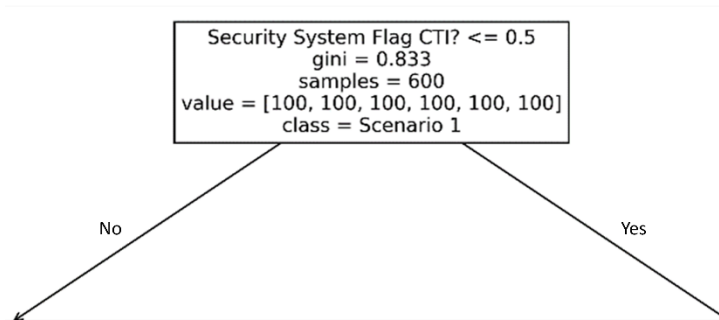


Figure 25. Top of Decision Tree Classifier

The flagging process determines which CTI to investigate, thereby guiding human participation toward specific CTIs. Consider, for example, the malicious act of intentionally submitting benign CTIs that the Security System nonetheless flags for investigation. Figure 26 illustrates this event sequence, in which a CTI is flagged and investigated. However, the investigation determines that the flagged CTI is benign.

This sequence emphasizes the need for accuracy in flagging and the potential consequences of erroneous flags. Such an act could quickly overwhelm the process of “Investigate Flagged CTI,” which requires human participation. Worse yet, if this pattern continues, Information Brokers and the Security System might interpret these activities as

normal and adjust the indicators responsible for flagging these benign CTIs. Should desensitization actions occur, the SoI's capability to identify analogous yet genuinely harmful activities in the future will correspondingly decrease.

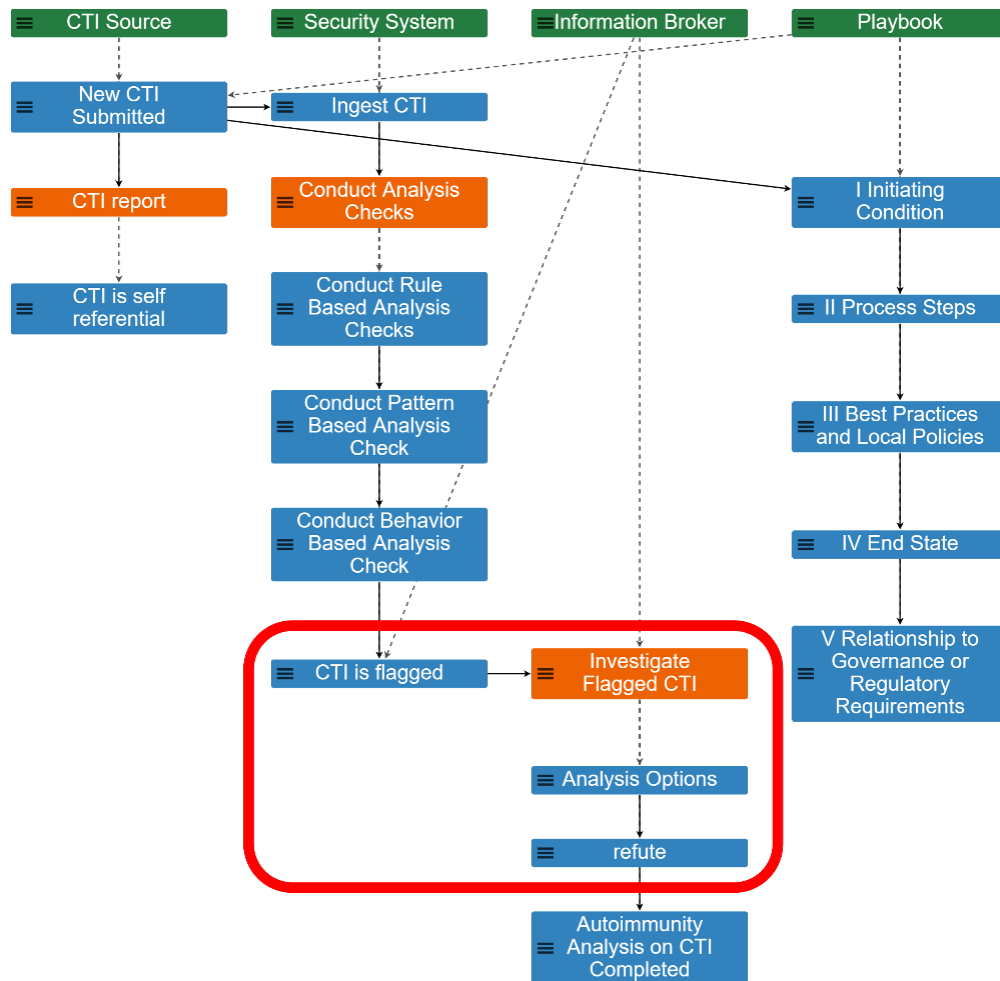


Figure 26. Event Trace Scope-2 Trace-3

This multi-stage analysis scrutinizes the SoI, pinpointing behavioral patterns and emergent propensities within the operational landscape. It emphasizes the effectiveness of MP in C-OPE activities. The investigation heightens situational awareness of the cyber environment through emergent behavior analysis, thereby bolstering commanders' decision-making within cyber operations. The subsequent chapter will synthesize the

findings coherently, proffer evidence-based recommendations, and outline directions for future inquiry. Collectively, these elements contribute to both academic understanding and practical application in cyber operations, aligning with the interests of scholars and engaged practitioners alike.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION

The significance of Cyber Operational Preparation of the Environment (C-OPE) lies in its ability to enhance commanders' situational awareness within the complex and perpetually evolving cyber environment. This study is an initial exploration of the application of MP to C-OPE activities; the analysis emphasizes MP's role in systems modeling, especially in identifying emergent behaviors. Through a comprehensive systems lens, clear insights are provided that contribute to improving commanders' situational awareness. Furthermore, the study applies MP as a means for a C-OPE activity, specifically its ability to enhance situational awareness within the cyber operational environment.

This study aims to examine and recommend effective methods and activities to support C-OPE. It acknowledges that the cyber environment, with its many interconnected systems, technologies, and protocols, is extensive and complex, reflecting the vastness and intricacy of modern digital interactions. This research uniquely contributes to this broad discipline by pursuing specific activities to support C-OPE.

A. SUMMARY OF KEY FINDINGS

This research demonstrated the practical application and value of MP in systems modeling, particularly in identifying emergent behaviors in the design of the 'Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted Cyber Threat Intelligence.' As detailed in Chapter IV and illustrated by the accompanying tables and figures, MP's ability to offer trace generation was instrumental in identifying behaviors one might otherwise overlook. For instance, the classification of 'Strong Negative' emergence and the discovery of the alert-driven nature of the "Investigate Flagged CTI" event revealed unexpected aspects of operational dynamics. Furthermore, this study exposed the importance of the Security System's flagging process in determining which CTIs to investigate, guiding human participation, and ensuring flagging accuracy.

These insights underscore the significance of apt cyber incident management and responsive action to alerts, encompassing the ramifications of incorrect flags. The scenario

delineated in the analysis, exemplified by the malevolent act of deliberately submitting benign CTIs, indicates that researchers should pursue further studies with this focus.

B. RECOMMENDATIONS

The findings of this research yield several helpful recommendations, each aimed at fostering a robust understanding of the cyber operational environment in the context of C-OPE activities.

1. **Establish Clear Criteria for Classifying Emergent Behaviors:** Developing a standardized and comprehensive set of criteria for identifying and categorizing emergent behaviors facilitates a more accurate analysis of complex cyber systems. A standardized uniform approach ensures that the modeling and analysis insights gained are consistent, measurable, and applicable across various operational contexts.
2. **Utilize Multiple Modeling Tools for Enriched Analysis:** Integrating multiple modeling tools in analyzing emergent behaviors ensures that the insights derived are correct and actionable. Utilizing diverse tools is helpful for capturing the multifaceted nature of cyber environments, allowing for a more comprehensive and nuanced understanding.
3. **Implement a Rigorous Verification and Refinement Process:** All modeling efforts must undergo rigorous verification and refinement. This step ensures the models are accurate and aligned with operational requirements, constraints, and minimizes potential errors, while enhancing overall efficiency.
4. **Identify, Understand, Model, and Analyze Key Processes, Activities, and Tasks:** The study calls for a detailed examination and optimization of the system's essential processes, activities, and tasks. This method could enhance accuracy in selecting CTIs for investigation, streamline human involvement, and minimize the risks of false alarms. Continued research in

this area may pinpoint vulnerabilities, recommend remedies, and aid in the continuous advancement of cybersecurity practices.

By executing these recommendations, researchers can attain a more comprehensive understanding of the cyber environment, leading to more accurate and effective contributions within C-OPE activities.

C. FUTURE RESEARCH

The exploration of emergent behavior, the utilization of MP in C-OPE activities, and other methodologies addressed in this research hold potential for further expansion and refinement. Future studies should consider designing and validating additional modeling tools to enhance emergent behavior analysis. The MP model developed in this research had few enough alternative branches that it was possible to fully map them out using a human-readable decision tree. The unexpected emergent behaviors found using MP could be seen also in the decision tree, because of the model's simplicity. A more complex behavior model would have a larger tree—perhaps too large for a human to process and correctly extract all unexpected behaviors. As a formal behavior modeling language that generates execution traces, MP automatically enumerates all possible paths through decision logic up to the user-defined scope. Manual enumeration (using a decision tree, for example) is prone to human errors, which are avoided by using automated enumeration (using MP, for example). As MP generates one trace for each path through the tree, it ensures full coverage through the given model that decision tree analysis does not provide. Future work involving more complex models should therefore leverage the automation provided by MP or other tools that come with a formal guarantee of execution trace set completeness. Collaborations with industry experts and integration with other frameworks are essential for future research, as they foster innovation and refine the alignment of C-OPE activities with real-world challenges, thereby building robust and adaptable capabilities that align with the constantly evolving cyber operational environment. Furthermore, the methodologies employed in this study can extend to areas beyond the cyber environment, enabling an evaluation of their effectiveness and applicability across various operational scenarios.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. SCHEMA

```
SCHEMA Advanced_Autoimmunity_Analysis_Playbook

/* Composite event that represents a CTI report. This event
can either be self-referential or not. */
CTI_report: ( CTI_is_self_referential | CTI_is_NOT_self_referential ) ;

/* Root named CTI_Source. This root starts with the initiating event. */
ROOT CTI_Source: {+ New_CTI_Submitted CTI_report +} ;

/* Composite event that represents the process of conducting analysis checks.
If CTI is not flagged then the playbook has met the end state. */
Conduct_Analysis_Checks:      Conduct_Rule_Based_Analysis_Checks
                                Conduct_Pattern_Based_Analysis_Check
                                Conduct_Behavior_Based_Analysis_Check
                                ( CTI_is_NOT_flagged
Autoimmunity_Analysis_on_CTI_Completed | CTI_is_flagged ) ;

/* Root named Security_System. This root can include any number of
Ingest_CTI and Conduct_Analysis_Checks events. */
ROOT Security_System: {* ( Ingest_CTI Conduct_Analysis_Checks ) *};

/* Coordinate the New_CTI_Submitted event from CTI_Source with the
Ingest_CTI event from Security_System, such that New_CTI_Submitted
always precedes Ingest_CTI. */
COORDINATE      $x: New_CTI_Submitted FROM CTI_Source,
                  $y: Ingest_CTI      FROM Security_System
DO              ADD $x PRECEDES $y; OD;

/* Composite event that represents the process of investigating a flagged CTI.
The completion of a investigating a flagged CTI meets the end state. */
Investigate_Flagged_CTI:      Analysis_Options
                                ( confirms | refute )
                                Autoimmunity_Analysis_on_CTI_Completed ;

/* Root named Information_Broker. This root can include any number of
CTI_is_flagged and Investigate_Flagged_CTI events. */
ROOT Information_Broker: {* ( CTI_is_flagged Investigate_Flagged_CTI ) *};

/* Make the CTI_is_flagged event shared between Security_System and Information_Broker.
Investigating a flagged CTI will only occur if CTI is flagged. */
Security_System, Information_Broker SHARE ALL CTI_is_flagged;

/* Define a root named Playbook. This root represents a sequence of
steps involved in handling a CTI. */
ROOT Playbook: ({* New_CTI_Submitted
                  I_Initiating_Condition
                  II_Process_Steps
                  III_Best_Practices_and_Local_Policies
                  IV_End_State
                  V_Relationship_to_Governance_or_Regulatory_Requirements
                  *});

/* Make the New_CTI_Submitted event shared between Playbook and CTI_Source.
This adds all five playbook components to the trace for each initiating condition */
Playbook, CTI_Source SHARE ALL New_CTI_Submitted;
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. ALTERNATE MODELING TOOL

```
"""
@article{scikit-learn,
  title={Scikit-learn: Machine Learning in {P}ython},
  author={Pedregosa, F. and Varoquaux, G. and Gramfort, A. and Michel, V.
    and Thirion, B. and Grisel, O. and Blondel, M. and Prettenhofer, P.
    and Weiss, R. and Dubourg, V. and Vanderplas, J. and Passos, A. and
    Cournapeau, D. and Brucher, M. and Perrot, M. and Duchesnay, E.},
  journal={Journal of Machine Learning Research},
  volume={12},
  pages={2825--2830},
  year={2011}
}
"""
import numpy as np
import pandas as pd
from sklearn.tree import DecisionTreeClassifier, plot_tree
import matplotlib.pyplot as plt

# 'Security System Flag CTI?' (0 or 1),
# 'CTI Self-referential?' (0 or 1)
# 'Info Brokers confirms flag?' (-1, 0, or 1)
data = [
    [0, 0, -1],
    [0, 1, -1],
    [1, 1, 1],
    [1, 1, 0],
    [1, 0, 1],
    [1, 0, 0],
]
# make the decision tree more robust and less likely to overfit the training data
data = data * 100

# Converting to DataFrame for better visualization
columns = ['Security System Flag CTI?', 'CTI Self-referential?', 'Info Brokers confirms flag?']
df = pd.DataFrame(data, columns=columns)

class_names = ["Scenario 1", "Scenario 2", "Scenario 3", "Scenario 4", "Scenario 5", "Scenario 6"]

# Targets (Outcomes)
targets = [1, 2, 3, 4, 5, 6] * 100

# Train the decision tree
clf = DecisionTreeClassifier(max_depth=3, random_state=1)
clf.fit(df, targets)

# Plot the decision tree
plt.figure(figsize=(16, 12))
plot_tree(
    clf,
    filled=True,
    feature_names=df.columns,
    class_names=class_names,
    fontsize=11,
)
plt.show()
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. SCOPE 2 TRACES

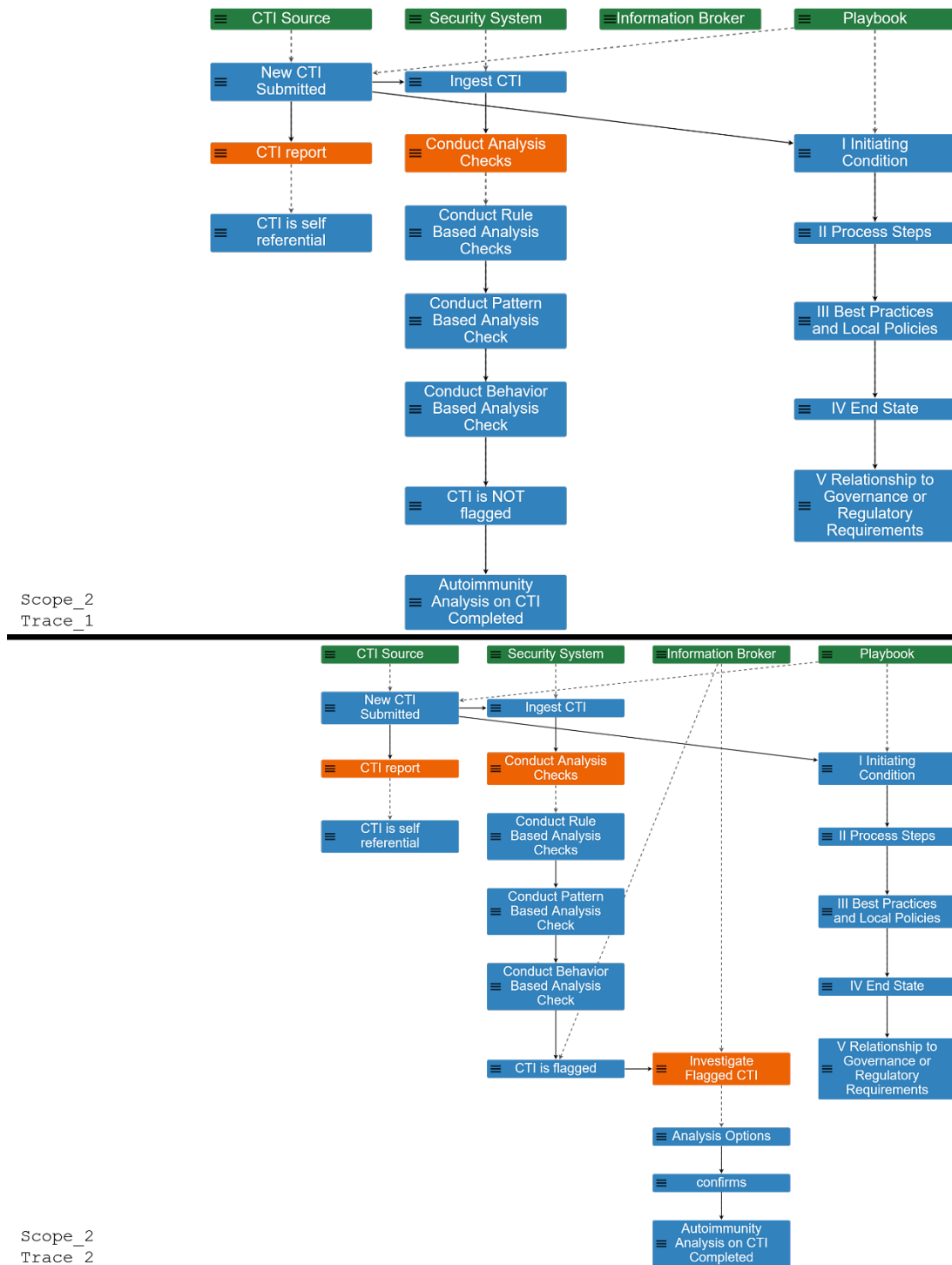


Figure 27. Event Trace Scope-2 Traces 1–2

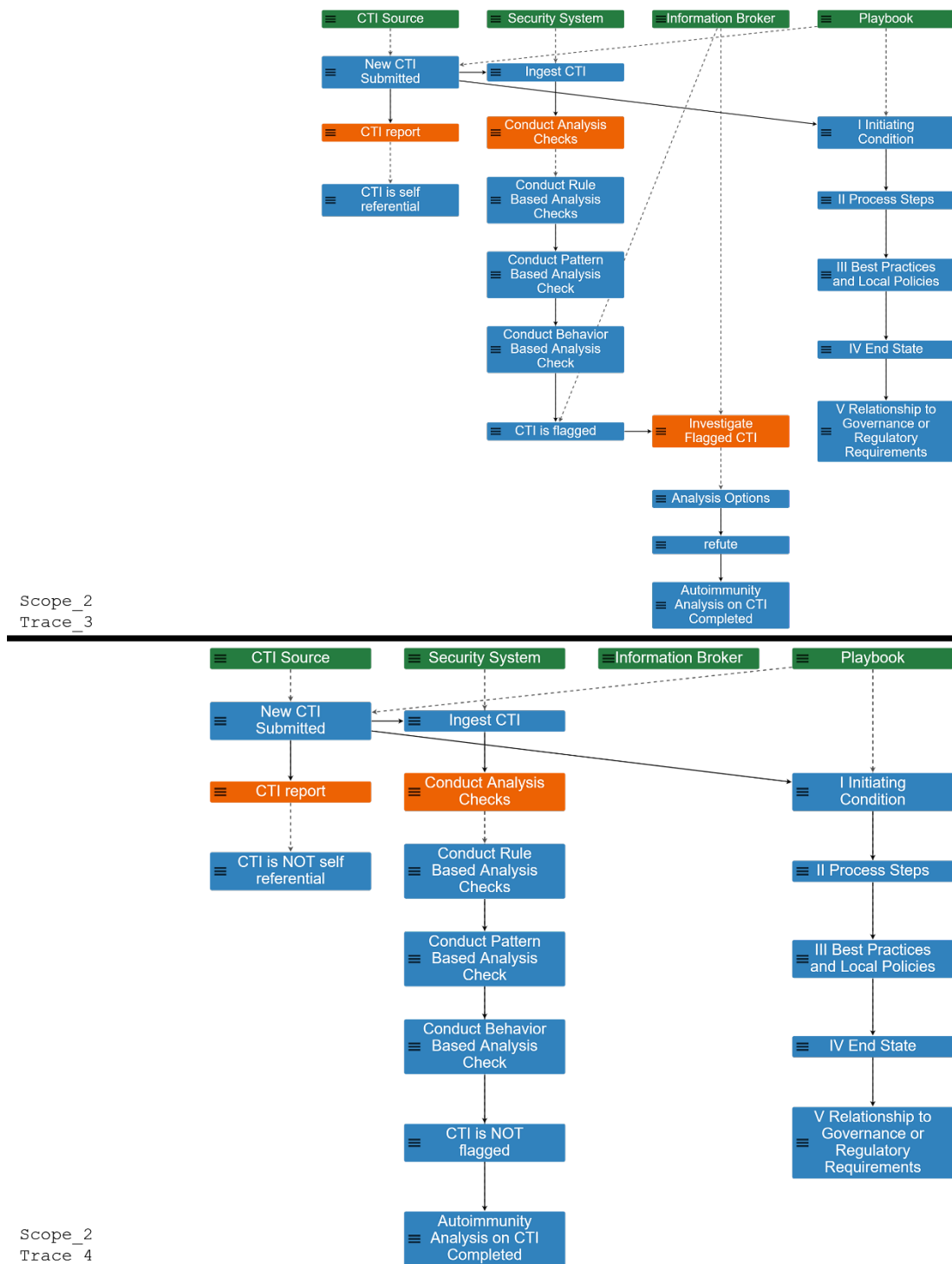


Figure 28. Event Trace Scope-2 Traces 3–4

Scope_2
Trace_5

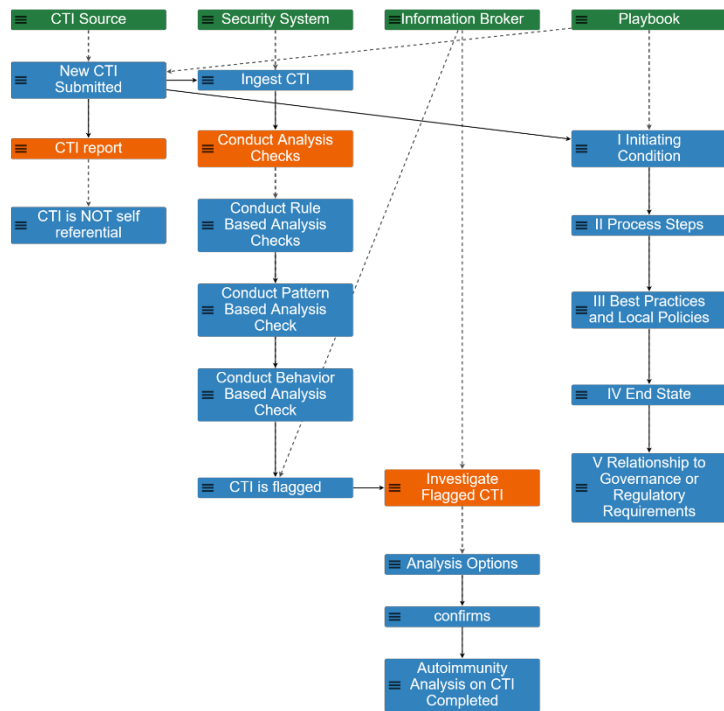


Figure 29. Event Trace Scope-2 Traces 5

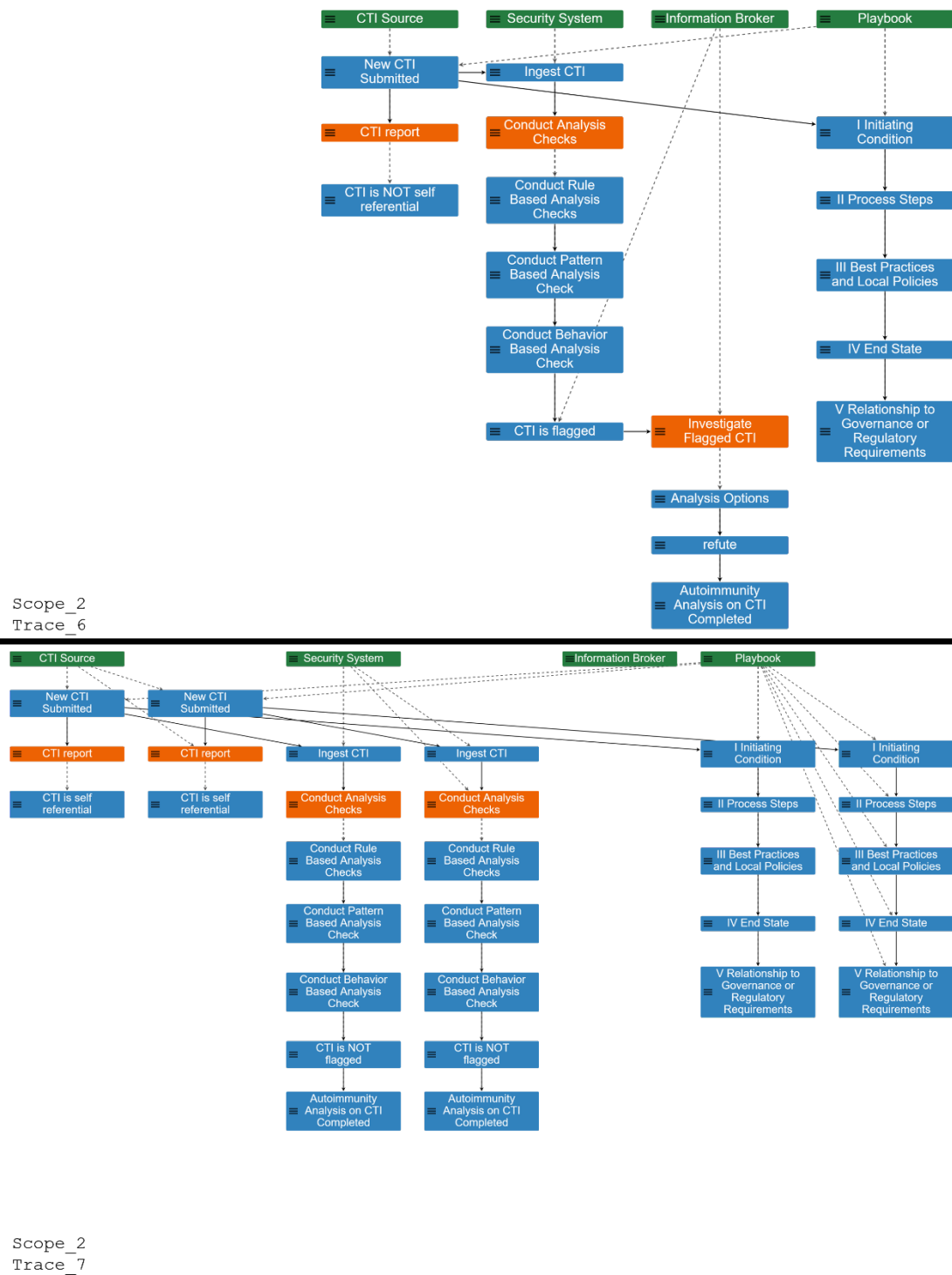
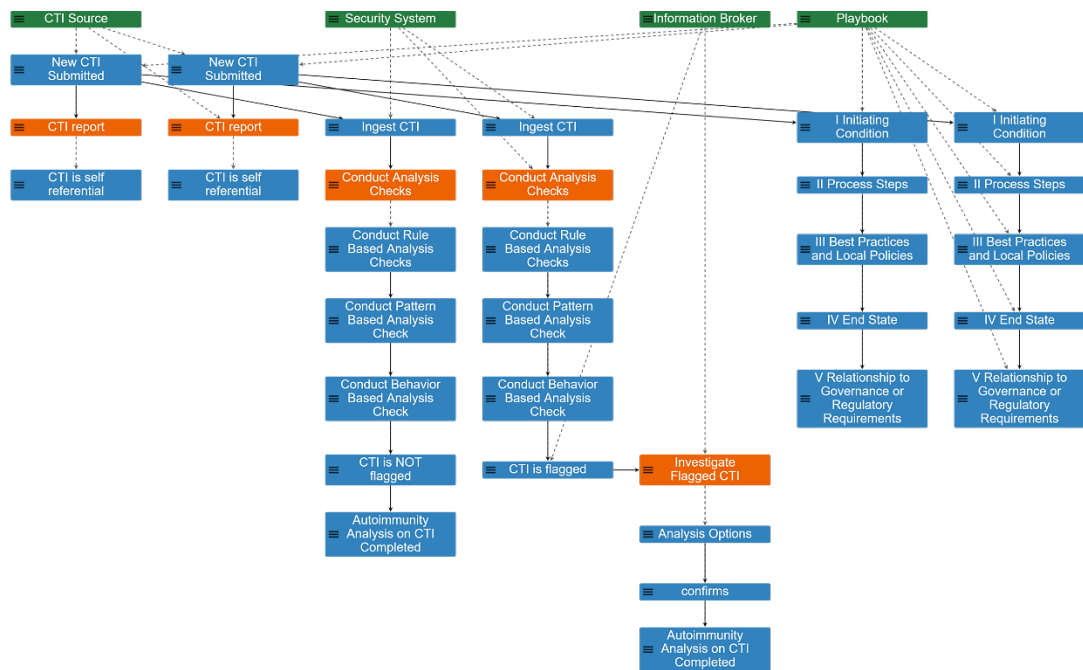
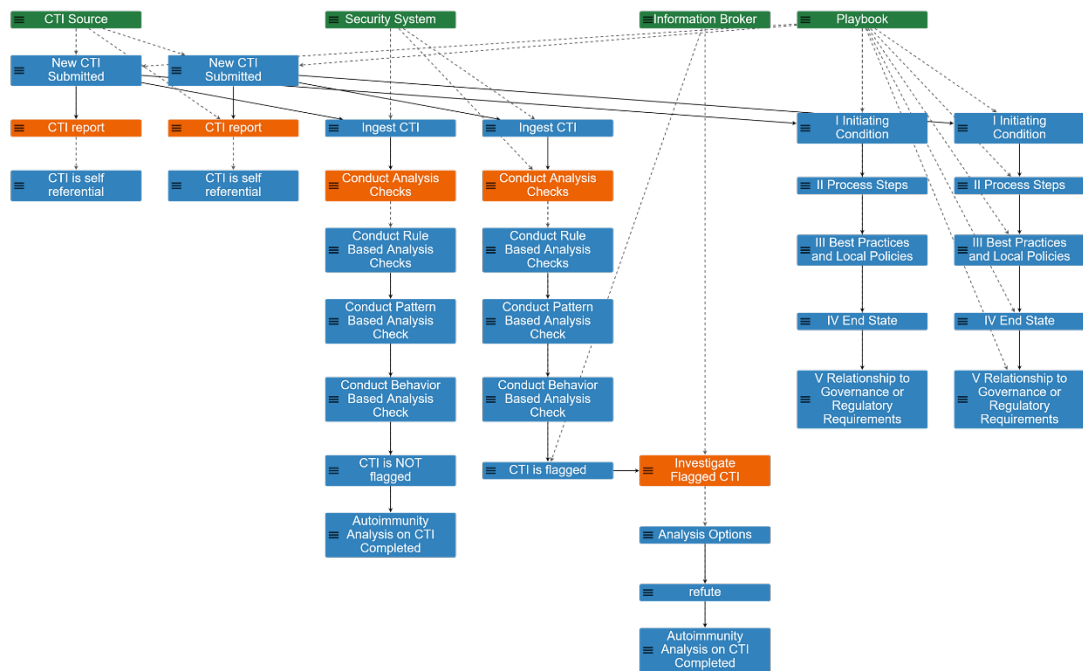


Figure 30. Event Trace Scope-2 Traces 6–7



Scope_2
Trace_8



Scope_2
Trace_9

Figure 31. Event Trace Scope-2 Traces 8–9

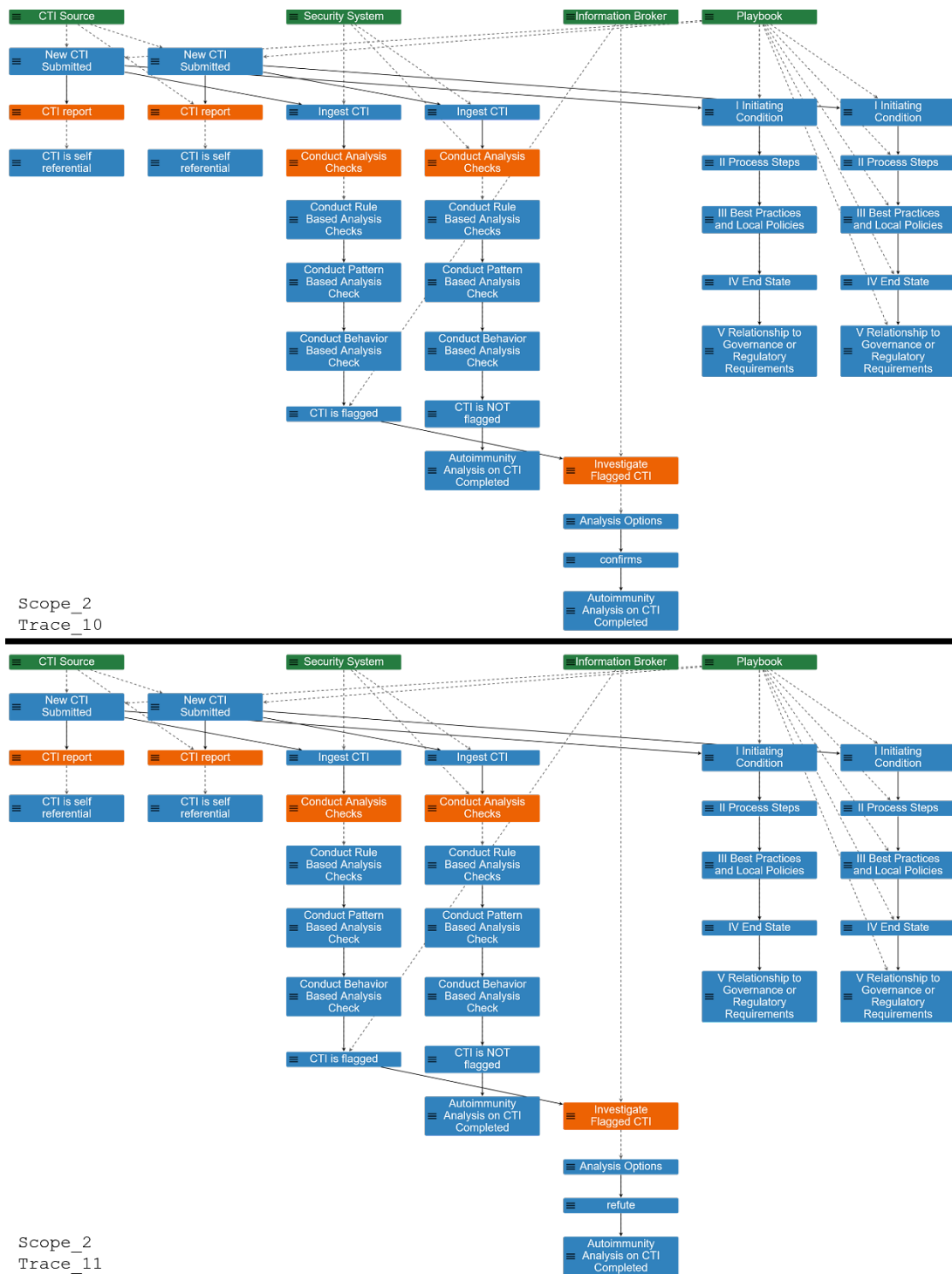
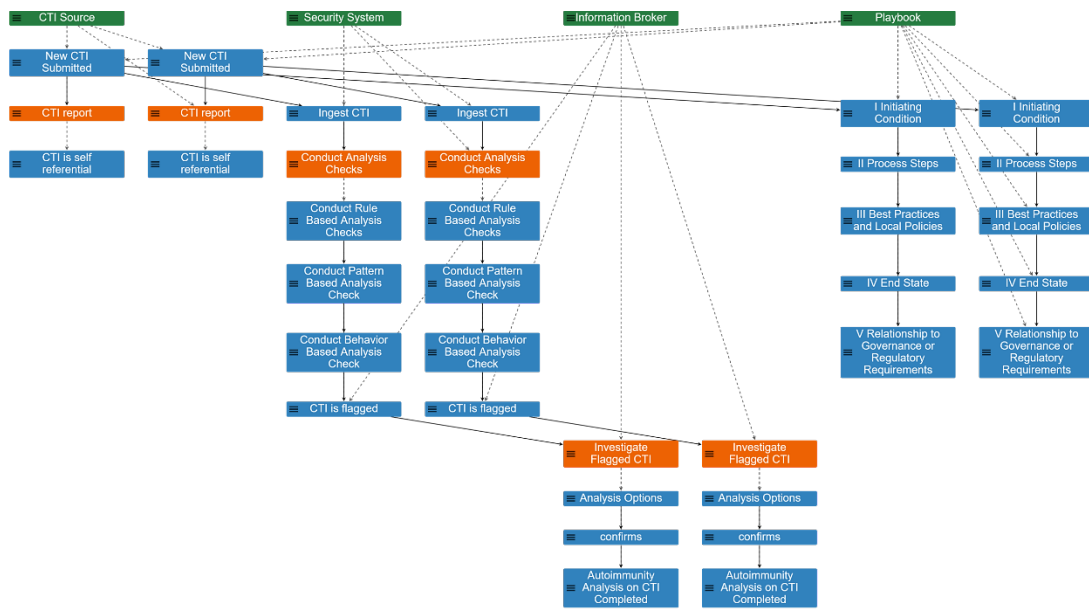
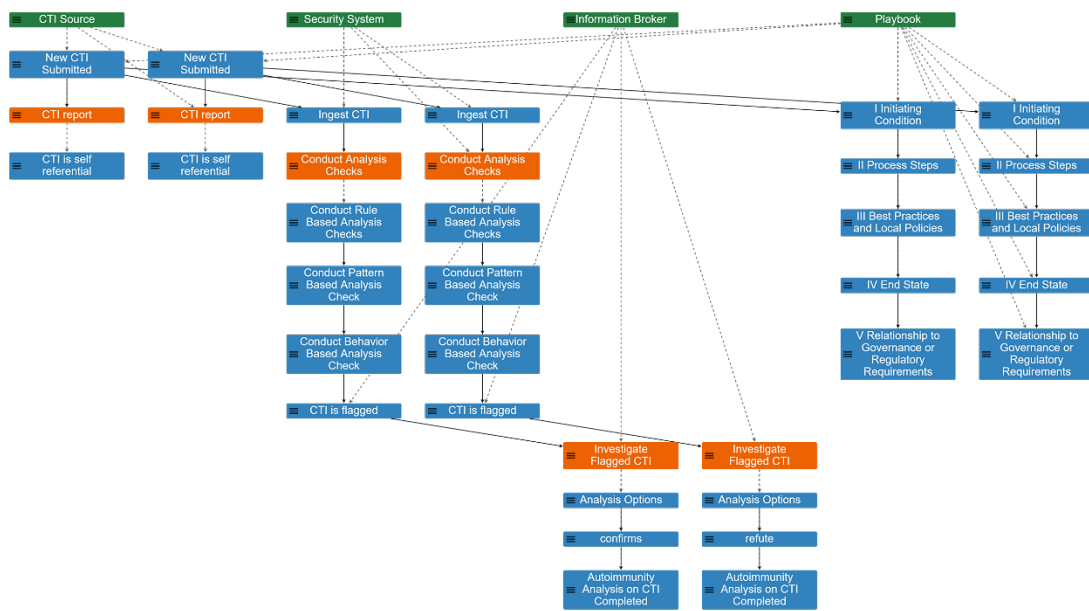


Figure 32. Event Trace Scope-2 Traces 10–11

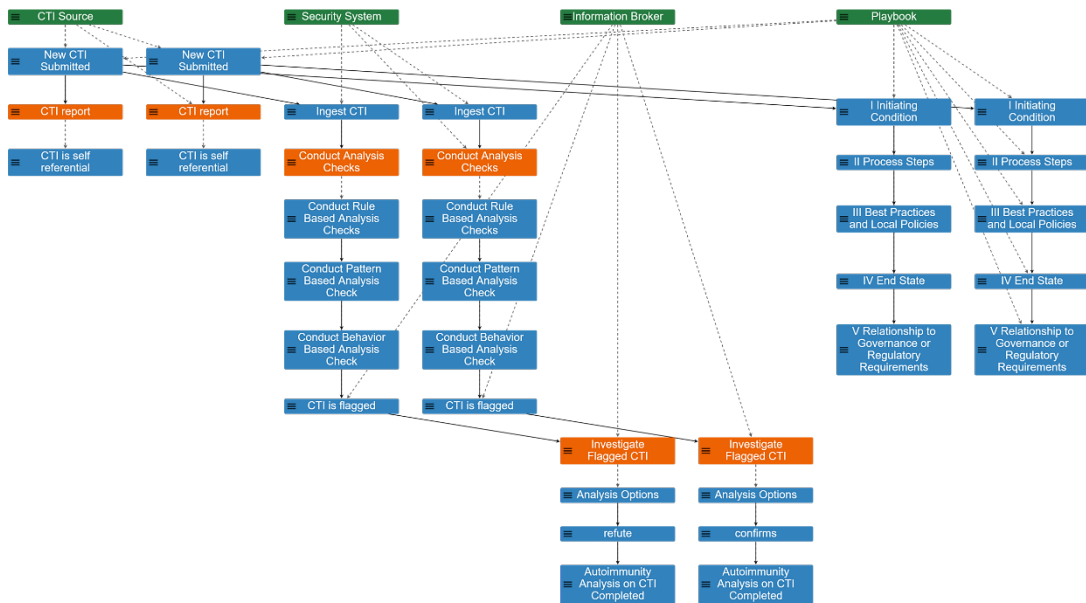


Scope_2
Trace_12

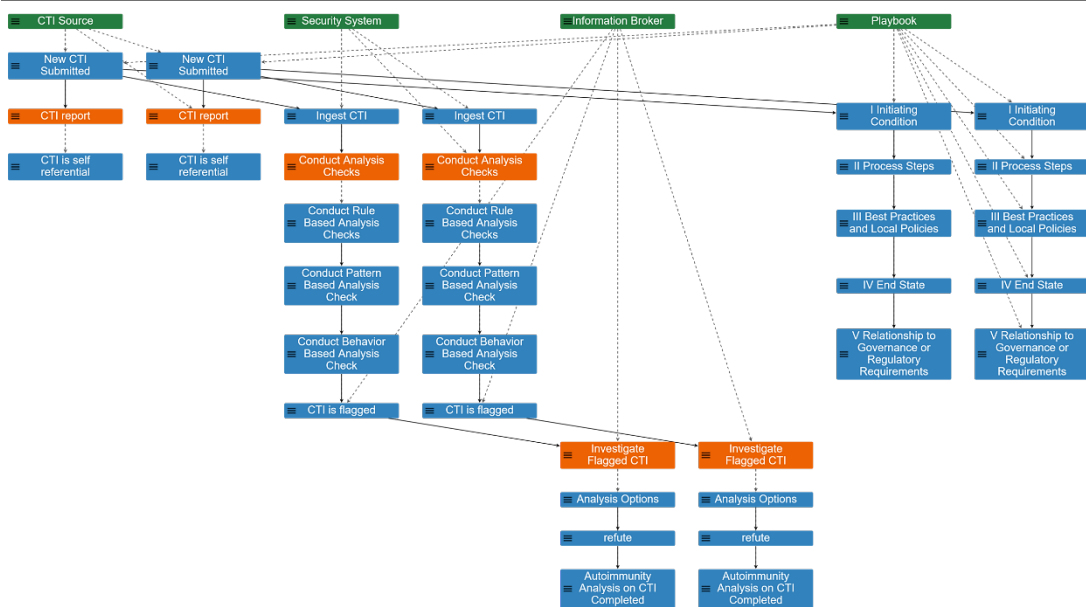


Scope_2
Trace_13

Figure 33. Event Trace Scope-2 Traces 12–13

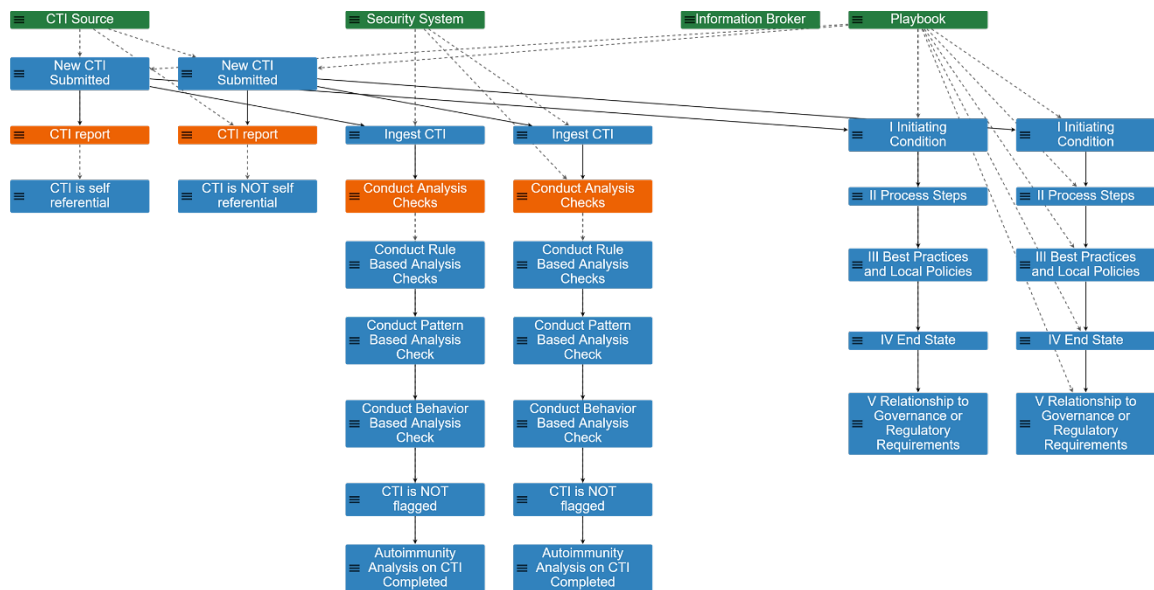


Scope_2
Trace_14

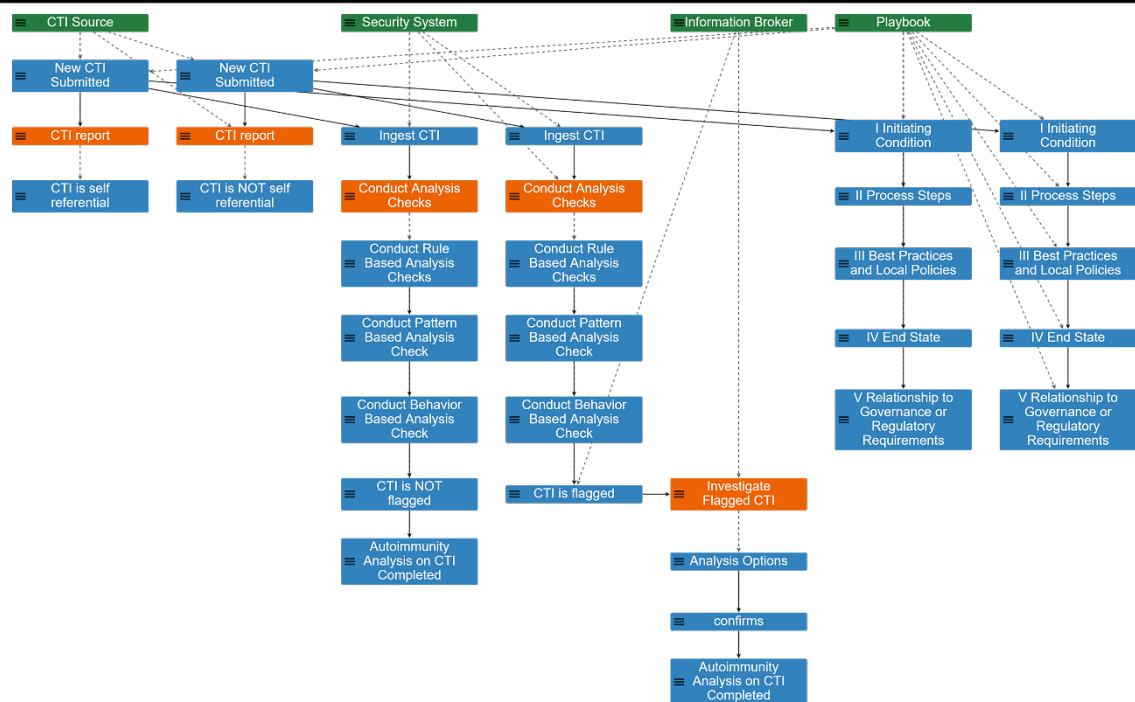


Scope_2
Trace_15

Figure 34. Event Trace Scope-2 Traces 14–15

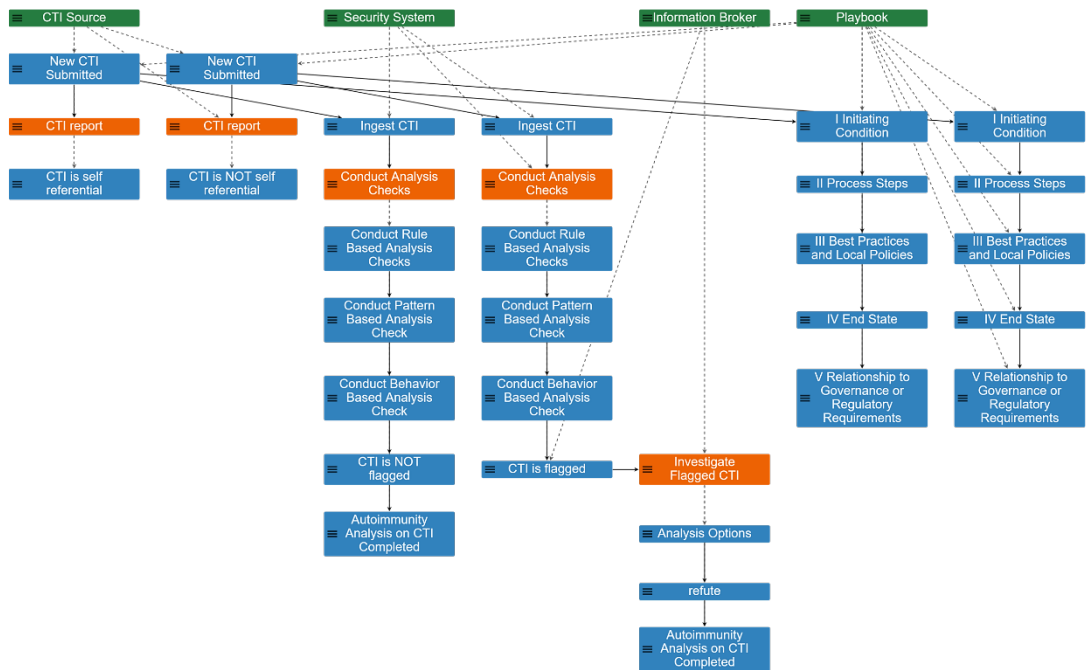


Scope_2
Trace_16

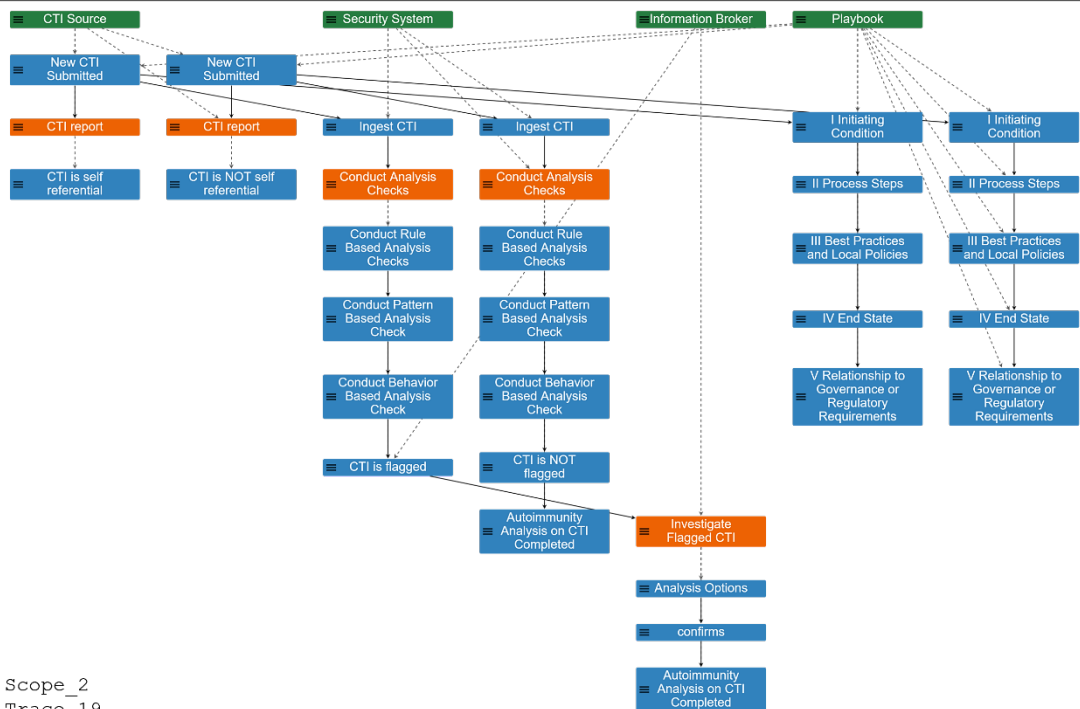


Scope_2
Trace_17

Figure 35. Event Trace Scope-2 Traces 16–17



Scope_2
Trace_18



Scope_2
Trace_19

Figure 36. Event Trace Scope-2 Traces 18–19

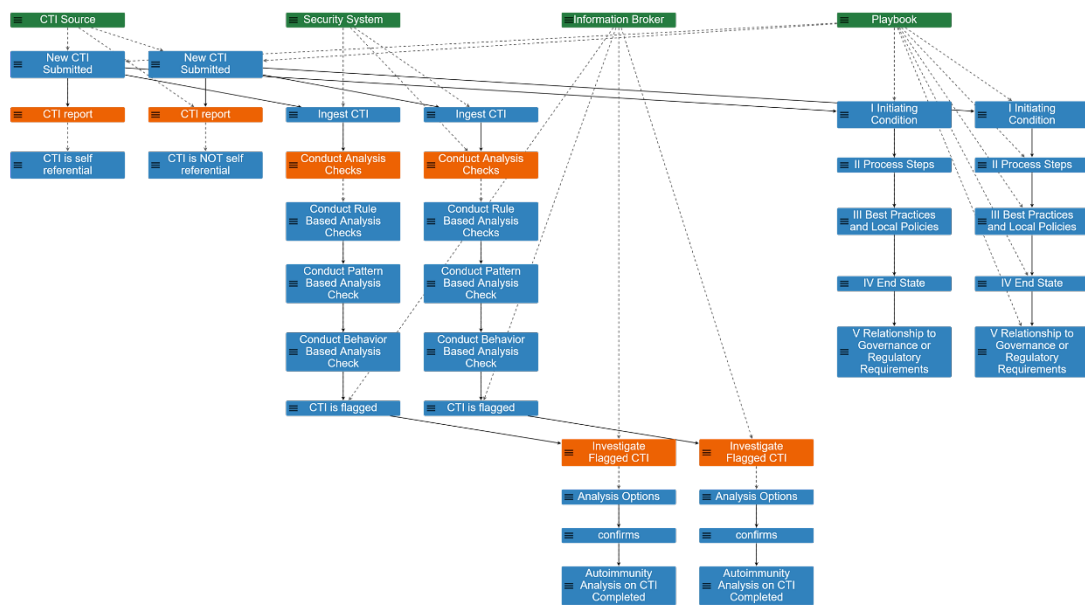
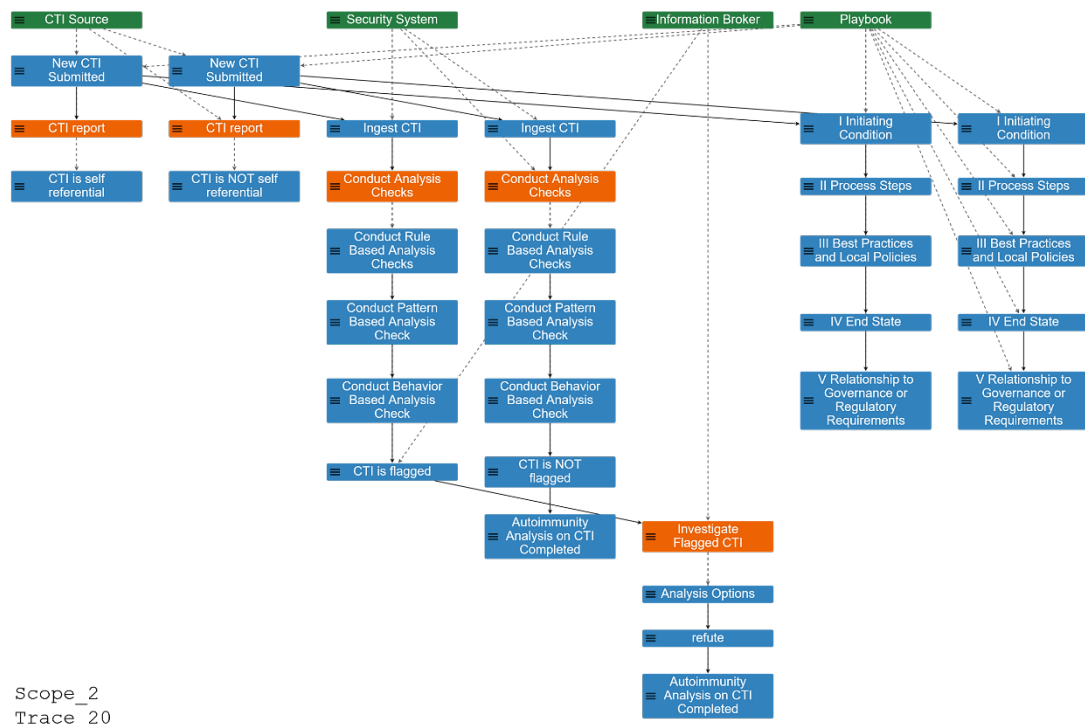
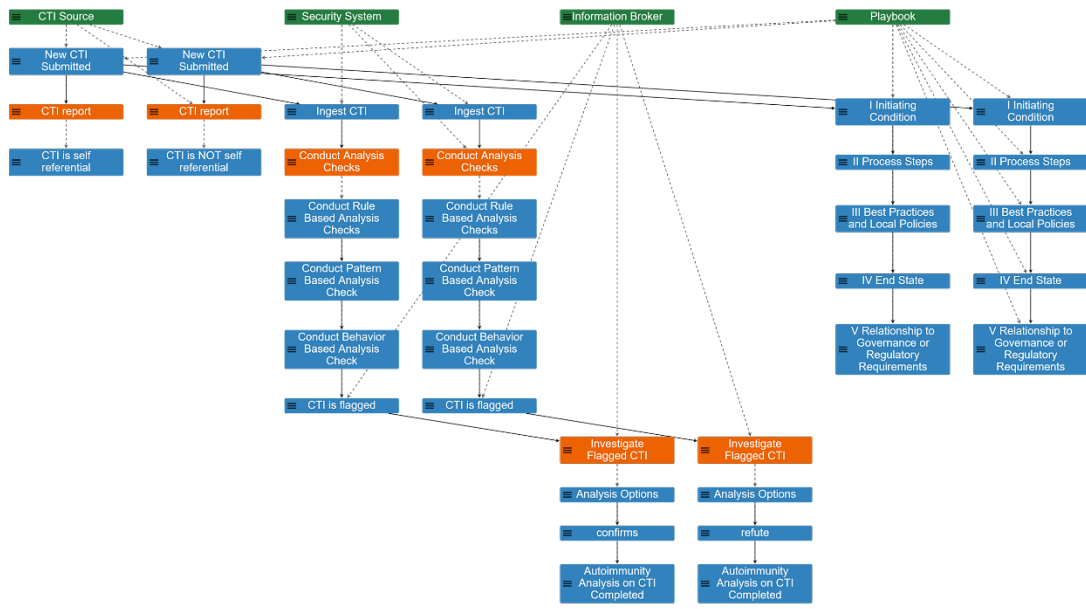
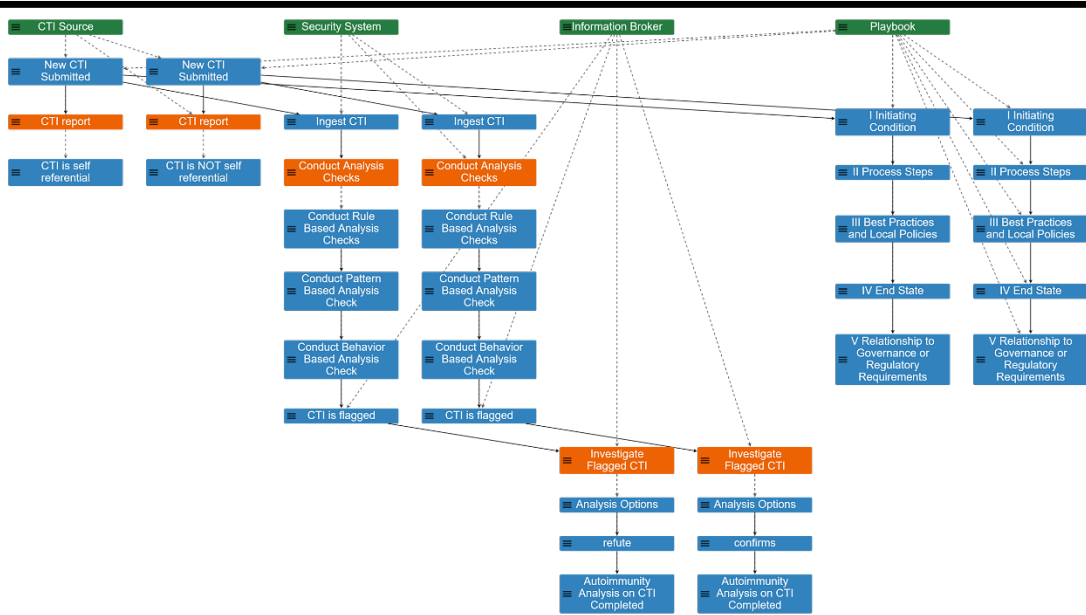


Figure 37. Event Trace Scope-2 Traces 20–21

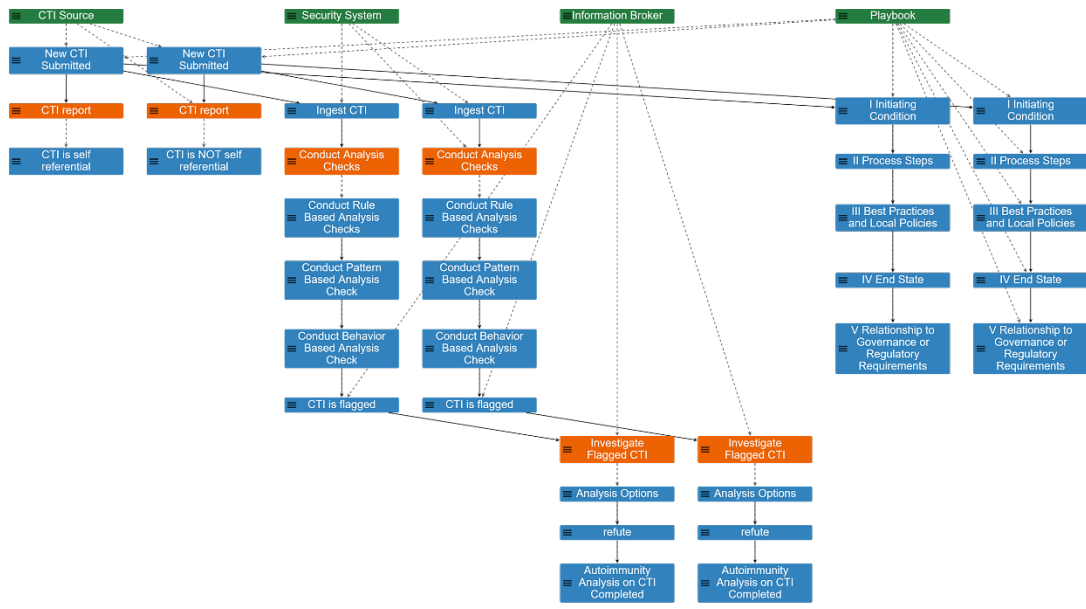


Scope_2
Trace_22

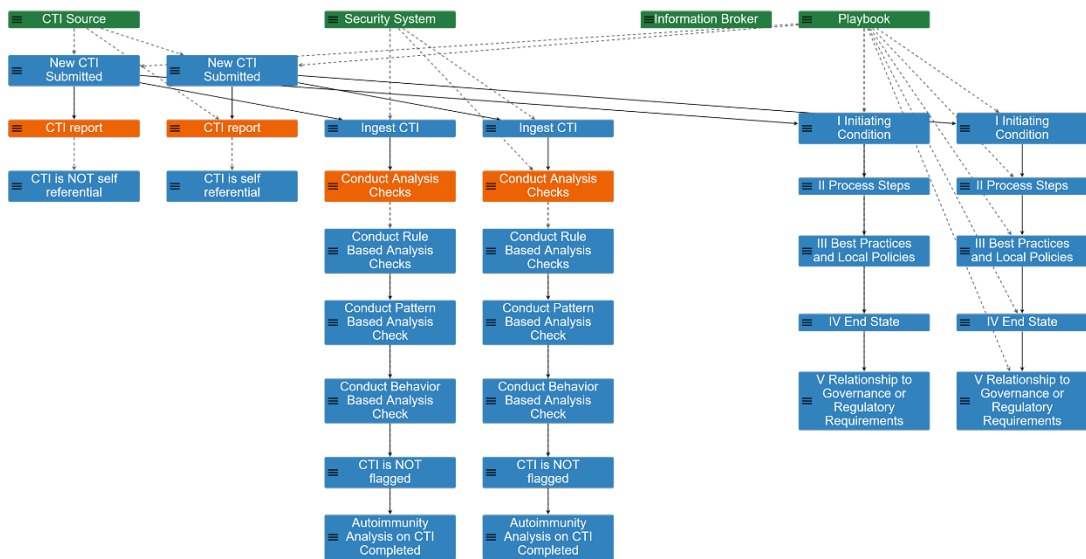


Scope_2
Trace_23

Figure 38. Event Trace Scope-2 Traces 22–23



Scope_2
Trace_24



Scope_2
Trace_25

Figure 39. Event Trace Scope-2 Traces 24–25

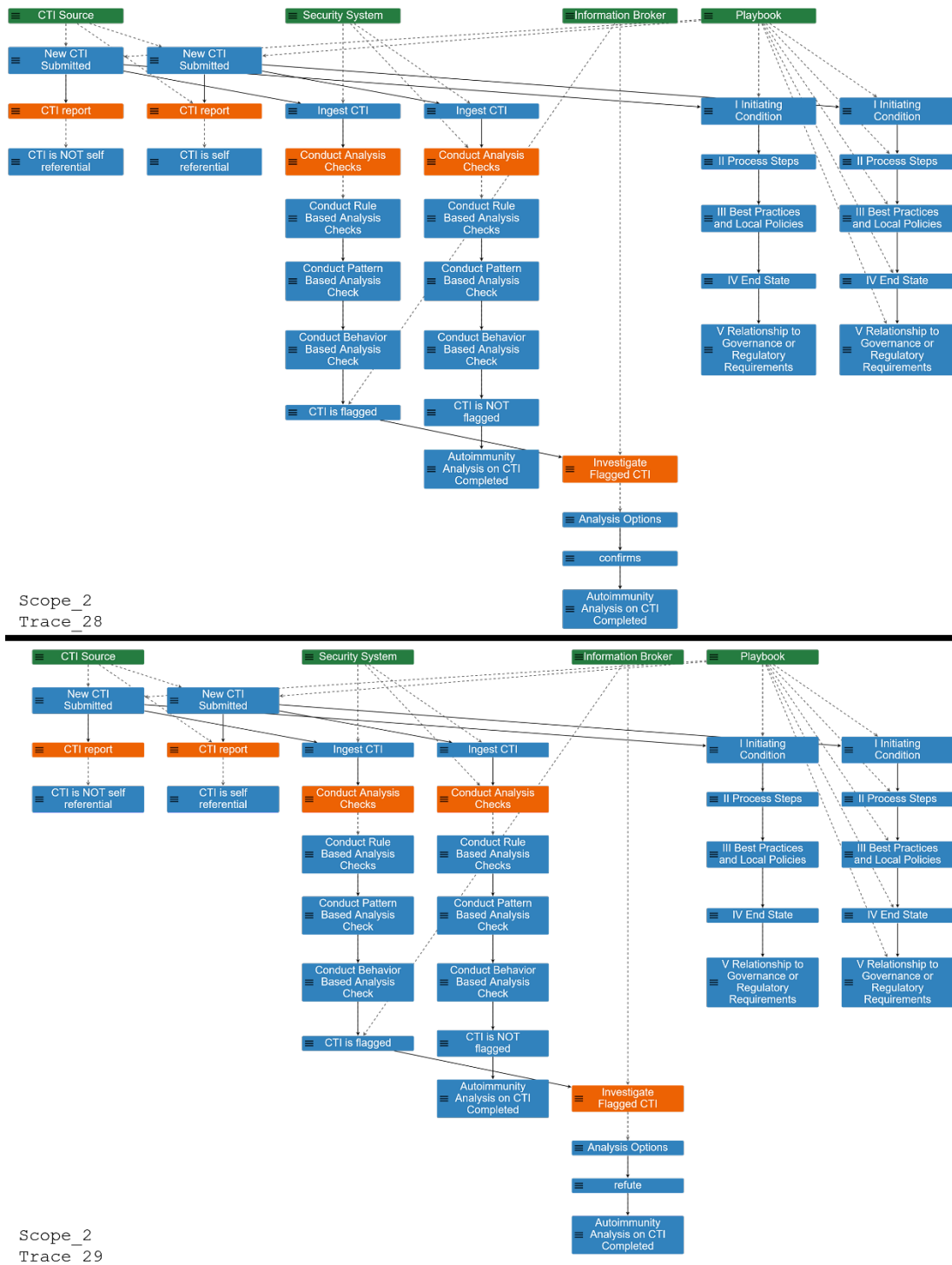
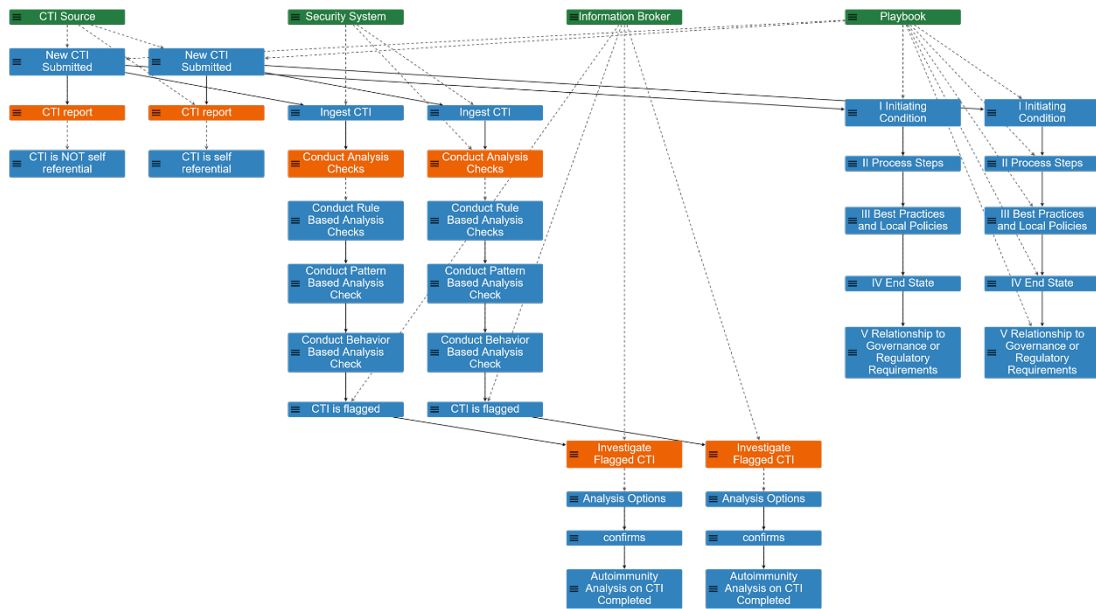
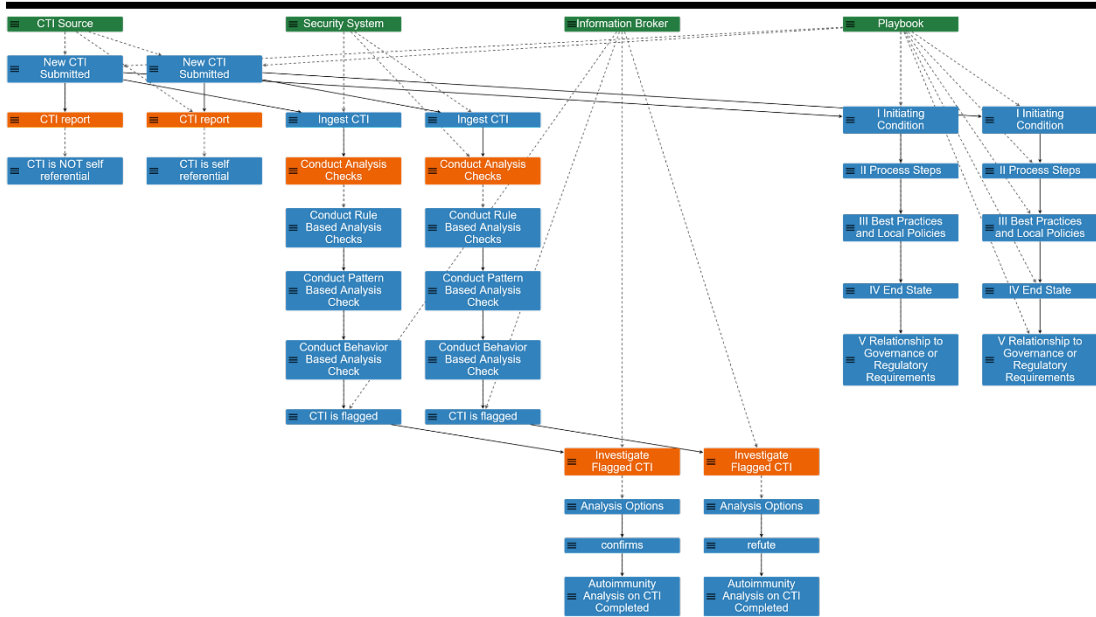


Figure 41. Event Trace Scope-2 Traces 28–29

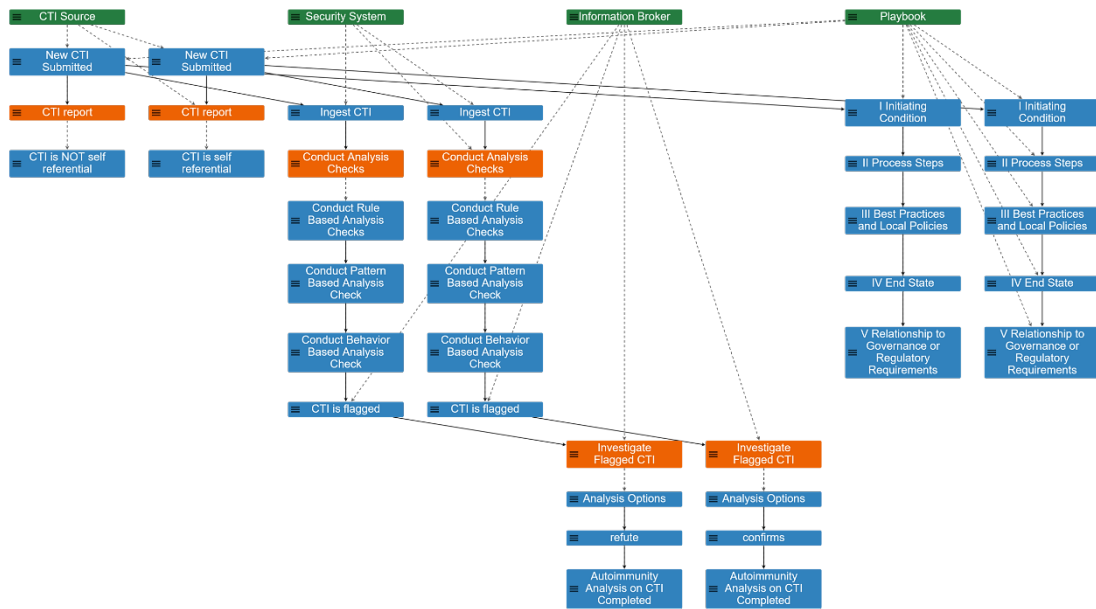


Scope_2
Trace_30

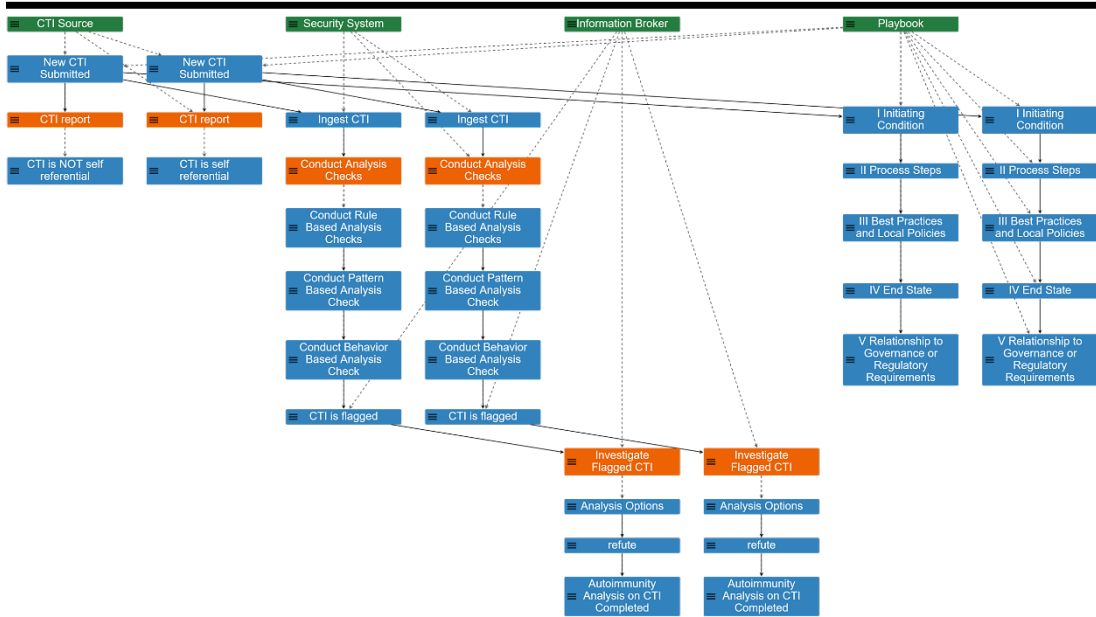


Scope_2
Trace_31

Figure 42. Event Trace Scope-2 Traces 30–31

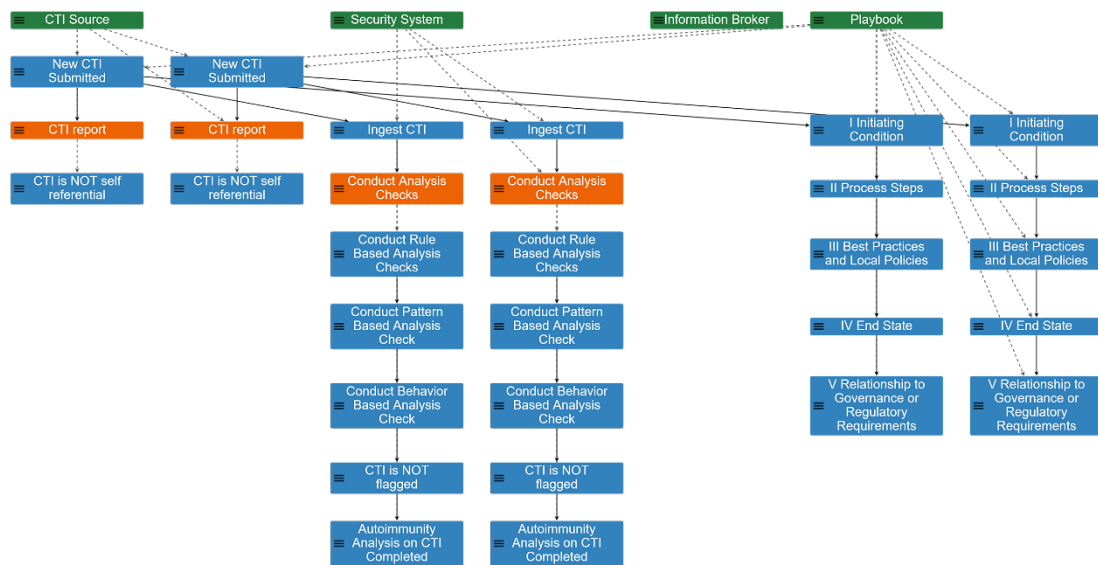


Scope_2
Trace_32

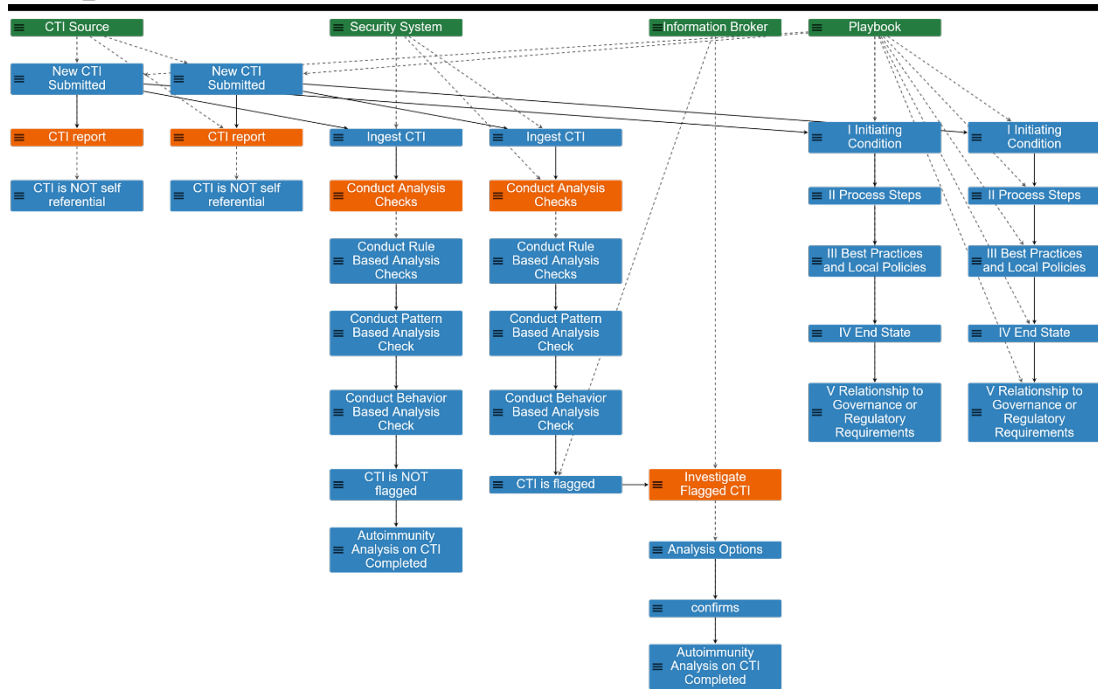


Scope_2
Trace_33

Figure 43. Event Trace Scope-2 Traces 32–33

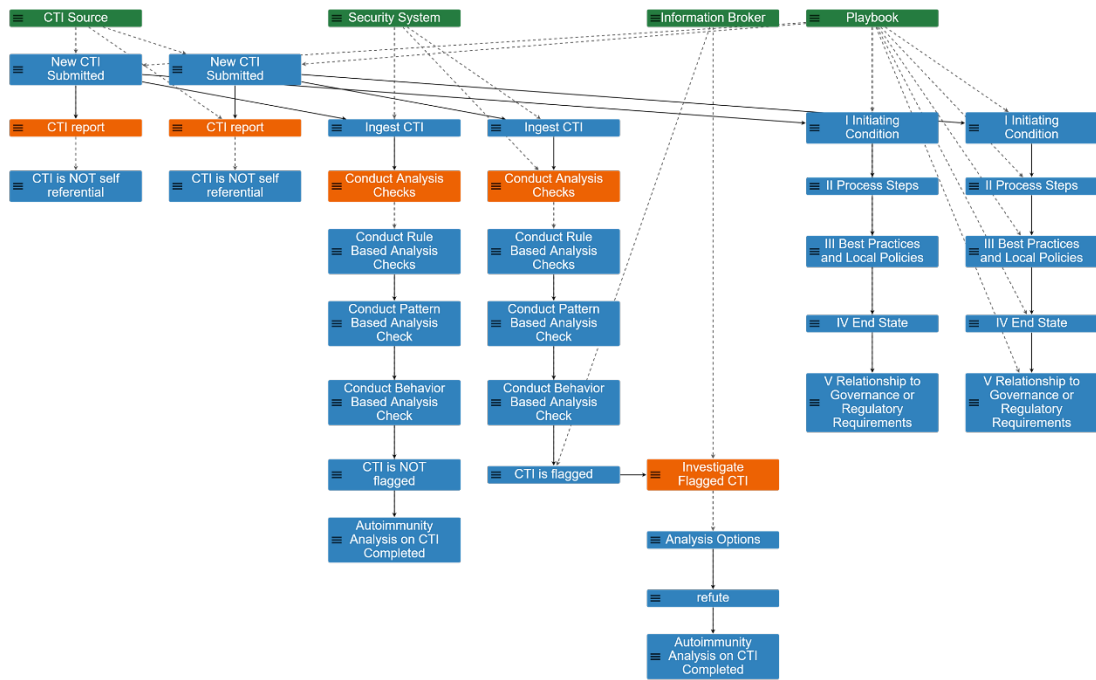


Scope_2
Trace_34

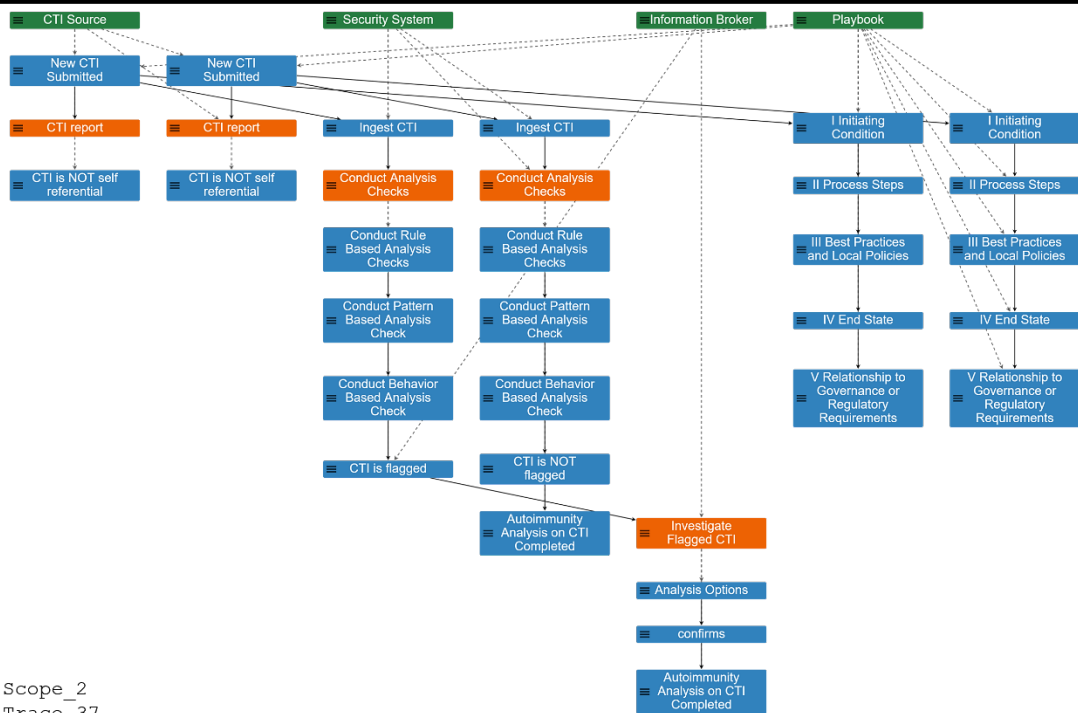


Scope_2
Trace_35

Figure 44. Event Trace Scope-2 Traces 34–35



Scope_2
Trace_36



Scope_2
Trace_37

Figure 45. Event Trace Scope-2 Traces 36–37

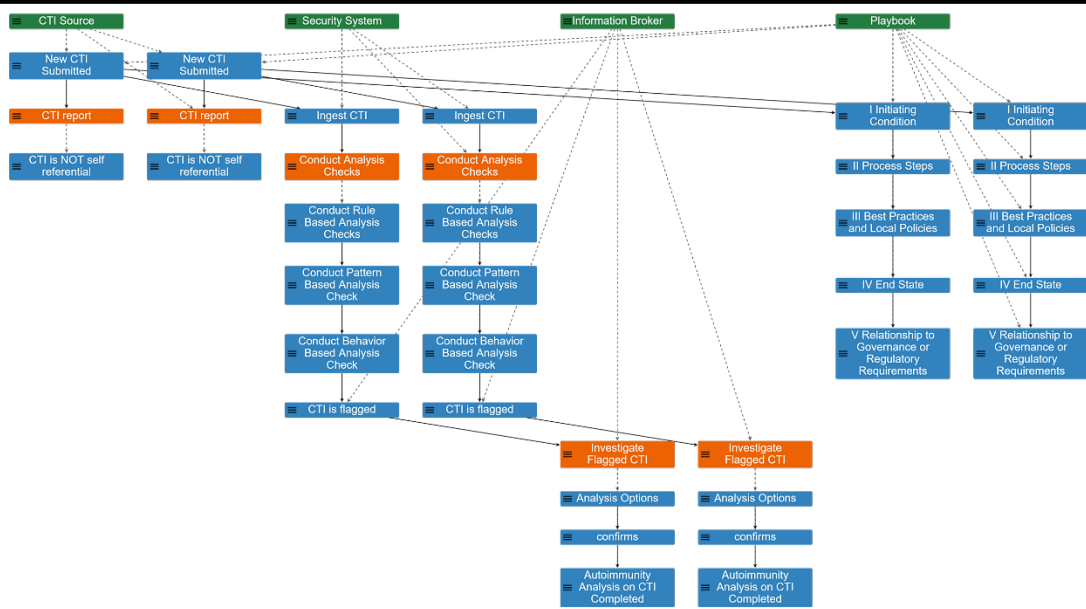
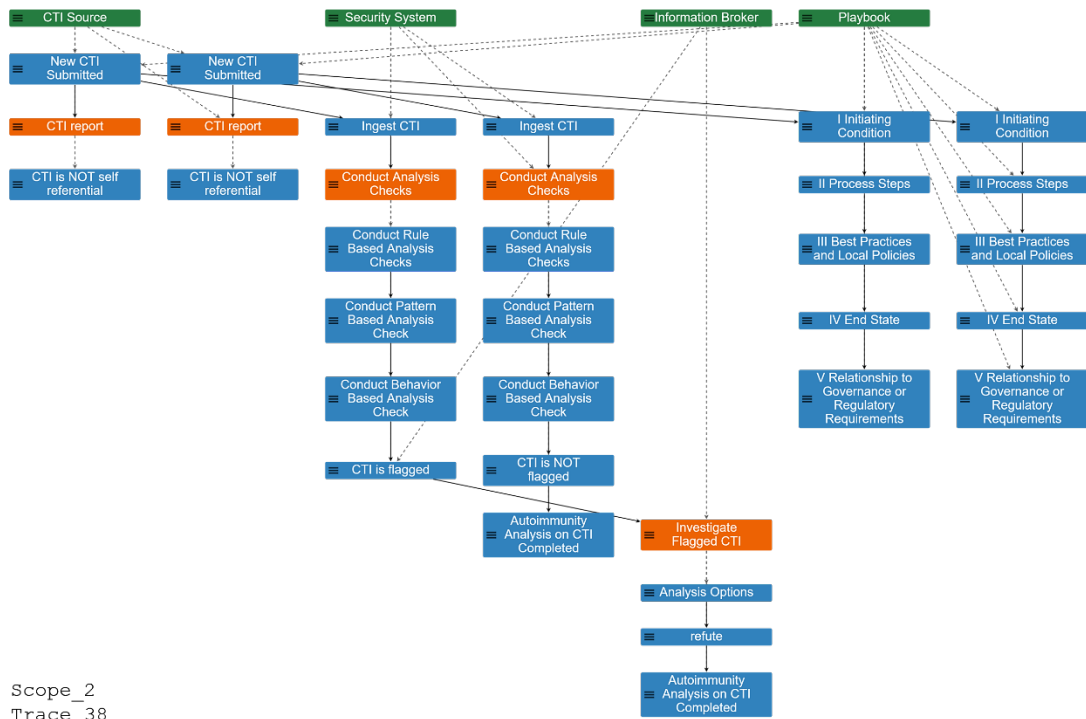
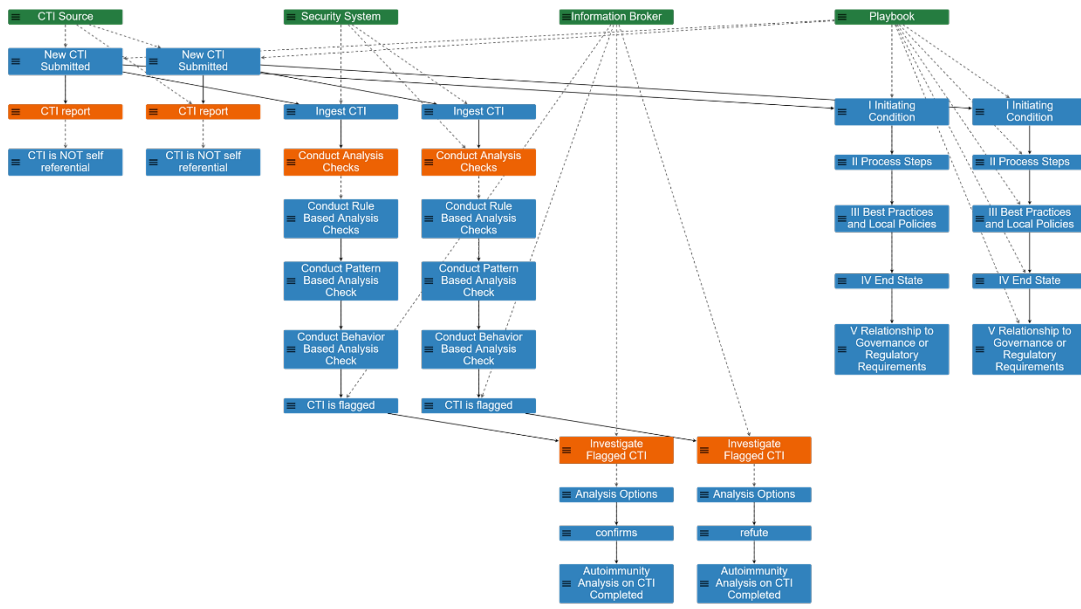
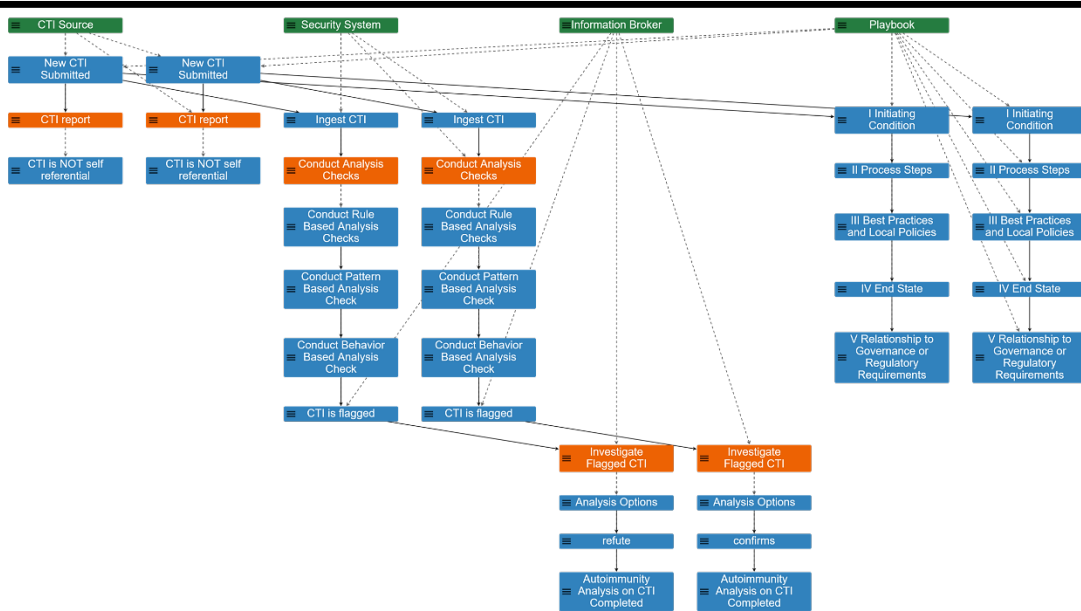


Figure 46. Event Trace Scop- 2 Traces 38–39

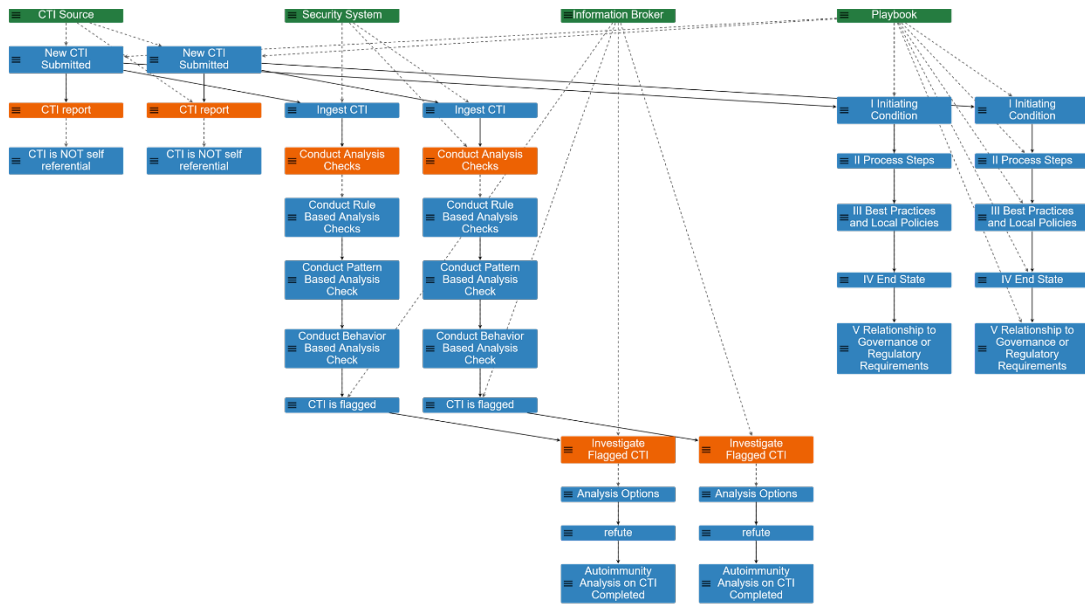


Scope_2
Trace_40



Scope_2
Trace_41

Figure 47. Event Trace Scope-2 Traces 40–41



Scope_2
Trace_42

Figure 48. Event Trace Scope-2 Traces 42

LIST OF REFERENCES

- Adcock, Rick, Scott Jackson, Dick Fairley, Janet Singer, Duane Hybertson. 2023. "Emergence." In *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*, v. 2.8, edited by R.J. Cloutier. Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed June 23. www.sebokwiki.org.
- Adcock, Rick, Brian Wells, Scott Jackson, James Martin. "Engineered System Context." in SEBoK Editorial Board. 2023. *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*, v. 2.8, edited by R.J. Cloutier. Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed 23 June 2023. www.sebokwiki.org.
- Adcock, Rick, Scott Jackson, Janet Singer, Duane Hybertson. 2023. "Principles of Systems Thinking." In *The Guide to the Systems Engineering Body of Knowledge (SEBoK)*, v. 2.8, edited by R.J. Cloutier. Hoboken, NJ: The Trustees of the Stevens Institute of Technology. Accessed 23 June 2023. www.sebokwiki.org.
- Antúnez, Juan Carlos. 2021. "Understanding the Operational Environment: The Human Dimension." Global Strategy Report. January 2021. <https://global-strategy.org/understanding-the-operational-environment-the-human-dimension/>.
- Aslanidis, Paris. 2018. "Populism as a Collective Action Master Frame for Transnational Mobilization." *Sociological Forum* (February) 33 (2): 443–64. <https://doi.org/10.1111/socf.12424>.
- Auguston, Mikhail, James Bret Michael, and Man-Tak Shing. 2006. "Environment Behavior Models for Automation of Testing and Assessment of System Safety." In *Information and Software Technology* 48: 971–980. <https://hdl.handle.net/10945/59398>.
- Auguston, Mikhail. 2009. "Monterey Phoenix, or How to Make Software Architecture Executable." In *Proceedings of the 24th ACM SIGPLAN Conference Companion on Object Oriented Programming Systems Languages and Applications*, 1031–40. <https://doi.org/10.1145/1639950.1640075>.
- Auguston, M. 2020. *System and Software Architecture and Workflow Modeling Language Manual (Version 4)*. Monterey, CA: Naval Postgraduate School. <https://wiki.nps.edu/display/MP/Documentation> (accessed August 8, 2023).
- Bell-Watson, Matt. "What Is Contextual Insight and Why Is It Important?" Priestley. Last modified May 17, 2021. <https://priestley.digital/2021/05/17/what-is-contextual-insight-why-contextual-insight-is-important/>.
- Berger III, Joseph. 2012. "Covert Action: Title 10, Title 50, and the Chain of Command," *JFQ*, no. 67 (4th Quarter): 32–39.

- Borghard, Erica D, and Shawn W Lonergan. 2019. "Cyber Operations as Imperfect Tools of Escalation," *Strategic Studies Quarterly*, 13: 122–45. <https://www.jstor.org/stable/10.2307/26760131>.
- Bremer, Peer-Timo, Tourassi, Georgia, Bethel, Wes, Gaither, Kelly, Pascucci, Valerio, and Xu, Wei. 2022. "Position Papers for the ASCR Workshop on Visualization for Scientific Discovery, Decision-Making, and Communication." OSTI.gov. <https://doi.org/10.2172/1843572>. <https://www.osti.gov/servlets/purl/1843572>.
- Buchanan, Ben. 2016. *The Cybersecurity Dilemma: Hacking, Trust, and Fear Between Nations*. Oxford: Oxford University Press.
- Buchanan, Ben, and Fiona S. Cunningham. 2023. "Preparing the Cyber Battlefield: Assessing a Novel Escalation Risk in a Sino-American Crisis." In *Cyberspace and Instability*, edited by Robert Chesney, 60–98. Edinburgh: Edinburgh University Press. <https://doi.org/10.1515/9781399512510-005>.
- Bykowski, Katie. 2023. "Automated Incident Response: Everything You Need to Know." Low-Code Security Automation & SOAR Platform | Swimlane. Last modified February 8, 2023. <https://swimlane.com/blog/automated-incident-response/>.
- Calder, Alan. 2018. *NIST Cybersecurity Framework: A Pocket Guide*. Ely, Cambridgeshire, UK: IT Governance Publishing.
- Cartwright, James. 2010. "Joint Terminology for Cyberspace Operations." Official memorandum. Washington, DC: Department of Defense. <https://info.publicintelligence.net/DOD-JointCyberTerms.pdf>.
- Choi, Byoung Kyu, and Donghun Kang. 2013. *Modeling and Simulation of Discrete-Event Systems*. Hoboken, NJ: Wiley. <https://doi.org/10.1002/9781118732793>.
- Cybersecurity and Infrastructure Security Agency. 2021. *Federal Government Cybersecurity Incident & Vulnerability Response Playbooks*. Washington, DC: CISA. <https://www.cisa.gov/sites/default/files/2023-02/federal-government-cybersecurity-incident-and-vulnerability-response-playbooks-508c.pdf>.
- Targeted News Service. 2018. "Collaborative Operations in Denied Environment Demonstrates Autonomy and Collaboration with Minimal Human Commands." 2018. <https://targetednews.com/>
- DeVine, Michael E. 2019. *Covert Action and Clandestine Activities of the Intelligence Community: Selected Definitions*. CRS Report No. R45175. Washington, DC: Congressional Research Service. <https://crsreports.congress.gov/product/pdf/R/R45196>.
- Elliott, Stephen N., Thomas R. Kratochwill, and Joan Cook Littlefield. 2000. *Educational Psychology: Effective Teaching, Effective Learning with Free*. 3rd ed. New York, NY: McGraw-Hill Higher Education.

- Few, Stephen. 2006. *Information Dashboard Design: The Effective Visual Communication of Data*. First Edition. O'Reilly Media, Inc.
- Finney, Nathan. 2008. *Human Terrain Team Handbook*. Fort Leavenworth, KS. <https://info.publicintelligence.net/humanterrainhandbook.pdf>.
- Forsberg Kevin J, Walden David D, Hamelin R. Douglas, Roedler Garry J, and Shortell Thomas M. 2015. *Systems Engineering Handbook – A Guide for System Life Cycle Processes and Activities (4th Edition)*. Hoboken, NJ: John Wiley & Sons.
- Geetha, T V, and S. Sendhilkumar. 2023. *Machine Learning: Concepts, Techniques and Applications*. Boca Raton, FL: CRC Press.
- Giammarco, Kristin. 2022. “Exposing and Controlling Emergent Behaviors Using Models with Human Reasoning.” In *Emergent Behavior in System of Systems Engineering*, by Larry B. Rainey and O. Thomas Holland, 1st ed., 23–61. Boca Raton: CRC Press. <https://doi.org/10.1201/9781003160816-4>.
- Graja, Imen, Slim Kallel, and Nawal Guermouche. 2019. “Modelling and Verifying Time-Aware Processes for Cyber-Physical Environments.” *The Institution of Engineering and Technology* 13 (1): 36–48. <https://doi-org.libproxy.nps.edu/10.1049/iet-sen.2018.5034>.
- GREAT. 2015. “A Fanny Equation: ‘I Am Your Father, Stuxnet.’” SECURELIST. February 17, 2015. <https://securelist.com/a-fanny-equation-i-am-your-father-stuxnet/68787/>.
- Hammond, Brian. et al. 2012. *ITU Calls for Public-Private Cooperation in Wake of Discovery of ‘Flame’ Malware*. New York: Aspen Publishers, Inc.
- Holland, John. 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. Cambridge, Massachusetts: MIT Press.
- Hunt, John. 2018. *Functional Programming in Scala. A Beginner’s Guide to Scala, Object Orientation and Functional Programming*. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-75771-1_22.
- Iannotta, Ben. 2022. “The Value of Failing Now and Then.” *Aerospace America* 60 (3): 4–4.
- Integrated Adaptive Cyber Defense. 2018. “Autoimmunity Playbook for Information Brokers: Autoimmunity Analysis of Submitted CTI.” <https://www.iacdautomate.org/playbook-and-workflow-examples>.
- Integrated Adaptive Cyber Defense. 2018. “Fast, Flexible, and Sustainable: Bringing “and” to Cybersecurity.” <https://www.iacdautomate.org/iacd101>.

- Integrated Adaptive Cyber Defense. 2017. “IACD Playbook Thin Specification: A Specification for Defining, Building and Employing Playbooks to Enable Cybersecurity Integration and Automation.” <https://www.iacdautomate.org/intro-to-playbooks-and-workflows>.
- Integrated Adaptive Cyber Defense. 2017. “IACD Playbook White Paper – Content Types.” <https://www.iacdautomate.org/intro-to-playbooks-and-workflows>.
- Integrated Adaptive Cyber Defense. 2017. “IACD Playbook White Paper – Introduction.” <https://www.iacdautomate.org/intro-to-playbooks-and-workflows>.
- Integrated Adaptive Cyber Defense. 2018. “How to Build an IACD Playbook.” <https://www.iacdautomate.org/playbook-and-workflow-examples>.
- Integrated Adaptive Cyber Defense. 2020. “Introduction to Integrated Adaptive Cyber Defense (IACD) Playbooks.” <https://www.iacdautomate.org/playbook-and-workflow-examples>.
- Integrated Adaptive Cyber Defense. 2020. “Operational Best Practices IACD Reference Workflow Template.” <https://www.iacdautomate.org/playbook-and-workflow-examples>.
- Integrated Adaptive Cyber Defense. 2020. “Types of Content within an IACD Playbooks.” <https://www.iacdautomate.org/playbook-and-workflow-examples>.
- Jackson, Daniel. 2006. *Software Abstractions: Logic, Language and Analysis*. Cambridge, Mass: MIT Press.
- Johns Hopkins University Applied Physics Laboratory. 2020. *Shareable Automation and Orchestration Workflows for scoring, sharing, and responding to Cyber Indicators of Compromise*. Laurel, MD: Johns Hopkins University Applied Physics Laboratory. <https://static1.squarespace.com/static/5a94b67ff93fd440f0516297/t/5f33e73f64211d31d004d321/1597237056194/shareable+workflows+for+scoring+and+responding+to+indicators+of+compromise.pdf>.
- Johnson, Chris, Lee Badger, and David Waltermire. 2016. *Guide to Cyber Threat Information Sharing*. Gaithersburg, MD: U.S. Department of Commerce, National Institute of Standards and Technology.
- Johnston, Nathaniel, and Dave Greene. 2022. *Conway’s Game of Life: Mathematics and Construction*. <https://conwaylife.com/book>.
- Joint Chiefs of Staff. 2014. *Information Operations*. JP 3-13. Washington, DC: Joint Chiefs of Staff. https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp3_13.pdf.
- Joint Chiefs of Staff. 2020. *Joint Doctrine for Special Operations*. JP 3-05. Washington, DC: Joint Chiefs of Staff. https://jdeis.js.mil/jdeis/new_pubs/jp3_05.pdf.

- Joint Chiefs of Staff. 2020. *Joint Planning*. JP 5-0. Washington, DC: Joint Chiefs of Staff. https://jdeis.js.mil/jdeis/new_pubs/jp5_0.pdf.
- Kaneko, Osamu. 2008. "On Linear Canonical Controllers Within the Unfalsified Control Framework." *IFAC Proceedings Volumes* 41 (2): 12279–84. <https://doi.org/10.3182/20080706-5-KR-1001.02079>.
- Kauffman, Stuart. 1993. *The Origins of Order Self-Organization and Selection in Evolution*. Oxford: Oxford University Press.
- Koestler, Arthur. 1967. *The Ghost in the Machine*. Macmillan.
- Kohnke, Anne, Ken Sigler, and Dan Shoemaker. 2017. *Implementing Cybersecurity*. Taylor & Francis Group.
- Lin, Herbert. 2012. "Escalation Dynamics and Conflict Termination in Cyberspace," *Strategic Studies Quarterly*, 6 (3): 46–70. <https://www.jstor.org/stable/26267261>.
- Malek, Zakiyabanu S, and Bhushan Trivedi. 2018. "User Behaviour Based Intrusion Detection System Overview." *Engineering Technology*, International Journal for Research, 6 (X): 152.
- McAfee, Andrew, and Erik Brynjolfsson. 2017. *Machine, Platform, Crowd: Harnessing Our Digital Future*. First edition. New York: W.W. Norton & Company.
- Mendling, Jan, and Matthias Weidlich, eds. 2012. *Business Process Model and Notation: 4th International Workshop, BPMN 2012, Vienna, Austria, September 12–13, 2012. Proceedings*. Lecture Notes in Business Information Processing, Vol. 125. Berlin, Heidelberg: Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-642-33155-8>.
- Miller, Greg. 2012. "CIA Is in Baghdad, Kabul for Long Haul: Large Covert Presence Part of U.S. Plan to Exert Power More Surgically." *The Washington Post*, February 7, 2012.
- Mittal, Saurabh, and Larry Rainey. 2015. "Harnessing Emergence: The Control and Design of Emergent Behavior in System of Systems Engineering." In *Proceedings of the Conference on Summer Computer Simulation*, 1–10. Society for Computer Simulation International.
- Nahmias, Steven, and Tava Olsen. 2015. *Production and Operations Analysis: Strategy, Quality, Analytics, Application*. 7th ed. Long Grove, Illinois: Waveland Press, Inc.
- Nielsen, Paul. "Systems Engineering and Software Engineering: Collaborating for the Smart Systems of the Future." *Carnegie Mellon University Software Engineering Institute's Insights* (blog). September 20, 2021. <https://insights.sei.cmu.edu/blog/systems-engineering-and-software-engineering-collaborating-for-the-smart-systems-of-the-future/>.
- Object Management Group. 2011. *Business Process Model and Notation (BPMN) Version 2.0*. Milford, MA. <http://www.omg.org/spec/BPMN/2.0>.

- Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel et al. 2011. “Scikit-Learn: Machine Learning in Python.” *Journal of Machine Learning Research*. <https://doi.org/10.5555/1953048.2078195>.
- Quartuccio, John J., and Kristin M. Giammarco. 2019. *Engineering Emergence*, 1st ed. Routledge. <https://doi.org/10.1201/9781138046412-19>.
- Quartuccio, John J. 2020. “Identification of Behavior Patterns in Systems-of-Systems Architectures.” PhD diss., Naval Postgraduate School. Calhoun (27550).
- Ragan, Ray K, and Timothy Strasser. 2020. “A Credit Union’s Introduction to Behavior-Based Cybersecurity.” *Credit Union Times*.
- Schubert, Johan, Joel Brynielsson, Mattias Nilsson, and Peter Svenmarck. 2018. “Artificial Intelligence for Decision Support in Command and Control Systems.” Paper presented at the 23rd International Command and Control Research & Technology Symposium, Pensacola, FL. https://www.researchgate.net/publication/330638139_Artificial_Intelligence_for_Decision_Support_in_Command_and_Control_Systems
- SendQuick. 2022. “Alert Management: Understanding Alerts, Events, And Incidents.” December 7, 2022. <https://www.sendquick.com/alert-management-understanding-alerts-events-and-incidents/>.
- Seidel, Stephan, Thomas Klotz, Ulrich Donath, and Jurgen Haufe. 2010. “Modelling the Real-Time Behaviour of Machine Controls Using UML Statecharts.” In *2010 IEEE 15th Conference on Emerging Technologies & Factory Automation (ETFA 2010)*, 1–8. Bilbao, Spain: IEEE. <https://doi.org/10.1109/ETFA.2010.5641254>.
- Sheth, Amit, Pramod Anantharam, and Cory Henson. 2013. “Physical-Cyber-Social Computing: An Early 21st Century Approach.” *IEEE Intelligent Systems* 28 (1): 78–82. <https://doi.org/10.1109/MIS.2013.20>.
- Sun, Xingming, Zhaoqing Pan, and Elisa Bertino, eds. 2018. *Cloud Computing and Security: 4th International Conference, ICCCS 2018, Haikou, China, June 8–10, 2018, Revised Selected Papers, Part II: Lecture Notes in Computer Science*. Springer Cham. <https://doi-org.libproxy.nps.edu/10.1007/978-3-030-00009-7>.
- Sutton, Richard, and Andrew Barto. 1998. *Reinforcement Learning an Introduction*. Cambridge, MA: MIT Press.
- Taha, Hamdy. 2017. *Operations Research: An Introduction*. 10th ed. Upper Saddle River, NJ: Prentice Hall.
- Turing, Alan. 1952. “The Chemical Basis of Morphogenesis.” *Biological Sciences*, 237: 37–72. <https://doi.org/10.1098/rstb.1952.0012>.

- Willems, J.C. 1997. "On Interconnections, Control, and Feedback." *IEEE Transactions on Automatic Control* 42 (3): 326–39. <https://doi.org/10.1109/9.557576>.
- Wolfram, Stephen. 2002. *A New Kind of Science*. Champaign, IL: Wolfram Media.
- Wong, Dona M. 2010. *The Wall Street Journal Guide to Information Graphics: The Dos and Don'ts of Presenting Data, Facts, and Figures*. 1st ed. New York: W.W. Norton & Company.
- Zeigler, Bernard P. 2016. "A Note on Promoting Positive Emergence and Managing Negative Emergence in Systems of Systems." *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology* 13 (1): 133–36. <https://doi.org/10.1177/1548512915620580>.
- Zetter, Kim. 2014. *Countdown to Zero Day: Stuxnet and the Launch of the World's First Digital Weapon*. New York: Crown Publishers.
- Zhang, Feng, Shijie Zhou, Zhiguang Qin, and Jinde Liu. 2003. "Honeypot: A Supplemented Active Defense System for Network Security." In *Proceedings of the Fourth International Conference on Parallel and Distributed Computing, Applications and Technologies*, 231–35. <https://doi.org/10.1109/PDCAT.2003.1236295>.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. United States Cyber Command
Ft. Meade, Maryland
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California



DUDLEY KNOX LIBRARY

NAVAL POSTGRADUATE SCHOOL

WWW.NPS.EDU

WHERE SCIENCE MEETS THE ART OF WARFARE