SIMULATION-BASED STUDY ON FALSE ALARMS IN INTRUSION DETECTION SYSTEMS FOR ORGANIZATIONS FACING DUAL PHISHING AND DOS ATTACKS

Jeongkeun Shin[a], L. Richard Carley[a], and Kathleen M. Carley[a b]

[a]Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh PA, United States
jeongkes@andrew.cmu.edu, *lrc@andrew.cmu.edu*
[b]Software and Societal Systems Department, Carnegie Mellon University, Pittsburgh PA, United States
kathleen.carley@cs.cmu.edu

ABSTRACT

Machine learning-based intrusion detection systems (IDS) have attracted considerable attention for their role in proactively identifying intrusion attempts and facilitating swift organizational response. While numerous researchers have conceptually discussed the potential negative impacts of high false alarms in machine learning-based IDS on organizations and have proposed various methods to reduce them, there is a shortage of studies that explore how these false alarms can exacerbate cyberattack damage in different organizational settings and in the face of various cyberattack campaigns. This paper introduces an agent-based modeling and simulation approach to assess false alarm consequences in machine learning-based IDS during dual Denial of Service (DoS) and phishing attacks. The IDS with distinct false positive rates, constructed using the KDD Cup 1999 dataset with diverse machine learning algorithms, were simulated to analyze how these varying false alarm rates affect the extent of damage caused by phishing and DoS attacks.

Keywords: cybersecurity, false alarm, intrusion detection system, human factors, phishing

1 INTRODUCTION

Researchers have employed a variety of machine learning algorithms to create robust intrusion detection systems (IDS) [1]. A well-trained ML-based IDS enables the early identification of intrusion attempts by cybercriminals, empowering organizations to respond promptly and to mitigate potential security breaches. However, ML-based IDS with high false positive rates can be problematic as they generate numerous false alarms, leading security teams to invest valuable time in investigating non-existent threats [2]. Given that security teams' attention is a limited cybersecurity resource within an organization, the time spent addressing false positives from the IDS hinders their ability to effectively mitigate or address other cybersecurity vulnerabilities. This may provide cybercriminals with opportunities to infiltrate the target organization through alternative means. To address this issue, machine learning researchers suggest various methods to reduce the false positives while maintaining high intrusion detection rates [3, 4]. However, there haven't been many studies examining how the high false positive rates of IDS can impact the magnitude of cyberattack damage across different organizational settings, cyberattack scenarios, availability of cybersecurity resources, and defense strategies. In this paper, we present the application of agent-based modeling and simulation methods to assess the effectiveness of a specific ML-based IDS in mitigating cyberattack damage within complex cyberattack scenarios. Through this simulation approach, our goal is to examine the negative impacts of high false positive rates in ML-based IDS when the virtual organization is subjected to a dual DoS and phishing attack. For this study, we conducted training to build multiple IDS using various machine learning algorithms and KDD Cup 1999 dataset [5]. We designed phishing, data exfiltration, ransomware, and

Proc. of the 2024 Annual Simulation Conference (ANNSIM'24), May 20-23, 2024, American University, DC, USA
C. Ruiz-Martin, B. Oakes and R. Cárdenas, eds.

DoS attacks based on MITRE ATT&CK tactics and techniques [6]. Additionally, we replicated a medium-sized organization [7] that experienced a phishing campaign into our simulation framework and modeled the behavior patterns of IT security officers. In this paper, we present two significant contributions to the field of cybersecurity and simulation research. First of all, we bridge empirical insights from studies on human susceptibility to phishing attacks [7] with computer simulations. This interdisciplinary approach allows simulation researchers to model the human element's vulnerability with good fidelity within complex cyber attack scenarios, enhancing the realism and efficacy of simulation models. Secondly, we propose a simulation-based approach to evaluate the potential costs associated with false alarms generated by IDS. By leveraging simulation technology, we demonstrate the capability to quantify the potential damage from false alarms from IDS for specific organizations, taking into account their unique cybersecurity resource constraints.

2 RELATED WORKS

There have been numerous previous studies that have utilized agent-based simulation to model and analyze cyberattack and defense scenarios. Kotenko employed agent-based modeling and simulation techniques to assess computer network security, examining the efficacy of security policies in countering various Distributed Denial of Service (DDoS) attacks [8]. Rajivan et al. employed agent-based modeling to simulate the behaviors of cyber defense analysts, with a specific focus on team collaboration, to analyze cyber defense performance under various collaboration strategies [9]. Kumar and Carley created an agent-based network simulation model to examine the patterns of Internet traffic flow during DDoS attack scenarios [10]. Dobson and Carley introduced the Cyber-FIT framework [11] to simulate the dynamics between attacker and defender teams in cyber warfare scenarios. Their work focused on evaluating the effectiveness of diverse defense strategies employed by military cyber forces against a range of cyberattacks, such as DoS, Phishing, and Routing Protocol Attacks. In Cyber-FIT model, Dobson et al. meticulously modeled cyberattack scenarios using the cyber-kill chain for attacker teams [12] and designed defender teams' cyber situational awareness perception [13]. Carley and Svoboda modeled the organizational landscape but overlooked the digital landscape and cyberattacks [14]. Additionally, while Dobson and Carley modeled the cyber response teams, the organizational impacts were not considered [15]. Addressing this gap, Shin et al. presented the OSIRIS framework [16, 17, 18], which simulates a human organization incorporating realistic behavior patterns of end-user agents and their social relationships. During the simulation, OSIRIS calculates the dynamically changing unique phishing susceptibility of end user agents by considering individual human factors assigned to each agent [19]. Modelers can employ logistic regression models derived from empirical studies that establish the correlation between human factors and phishing susceptibility. This allows OSIRIS to calculate the phishing susceptibility of each end user agent precisely. OSIRIS designed cybercriminal agents to execute diverse cyberattack scenarios, including phishing for data exfiltration [18], ransomware [20], and DoS [16], based on MITRE ATT&CK tactics and techniques [6]. The framework has been employed to evaluate the effectiveness of human firewall defense strategy against phishing attacks [18] and various types of ML-based IDS against DoS attacks [16]. In this paper, we use the OSIRIS framework [16, 18, 19] to model the interaction between the human organization and cyberattack campaigns. Our objective is to observe and assess the negative impact of false positive rates in machine learning-based IDS.

3 CYBERATTACK CAMPAIGN MODEL

In this section, we elucidate the details of cyberattack campaigns simulated within our study. OSIRIS framework provides the cybercriminal agent [16, 18], responsible for executing the cyberattack campaign based on MITRE ATT&CK cyberattack tactics and techniques [6]. The duration of the attack spans 11 days, which corresponds to 15,840 ticks in the simulation environment (1 Tick = 1 Minute). The cyberattack campaign involves two cybercriminal agents. One agent orchestrates five different types of DoS attacks on the server agent within the target virtual organization, aiming to disrupt the organization's services. Simultaneously, the other cybercriminal agent conducts a phishing attack with the objective of exfiltrating and encrypting data on the computing device of the targeted end user agents.

3.1 Denial of Service (DoS) Campaign

The OSIRIS framework offers five distinct types of DoS attacks: Neptune, Smurf, Land, Mailbomb, and Back attack [16]. All of these attack types are included in the KDD Cup 1999 dataset [5]. If the modeler specifies the time interval between each attack before initiating the simulation, the cybercriminal agent will randomly choose one of the five DoS attack types to execute against the target server agent at each interval.

3.2 Phishing Campaign for Data Exfiltration & Ransomware

In prior studies, OSIRIS has modeled and simulated phishing campaigns for both data exfiltration [18] and ransomware attacks [20], leveraging the MITRE ATT&CK tactics and techniques [6]. This paper extends this work by modeling phishing campaigns based on real cyberattack scenarios, drawing insights from Digital Forensics and Incident Response (DFIR) reports that document two distinct incidents [21, 22]. In the first case, cybercriminals employed a phishing attack utilizing a VBA macro to deceive end users within an organization [21]. The objective was to gain unauthorized access, collect data through keylogging techniques, and exfiltrate the obtained data using a command and control (C2) server [21]. In the second case, cybercriminals executed a phishing attack employing a malicious macro embedded in a Microsoft Word document [22]. The goal was to deceive end users into granting access, leading to the encryption of the organization's systems with ransomware [22]. Both reports illustrate the list of MITRE ATT&CK techniques [6] involved in comprehensive cyberattack campaigns [21, 22]. We meticulously identified and incorporated missing techniques, focusing particularly on those related to Reconnaissance and Resource Development tactics. Our approach involved establishing connections between each cybercriminal's cyber operation and the corresponding MITRE ATT&CK techniques [6], thereby modeling the entire cyberattack campaign as a sequence of these techniques. In crafting the 11-day cyberattack campaign, we amalgamated the data exfiltration campaign from the first report [22], which utilized keylogging and C2 server, with the second report's campaign that involved exfiltrating data to the MEGA cloud and encrypting it using ransomware [21]. This consolidated campaign is summarized in Figure 1. Below, we provide a brief description of each day's attack scenario, wherein the cybercriminal agent of OSIRIS follows this sequence of MITRE ATT&CK techniques [6] to conduct the virtual cyber scenario during the simulation. Detailed information on the entire cyberattack scenarios can be referenced in the respective DFIR reports [21, 22].

Day 0 : To make the phishing emails more persuasive, cybercriminal agents gather information about end users' identities **(T1589)** from sources such as social media and other publicly accessible websites **(T1593)**. Additionally, they acquire details about the target organization **(T1591)** and the websites operated by the target organization **(T1594)**. Subsequently, cybercriminals procure various tools **(T1588.002)** and malware **(T1588.001)** for use in their cyberattack campaigns. These include phishing emails, malicious Word documents with macros, C2 servers, MEGA cloud storage, AutoHotkey, Powersploit framework, and RClone.

Day 1 : The cybercriminal agent spreads spear-phishing emails targeted at end user agents within the designated organization **(T1566.001)**. In the event that an end user agent falls victim to the phishing email and opens the attached Word document, malicious scripts are deployed onto their computing device **(T1204.002)**. These scripts, executed through PowerShell **(T1059.001)**, establish persistence by creating scheduled tasks **(T1053)**. Subsequently, the scripts establish a connection to the Command and Control (C2) server **(T1573.001)**.

Day 2 : The cybercriminal agent begins by using the Windows Management Instrumentation Command-Line (WMIC) to gather information about logical disks in a specific location **(T1047)**. Following this, it retrieves details about files and directories **(T1083)**. Subsequently, the cybercriminal agent obtains network configuration information and fetches data about current TCP connections on the system **(T1049)**. Moving forward, the cybercriminal agent collects a comprehensive set of system-related information, including details about the operating system, hardware, installed software, and current users **(T1082 / T1033)**. Finally, it gathers information about the running processes on the system **(T1057)**. The cybercriminal agent proceeds to execute the Powersploit to acquire domain user data **(T1087.002)**. Subsequently, it compresses the collected data into a zip file **(T1560.001)** and exfiltrates it through the command and control (C2) server **(T1041)**. Following the data collection process, the cybercriminal agent takes steps to clean up discovery files, removing any indicators that may reveal its presence **(T1070.004)**.

The cybercriminal agents initiate a series of actions, including retrieving the current system time **(T1124)**, discovering network infrastructure details **(T1016.001)**, enumerating local accounts **(T1087.001)**, gathering information about files and directories in various system locations **(T1083)**, and employing a PowerShell script **(T1059.001)** to extract security account manager information **(T1087.002)**. Subsequently, the cybercriminal agent compresses the gathered data into a CAB file **(T1560.001)** and proceeds to exfiltrate this CAB file through the Command and Control (C2) server **(T1041)**.

Day 3 : The cybercriminal agent gathers data from the local system and consolidates it into a centralized location **(T1074.001)**. Subsequently, it exfiltrates the collected data to MEGA cloud storage using RClone, with a bandwidth limit set to 10 megabytes per second **(T1567.002)**.

Day 4 : The cybercriminal agent establishes a scheduled task named MicrosoftEdgeUpdateTaskMachineUC **(T1053.005)**, designed to run a keylogging script. This keylogging script captures and records the inputs made by the targeted end user agent **(T1056.001)**. The logged keystrokes are then stored in a designated location within the registry **(T1112)**.

Day 5 : The scheduled task named MicrosoftEdgeUpdateTaskMachineUC executes a keylogging script to captures and records the inputs made by the targeted end user agent **(T1056.001)**. The logged keystrokes are then stored in a designated location within the registry **(T1112)**. Then, by using PowerShell **(T1059.001)**, the cybercriminal agent gathers information about the currently running processes on the system **(T1057)**. Additionally, it retrieves the contents of the temporary directory on the system **(T1083)**.

Day 6 : The cybercriminal agent gathers a list of files and directories within the temporary directory, converting the output to a string format **(T1083)**. The cybercriminal agent then compresses the collected data into a CAB file **(T1560.001)** and proceeds to exfiltrate this CAB file through the Command and Control (C2) server **(T1041)**. Subsequently, it utilizes PowerShell **(T1059.001)** to transfer the screen capture script onto the local system **(T1185)**, capturing the screen content of the targeted end user agent's computing device **(T1113)**. Finally, collected data is exfiltrated through the Command and Control (C2) server **(T1041)**. The cybercriminal agent gathers information about the currently logged-in user's username **(T1033)**. Additionally, it checks the status of the Windows Defender service **(T1007)** and retrieves the antivirus and antimalware status on the computer **(T1518.001)**.

Day 7 : The scheduled task named MicrosoftEdgeUpdateTaskMachineUC executes a keylogging script to captures and records the inputs made by the targeted end user agent **(T1056.001)**. The logged keystrokes are then stored in a designated location within the registry **(T1112)**. The cybercriminal agent then compresses the collected data into a CAB file **(T1560.001)** and proceeds to exfiltrate this CAB file through the Command and Control (C2) server **(T1041)**.

Day 8 : The cybercriminal agent gathers data from the local system and consolidates it into a centralized location **(T1074.001)**. Subsequently, it exfiltrates the collected data to MEGA cloud storage using RClone, with a bandwidth limit set to 10 megabytes per second **(T1567.002)**.

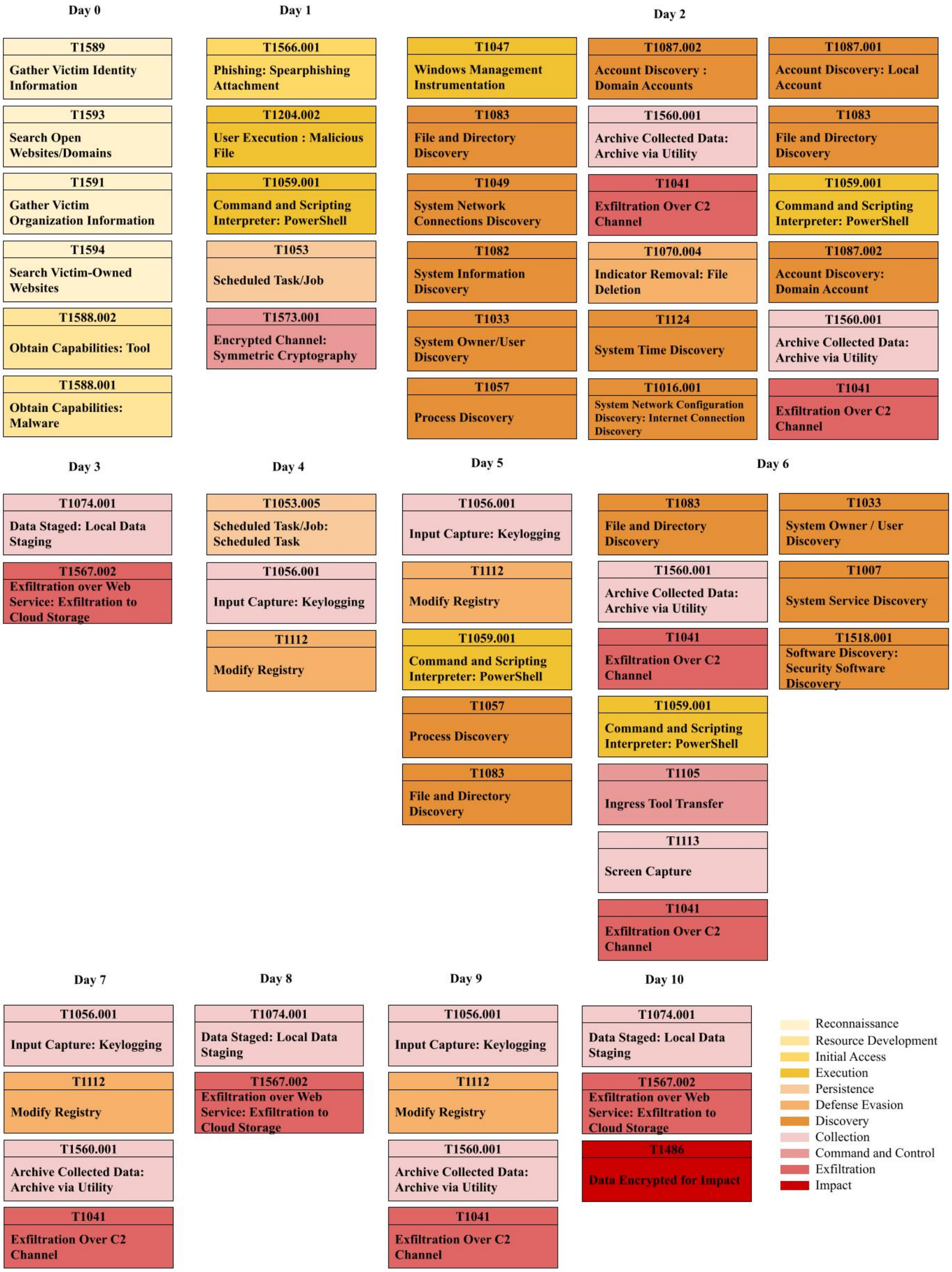

Day 0
T1589
Gather Victim Identity Information
T1593
Search Open Websites/Domains
T1591
Gather Victim Organization Information
T1594
Search Victim-Owned Websites
T1588.002
Obtain Capabilities: Tool
T1588.001
Obtain Capabilities: Malware
Day 1
T1566.001
Phishing: Spearphishing Attachment
T1204.002
User Execution : Malicious File
T1059.001
Command and Scripting Interpreter: PowerShell
T1053
Scheduled Task/Job
T1573.001
Encrypted Channel: Symmetric Cryptography
Day 2
T1047
Windows Management Instrumentation
T1083
File and Directory Discovery
T1049
System Network Connections Discovery
T1082
System Information Discovery
T1033
System Owner/User Discovery
T1057
Process Discovery
T1087.002
Account Discovery : Domain Accounts
T1560.001
Archive Collected Data: Archive via Utility
T1041
Exfiltration Over C2 Channel
T1070.004
Indicator Removal: File Deletion
T1124
System Time Discovery
T1016.001
System Network Configuration Discovery: Internet Connection Discovery
T1087.001
Account Discovery: Local Account
T1083
File and Directory Discovery
T1059.001
Command and Scripting Interpreter: PowerShell
T1087.002
Account Discovery: Domain Account
T1560.001
Archive Collected Data: Archive via Utility
T1041
Exfiltration Over C2 Channel
Day 3
T1074.001
Data Staged: Local Data Staging
T1567.002
Exfiltration over Web Service: Exfiltration to Cloud Storage
Day 4
T1053.005
Scheduled Task/Job: Scheduled Task
T1056.001
Input Capture: Keylogging
T1112
Modify Registry
Day 5
T1056.001
Input Capture: Keylogging
T1112
Modify Registry
T1059.001
Command and Scripting Interpreter: PowerShell
T1057
Process Discovery
T1083
File and Directory Discovery
Day 6
T1083
File and Directory Discovery
T1560.001
Archive Collected Data: Archive via Utility
T1041
Exfiltration Over C2 Channel
T1059.001
Command and Scripting Interpreter: PowerShell
T1105
Ingress Tool Transfer
T1113
Screen Capture
T1041
Exfiltration Over C2 Channel
T1033
System Owner / User Discovery
T1007
System Service Discovery
T1518.001
Software Discovery: Security Software Discovery
Day 7
T1056.001
Input Capture: Keylogging
T1112
Modify Registry
T1560.001
Archive Collected Data: Archive via Utility
T1041
Exfiltration Over C2 Channel
Day 8
T1074.001
Data Staged: Local Data Staging
T1567.002
Exfiltration over Web Service: Exfiltration to Cloud Storage
Day 9
T1056.001
Input Capture: Keylogging
T1112
Modify Registry
T1560.001
Archive Collected Data: Archive via Utility
T1041
Exfiltration Over C2 Channel
Day 10
T1074.001
Data Staged: Local Data Staging
T1567.002
Exfiltration over Web Service: Exfiltration to Cloud Storage
T1486
Data Encrypted for Impact
Reconnaissance
Resource Development
Initial Access
Execution
Persistence
Defense Evasion
Discovery
Collection
Command and Control
Exfiltration
Impact

Figure 1: Phishing attack campaign for data exfiltration and ransomware.

Day 9 : The scheduled task named MicrosoftEdgeUpdateTaskMachineUC executes a keylogging script to captures and records the inputs made by the targeted end user agent **(T1056.001)**. The logged keystrokes are then stored in a designated location within the registry **(T1112)**. The cybercriminal agent then compresses the collected data into a CAB file **(T1560.001)** and proceeds to exfiltrate this CAB file through the Command and Control (C2) server **(T1041)**.

Day 10 : The cybercriminal agent gathers data from the local system and consolidates it into a centralized location **(T1074.001)**. Subsequently, it exfiltrates the collected data to MEGA cloud storage using RClone, with a bandwidth limit set to 10 megabytes per second **(T1567.002)**. Finally, the cybercriminal agent encrypts the files on the end user agent's local system using ransomware **(T1486)**.

4 VIRTUAL ORGANIZATION MODEL

In this section, we demonstrate the modeling of the virtual organization targeted by cybercriminal agents during the simulation. OSIRIS [16, 18] offers the user interface that allows modelers to customize their organization. This involves deploying various end-user agents with unique human factor information, assigning diverse computing devices to each end-user agent, establishing formal or informal social networks among them, and creating connections between server agents and computing devices within the virtual organization [17, 19]. Additionally, various ML-based IDS can be integrated into the server agent to assess their effectiveness against intrusion attempts [16]. IT security agents can also be deployed, with customizable cyberattack defense policies assigned to them.

4.1 End User Agent

Eftimie et al. conducted an empirical phishing simulation involving a software development company with 235 employees [7]. The purpose of this study was to explore the relationship between an individual's susceptibility to phishing attacks and various human factors such as age, gender, and the Big Five personality traits (Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism). This investigation spanned both pre and post-cybersecurity education phases.

In our study, we replicated this software company, consisting of 235 employees, within the OSIRIS framework. This virtual organization became the target of cybercriminal agents during simulated cyberattack campaigns. Unfortunately, Eftimie et al.'s empirical study did not disclose demographic information and Big Five personality scores for each individual [7]. To address this, we utilized age range and gender distribution information, along with mean and standard deviation data for each Big Five Personality trait across the entire employee population. Leveraging this information, we generated virtual age, gender, and Big Five Personality information for each end user agent in OSIRIS using normal distribution.

In the initial two phishing campaigns conducted prior to cybersecurity education in Eftimie et al.'s empirical study, approximately 14.3% of employees fell victim to phishing attempts [7]. Leveraging the Beta values from Eftimie et al.'s logistic regression model [7], which predicts phishing susceptibility based on age, gender, and Big Five Personality scores, OSIRIS autonomously computes the unique phishing susceptibility for each end user agent based on its assigned human factor information (Age, Gender, Big Five Personality Score) [19]. Subsequently, we simulated phishing campaigns within OSIRIS, mirroring Eftimie et al.'s empirical phishing simulation [7]. We calibrated the demographic and Big Five Personality scores of the virtual end user agents until the overall phishing susceptibility matched that of the empirical study. Each end user agent was assigned a personal computer device agent and adhered to an 8-to-5 weekday work schedule during the simulation.

4.2 Server Agent with Intrusion Detection System (IDS)

In OSIRIS, the server agent can be deployed to the virtual organization, generating network traffic data at each tick. This study assumes that the virtual organization's network environment mirrors the setting in which the KDD Cup 1999 dataset [5] was collected. The KDD Cup 1999 dataset [5] was split into two parts, with 80% utilized as the training set and 20% as the test set, following the methodology of prior work in OSIRIS [16].

The training set was employed to construct an IDS using various machine learning algorithms. Meanwhile, the test set serves the purpose of producing network traffic data from the server agent during the simulation [16]. As depicted in Figure 2, in the absence of intrusion attempts on the server agent, it randomly selects an array of data from the test set labeled as 'normal' and generates that data. Conversely, when a specific intrusion attempt occurs, the server agent randomly chooses an array of data from the test set with the corresponding attack label. For instance, in the case of a 'Back' DoS attack on the server agent, it randomly selects an array of data labeled as 'Back' from the test set and generates the corresponding network traffic data.

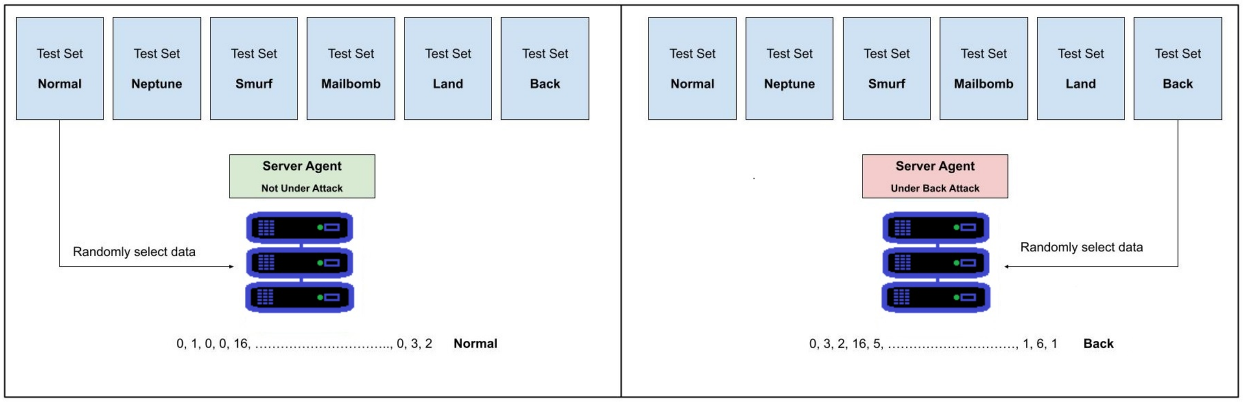


Figure 2: Mechanism for generating network traffic data in the server agent.

We employed Weka software [23, 24] to construct an IDS, leveraging its diverse set of machine learning algorithms. Specifically, we selected the Naive Bayes [25], Naive Bayes Multinomial [26], and Bayes Net [27] algorithms to develop the IDS. The training set was employed to train the model, and its performance was evaluated on the test set. Table 1 presents the performance metrics of each IDS, including the percentage of accurately identified 'Normal' data as 'Normal', the percentage of incorrectly recognizing 'Normal' network traffic data as 'DoS', and the percentage of accurately capturing the five different types of DoS attacks. After integrating the IDS with the server agent, if the IDS detects any DoS attack during simulation, it promptly triggers an alarm to alert the IT security agent. However, IDS cannot monitor phishing attempts targeting individual end user agents' computing devices.

Table 1: IDS performance with Naive Bayes, Naive Bayes Multinomial, and Bayes Net algorithms.

	Normal Accuracy	False Alarm Rate (Normal→DoS)	Neptune Accuracy	Smurf Accuracy	Mailbomb Accuracy	Land Accuracy	Back Accuracy
Naive Bayes	38.8%	22.87%	99.2%	99.9%	97.8%	100%	96.1%
Naive Bayes Multinomial	14.0%	16.14%	99.2%	100%	100%	83.3%	99.4%
Bayes Net	93.7%	0.049%	99.8%	100%	100%	100%	99.7%

4.3 IT Security Officer Agent

In this study, a single IT security officer agent is deployed. The primary responsibility of the IT security officer agent is to conduct periodic inspections of computing devices belonging to end user agents. If any malicious activity is detected during the inspection, the IT security officer agent takes corrective actions and disconnects the access of the cybercriminal agent. Additionally, in the event of a DoS attack alert from the IDS, the IT security officer agent promptly examines the network traffic of the server agent. It verifies the legitimacy of the alarm, a process that typically takes 2 to 4 minutes, and if confirmed as a genuine DoS attack by the cybercriminal agent, initiates measures to mitigate the attack and restore service. If an IT security officer receives an alert from the IDS while inspecting an individual computing device belonging to an end user agent, it will promptly halt the inspection and prioritize checking and mitigating any potential DoS attacks. Considering variations in cybersecurity capabilities among IT security officers and diverse organizational cybersecurity policies, the modeler should set two factors pertaining to the IT security agent before initiating the simulation: 1) the inspection frequency of end user agents' computing devices and 2) the probability of accurately identifying and correcting the malicious activities from end user agents' computing devices. In this paper, we assume that IT security officers spend one hour inspecting each computing device, and any detected malicious activity is promptly corrected within that timeframe.

5 MODEL VALIDATION

In this section, we demonstrate the validation process for both the end user agents' phishing susceptibility model and the network traffic generation mechanism in the server agent. As explained in section 4.1, demographic and personality information for each end user agent is randomly generated, relying on provided statistics from Eftimie et al.'s empirical study [7]. This approach is adopted due to the unavailability of individual human subjects' demographic and personality information. Given the difference between the generated human factor data and the actual human subjects' human factor data, there exists an inherent discrepancy in the phishing susceptibility of each end user agent compared to real human subjects. To address this gap and validate the phishing susceptibility of end user agents, we employ the calibration method [28]. This involves conducting a virtual phishing campaign in the simulation model, calculating the overall phishing susceptibility, and adjusting the human factor information of end user agents. The calibration process iterates until the overall phishing susceptibility of end user agents aligns with the results from the empirical study, indicating a susceptibility rate of 14.3% [7]. The parameter we calibrate is the 'age' range of the end user agents. In Eftimie et al.'s paper [7], human subjects' ages ranged from 21 to 56 years old. The age was the only variable for which only the range is disclosed, while the mean and standard deviation are not provided. In our calibration process, we consider the upper limit of age (56) as an outlier, narrow down the age range for each calibration iteration, and randomly assign the age of one end user agent to 56. Ultimately, the virtual organization, with an age range from 21 to 35 and one outlier at 56, yields the optimal overall phishing susceptibility close to the empirical result after running 100 different virtual phishing campaigns (Mean = 14.22%, SD = 1.93%).

In an empirical study, it was observed that small and medium-sized companies, on average, take 9 minutes to detect a DDoS attack [29]. Given that such companies typically lack an IDS, we adjusted our simulation model to reflect a recognition time range of 7 to 11 minutes for identifying a DoS attack when no IDS is integrated into the server agent [16]. In our simulation model, we conducted 13,629 DoS attacks on a target server without any IDS. On average, it took approximately 9.015 minutes for the organization to recognize the attack, with a standard deviation of 1.415 minutes. This finding closely aligns with the result obtained from the empirical DDoS campaign [29].

As outlined in Section 4.2, the server agent generates simulated network traffic data based on the test set of the KDD 1999 Cup dataset [5]. To validate this synthetic data, we assessed the performance of the same machine learning algorithms on a simulated dataset. A one-year (525,960 ticks) simulation was conducted with a virtual organization exposed to five different types of DoS attacks. Throughout the simulation, we recorded the performance of each IDS on the simulated network traffic data, which is summarized in Table 2. Comparing the performance of the IDS on our simulated data (Table 2) to its performance on the real test set (Table 1), the observed difference is within 1%. This validates that our simulated network traffic data accurately replicates the conditions under which the KDD Cup 1999 dataset [5] was originally collected.

Table 2: IDS performance on simulated network traffic data.

	Normal Accuracy	False Alarm Rate (Normal → DoS)	Neptune Accuracy	Smurf Accuracy	Mailbomb Accuracy	Land Accuracy	Back Accuracy
Naive Bayes	38.81%	23.54%	99.24%	99.93%	97.45%	100%	95.77%
Naive Bayes Multinomial	13.92%	16.52%	99.27%	100%	100%	82.90%	99.04%
Bayes Net	93.68%	0.045%	99.73%	100%	100%	100%	99.71%

6 VIRTUAL EXPERIMENTS

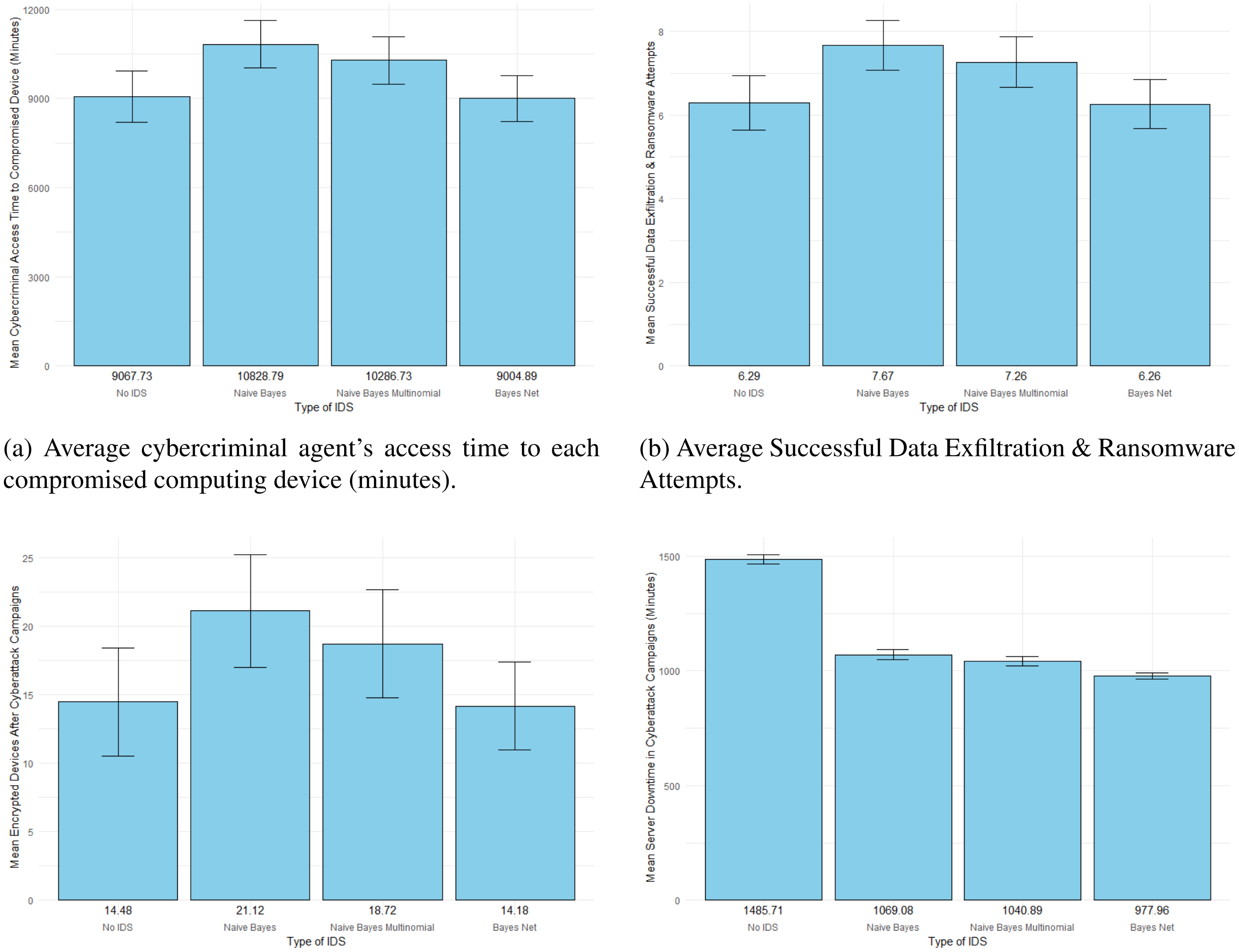
In this section, we describe the design of our virtual experiments, a dual phishing and Denial of Service campaign spanning 11 days (15,840 ticks), with each tick equivalent to 1 minute. We constructed a virtual organization comprising 235 end-user agents with computing device agents, one server agent, and one IT security officer agent, as detailed in Section 4. We assumed that the deployed IT security officer is proficient and possesses the ability to identify and rectify phishing attempts by cybercriminal agents for data exfiltration and ransomware, ensuring a 100% inspection success rate. Two distinct types of cybercriminal agents were deployed, as explained in Section 3. One cybercriminal agent executed phishing attacks for data exfiltration and ransomware, while the other performed one of five DoS attacks every 180 minutes (ticks). For each simulation, we equipped the server agent with one of four different options (No IDS, Naive Bayes IDS, Naive Bayes Multinomial IDS, Bayes Net IDS). When the IDS triggered an alert about a DoS attack, the IT security agent promptly inspects server agent to determine the validity of the alarm and mitigated the attack if it is confirmed. Following empirical study results indicating that small and medium-sized companies take 9 minutes to recognize a DoS attack and 13 minutes to counter DoS attack after detection [29], we set the organization's recognition time to be between 7 and 11 minutes in scenarios where the IDS does not exist or fails to recognize the attack and 13 minutes to counter and recover the system from the DoS attack. We executed 100 simulations for each cell (No IDS, Naive Bayes, Naive Bayes Multinomial, Bayes Net), resulting in a total of 400 simulations. Throughout the simulation, we measured four distinct outcomes:

Average Cybercriminal Agent's Access Time: This metric gauges the duration in minutes that cybercriminal agents maintained their connection to each compromised computing device before losing it following the IT security officer agent's inspection.

Average Successful Data Exfiltration and Ransomware Attempts: As illustrated in Figure 1, there were ten distinct attempts of data exfiltration and ransomware attempts throughout the 11-day cyberattack campaigns. This metric represents the average number of successful attempts for each compromised computing device agent over the 11-day cyberattack campaign.

Average Number of Encrypted Computing Devices: This metric depicts the total number of encrypted computing device agents after the 11-day cyberattack campaign, indicating the number of compromised devices that the IT security agent failed to inspect during the campaign.

Average Server Downtime in Cyberattack Campaigns: This quantifies the duration in minutes during which the server agent was down due to the DoS attack campaign throughout the 11-day cyberattack campaign.



(a) Average cybercriminal agent's access time to each compromised computing device (minutes).

(b) Average Successful Data Exfiltration & Ransomware Attempts.

(c) Average Number of Encrypted Computing Devices After Cyberattack Campaign.

(d) Average Server Downtime in Cyberattack Campaigns (Minutes).

Figure 3: Virtual experiment results.

The simulation results are presented in Figure 3. As illustrated in Figure 3d, the integration of any ML-based IDS (Naive Bayes IDS, Naive Bayes Multinomial IDS, and Bayes Net IDS) into the server agent proves effective in mitigating total server downtime caused by DoS attacks. However, it is discerned from Figures 3a, 3b, and 3c that deploying an IDS built with the Naive Bayes algorithm or Naive Bayes Multinomial algorithm amplifies the damage from the phishing campaign compared to the scenario without an IDS. The severity of damage escalation from phishing follows a hierarchy, with Naive Bayes exhibiting the most significant impact, followed by Naive Bayes Multinomial, while Bayes Net shows minimal influence. Referencing Table 1, the false alarm rates, indicating the misidentification of normal signals as DoS, are 22.87%, 16.14%, and 0.049% for Naive Bayes IDS, Naive Bayes Multinomial IDS, and Bayes Net IDS, respectively. Considering the relationship between false alarm rates and the magnitude of worsening damage from phishing, we can deduce that false alarms generated by the IDS consume valuable time for IT security officials to ascertain the validity of alarms. This time could otherwise be utilized to inspect end-user agents' computing devices, countering phishing attempts for data exfiltration and ransomware attacks. The Cybercriminal agent conducting phishing attacks exploit this wasted time as an opportunity to prolong access time to compromised computing devices (Figure 3a), achieve a more data exfiltration and ransomware attempts (Figure 3b), and accomplish to encrypt a greater number of compromised computing devices (Figure 3c).

However, in the case of Bayes Net IDS, deploying the ML-based IDS with a low false alarm rate leads to a reduction in server downtime caused by DoS attacks without exacerbating the magnitude of damage from phishing campaign. Therefore, based on this simulation study, it becomes apparent that as the false alarm rate of the ML-based IDS increases under the dual Denial of Service and phishing attack scenario, the severity of damage from phishing tends to escalate.

7 DISCUSSION AND CONCLUSION

In this paper, we have demonstrated the adverse impact of deploying an ML-based IDS with a high false alarm rate on the cybersecurity resilience of an organization. Our approach employs agent-based modeling and simulation to examine the intricate dynamics, particularly when faced with human resource constraints for inspecting computing devices and mitigating various cyberattacks. While a high-accuracy IDS can effectively reduce server downtime caused by DoS attacks, the associated high false alarm rate proves to be a double-edged sword. The substantial time consumed by IT security officials in verifying false alarms ultimately results in a trade-off, amplifying the damage from other types of attacks, such as phishing for data exfiltration and ransomware. On the contrary, our findings reveal that deploying an ML-based IDS with an optimized false alarm rate efficiently minimizes server downtime due to DoS attacks. This reduction is achieved without exacerbating the magnitude of damage from other concurrent attacks, specifically phishing attempts.

Our study has several limitations. Firstly, the simulated network traffic data is generated based on the KDD 1999 Cup dataset [5], which is considered outdated and includes some attacks that are not prevalent in current times. Secondly, the simulation outcomes may vary depending on the proficiency level of IT security officers and an organization's policy regarding the frequency of inspecting computing devices. Lastly, our current cyberattack model does not specify which system or human vulnerabilities are exploited by each attack technique during the cyberattack campaign.

Future work should address these limitations. Specifically, incorporating more recent network security and intrusion detection datasets would enhance the model's ability to reflect the contemporary network traffic environment. Additionally, exploring the proposed dual cyberattack scenario with different options for the number of IT security officers, their proficiency levels, and the frequency of computing device inspections would provide insights into how each factor influences the overall magnitude of cyberattack damage. Moreover, supplementing the cyberattack model by illustrating how each attack technique exploits specific vulnerabilities, with reference to resources like MITRE CVE list [30] or the system and human vulnerability list from CASOS technical report [31], would make the model more realistic and comprehensive.

ACKNOWLEDGMENTS

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This research was supported in part by the Minerva Research Initiative under Grant #N00014-21-1-4012 and by the Center for Computational Analysis of Social and Organizational Systems (CASOS) at Carnegie Mellon University. The views and conclusions are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research or the US Government.

REFERENCES

[1] T. Saranya, S. Sridevi, C. Deisy, T. D. Chung, and M. A. Khan, "Performance analysis of machine learning algorithms in intrusion detection system: A review," *Procedia Computer Science*, vol. 171, pp. 1251–1260, 2020.

[2] R. P. Lippmann, D. J. Fried, I. Graf, J. W. Haines, K. R. Kendall, D. McClung, D. Weber, S. E. Webster, D. Wyschogrod, R. K. Cunningham *et al.*, "Evaluating intrusion detection systems: The 1998 darpa off-line intrusion detection evaluation," in *Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00*, vol. 2. IEEE, 2000, pp. 12–26.

[3] G. P. Spathoulas and S. K. Katsikas, "Reducing false positives in intrusion detection systems," *computers & security*, vol. 29, no. 1, pp. 35–44, 2010.

[4] O. Abouabdalla, H. El-Taj, A. Manasrah, and S. Ramadass, "False positive reduction in intrusion detection system: A survey," in *2009 2nd IEEE International Conference on Broadband Network & Multimedia Technology*. IEEE, 2009, pp. 463–466.

[5] S. Stolfo, W. Fan, W. Lee, A. Prodromidis, and P. Chan, "KDD Cup 1999 dataset," *UCI KDD Repository*, 1999, https://archive.ics.uci.edu/dataset/130/kdd+cup+1999+data, accessed 20th January.

[6] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and philosophy," The MITRE Corporation, McLean, VA, Tech. Rep. 10AOH08A-JC, 2018.

[7] S. Eftimie, R. Moinescu, and C. Răcuciu, "Spear-phishing susceptibility stemming from personality traits," *IEEE Access*, vol. 10, pp. 73 548–73 561, 2022.

[8] I. Kotenko, "Agent-based modeling and simulation of cyber-warfare between malefactors and security agents in internet," in *19th European Simulation Multiconference "Simulation in wider Europe*, 2005.

[9] P. Rajivan, M. A. Janssen, and N. J. Cooke, "Agent-based model of a cyber security defense analyst team," in *Proceedings of the human factors and ergonomics society annual meeting*, vol. 57, no. 1. SAGE Publications Sage CA: Los Angeles, CA, 2013, pp. 314–318.

[10] S. Kumar and K. M. Carley, "Simulating DDOS attacks on the us fiber-optics internet infrastructure," in *2017 Winter Simulation Conference (WSC)*. IEEE, 2017, pp. 1228–1239.

[11] G. B. Dobson and K. M. Carley, "Cyber-FIT: an agent-based modelling approach to simulating cyber warfare," in *Social, Cultural, and Behavioral Modeling: 10th International Conference, SBP-BRiMS 2017, Washington, DC, USA, July 5-8, 2017, Proceedings 10*. Springer, 2017, pp. 139–148.

[12] G. Dobson, A. Rege, and K. Carley, "Informing active cyber defence with realistic adversarial behaviour," *Journal of Information Warfare*, vol. 17, no. 2, pp. 16–31, 2018.

[13] G. B. Dobson and K. M. Carley, "A computational model of cyber situational awareness," in *Social, Cultural, and Behavioral Modeling: 11th International Conference, SBP-BRiMS 2018, Washington, DC, USA, July 10-13, 2018, Proceedings 11*. Springer, 2018, pp. 395–400.

[14] K. M. Carley and D. M. Svoboda, "Modeling organizational adaptation as a simulated annealing process," *Sociological methods & research*, vol. 25, no. 1, pp. 138–168, 1996.

[15] G. B. Dobson and K. M. Carley, "Towards agent validation of a military cyber team performance simulation," in *Social, Cultural, and Behavioral Modeling: 13th International Conference, SBP-BRiMS 2020, Washington, DC, USA, October 18–21, 2020, Proceedings 13*. Springer, 2020, pp. 182–191.

[16] J. Shin, L. R. Carley, G. B. Dobson, and K. M. Carley, "Beyond Accuracy: Cybersecurity Resilience Evaluation of Intrusion Detection System against DoS Attacks using Agent-based Simulation," in *2023 Winter Simulation Conference (WSC)*. IEEE, 2023, pp. 118–129.

[17] J. Shin, G. B. Dobson, K. M. Carley, and L. R. Carley, "OSIRIS: Organization Simulation in Response to Intrusion Strategies," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 2022, pp. 134–143.

[18] J. Shin, L. R. Carley, G. B. Dobson, and K. M. Carley, "Modeling and Simulation of the Human Firewall Against Phishing Attacks in Small and Medium-Sized Businesses," in *2023 Annual Modeling and Simulation Conference (ANNSIM)*. IEEE, 2023, pp. 369–380.

[19] J. Shin, K. M. Carley, and L. R. Carley, "Integrating Human Factors into Agent-Based Simu-

lation for Dynamic Phishing Susceptibility," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 2023, pp. 169–178.

[20] J. Shin, L. R. Carley, G. B. Dobson, and K. M. Carley, "Leveraging OSIRIS to simulate real-world ransomware attacks on organization," in *2022 Winter Simulation Conference (WSC) Poster Session*.

[21] "Continuing the bazar ransomware story," Nov 2021. [Online]. Available: https://thedfirreport.com/2021/11/29/continuing-the-bazar-ransomware-story/

[22] "Collect, exfiltrate, sleep, repeat," Feb 2023. [Online]. Available: https://thedfirreport.com/2023/02/06/collect-exfiltrate-sleep-repeat/

[23] G. Holmes, A. Donkin, and I. H. Witten, "WEKA: A Machine Learning Workbench," in *Proceedings of ANZIIS'94-Australian New Zealnd Intelligent Information Systems Conference*. IEEE, 1994, pp. 357–361.

[24] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10–18, 2009.

[25] G. H. John and P. Langley, "Estimating continuous distributions in bayesian classifiers," in *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, 1995, pp. 338–345.

[26] A. McCallum, K. Nigam *et al.*, "A comparison of event models for Naive Bayes text classification," in *AAAI-98 workshop on learning for text categorization*, vol. 752, no. 1. Madison, WI, 1998, pp. 41–48.

[27] G. F. Cooper and E. Herskovits, "A Bayesian method for the induction of probabilistic networks from data," *Machine learning*, vol. 9, pp. 309–347, 1992.

[28] K. M. Carley, "Validating computational models," *CASOS technical report (2017)*, 1996. [Online]. Available: http://www.casos.cs.cmu.edu/publications/papers/CMU-ISR-17-105.pdf

[29] S. Park, "Small and medium companies take average 9 minutes to detect cyberattack: Simulation data," 2021. [Online]. Available: https://www.ajudaily.com/view/20210706153945251

[30] MITRE, "The Common Vulnerabilities and Exposures (CVE) Initiative." [Online]. Available: https://cve.mitre.org/

[31] J. Shin, G. B. Dobson, L. R. Carley, and K. M. Carley, "Revelation of System and Human Vulnerabilities Across MITRE ATT&CK Techniques with Insights from ChatGPT," *CASOS Technical Report (2023)*.

AUTHOR BIOGRAPHIES

JEONGKEUN SHIN is a Ph.D student in the Department of Electrical and Computer Engineering at Carnegie Mellon University in Pittsburgh, Pennsylvania, United States. His research interests includes the agent-based modeling and simulation with cyberattack and defense scenarios and human factors in cybersecurity. His email address is jeongkes@andrew.cmu.edu.

L. RICHARD CARLEY is the Professor of Electrical and Computer Engineering Department at Carnegie Mellon University in Pittsburgh, Pennsylvania, United States. His research interests include analog and RF integrated circuit design in deeply scaled CMOS technologies, and algorithms and methodology for analyzing over-time social media network data. His email address is lrc@andrew.cmu.edu.

KATHLEEN M. CARLEY is a Professor of Societal Computing, Software and Societal Systems Department (S3D), Carnegie Mellon University, Director of the Center for Computational Analysis of Social and Organizational Systems (CASOS), and CEO of Netanomics. Her research blends computer science and social science to address complex real world issues such as social cybersecurity, disinformation, disease contagion, disaster response, and terrorism from a high dimensional network analytic, machine learning, and natural language processing perspective. Her email address is kathleen.carley@cs.cmu.edu.