IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 20, 2022, accepted June 3, 2022, date of publication June 13, 2022, date of current version June 16, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3182383

ICSTASY: An Integrated Cybersecurity Training System for Military Personnel

DONGHWAN LEE[1,2], (Graduate Student Member, IEEE), DONGHWA KIM[1], CHANGWON LEE[1], MYUNG KIL AHN[1], AND WONJUN LEE[2], (Fellow, IEEE)
[1]Cyber/Network Technology Center, Agency for Defense Development, Seoul 05771, Republic of Korea
[2]School of Cybersecurity, Korea University, Seoul 02841, Republic of Korea

Corresponding author: Wonjun Lee (wlee@korea.ac.kr)

This work was supported in part by the Agency of Defense Development, Republic of Korea; and in part by the National Research Foundation (NRF) of Korea Grant by the Korean Government through the Ministry of Science and ICT (MSIT) under Grant 2019R1A2C2088812.

ABSTRACT Cyberwarfare can occur at any moment, anywhere on the planet, and it happens more often than we realize. The new form of warfare is wreaking havoc on not only the military but also on every aspect of our daily lives. Since cybersecurity has only recently established itself as a critical element of the military, the military community relies heavily on the private sector to ensure cyber mission assurance. Given the military's secrecy, such reliance may increase the danger of mission degradation or failure. To address this issue, the military has attempted to build a dedicated cybersecurity training system for the purpose of internalizing cybersecurity training. However, existing cybersecurity training systems frequently lack comprehensive support for effective and efficient cybersecurity training. In this study, we propose ICSTASY, a scenario-based, interactive, and immersive cybersecurity training platform that supports a variety of training features holistically. The primary requirements and design principles required to overcome the challenges inherent in developing a cyber training system were offered based on a review of prior work. Through the demonstration of our prototype, we have proven the feasibility of efficient and truly realistic cyber training, not only for the military environment but also for the private sector.

INDEX TERMS Cybersecurity training, cybersecurity training system, cyber trainer, prototype demonstration.

I. INTRODUCTION

Cybersecurity is indispensable to the attainment of success in military operations nowadays. According to the results of a recent survey of military professionals, over the next five years, cyber attacks will be the greatest concern for the national security enterprise [1]. One of the biggest challenges lying ahead of us is a dearth of the forces capable of repelling enemy attacks. There are highly trained, intellectual criminals behind cyber attacks whereas defense's human resource pool is limited and heavily relies on automated devices for the majority of their defensive activities. Due to these constraints, the defensive operations in cyberwarfare can barely protect critical assets, and other actions such as backtracking and identifying threat actors are practically impossible. Many organizations including the military have attempted to address this issue by introducing a cybersecurity training program that is specifically tailored to train and educate their defense forces.

Cybersecurity training is a critical prerequisite to the military being fully cyberized. The military has attempted to build its own specialized cybersecurity training system, or the cyber range. SIMTEX (Simulator Training Exercise Network) [2] is one of the earliest examples of military-developed cybersecurity training systems. SIMTEX was designed initially for the US Air Force's training purposes and later selected as the operational platform for US military-hosted cybersecurity exercises such as Black Daemon and Cyber Flag. SIMTEX offered virtualized hosts to simulate the information assets targeted by cyber attacks and a VPN tunnel to isolate attack flow across remote sites for the exercises.

CAAJED (Cyber And Air Joint Effects Demonstration) [3] is another USAF effort that combines a commercial wargame simulator called MAP (Modern Air Power) and a cyber simulation model called SECOT (Simulated Enterprise for

The associate editor coordinating the review of this manuscript and approving it for publication was Laxmisha Rai.

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

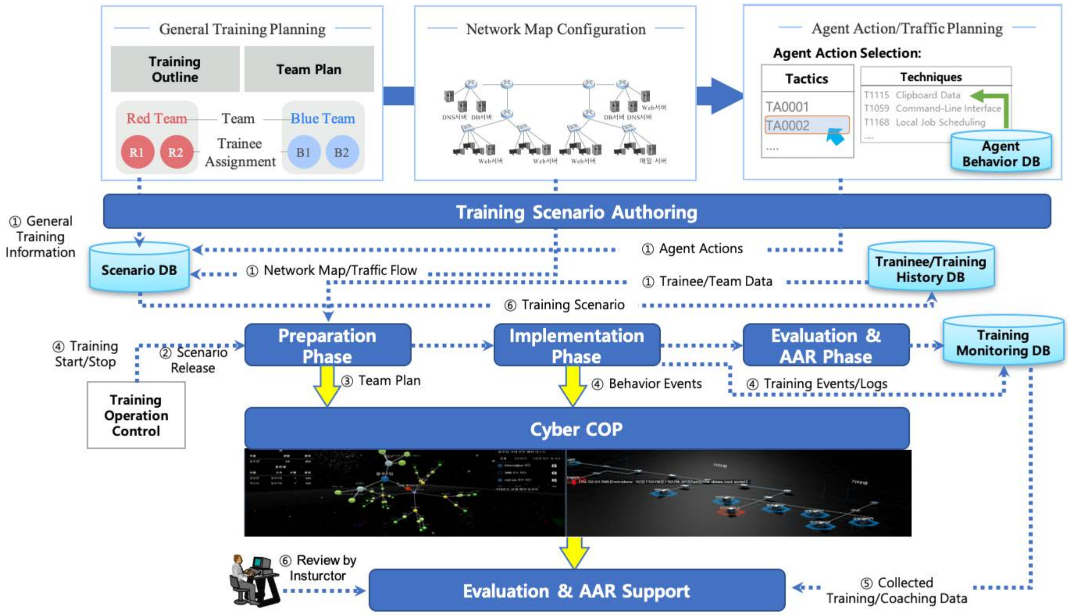


FIGURE 1. Operational concept and procedure of ICSTASY.

Cyber Operations Training). CAAJED was utilized in Cyber Defense Exercise 2007 (CDX 2007), a cyber exercise based on capture-the-flag tactics. During the exercise, red team members conduct simulated cyber attacks, and once the attacks are successful, SECOT calculates the cyber attack's impact on mission performance in the kinetic world.

SAST (Security Assessment Simulation Toolkit) [4] was developed by Pacific Northwest National Laboratory to provide high-level, specialized training to USAF CNO personnel. SAST provides an isolated network that simulates a large network under cyber attack. Additionally, SAST comprises MUTT, a Multi-User Training Tool that generates millions of simulated users to simulate realistic background traffic, and CAT, a Coordinated Attack Tool that incorporates cyber attacks into simulations.

StealthNet [5] is a US Army-funded LVC (Live-Virtual-Constructive) platform for cyber-related testing, evaluation, and training on the Army's tactical networks. StealthNet contains emulation models of cyber attacks such as jamming, DDoS, and worm propagation to determine the impact of cyber threats on tactical networks. StealthNet, in particular, provides simulation models for wireless tactical networks, enabling LVC co-simulation for cybersecurity training in tactical networks.

However, military-developed cybersecurity training programs are constrained in two ways: To begin, they were created for large-scale, short-term exercises such as capture-the-flag drills or cyber wargames. While these events may be beneficial for strengthening capabilities for existing cybersecurity responsibilities, they are detrimental in developing the highly qualified individuals required in the long run. Second, they frequently lack the capabilities necessary for training or are overly focused on tactical applications. These restrictive, overly-specific cybersecurity training systems have hampered the development of a realistic and effective cybersecurity training process.

Meanwhile, in the private sector, many research efforts have been conducted in recent days to build more comprehensive training systems that would address more fundamental, long-term cybersecurity training demands. These modern cybersecurity training systems include graphical user interfaces for configuring the training environment, autonomous red/blue team agents, and automated scoring. However, private-developed cybersecurity trainers have limitations in that such features are not fully integrated, limiting the ability to provide comprehensive, practical cybersecurity training. To overcome these shortcomings, we present a novel cybersecurity training platform, ICSTASY, in this work. ICSTASY provides holistic support for a variety of training capabilities.

To ascertain prerequisites and capability gaps and to develop a blueprint concept for a fully integrated cybersecurity training system, we begin by formulating the desired training system's operational concept. ICSTASY's operational concept and procedure are depicted in Fig. 1. To begin, the initial (preparation) phase defines all of the preliminary information for cybersecurity training, such as a team plan, network map configuration, and agent actions. The following step (implementation) manages a training session, via which trainees interact with the system and associated tasks such as user/agent behavior monitoring and progress/situation visualization. The last (evaluation & AAR) phase of the system facilitates the instructor's assessment and After-Action-Review (AAR) activities by consolidating training logs into trainees' scores and providing reports and replays of completed sessions. The considerations identified with the operational concept are condensed to the ICSTASY design requirements, which we will use to demonstrate that our prototype is developed in accordance with our initial goals and conceptions throughout the development process. The contribution of our study is as follows:

- The operating concept and procedure were suggested to develop a fully integrated cybersecurity training platform for the military environment that requires more discreet but realistic and comprehensive cybersecurity training.
- We defined requirements and specifications and presented a system architecture that enables the implementation of the operating concept and procedure.
- Finally, we built a prototype of the desired platform, ICSTASY, and demonstrated its capabilities of accommodating the numerous features necessary to deliver effective and realistic cyber training dedicated to (but not limited to) the military.

The rest of this paper is arranged as follows: Section II introduces related studies. Section III provides the design concepts and requirements for ICSTASY, and Section IV elaborates on the overall architecture and system design of ICSTASY by expending on the previously stated design principles and requirements. Section V illustrates the development process through several, detailed screenshots of ICSTASY and compares our cybersecurity training system to others. The concluding section recaps and summarizes this paper.

II. RELATED WORK

As with the military, there is little completed research on integrated cybersecurity training systems in the private sector, including academia; nonetheless, there is some notable work on each technological part of cybersecurity training systems. This section will highlight some of the essential work proposed in the private sector.

CyRIS [6] is a cyber range instantiation system developed by JAIST in which KVM-based virtual hosts are set up and created automatically following a script-based scenario file. Additionally, the scenario script specifies the types of emulated attacks to be executed and the target nodes. Their latest cybersecurity training system, CyTrOne [7], includes these technologies.

Nautilus [8] devised its own script language called SDL (Scenario Description Language) for automating the deployment and configuration of a virtualization-based cybersecurity training environment. As with CyRIS, SDL specifies virtual hosts and network configurations for a training environment, but instead of emulated attacks, it defines vulnerabilities embedded in hosts. A CVE (Common Vulnerability Enumerator) code [9] identifies each vulnerability and, based on a predetermined script, automatically plants it upon the instantiation of the vulnerable host.

ASL (Attack Specification Language) [10] provides an integrated representation of cyber threat scenarios for cybersecurity trainers. Considering the dynamic nature of the cyber threat scenarios, ASL is built with the innate feature to deduce the most advantageous attack technique given the conditions using machine learning based inference. Taking a step forward, GHOSTS [11] introduced the concept of a Non Player Character (NPC) into cyber training systems, which aims to emulate the hostile behaviors of an enemy and the benign activities of regular users.

CybOrg [12] is a cyber gym platform dedicated to the training of autonomous agents. The platform is built on a commercial cloud platform, AWS, and intends to provide a repeating training environment for autonomous agents to practice cyber attack and defense techniques using reinforcement learning. Each repeat uses a YAML-based script to duplicate and diversify the episodes given to the agents. Agents trained in this manner get deployed as red and blue team agents that face off against trainees.

However, the linked work discussed above concentrated on specific technological aspects rather than proposing a comprehensive platform. The Swedish research agency FOI launched CRATE (Cyber Range and Training Environment), a pioneering cybersecurity training platform [13], [14]. In contrast to the other previous effort, CRATE's objective was to create an integrated cybersecurity training platform by combining the fragmented technology elements. For example, CRATE's NodeAgent and Core API services facilitate the configuration and deployment of virtualized hosts and networks. Its CRATE Exercise Control (CEC) platform enables situational monitoring and evaluation of cybersecurity training [15]. Additionally, SVED (Scanning, Vulnerabilities, Exploits, and Detection) identifies vulnerabilities in a training environment and assists automated red-team agents with attack planning. [16]. Although a significant portion of the features rely on commercial off-the-shelf software such as OpenVAS [17], snort [18], or TCPdump and thus provide only partial, limited capabilities, CRATE retains meaning as the initial attempt to integrate the technology elements of a cybersecurity training system.

KYPO [19] is another notable study that takes into account the exhaustive design principles of a cybersecurity training system. KYPO acknowledges the importance of real-time monitoring and evaluation (so-called post-mortem analysis) by suggesting a highly fine-grained log production and collection architecture.

III. DESIGN CONCEPTS AND REQUIREMENTS

A detailed assessment of existing cyber trainers showed that most of their systems could not handle the inclusion of additional elements needed for full-fledged cybersecurity training. Our novel training system addresses these limitations by being developed according to the standard V model, i.e., based on identified capability gaps. We first developed a set of principles and requirements to consider when designing a novel cybersecurity training system. The system design, implementation, and evaluation follow in due order.

A. EDITABLE AND REUSABLE SCENARIO WITH TEMPLATES

A scenario is a critical component of any cybersecurity training system. A training session is essentially a reproduction of a training scenario, and a robust cyber training system is one with rich scenarios. However, many cyber trainers supply scenarios as a bundled package which usually does not allow instructors to change the scenarios. This precludes the trainer from diversifying training scenarios and providing variance within a single training session. As a result, a novel cyber training system must enable an editable and reusable scenario. To ensure that this design principle is adhered to, we propose the following requirements.

1) A scenario should contain all of the elements necessary to conduct a training session, such as host and network configuration, agent behavior schedule, expected user events, and other relevant information.
2) A scenario should be exportable and re-importable as an editable script or markup language.
3) A scenario should have a layered structure with numerous layers, enabling the training system and scenario editor to access and locate required data.

B. AUTONOMOUS OPPONENT FOR TRAINEE INTERACTION

In a typical cyber training system, interactive experiences are confined to engagement with a human opponent or to unidirectional activities outlined in a script file [20]. This prevents trainees from encountering a variety of situations that can arise in cyberspace. More intelligent *agents* capable of interacting with learners in a training environment is necessary to offer as many situations as possible. Additionally, trainees can benefit from a variety of cybersecurity experiences if an autonomous blue team and an autonomous red team are available, which is the only form of team permitted in the majority of conventional cyber trainers. In summary, we propose the following requirements for this design principle.

1) Autonomous agents capable of varying their behavior in response to changing variables in the training environment should be provided.
2) A user should be able to plan and edit the essential actions of agents throughout the scenario authoring process.
3) Agents from either the red or blue teams should be able to be chosen for an autonomous opponent team.

C. FULL VISIBILITY INTO TRAINING SESSIONS

Because comprehensive situation awareness in cyberspace has been one of the most significant issues in the cybersecurity area, a question-and-answer-based test for trainees was an indirect method used to provide visibility into the training environment. We can promote productive interactions between a trainee and their instructor if we can automatically recognize and notify the instructor about the trainees' behavior. This enables an instructor to adjust their teaching methods while keeping a close eye on the trainee's progress. The following requirements would provide complete visibility into training sessions.

1) The training system should recognize and collect all potential events associated with trainee activity into raw logs, which are the system's most granular logs.
2) The expected training event for a training session should be definable during the scenario creation phase using the logical operations of the raw log events.
3) The training system should notify the instructor immediately upon detecting expected training events, using a visually effective method such as a dashboard and/or a Common Operational Picture (COP) for the cyber training environment.

D. AUTOMATED EVALUATION AND AFTER-ACTION-REVIEW

As indicated previously, a question-and-answer-based test has typically been the primary approach for enabling visibility into the training environment. For instance, if a trainee responds with a string only obtained through successive privilege escalation, it implies the trainee successfully executed the privilege escalation. The evaluation process is identical in the majority of conventional cyber trainers. However, if we can automatically detect and analyze trainees' behaviors, we will be able to evaluate trainee behavior as well. The following requirements are needed to substantiate our cyber training system's automatic evaluation and AAR features.

1) The trainee behavior events that occur during a training session should be recorded in a database to be utilized post-session by the evaluation and AAR features.
2) The training system should provide an interface via which an instructor can assign scores to observed behaviors ahead of time so that the score is automatically attributed when the trainee exhibits that expected behavior.
3) When the system detects important behaviors, the main screens, such as the dashboard/COP screen and the trainee's screens, should be captured as screencasts for the AAR phase debriefing.

IV. OVERALL ARCHITECTURE AND SYSTEM DESIGN

This section presents the proposed training system's overall architecture and system design based on the principles and requirements described in the preceding section. We begin by proposing the system's overall architecture, followed by a description of the system's design in three primary components.

A. OVERALL ARCHITECTURE

To begin, we determine the operational procedures for our training system to create a design for the overall architecture.

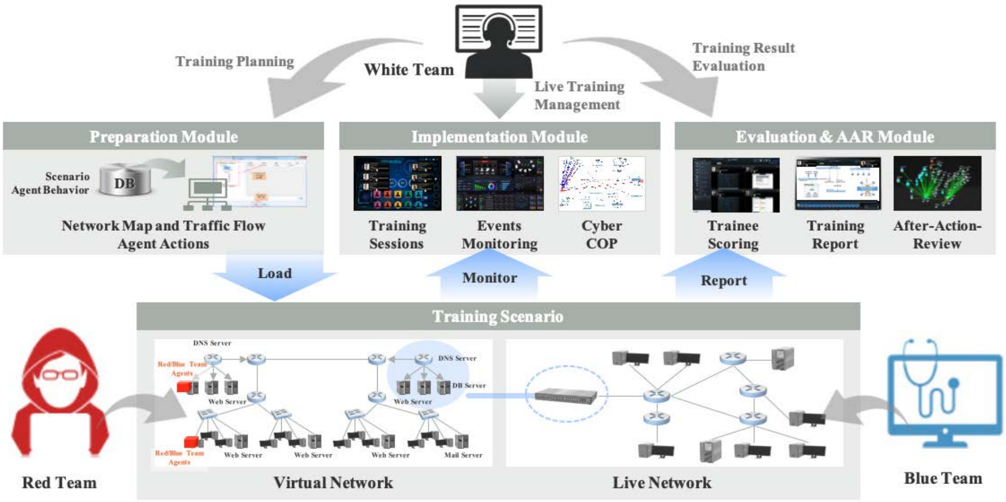


FIGURE 2. System architecture and modules of the ICSTASY prototype.

When considering the cybersecurity training system's use cases, the key user is the instructor. They create the scenario required for a training session, conduct the session, and lead and evaluate trainees. These tasks may be accomplished collaboratively by members of several teams, such as white, yellow, and green. Unless otherwise specified, we refer to a user who can participate in any of these teams as an instructor. As briefly mentioned above, the ICSTASY operational procedure has three phases, mainly from the instructor's perspective:

1) The preparation phase: contains activities such as scenario creation, network map/virtual machine configuration, agent behavior design, and training session management.
2) The implementation phase: includes initiating, managing, and terminating a training session. Automated agents, live monitoring, and coaching activities are performed throughout the session.
3) The evaluation and AAR phase: the final stage of training, during which an instructor can assess trainees' progress and advise them based on the information acquired throughout the assessment.

Given the operational procedure stated above, ICSTASY has three modules that correspond to the three phases: the preparation, implementation, and evaluation & AAR modules. As shown in Fig. 2, each module performs the functionality necessary for each operational phase.

B. MODULE-WISE FEATURES AND SYSTEM DESIGN

This section details the features and specifications of each module focused on meeting the aforementioned significant requirements.[1]

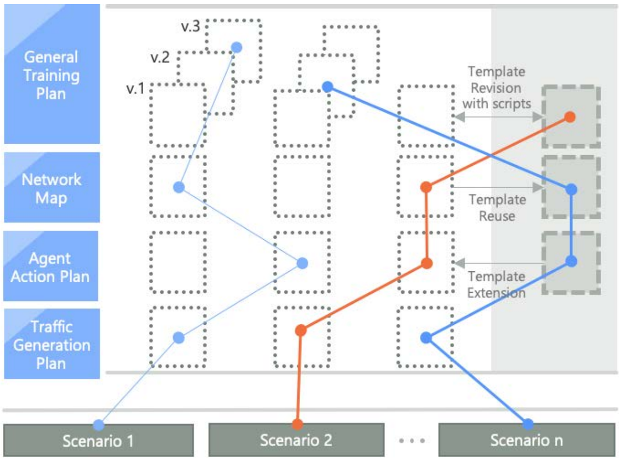


FIGURE 3. Layered structure of the ICSTASY training scenario.

1) PREPARATION MODULE: SCENARIO AUTHORING/MANAGEMENT

First of all, we divided the scenario authoring process into multiple steps to accommodate the multi-layered structure of a training scenario (Requirements III-A-1 and III-A-3): 1) Defining general training concepts and organizing teams 2) Configuring network map/virtual hosts; 3) Scheduling the actions of agents and listing expected events. Each step intends to aid instructor teams in their preliminary work. For instance, in the second step, ICSTASY provides a drag-and-drop UI that enables the instructor to easily and efficiently build virtualized infrastructure, which is distinct green team work. In this manner, based on the objectives of training, the instructor can readily deploy security appliances: from virtualizable IDS/IPS/firewalls like pfSense, snort, Suricata, and Bro to any hardware-type appliances supporting IP networks such as firewalls, IDS/IPSes, and the anti-DDoS and anti-spam devices. The third permits instructors to more easily

[1] The requirements are referred to by their section and item numbers, for example, III-A-1 refers to the first requirement in Section III.A.

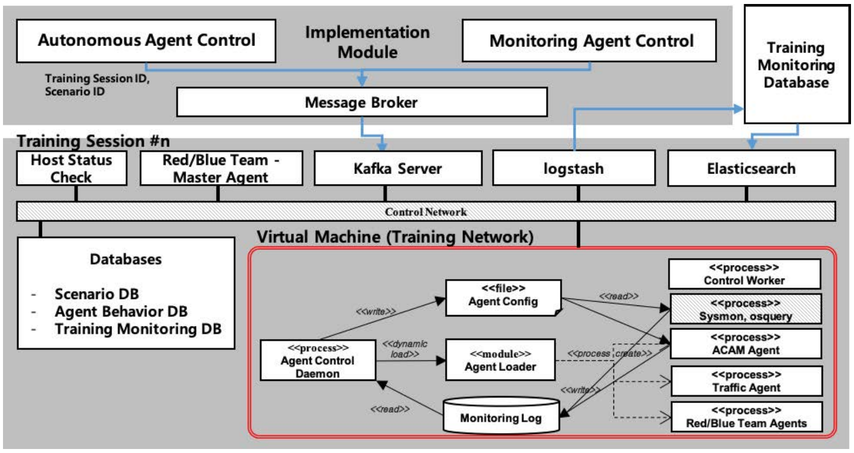


FIGURE 4. Structure and data flow of the ICSTASY log collection.

monitor and evaluate trainees' actions and performance, which may be related to the work of a yellow or white team.

To meet the scenario's requirements (Requirements III-A-2, III-C-2, and III-D-2), we designed a scenario as an editable XML file containing gathered data from each step in each layer. A proficient instructor may quickly locate and edit specific sections of a scenario file, resulting in a more sophisticated scenario than one prepared via the GUI. Fig. 3 illustrates a scenario file reflecting the layered structure of the ICSTASY training scenario. Scenarios are saved as templates in each layer, enabling scenario reuse and editing on a template-by-template basis.

We added a session management step before the implementation phase, in addition to the scenario authoring activity. An instructor must complete this step by creating a session where a selected scenario will be loaded. This enables us to build several sessions from a single scenario and easily manage temporal data such as trainee logs.

2) PREPARATION MODULE: AGENT ACTION PLANNING

The agent action planning feature is one of the most prominent features of the preparation module. The agents' actions are produced and structured automatically by specifying a few parameters, such as the starting and ending points of attack (Requirement III-B-2). Each activity of a red team agent is associated with a Technique Instance (TI), which is defined as an instantiated technique in MITRE's ATT&CK framework [21]. Each TI has pre-and-post conditions that allow us to simulate the attack path before training and pre-determine the agent's availability. Our prior work [22] and [23] have the particular automation strategies upon which our red team and blue team agents were built, respectively. Thus, an instructor can assign agents alternative roles and courses of action according to the training objective, giving a high level of diversity and flexibility for an advanced cyber training experience (Requirements III-B-1 and III-B-3). It eliminates the need for a costly white/green team and allows for the potential of a one-person white team, whereas many existing cyber trainers rely on pre-determined, immutable agent activities. Appendix contains the complete list of TIs included in ICSTASY.

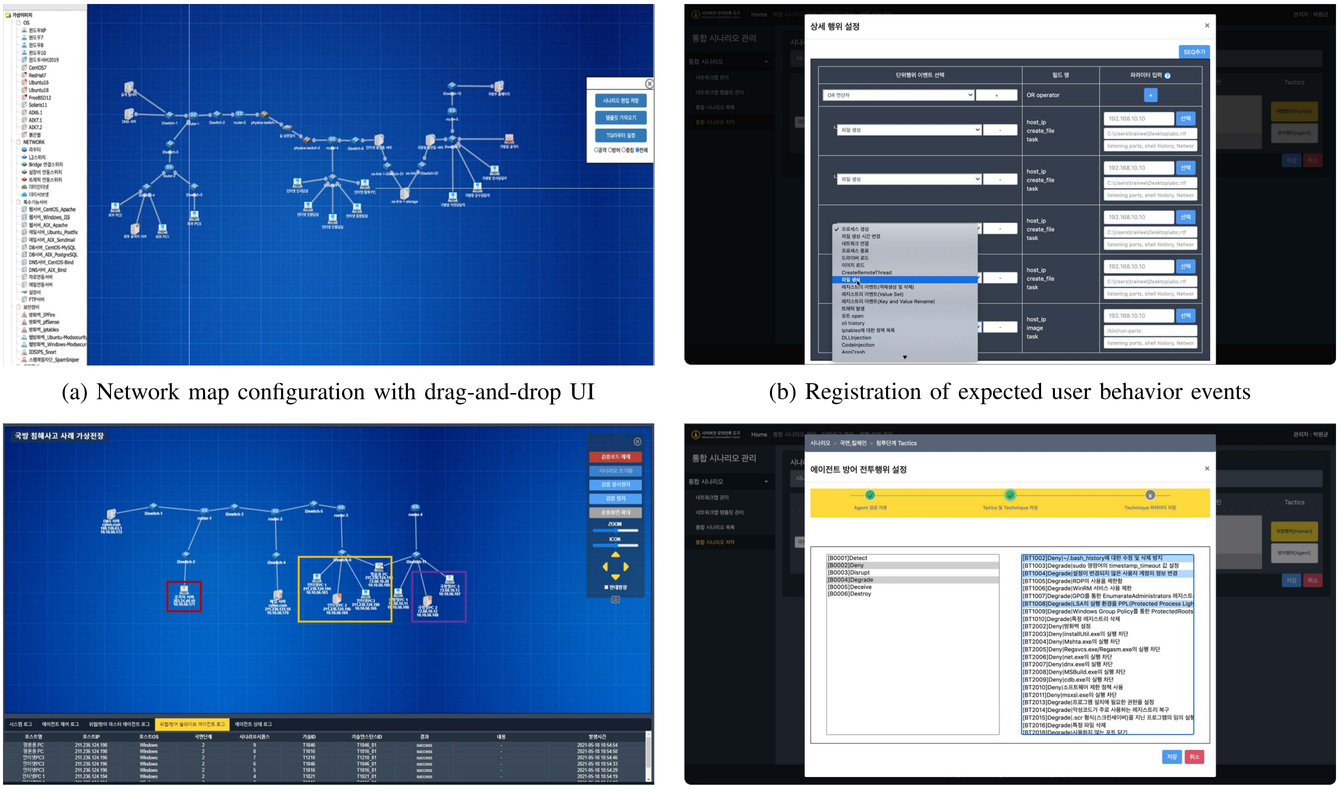
In addition to the autonomous agents, we enabled ICSTASY to imitate background and network traffic using pre-stored pcap files [24]. The preparation phase allows for the configuration of source-and-destination pairs for traffic creation. The source and destination can be hosts in a simulation or a real-world environment and in [25], indicating that our training system is LVC interoperation ready.

3) IMPLEMENTATION MODULE

The implementation module includes a variety of features for managing live training sessions. The implementation module's primary feature is session management, which enables an instructor to control the flow of a training session via an initiation/pause/termination interface. A notable feature of ICSTASY is the ability to pause a session, which is rarely available in other cyber trainers. This capability suspends all system activities, including the log collection for the session and all associated virtual machines. We integrated VMware vShpere API [26] and IBM PowerVC API [27] into ICSTASY connecting the session management feature to the backend that manages all the virtual machines.

Another critical feature included in the implementation module is the ability to visualize training sessions. The visualization function provides a visual representation of the session statuses and enables event-driven monitoring of learner behavior. The implementation module utilizes Logstash [28] to capture all data associated with a training session, accumulating it in the monitoring database (Requirements III-C-1 and III-D-1). Elasticsearch is used to retrieve and analyze the stored logs [29].

The key concept behind the visualization feature is a *behavior event*, which is pre-defined metadata during the preparation process. It provides expected user/agent behaviors during a training session to meet the objectives. In this manner, we may focus our search on a subset of the massive



(a) Network map configuration with drag-and-drop UI

(b) Registration of expected user behavior events

(c) Action/attack path planning for a red team agent

(d) Action planning for a blue team agent

FIGURE 5. Demonstrative screenshots of the preparation phase.

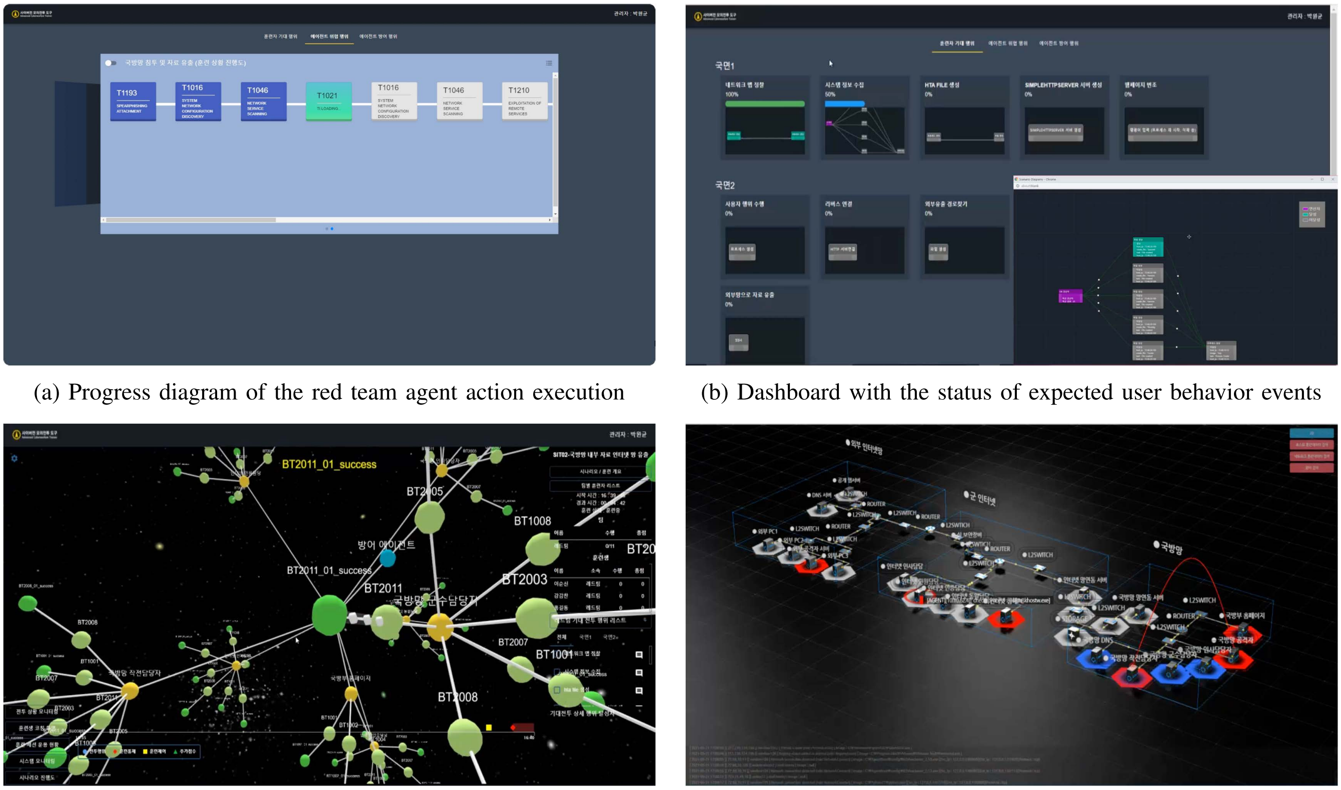
amount of logs. On the other side, we ensured that we collected as many fine-grained logs as possible from hosts and networks, referred to as an emphatomic event. In addition to the usual IDS-based detection method, live forensic/EDR (End-Host Response)-based techniques were incorporated. For instance, Microsoft's Sysinternals Suite [30] and Facebook's OSquery [31], as well as a self-developed mini-filter driver named ACAM (Advanced Cyber Activity Monitoring), collected a large amount of host-related data. These are deployed in each host, enabling highly sensitive detection of kernel level changes such as privilege escalation, driver loading/unloading, process crashes, and DLL/code injections. The atomic logs are stratified, so at least one comprises an *interim event*, and one or more interim events constitute a behavior event at the highest level. For visualization purposes, Fig. 4 illustrates the hierarchical structure of the log collection. The resulting behavioral events appear in the form of a cyber COP (Common Operational Picture) (Requirement III-C-3). ICSTASY can efficiently and precisely detect trainee/agent behaviors using this approach, whereas the majority of existing trainers rely on the instructor's skill.

Visualization also requires the log collection of autonomous agents. Once the training session initiates, the implementation module triggers automated agents to begin trace the pre-programmed path planned in the preparation phase. Unlike human behaviors, agents report agent behaviors and hence do not require the module to gather granular logs and detect them. After determining the success or failure of an action, an agent generates a behavior log.

The module's final feature is live coaching, which provides an engaging experience for trainees and increases training efficiency. To avoid possible intrusion into the training world while conducting coaching activities, we isolated the training network from the 10 GbE network. All data not used for training, including atomic logs for visualization, flows up through this network. The coach can monitor each trainee's shared screen guide any trainees using the live coaching feature.

4) EVALUATION AND AAR MODULE

The module for evaluation and AAR relies heavily on the implementation module. From a design standpoint, the evaluation and AAR module can be defined as an implementation module that uses archived data rather than real-time data. After a training session, the instructor can playback recorded COP and trainee screens and examine saved behavior events and other records (Requirements III-D-2 and III-D-3). To facilitate evaluation and AAR, we first built a central time server and timestamped all visualization and coaching data collected during a training session. This simplifies data synchronization and enables instructors to



(a) Progress diagram of the red team agent action execution

(b) Dashboard with the status of expected user behavior events

(c) Cyber COP in the force-directed graph expression

(d) Cyber COP in the network map expression

FIGURE 6. **Demonstrative screenshots of the implementation phase.**

navigate trainees' training records by moving around a single timeline. Second, we assured that the module visualized COP using the latest web standards, including CSS3, and that the visualization data was stored after time stamping. As a result, a more informative and lightweight COP-centric replay feature emerged, especially when compared to video recording techniques that consume considerable system resources. Thus, an instructor can review the training situation collectively and conveniently for the chosen time period without losing any knowledge.

V. DEVELOPMENT RESULTS

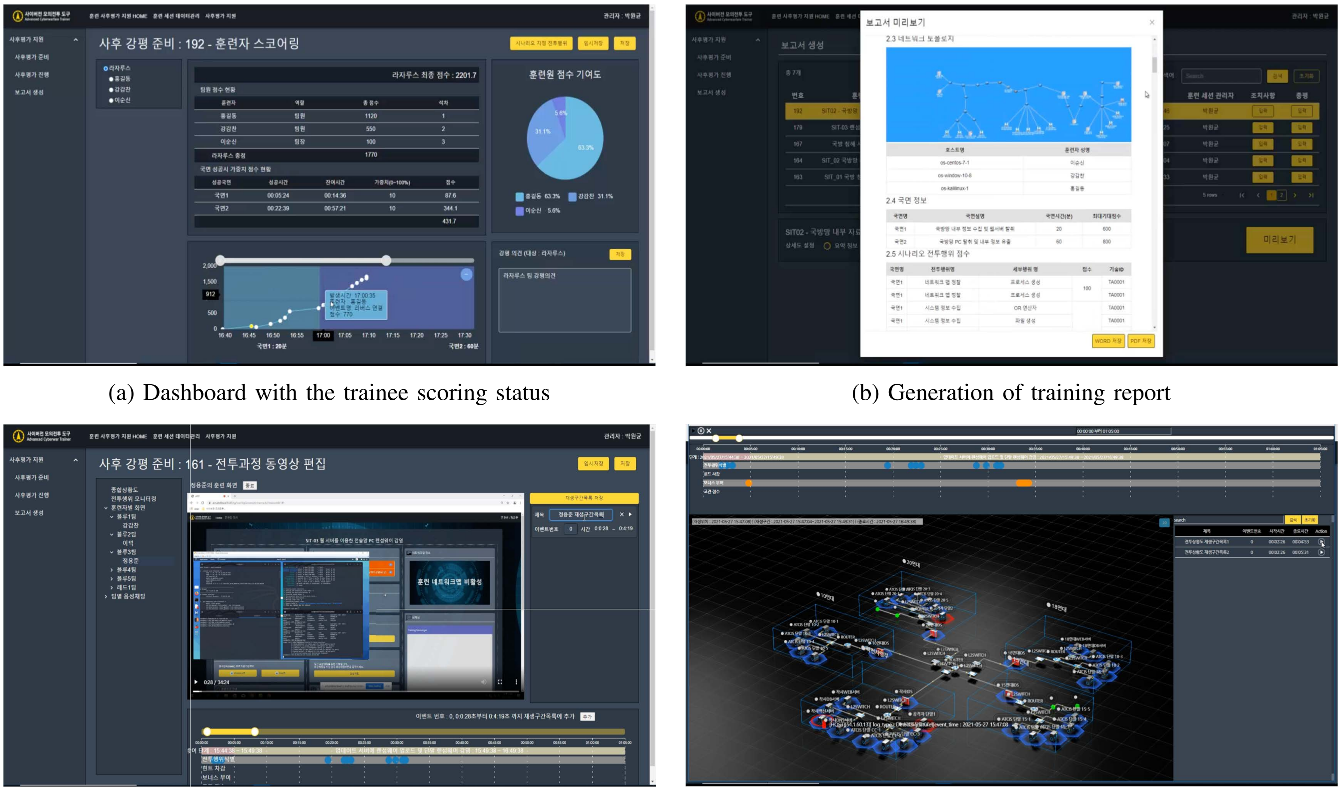
We developed a prototype of ICSTASY based on the module design described previously to illustrate the feasibility and usability of an advanced, immersive cyber training experience. We cannot give detailed training situations due to the possibility of disclosing confidential information; nonetheless, we have attempted to provide as many different screenshots as possible to understand our training system.

A. PREPARATION PHASE

Given that the preparation phase is the most labor-intensive, we placed focus on the instructor interface during the development process. As specified in the module design, drag-and-drop-based UI for configuring the network map/virtual hosts was implemented. Refer to Fig. 5a for a screenshot of the network map configuration tool. An instructor can drag and drop the desired host template from the tool's left side panel to the tool's main panel. The host is then instantiated, allowing the instructor to update the host's different metadata, including the network configuration and user account/credential information. The metadata is initially recorded in the scenario database and is then simultaneously sent to VMware vCenter and IBM PowerVM via the vSphere API and the PowerVC API.

Fig. 5b shows the expected trainee behavior events listed in the preparation phase. As of the prototyping stage, 79 different types of atomic logs are available for an instructor to select an interim log. Given that only the AND operation is permitted for combining atomic logs, combinations can generate $(79-1)(79-2)/2 = 3003$ interim logs. As a two-step logical operation using AND or OR is permitted while composing a behavior log, $(3002 \times 3001 - 1) \times (3002 \times 3001 - 2) \approxeq 8.11 \times 1013$, i.e., a nearly infinite number of behavior logs, can be constructed in our training system. However, providing all of the behavior logs expected for a training session might be challenging for an instructor. Therefore, we enabled the prototype to reuse behavior logs utilized in prior sessions to address this issue.

Fig. 5c and Fig. 5d illustrate the action planning processes of the red and blue team agents, respectively. As for a red team agent, the network map built in the previous process


사후 강평 준비 : 192 - 훈련자 스코어링
훈련원 점수 기여도
보고서 미리보기
사후 강평 준비 : 161 - 전투과정 동영상 편집
훈련 네트워크별 비활성

(a) Dashboard with the trainee scoring status

(b) Generation of training report

(c) Editing of a trainee screencast for After-Action-Review

(d) Playback of a cyber COP screencast for After-Action-Review

FIGURE 7. Demonstrative screenshots of the evaluation and AAR phase.

allows an instructor to automatically configure the attack path and assign offensive TIs utilized in the attack by selecting the attack's start and end points. As indicated previously, the walk-through feature is used to rehearse with the autonomous red team agents. At the bottom of Fig. 5c, we can see the logs generated by the red team agents as they walk through a penetration scenario in which an external host (red square) infiltrates victim hosts (violet square) via intermediary nodes (amber square). In the case of a blue team agent, we supplied a detailed configuration UI that allowed an instructor to fine-tune the blue team's behaviors using a variety of defensive TIs in addition to the fundamental defense measures that may be done automatically with a few inputs. On the right side of the panel in Fig. 5d, we can confirm the many defensive TIs provided under the 6D categories of defense course-of-actions: detection, deny, disrupt, degrade, deceive, and destroy [32].

B. IMPLEMENTATION PHASE

The implementation phase focused on the visualization of COP, which enables instructors to assess the training situation and trainee progress quickly. Because the achievement of expected behavior events is the primary indicator of the flow of the training process and the trainee progress, we developed and organized the forms of COPs expected to be the most effective at representing the state of behavior events.

The graphic in Fig. 6a depicts the progress of agent behavior events, specifically the status of TI executions. TIs are activated per the execution route determined by the pre-and post-conditions specified. With the diagram, an instructor can verify that each TI was successfully run and quickly determine which TI caused the flow to fail to complete as expected. The progress diagram for the blue team agent action execution is constructed similarly to the red team agent's but is displayed in parallel, as the blue team agent's activities are not serialized as the red team agent's actions are.

The dashboard seen in Fig. 6b monitors the status of expected behavior occurrences. Once a trainee's actions identify interim events, the progress bar for the corresponding behavior event reflects the percentage of completed interim events. By clicking on any behavior event, an instructor can view the detailed state of event detection and the logical breakdown of that behavior event.

Fig. 6c and 6d illustrate the two primary Cyber COP displays produced for ICSTASY. The first is a cyber COP with a force-directed graph, which serves as the primary COP for assisting an instructor's situational awareness via a conceptual data model. Specifically, when an agent or trainee node has a new behavior event as a child node, it is added to the center node. The child nodes are added up whenever the agent or trainee experiences a new behavior event. If the conditions between events are dependent, the event nodes have a

TABLE 1. Comparison between cybersecurity training system/platforms.[2]

Phase	Features	Cybersecurity Training System/Platforms					
		CyRIS [6]	Nautilus [8]	CybOrg [12]	CRATE [15]	KYPO [19]	ICSTASY
Preparation	Script/markup language-based training environment configuration (III-A-1, III-A-2)	Yes	Yes	Yes	Yes	No	Yes
	GUI-based training environment configuration (III-A-1, III-A-3)	No	Yes	No	No	No	Yes
	Automated agent action planning (III-B-1, III-B-2)	P/S	No	Yes	Yes	No	Yes
Implementation	Automated training environment provisioning (III-1-1, III-1-3)	Yes	Yes	Yes	Yes	Yes	Yes
	IDS-based basic event monitoring (III-C-1)	No	No	No	Yes	Yes	Yes
	Dedicated agent-based fine-grained event monitoring (III-C-1, III-C-2)	No	No	No	No	N/A	Yes
	Visualization via cyber COP (III-C-3)	No	No	No	P/S	P/S	Yes
	Autonomous red/blue team agents (III-B-3)	P/S	No	Yes	Yes	No	Yes
	Background traffic generation/traffic injection (III-B-1)	No	No	No	P/S	No	Yes
Evaluation & AAR	Automated trainee scoring (III-D-1, III-D-2)	No	No	No	Yes	Yes	Yes
	Screen recording & replay (III-D-3)	No	No	No	No	No	Yes
	Training report generation (III-D-1, III-D-2)	No	No	No	Yes	N/A	Yes

subordinate connection. On the right side of the cyber COP, trainees' achieved behavior events are also listed. By selecting an event from the list, a video with the trainee's screencast at the time of the event will play. ICSTASY accomplishes this by maintaining video recordings of trainees' screencasts with a 30-second window size.

The second is a cyber COP with a conventional network map diagram, which serves as a secondary COP to aid intuitive network plane knowledge. The red and blue hexagonal loops surrounding the nodes in Fig. 6d denote the region of the red and blue teams, respectively. The white team designates the color-coded information prior to the train session and can swap to another color when an instructor confirms a trainee's occupation report. The red parabolic line in the figure represents the network flow associated with a cyber attack, connecting the attack's origin and destination. These visual elements provide instructors and observers of cyber training with an instantaneous perception of a training situation and create a highly immersive, competition-like (i.e., gamified) environment for trainees when combined with the varied coaching experience supported by various media.

C. EVALUATION AND AAR PHASE

In terms of user experience, the assessment and AAR phase is divided into two distinct components: evaluation and AAR. Fig. 7a and 7b illustrate the trainee scoring and training report generation features, respectively, which are mostly used for the instructor's evaluation work. The trainee scoring dashboard summarizes and displays the current training session's point-scoring and learning progress. For instance, an instructor can use the dashboard to determine how trainees earned scores and which trainee contributed the most to their team's point total. The training history saved in the training monitoring database, along with relevant data contained in other databases, is assembled into a single training report, including the scoring data. Additionally, the training report includes additional statistics about the training that are not displayed in the UI, such as the status of file/network/process access and privilege escalation, as well as the CPU/RAM consumption on each host.

Regarding the AAR part shown in Fig. 7c and 7d, we attempted to maximize the use of screencasts of trainees' screens and cyber COPs captured during a training session. However, because retaining complete screencasts of all the screens displayed during a session could result in an enormous strain on the ICSTASY system's storage, the editing process for the screencasts to be used in AAR was added immediately upon the session's conclusion. As illustrated in Fig. 7c, only the partial, selected segments of the trainees' screencasts required for AAR remain after a training session. Although the screencasts of cyber COPs are editable in the same way as those of trainees, they are substantially more lightweight to process since only the visualization data for each COP is recorded and replayed.

D. COMPARISONS WITH OTHER TRAINING SYSTEMS

Table 1 outlines and contrasts the primary features of each work. We can certify that ICSTASY offers the most comprehensive features over any other training solution. While certain training systems, such as CRATE, may contain a number

[2] P/S and N/A denote *Partially Supported* and *Not Available* (unknown), respectively.

TABLE 2. List of Technique Instances for Red Team Agents.

No.	Tactic	Technique	TI ID	Target Platforms	Description
1	TA0001 (Initial Access)	Valid Accounts	T1078	Linux, Windows	Using ordinary user accounts
2		Spearphishing Link	T1192	Linux, Windows	Spearphishing using URL links
3		Spearphishing Attachment	T1193	Linux, Windows	Spearphishing using file attachments
4	TA0002 (Execution)	Service Execution	T1035	Windows	Windows service execution
5		Scheduled Task	T1053	Windows	Execution of a program via scheduled task
6		Command-Line Interface	T1059	Linux, Windows	Execution of an executable file with CLI
7		PowerShell	T1086	Windows	Execution with Windows PowerShell
8		Clipboard Data	T1115	Linux, Windows	Collecting data stored in clipboard
9		Space after Filename	T1151	Linux	Adding a space after a filename
10		Source	T1153	Linux	Execution of a function or a file using the source command
11		Local Job Scheduling	T1168	Linux	Registering a task on the cron daemon
12		User Execution	T1204	Linux, Windows	Execution of an executable file with the ordinary user privilege
13	TA0003 (Persistence)	Winlogon Helper DLL	T1004	Windows	Registering DLL on Windows Registry (e.g., HKLM\Software [Wow6432Node]Microsoft\Windows NT\CurrentVersion\Winlogon\ and HKCU\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\)
14		Port Monitors	T1013	Windows	Manipulating a Windows Registry key to modify the DLL path called by spoolsv.exe
15		Modify Existing Service	T1031	Windows	Modifying sc.exe or a Windows Registry key to change binPath of a running service
16		New Service	T1050	Windows	Registering a new service with sc.exe
17		Service Registry Permissions Weakness	T1058	Windows	Modifying binPath or imagePath of services registered on Windows Registry (HKLM\SYSTEM\ CurrentControlSet\Services)
18		Registry Run Keys / Startup Folder	T1060	Windows	Adding/Modifying Windows Registry key related to starting programs
19		AppInit DLLs	T1103	Windows	Modifying a Windows Registry key (HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Windows) used by AppInit
20		Netsh Helper DLL	T1128	Windows	Modifying a Windows Registry key (HKLM\SOFTWARE\Microsoft\Nets) used by NetSH
21		Authentication Package	T1131	Windows	Modifying a Windows Registry key (HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Windows) used by Local Security Authority (LSA)
22		Create Account	T1136	Linux, Windows	Creation of an account

TABLE 2. List of Technique Instances for Red Team Agents.

23		.bash_profile and .bashrc	T1156	Linux	Adding a script code in .bash_profile or .bashrc
24		AppCert DLLs	T1182	Windows	Modifying a Windows Registry key (HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager) used by AppCert DLL
25		Port Knocking	T1205	Linux	Conduct of network port scanning
26		Time Providers	T1209	Windows	Modifying a Windows Registry key (HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\W32Time\TimeProviders\) used by the W32Time service
27	TA0004 (Privilege Escalation)	Sudo	T1169	Linux	Execution of a program with the administrator privilege
28		Sudo Caching	T1206	Linux	Using the privilege of root-privileged executable before its returning to the normal user privilege
29	TA0005 (Defense Evasion)	Masquerading	T1036	Linux, Windows	Disguise the names of malicious programs/processes as normal ones
30		File Deletion	T1107	Linux, Windows	Deletion of files using a normal delete operation
31		Modify Registry	T1112	Windows	Creation/modification/deletion/hiding of a Windows Registry entry/key
32		Clear Command History	T1146	Linux	Deletion of CLI command history
33		File Permissions Modification	T1222	Linux, Windows	Modifying the file permissions
34	TA0006 (Credential Access)	Credential Dumping	T1003	Linux, Windows	Dumping of the account credentials
35		Credentials in Files	T1081	Linux, Windows	Using the contents in stored files for account access management (e.g., xml files used for FileZilla to manage sessions)
36		Bash History	T1139	Linux	Accessing the .bash_history file
37		Exploitation of Remote Services	T1210	Linux, Windows	Penetrate into a host using vulnerability of the SMB protocol (e.g., Eternal Blue)
38	TA0007 (Discovery)	System Service Discovery	T1007	Windows	Collecting the system service information
39		Application Window Discovery	T1010	Windows	Accessing the application list
40		System Network Configuration Discovery	T1016	Linux, Windows	Accessing the system network configurations
41		System Owner/User Discovery	T1033	Linux, Windows	Accessing the system administrators/users informations
42		Network Service Scanning	T1046	Linux, Windows	Scanning the network services
43		System Network Connections Discovery	T1049	Linux, Windows	Accessing informations on the active network connections
44		Process Discovery	T1057	Linux, Windows	Accessing the running processes list

TABLE 2. List of Technique Instances for Red Team Agents.

45		System Information Discovery	T1082	Linux, Windows	Accessing the system informations
46		File and Directory Discovery	T1083	Linux, Windows	Exploring files and directories
47		Account Discovery	T1087	Linux, Windows	Accessing the user accounts list
48		System Time Discovery	T1124	Windows	Accessing the system time information
49		Network Share Discovery	T1135	Windows	Scanning shared networks
50		Remote Service	T1021	Windows	Making a connection using a remote service
51		Taint Shared Content	T1080	Windows	Uploading a malware on a shared file server
52	TA0009 (Collection)	Data Staged	T1074	Linux, Windows	Storing data in a temporary space
53		Automated Collection	T1119	Linux, Windows	Collecting files in an automatic manner
54		Data from Information Repositories	T1213	Linux, Windows	Acquiring data via information repositories such as databases and file servers
55	TA0010 (Command & Control)	Data Compressed	T1002	Linux, Windows	Conducting the compression of data
56		Data Encrypted	T1022	Linux, Windows	Conducting the encryption of data
57		Exfiltration Over Command and Control Channel	T1041	Linux, Windows	Exfiltration of data to a C2 server with the commonly used communication protocols
58		Exfiltration Over Alternative Protocol	T1048	Linux, Windows	Exfiltration of data to a C2 server with a special communication protocol
59	TA0011 (Exfiltration)	Standard Cryptographic Protocol	T1032	Linux, Windows	Conducting C&C communication for data exfiltration with the standard encryption techniques
60		Commonly Used Port	T1043	Linux, Windows	Conducting C&C communication for data exfiltration with the commonly used ports
61		Uncommonly Used Port	T1065	Linux, Windows	Conducting C&C communication for data exfiltration with non-commonly used ports
62		Standard Application Layer Protocol	T1071	Linux, Windows	Conducting C&C communication for data exfiltration with the standard application layer protocols
63		Data Encoding	T1132	Linux, Windows	Conduct of data encoding for data exfiltration
64	TA0040 (Impact)	Data Destruction	T1485	Linux, Windows	Deleting all the data in a storage
65		Data Encrypted for Impact	T1486	Linux, Windows	Conducting data encryption for sabotage (e.g., ransomware)
66		Service Stop	T1489	Windows	Termination of a running service
67		Stored Data Manipulation	T1492	Linux, Windows	Modifying stored data such as documents, email files and databases

of features comparable to ICSTASY, given the publicly available data, the technological maturity of each feature appears to be less than that of ICSTASY.

VI. CONCLUSION

This paper introduces ICSTASY, a novel cybersecurity training system for military personnel. It outlines the essential requirements and design architectures that must be met for trainees to have an immersive training experience and to facilitate instructors in their capacity to coach and manage cybersecurity training effectively. The development outcome of ICSTASY as a prototype proved that design concepts and requirements were concretely represented and incorporated into the system, demonstrating the feasibility of integrated, comprehensive cybersecurity training. Our next effort will include integrating LVC interoperability with ICSTASY.

APPENDIX

LIST OF TECHNIQUE INSTANCES FOR RED TEAM AGENTS

See Table 2.

REFERENCES

[1] (Oct. 2021). A. Mehta. *Cyber Concerns, Classification Disagreements Lead Space Survey Results*. Breaking Defense. [Online]. Available: https://breakingdefense.com/2021/10/cyber-concerns-classification-disagreements-lead-space-survey-results/

[2] M. G. Wabiszewski, T. R. Andel, B. E. Mullins, and R. W. Thomas, ''Enhancing realistic hands-on network training in a virtual environment,'' in *Proc. Spring Simul. Multiconf. (SpringSim)*, San Diego, CA, USA, Mar. 2009, pp. 1–8.

[3] R. S. Mudge and S. Lingley, ''Cyber and air joint effects demonstration (CAAJED),'' Inf. Directorate, Air Force Res. Lab, Rome, NY, USA, Tech. Rep. AFRL-RI-RS-TM-2008-12, Mar. 2008.

[4] W. D. Meitzler, S. J. Ouderkirk, and C. O. Hughes, ''Security assessment simulation toolkit (SAST) final report,'' Pacific Northwest Nat. Lab. (PNNL), Richland, WA, USA, Tech. Rep. PNNL-18964, Nov. 2009.

[5] G. Torres, K. Smith, J. Buscemi, S. Doshi, H. Duong, D. Xu, and H. K. Pickett, ''Distributed stealthnet (D-SN): Creating a live, virtual, constructive (LVC) environment for simulating cyber-attacks for test and evaluation (T&E),'' in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2015, pp. 1284–1291.

[6] C. Pham, D. Tang, K.-I. Chinen, and R. Beuran, ''CyRIS: A cyber range instantiation system for facilitating security training,'' in *Proc. 7th Symp. Inf. Commun. Technol.*, Ho Chi Minh, Vietnam, Dec. 2016, pp. 251–258.

[7] R. Beuran, D. Tang, C. Pham, K.-I. Chinen, Y. Tan, and Y. Shinoda, ''Integrated framework for hands-on cybersecurity training: CyTrONE,'' *Comput. Secur.*, vol. 78, pp. 43–59, Sep. 2018.

[8] G. Bernardinetti, S. Iafrate, and G. Bianchi, ''Nautilus: A tool for automated deployment and sharing of cyber range scenarios,'' in *Proc. 16th Int. Conf. Availability, Rel. Secur.*, Vienna, Austria, Aug. 2021, pp. 1–7.

[9] S. Christey and R. A. Martin, ''Vulnerability type distributions in CVE,'' MITRE, McLean, VA, USA, Tech. Rep., May 2007. [Online]. Available: https://cwe.mitre.org/documents/vuln-trends/vuln-trends.pdf

[10] S. Arshad, M. Alam, S. Al-Kuwari, and M. H. A. Khan, ''Attack specification language: Domain specific language for dynamic training in cyber range,'' in *Proc. IEEE Global Eng. Educ. Conf. (EDUCON)*, Apr. 2021, pp. 873–879,

[11] D. D. Updyke, G. B. Dobson, T. G. Podnar, L. J. Osterritter, B. L. Earl, and A. D. Cerini, ''Ghosts in the machine: A framework for cyber-warfare exercise npc simulation,'' Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU/SEI-2018-TR-005, Dec. 2018.

[12] M. Standen, M. Lucas, D. Bowman, T. J. Richer, J. Kim, and D. Marriott, ''CybORG: A gym for the development of autonomous cyber agents,'' in *Proc. 1st Int. Workshop Adapt. Cyber Defense*, Aug. 2021, pp. 1–7. [Online]. Available: https://arxiv.org/html/2108.08476v1

[13] T. Sommestad, ''Experimentation on operational cyber security in CRATE,'' in *Proc. NATO STO-MP-IST Spec. Meeting*, Copenhagen, Denmark, 2015, pp. 7:1–7:12. [Online]. Available: http://www.sommestad.com/teodor/

[14] T. Gustafsson and J. Almroth, ''Cyber range automation overview with a case study of CRATE,'' in *Proc. 25th Nordic Conf. Secure IT Syst. (NordSec)*, Nov. 2020.

[15] J. Almroth and T. Gustafsson, ''CRATE exercise control—A cyber defense exercise management and support tool,'' in *Proc. 5th IEEE Eur. Symp. Secur. Privacy Workshops (EuroS&PW)*, Sep. 2020, pp. 37–45.

[16] H. Holm and T. Sommestad, ''SVED: Scanning, vulnerabilities, exploits and detection,'' in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Baltimore, MD, USA, Nov. 2016, pp. 976–981.

[17] Greenbone Networks GmbH. *OpenVAS–Open Vulnerability Assessment Scanner*. Accessed: Nov. 13, 2021. [Online]. Available: https://www.openvas.org

[18] M. Roesch, ''Snort–lightweight intrusion detection for networks,'' in *Proc. 13th USENIX Large Installation Syst. Admin. Conf. (LISA)*, Seattle, WA, USA, Nov. 1999, pp. 1–11.

[19] P. Čeleda, J. Čegan, J Vykopal, and D Tovarňák, ''KYPO—A platform for cyber defence exercises,'' in *Proc. Modelling Simulation Support Oper. Tasks Including War Gaming, Logistics, Cyber Defence (NATO STO-MP-MSG)*, Munich, Germany, Oct. 2015. [Online]. Available: https://www.sto.nato.int/publications/STOMeetingProceedings/STO-MP-MSG-133/MP-MSG-133-COVER.pdf

[20] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, and M. A. Ferrag, ''Cyber ranges and TestBeds for education, training, and research,'' *Appl. Sci.*, vol. 11, no. 4, p. 1809, Feb. 2021.

[21] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, ''MITRE ATT&CK: Design and philosophy,'' MITRE, McLean, VA, USA, Tech. Rep. MP180360R1, Jul. 2018.

[22] S. Y. Enoch, Z. Huang, C. Y. Moon, D. Lee, M. K. Ahn, and D. S. Kim, ''HARMer: Cyber-attacks automation and evaluation,'' *IEEE Access*, vol. 8, pp. 129397–129414, 2020.

[23] S. Y. Enoch, C. Y. Moon, D. Lee, M. K. Ahn, and D. S. Kim, ''A practical framework for cyber defense generation, enforcement and evaluation,'' *Comput. Netw.*, vol. 208, May 2022, Art. no. 108878.

[24] C. Lee, ''Method for providing background traffic using IP random assigning in cyber range,'' *Electron. Lett.*, vol. 57, no. 6, pp. 261–263, Feb. 2021.

[25] D. Lee, D. Kim, M. K. Ahn, W. Jang, and W. Lee, ''Cy-through: Toward a cybersecurity simulation for supporting live, virtual, and constructive interoperability,'' *IEEE Access*, vol. 9, pp. 10041–10053, 2021.

[26] VMware. *vSphere Automation API Reference*. Accessed: Nov. 13, 2021. [Online]. Available: https://developer.vmware.com/apis/vsphere-automation/latest

[27] *IBM Power Virtualization Center APIs*. Accessed: Nov. 13, 2021. [Online]. Available: https://www.ibm.com/docs/en/powervc/1.4.3?topic=power-virtualization-center-apis

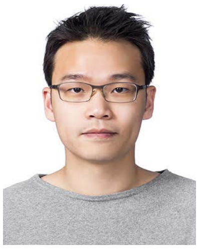
[28] J. Turnbull, *The Logstash Book*. Research Triangle, NC, USA: Lulu Press, 2013.

[29] C. Gormley and Z. Tong, *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics engine*. Sebastopol, CA, USA: O'Reilly Media, 2015.

[30] *Sysinternals Suite*. Accessed: Nov. 13, 2021. [Online]. Available: https://docs.microsoft.com/en-us/sysinternals/downloads/sysinternals-suite

[31] *Osquery*. Accessed: Nov. 13, 2021. [Online]. Available: https://github.com/osquery/osquery

[32] E. M. Hutchins, M. J. Cloppert, and R. M. Amin, ''Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains,'' *Leading Issues Inf. Warfare Secur. Res.*, vol. 1, no. 1, pp. 80–106, Apr. 2011.



DONGHWAN LEE (Graduate Student Member, IEEE) received the B.E. degree in industrial engineering and the M.S. degree in computer science and engineering from Korea University, Seoul, Republic of Korea, in 2006 and 2008, respectively, where he is currently pursuing the Ph.D. degree in cybersecurity. He is a Senior Researcher at the Cyber/Network Technology Center, Agency for Defense Development, Seoul. His research interests include wireless communication, parallel and distributed computing, wireless security, and virtualization technologies for cybersecurity.

DONGHWA KIM received the B.S. and M.S. degrees from the School of Electrical Engineering, Korea University, Seoul, Republic of Korea, in 2004 and 2007, respectively. He is currently a Senior Researcher at the Cyber/Network Technology Center, Agency for Defense Development, Seoul. His research interests include cybersecurity training systems and red team automation.

CHANGWON LEE received the B.S., M.S., and Ph.D. degrees in electronics and computer engineering from Hanyang University, in 1999, 2001, and 2019, respectively. He is currently a Principal Researcher at the Cyber/Network Technology Center, Agency for Defense Development, Seoul, Republic of Korea. His current research interests include cyber security and hardware security.

MYUNG KIL AHN received the B.S. degree in information and communication engineering from Chungnam National University, Daejeon, Republic of Korea, in 1997, the M.S. degree in computer engineering from Sogang University, Seoul, Republic of Korea, in 2003, and the Ph.D. degree in electrical and electronics engineering from Chung-Ang University, Seoul, in 2021. She is currently a Principal Researcher at the Cyber/Network Technology Center, Agency for Defense Development, Seoul. Her research interests include computer security and cyberwarfare modeling and simulation.

WONJUN LEE (Fellow, IEEE) received the B.S. and M.S. degrees in computer engineering from Seoul National University, Seoul, Republic of Korea, in 1989 and 1991, respectively, the M.S. degree in computer science from the University of Maryland, College Park, MD, USA, in 1996, and the Ph.D. degree in computer science and engineering from the University of Minnesota, Minneapolis, MN, USA, in 1999. In 2002, he joined the Faculty of Korea University, Seoul, where he is currently a Professor with the School of Cybersecurity. He has authored or coauthored over 220 papers in refereed international journals and conferences. His research interests include communication and network protocols, optimization techniques in wireless communication and networking, security and privacy in mobile computing, and RF-powered computing and networking. He has served as the TPC and/or an Organizing Committee Member for IEEE INFOCOM, from 2008 to 2023, the PC Vice Chair for IEEE ICDCS 2019 and the ACM MobiHoc, from 2008 to 2009, and over 130 international conferences.