Novel Architecture for Naval Cyber-Kinetic Training

Omar Hasan, Ph.D., Derek Crane, W. Jeremy RiCharde
Dignitas Technologies
Orlando, Florida

MODSIM World 2025

"The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government."

Novel Architecture for Naval Cyber-Kinetic Training

Omar Hasan, Ph.D., Derek Crane, W. Jeremy RiCharde
Dignitas Technologies
Orlando, Florida
ohasan@dignitastech.com, dscrane@dignitastech.com, jricharde@dignitastech.com

ABSTRACT

To disrupt naval mission operations, adversaries regularly conduct internet protocol (IP)-based cyberattacks, including denial of services, data exfiltration, and spoofing. Cyber Mission Force (CMF) teams counter these threats by performing offensive and defensive cyberspace operations in support of combatant commands. Cyber Protection Teams (CPT) defend Naval key resources from threat actions, while Cyber Combat Mission Teams (CCMT) conduct military cyber operations to support operational objectives. These cyber teams require collective cyber-kinetic training to ensure they work effectively with commanders across the Services and Joint Force to achieve battlespace advantage. Cyber-kinetic training is hindered because Live, Virtual, and Constructive (LVC) systems used for command staff training are not developed to communicate directly with cyber ranges used for cyber team training. Manual coordination of cyber effects between these training environments is performed, which reduces realism and is error-prone. We describe a novel system architecture developed to automate the communication of cyber effects between a cyber range and LVC systems. The system utilizes cyber range sensors to determine cyber Battle Damage Assessment (BDA) due to operator actions within the range that cause changes to network and system states. This cyber BDA is communicated to the LVC training environment so that generated cyberspace effects have an operational impact on shipboard systems and Maritime Operations Center (MOC) workstations. We demonstrate our approach though a prototype that coordinates several cyberspace effects between the cyber range and LVC environment. This approach represents a significant improvement for cyber-kinetic training to increase readiness for conducting multidomain operations.

ABOUT THE AUTHORS

Dr. Omar Hasan is the Chief Technology Officer (CTO) at Dignitas Technologies, where he also serves as the principal investigator on cyberspace-related research efforts. Dr. Hasan has 25 years of experience in software development, focusing on the Modeling and Simulation (M&S) areas of simulator interoperability, distributed simulation, and simulation architecture and infrastructure. He has extensive experience in object-oriented software analysis and design, open-source technologies and methodologies, and collaborative software development. Dr. Hasan has held architect and software engineering lead positions on both the One Semi-Automated Forces (OneSAF) and Joint Land Component Constructive Training Capability (JLCCTC) programs. He has also supported software development and cyber test event execution activities for the National Cyber Range Complex (NCRC). Dr. Hasan holds a B.S. and M.S. in Engineering from Columbia University and a Ph.D. in Engineering from Rutgers University.

Derek Crane is the technical lead for this research. He has 15 years of experience with system/software development for military modeling, simulation, and training systems. He has significant experience with Development Operations (DevOps) principles, including containerization using Docker and Podman, automation using Ansible, and is experienced with Linux variants. Mr. Crane has leveraged containerization to run infrastructure monitoring tools, host web services, and to create build environments for large Army Programs of Record (PoR), including the Aviation Combined Arms Tactical Trainer (AVCATT). Mr. Crane holds a B.S. in Computer Science with a minor in Mathematics from the University of Central Florida.

W. Jeremy RiCharde is a software developer on this research effort. He has 7 years of experience with system/software development for military modeling, simulation, and training systems, and additional experience with general software development. He has worked as a full stack developer with several programming languages and is experienced with containerization. Mr. RiCharde holds B.S degrees in both Physics and Computer Science from the University of Central Florida.

Novel Architecture for Naval Cyber-Kinetic Training

Omar Hasan, Ph.D., Derek Crane, W. Jeremy RiCharde
Dignitas Technologies
Orlando, Florida
ohasan@dignitastech.com, dscrane@dignitastech.com, jricharde@dignitastech.com

INTRODUCTION

In the modern battlespace, the traditional fight within the warfighting domains of air, land, sea, and space has expanded to the cyberspace domain. Within cyberspace, adversaries actively pursue Internet Protocol (IP)-based cyber attacks to affect operational missions in all domains. The U.S. Navy and other Services utilize Cyber Mission Force (CMF) teams to direct, synchronize, and coordinate cyberspace operations in defense of U.S. national interests. Within the CMF, Cyber Protection Teams (CPT) defend critical infrastructure and key resources from threat actions, while Cyber Combat Mission Teams (CCMT) conduct military cyber operations in support of combatant commands. To maximize their effectiveness for multidomain operations, these cyber teams require collective cyber-kinetic training (combined cyber training with kinetic-focused training) to ensure they work effectively with commanders across the Services and Joint Force to accomplish their assigned missions and achieve information advantage in the battlespace.

During Fleet Synthetic Training (FST) events, Naval trainees within virtual and live ships interact with actors simulated by constructive systems within the Navy Continuous Training Environment (NCTE). However, the Live, Virtual, and Constructive (LVC) systems within the NCTE are not developed to communicate directly with cyber ranges used for CMF training. Coordination of cyber effects between these training environments is performed manually (i.e., white cards, swivel chair). This manual coordination is cumbersome, error-prone, and limits the realism for the training audience. To better prepare for multi-domain operations (MDO) in the current battlespace, Naval training systems need to be developed to automatically communicate cyberspace information across the entire training environment, so a unified operational picture is provided to all trainees (cyber and non-cyber).

APPROACH

Our work provides an approach to develop an architecture supporting *cyber-kinetic* training, in which actions within a cyber range result in the communication and application of cyberspace effects within a connected simulation environment. This unified environment allows command staff to train concurrently with CMF in both mitigating threat actions affecting their operational systems and to effectively perform offensive cyber actions against threat systems. In this training architecture, cyber range operators acting as a CPT or CCMT, attack or defend emulated systems that represent operational systems within the training scenario. Those systems are also represented as simulated or real devices within the simulation environment. Cyber battle damage is assessed from the results of activities within the cyber range and that damage is injected into the simulation environment as a cyberspace effect on the simulated or real devices within the simulation environment. This use case represents *collective* cyberspace training, where the cyber range environment is used to train CPT or CCMT members and the simulation environment is used to simultaneously train military command staff and systems operators. This approach brings significant improvements over the current methods used for cyber-kinetic training, which are error prone and limit training realism due to manual coordination between the two disparate training environments.

To support naval cyber-kinetic training, a novel system architecture was developed to automate the communication of cyber effects between a cyber range, an LVC simulation environment, and connected trainee systems. We considered two types of trainee systems in this work: 1.) representative shipboard Command, Control, Communications, Computers, and Intelligence (C4I) systems, and 2.) representative Maritime Operations Center (MOC) workstations. Our architecture utilizes cyber range sensors that detect changes to network and system state due to range operator actions and perform cyber Battle Damage Assessment (BDA). This cyber BDA is

communicated to the simulation environment using a cyberspace brokering architecture, so that generated cyberspace effects have an impact on connected LVC simulations, C4I systems, and MOC workstations. The feasibility of this approach was demonstrated though a prototype that coordinates cyberspace effects between the cyber range, simulation environment, and connected C4I systems and MOC workstations. This approach can significantly improve naval cyber-kinetic training in various areas of the Navy operational environment, increasing warfighter readiness for conducting MDO.

ARCHITECTURE

This section describes the high-level architecture developed to communicate cyberspace effects between a cyber range, simulation environment, C4I systems, and operator workstations. This architecture, depicted in Figure 1, consists of a cyber range, used for cyber offensive and defensive team training, and constructive simulations, C4I systems and operator workstations, used for command staff training. Cyberspace domain-related information is communicated between the systems within the training environment using a cyberspace brokering architecture, which provides cyberspace effect models as well as user interfaces utilized by exercise facilitators. A network guard (not shown) is optionally used to restrict the data flow between the cyber range and the simulation environment in multi-level security (MLS) environments. A description of each of the components of this architecture is given in Table 1.

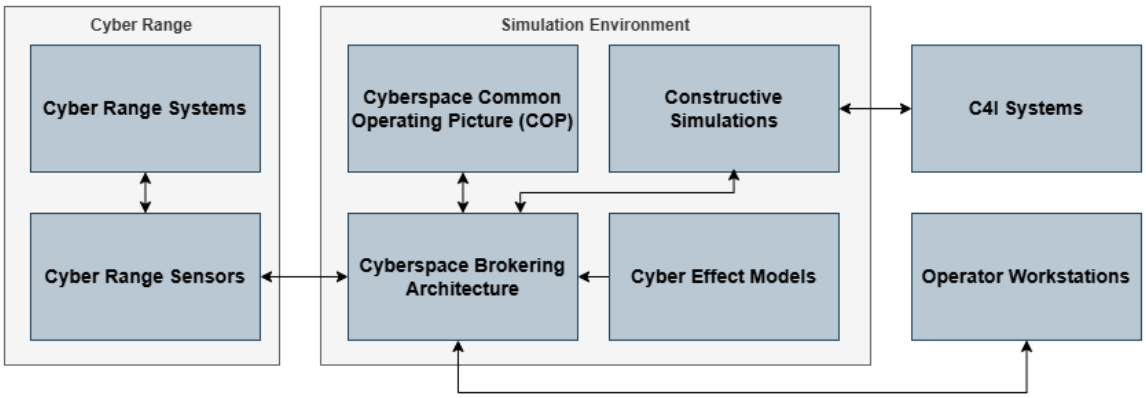

Figure 1. High-level architecture used to communicate cyberspace effect information between a cyber range, simulation environment, C4I systems, and operator workstations.

Table 1. Components within the combined cyber range and simulation environment architecture.

Architecture Component	Description
Cyber Range	Provides a virtual environment that contains emulated systems and networks, as well as training content used for training Cyber Protection Teams and Cyber Combat Mission Teams
Cyber Range Sensors	Provides software applications that assess changes within cyber range systems, assess corresponding cyber battle damage, and communicate resulting cyberspace effects to simulation environment
Cyberspace Brokering Architecture	Provides data model and communication mechanism for communicating cyberspace-related information between connected systems
Cyberspace Common Operating Picture (COP)	Provides exercise facilitator / white cell functionality to control and monitor cyberspace effects within the training environment
Constructive Simulations	Provides models of friendly and threat actors, cyberspace devices, cyberspace operations and effects
Cyber Effect Models	Provides modeling of cyber and Electromagnetic Warfare (EW) effects, which can be based on specific Blue Force (BLUFOR) and adversarial Tactics, Techniques, and Procedures (TTP)

C4I Systems	Tactical interfaces utilized by the command staff during training
Operator Workstations	Workstations utilized by operations support staff during training

Each of these components is described in more detail in the following sections.

Cyber Range

Cyber ranges are comprised of interactive, emulated platforms and representations of networks, systems, tools, and applications. They emulate an organization's network, systems, and services in a safe and controlled virtual environment for cybersecurity training. Within Department of Defense (DoD) service branches, CPTs and CCMTs utilize cyber ranges for training on complex TTPs required for offensive and defensive military cyberspace operations to support mission objectives. For example, a cyber range may be used by trainees role playing as a threat cyber red team to perform cyberspace operations against emulated BLUFOR systems or military or civilian Industry Control Systems (ICS). These emulated systems, implemented as Virtual Machines (VM) or software containers within the cyber range, can represent a variety of real-world systems pertinent to military missions, including tactical systems in a command post or on a Navy ship, MOC workstations, or power facility control systems. Trainees within the cyber range perform offensive or defensive cyberspace operations on the emulated devices and software-defined networking within the range to simulate those actions on corresponding real-world systems. Cyber ranges used for DoD training include dedicated infrastructure such as the Persistent Cyber Training Environment (PCTE), operated by the United States Cyber Command (USCYBERCOM) or the National Cyber Range Complex (NCRC) operated by the Department of Defense (DoD) Test Resource Management Center (TRMC).

Cyber Range Sensors

Currently, there are no automated mechanisms used within cyber-kinetic training exercises to analyze actions occurring within the cyber range and to programmatically impart related cyberspace effects on the simulation and connected systems used by the battle staff being trained. During our investigation, we analyzed, leveraged, and developed technologies that can act as cyber range sensors, automatically mining the cyber range for information about ongoing operator activities against range systems that emulate C4I systems, operator workstations, and other systems relevant to the military scenario. The sensors perform cyber BDA on those emulated systems and derive appropriate cyberspace effects based on range operator activities. The sensors automatically communicate the cyberspace effect information between the cyber range and the constructive simulation system, reducing manpower requirements (i.e., white carding, swivel chair synchronization) and providing more realistic cyberspace effects to the trainees.

To assess cyber BDA and communicate the resultant cyber effect, the cyber range sensors utilize a four-step process:

1. Cyber range operators, role playing as threat cyber actors or executing BLUFOR offensive cyber operations, perform actions (attacks) against systems in the cyber range that represent C4I systems, operator workstations, and other systems relevant to the scenario.
2. These actions change the state of the emulated systems (e.g., increased network usage, central processing unit (CPU) spikes, service disruptions) and/or leave *breadcrumbs* within the filesystem on the emulated systems (e.g., system logs, added malware).
3. Cyber sensors within the range detect these state changes and filesystem changes and perform cyber BDA by determining the appropriate cyberspace effect, if any, that results from these changes.
4. The resultant cyberspace effect is communicated to the simulation systems and connected systems (i.e., C4I systems, operator workstations) for implementation of the appropriate cyberspace effect.

A similar process is used to communicate the mitigation of a cyber effect from the cyber range to the LVC environment, for example due to BLUFOR defensive cyber operations that remediate a vulnerability within a cyber range system. In this case, an ongoing simulated threat cyberattack on BLUFOR systems is stopped due to BLUFOR defensive actions within the cyber range environment.

Our work considered various means by which the cyber range sensors can determine cyber BDA due to changes in the range systems. The cyber range sensors can query and monitor the range systems directly, or they can utilize existing Open Source Software (OSS) and/or Commercial-Off-The-Shelf (COTS) tools such as Network Security Monitoring (NSM) systems, Intrusion Detection Systems (IDS), and Security Information and Event Management (SIEM) systems. For example, these systems can perform Network Behavior Anomaly Detection (NBAD) by examining individual network packet signatures for anomalies to help detect attacks such as spoofing. Detection of cyberspace attacks using range sensors must be done carefully, since detection methods are highly dependent on specific attack vectors and circumstances. Once a cyberattack has been detected, cyber BDA is performed to generate an appropriate cyberspace effect which is communicated to the simulation and connected systems. Mappings were developed between cyberspace attack operations within the cyber range and the cyberspace effect that is generated upon cyber BDA. Our analysis found that there is a many-to-one relationship between cyberspace attack types and the respective generated cyberspace effect. That is, multiple types of cyberspace attacks may result in the generation of the same cyberspace effect. In our analysis, we identified some example attack vectors within the cyber range that would cause a particular cyberspace effect, providing input to the symptoms the cyber range sensors should monitor for that effect. There are many combinations of possible ways to generate cyberspace effects, however, and other attack vectors will be explored in future work.

Cyberspace Brokering Architecture

To meet this training need, we developed a system architecture, which incorporates simulated cyberspace effects within this complex environment using a flexible integration approach. Our team is developing and prototyping the Cyber Simulation TRaining for Impacts to Kinetic Environment (CyberSTRIKE) architecture through a Small Business Innovation Research (SBIR) effort under the Office of Naval Research (ONR). CyberSTRIKE is government purpose rights (GPR) software that communicates cyberspace effects requested by the white cell or due to cyber range battle damage assessment (BDA) to Navy simulation systems and connected shipboard command, control, communications, computers, and intelligence (C4I) systems. [1] The CyberSTRIKE cyberspace brokering architecture provides a Cyberspace Data Model (CDM), software interfaces, cyberspace operations and effects models, and user interfaces to communicate cyberspace elements and effects between simulation systems and other cyberspace toolsets. CyberSTRIKE builds upon the Cyberspace Battlefield Operating System Simulation (CyberBOSS) system architecture that our team develops under the US Army Combat Capabilities Development Command – Soldier Center (DEVCOM SC) Simulation and Training Technology Center (STTC) [2]. The CyberSTRIKE system architecture is a microservices based system in a Service Oriented Architecture (SOA) that uses well-defined software interfaces and protocols to facilitate system integration and expansion to other systems. [3] [4] This system architecture employs an open and transparent hub-and-spoke approach where client applications connect into a common, federated data bus that is managed by a centralized server. Services maintain the model of the state of the cyberspace terrain across the training environment to provide a common and consolidated view for all connected client applications. Client applications communicate using CDM representations to specify cyberspace-specific information (e.g., cyberattacks, cyber effects, cyber status). [5] The CDM builds upon previous cyberspace data models such as Cyber Operational Architecture Training System (COATS) [6] and is compliant with emerging cyberspace data standards, such as the recently released Simulation Interoperability Standards Organization (SISO) Cyber Data Exchange Model (CyberDEM) (SISO-STD-025-2023). A wide variety of system types may interoperate through the CyberBOSS system architecture, including LVC systems, cyber ranges, cyberspace operations and effects models, and cyberspace effects tools. For the purposes of this work, this architecture was utilized to broker cyber effects from the cyber range to Constructive simulation, C4I systems, and operator workstations.

Cyberspace COP

The cyberspace COP provides user interfaces and other tools that exercise facilitators and white cell controllers use to inject and monitor cyber and EW effects within the training environment. The cyberspace COP can provide a visualization of cyberspace domain objects and effects using two- or three-dimensional maps and table views. In this architecture, the cyberspace COP provides two main areas of functionality: 1.) visualizing the state of simulated and emulated devices across the training environment (i.e., cyber range VMs, constructive device models), and 2.) monitoring of cyberspace effects resulting from actions within the cyber range.

Constructive Simulations

Within the simulation environment, the Constructive simulations provide modeling of BLUFOR, threat, and civilian actors and organizations. These simulations provide modeling of kinetic activities (i.e., moving, sensing, shooting) of these forces during simulated military operations. Within this architecture, interfaces were developed between the Constructive simulations and the cyberspace brokering architecture to communicate cyberspace and EW effects. Depending on the effect type, each effect can be applied in specific ways to models within the Constructive simulation to affect the modeling of kinetic activities within the simulation. For example, for effects disrupting or altering simulated Global Positioning System (GPS) signals used by constructive actors, simulated GPS signal data can be removed or modified within constructive mobility or firing models, changing the output of those models within the simulation and causing differences in the simulated movement or firing capability of the simulated actors.

Cyber Effect Models

Within the simulation environment, the Cyber Effect Models provide modeling of cyber and EW effects for BLUFOR and threat operations within cyberspace. These models receive cyberspace effect requests from other systems in the training environment, such as LVC simulations, and provide effect results while effects are on-going. The cyber effect models consist of both models of IP-based attacks, such as denial of service (DoS), data exfiltration, and credential compromise, as well as Radio Frequency (RF)-based attacks, such as GPS or radio communications jamming.

C4I Systems

Within the training environment, C4I systems are stimulated with tactical messages sent from the Constructive simulations. This simulated data is communicated using a variety of military protocols, depending on the targeted C4I system. During training, C4I system operators, including military command staff, view the kinetic operations modeled within the simulation using C4I system interfaces. In this architecture, cyberspace and EW effects received by the Constructive simulation from the cyberspace brokering architecture can be applied to tactical messages communicated to the C4I systems to have an operational impact on those systems. [7] For example, a jamming effect can cause information to disappear from the C4I system interface, while a data injection effect can cause erroneous information to be displayed on the C4I system interface. These effects are typically manifested by adding or removing tactical messages sent between the constructive simulation and the C4I systems, or by altering specific fields in those tactical messages to change the information received by the C4I systems.

Operator Workstations

Within the training environment, operator workstations are used by warfighting operations staff to monitor and support battlespace activities. These workstations can represent a wide variety of assets within the warfighting operational environment, including MOC and Network Operations Center (NOC) workstations, shipboard terminals, and logistics systems. Using this architecture, cyber and EW effects are placed on these systems to affect the ability of the trainees to use these systems to support warfighting functions.

PROTOTYPING EFFORTS

This section provides details on our prototyping efforts using the above architecture to demonstrate the communication of cyberspace effect information between a cyber range, simulation environment, C4I systems, and operator workstations. Our prototyping activities focused on the design and development of the cyber range sensors and associated applications used to assess changes to cyber range systems, determine resulting cyber BDA, and communicate that BDA to the connected cyberspace brokering architecture. We continue to build upon this protype with our on-going research efforts to meet the emerging future work requirements we discuss below.

Cyber Range

The cyber range infrastructure used in our prototyping was comprised of a set of Podman software containers running on a Red Hat Enterprise Linux (RHEL) 8.10 host system. The Podman containers were used to execute various emulated systems within the cyber range, including both base image Alpine Linux 3.20.3 systems and images of Alpine Linux with specialized software applications/services such as the Apache Hypertext Transfer Protocol (HTTP) Server (i.e,, httpd), OpenSSH, and MediaWiki. Internal Podman networks were set up to allow communication between the Podman containers within the range as well as communication with the CyberSTRIKE cyberspace brokering architecture. A *lightweight*, container-based cyber range was highly useful in our prototyping work since it could run on smaller hardware footprint than needed by a larger, traditional cyber range. However, the work performed prototyping with this smaller range could be expanded in future work to larger ranges. As an example, we continue to integrate our cyber range sensors with the PCTE [8].

Cyber Range Sensors Design

For our prototyping efforts, we developed a design for the cyber range sensors used within our architecture. As described above, sensors connected to the cyber range monitor the state of the systems in the range and the activity of operators as inputs for determining cyber BDA. Figure 2 depicts our overall design of the cyber range sensors. In this design, a cyber-range operator role-plays the actions of a cyber red team, performing actions and cyberattacks on emulated systems within the cyber range. A sensor host machine runs within or is connected to the cyber range environment and contains various components that collect information, perform BDA, and transmit cyberspace effect data to CyberSTRIKE based on the assessed battle damage. Within our design, two types of applications run within the sensor host machine:

1. **Cyber Range Sensor application(s).** One or more cyber range sensor applications run within the sensor host machine to monitor the state of VMs within the cyber range. These various monitoring applications sense changes in the state of the emulated systems due to range operator actions, assessing for system damage. In Figure 2, the OSS LibreNMS is shown as a representative monitoring tool (sensor). However, our architecture uses open Application Programming Interfaces (API) to support various sensor back-end implementations to work in tandem. These back-ends act as a plugin architecture, allowing various configurations of monitoring services to run depending on the desired functionality. This also provides loose coupling between the specific sensor technology back-ends (e.g., LibreNMS) and the other components of our architecture, promoting flexibility and scalability.
2. **Cyber Effect Generator application.** If the sensors detect a significant change in an emulated system, an alert is sent to the cyber effect generator application. The cyber effect generator collects information from the sensors, performs cyber BDA, and transmits corresponding cyberspace effects to CyberSTRIKE. CyberSTRIKE then communicates the effects to the simulation and connected systems. The cyber effect generator application contains two main components: 1.) an alert bus, used to receive and normalize alerts from the sensor applications, and 2.) the cyber effect resolver, used to perform cyber BDA based on the sensor alerts, generate a resulting cyberspace effect, and communicate that effect to the simulation environment. The cyber effect resolver contains cyber BDA models and a CyberBOSS Interface Framework (CIF) connection for communication with the CyberSTRIKE infrastructure, including the CyberBOSS Server.

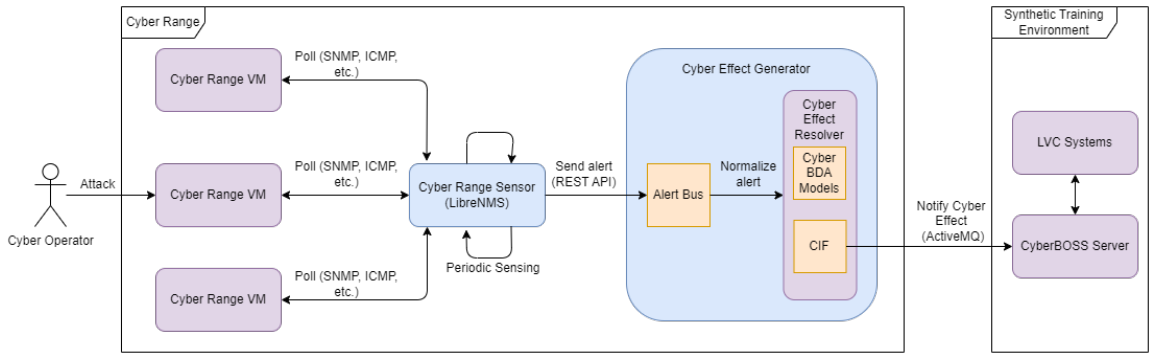

Figure 2. Design of cyber range sensors, showing LibreNMS as an example monitoring tool.

Prototyping of Cyber Range Sensor Applications

In our work, we prototyped the use of cyber range sensor applications described in the above design. As mentioned, if the sensors detect a significant change in an emulated system, an alert is sent to the cyber effect resolver application. In our prototyping, LibreNMS [9] was used as a cyber range sensor. LibreNMS was chosen since it is OSS and there is community support for development of a library of alert rules [10] and supporting macros [11] that can be reused or extended for this work. In our prototyping, these existing alert rules were utilized to monitor cyber range VMs that were not responsive to Internet Control Message Protocol (ICMP) messages (pings). These alert rules were also used to monitor cyber range VMs for which a particular service (i.e., ssh, http) was not responsive. These alerts were sent from the cyber range sensor application (LibreNMS) to the Alert Bus component of the Cyber Effect Generator application. The Alert Bus contains a Representational State Transfer (REST) endpoint that receives Hypertext Transfer Protocol (HTTP) POST messages from the LibreNMS alert transport capability when an alert occurs.

Prototyping of Cyber Effect Generator Component

In our prototyping, components within the Cyber Effect Generator application were developed to receive the alerts from the cyber sensors (LibreNMS), normalize the alerts, and convert them into a format that is consumable by the cyberspace brokering architecture (CyberSTRIKE). As mentioned above, the Alert Bus contains a Java Spring Boot REST endpoint that receives HTTP POST messages from the LibreNMS alert transport capability when an alert occurs. After receiving an alert from a sensor, the Alert Bus first normalizes the alert information a common data model. Normalizing the alerts allows for future flexibility and scalability if other types of sensors are utilized in future work. After normalization, the Alert Bus then passes the alert to one of the cyber BDA models within the Cyber Effect Resolver. After normalization of the alert, cyber BDA models then utilize information in the alert to determine what, if any, cyberspace effect should result based on the alert. The cyber BDA models are responsible for analyzing the normalized sensor alert data and assessing battle damage. The cyber BDA models are stateful components that track the history of sensor alert data, as well as on-going cyberspace effects, to determine if cyberspace effects should be created or removed from the training environment. If a cyber BDA model determines the emulated system is damaged (e.g., disabled, compromised, disrupted), a corresponding request for a cyberspace effect is generated. For example, in our prototyping, cyber range systems not responding to ICMP messages (pings) were mapped as being under a hardware damage cyberspace effect.

Prototyping the Communication of Cyber Effects to the Cyberspace Broker

As described above, the cyber BDA models send the cyberspace effect status messages to the cyberspace brokering architecture (CyberSTRIKE) for communication to the connected simulation environment, C4I systems, and operator workstations. Those systems can receive the cyberspace effect information and implement the effect in a manner applicable to the receiving system. For example, upon receipt of a DoS effect, an adapter to the C4I systems, such as the Cyber Operations Battlefield Web Services (COBWebS) [12] or the Joint Bus (JBUS), may stop tactical messaging corresponding to the targeted system from being communicated to the C4I system, resulting in the targeted system disappearing as a track on the C4I system. Similarly, upon receipt of a DoS effect, systems such as the Network Effects Emulation System (NE2S) [13] can be used to impart results on target operator workstations, such as displaying a *Blue Screen of Death (BSOD).* The cyberspace effect is also received by the cyberspace COP (i.e., the CyberSTRIKE Control Tool), where it can be monitored by exercise facilitators or white cell personnel. In our prototyping, a mechanism was developed to communicate the cyberspace effect information between the cyber BDA models and the cyberspace brokering architecture. The CyberBOSS CDM was utilized to communicate cyberspace effect information as JavaScript Object Notation (JSON) messages over an ActiveMQ message bus. These messages were received by the CyberBOSS Server, which communicated the cyberspace effect message to other CyberSTRIKE federates, including the Joint Simulation Bus (JBUS) used in our prototype to communicate the cyberspace effects to C4I systems connected to the synthetic battlespace.

EXPERIMENTAL RESULTS

To demonstrate the feasibility of our approach to communicate cyberspace effects between a cyber range, simulation environment, C4I systems, and operator workstations, we developed and experimented with a representative scenario that was implemented across the systems depicted in the above architecture. As an example of combined cyber-kinetic operations, this scenario involves coordination between a BLUFOR CPT and command staff to mitigate cyber threats against shipboard C4I systems and MOC workstations. In this scenario, the threat performs cyberattacks against shipboard C4I systems and networks with the goal of producing a disrupted or incorrect view of the operational picture for the naval command staff, providing the threat advantage within the battlespace. Similarly, the threat performs cyberattacks against MOC workstations with the goal of affecting the ability of the MOC to support Navy battle operations. The BLUFOR CPT coordinates with command staff and conducts Hunt/Clear/Harden/Assess operations to locate and fix the problems in the shipboard C4I network and within the MOC operator workstations, restoring the correct operational view to the command staff and the ability of the operations team to support warfighting.

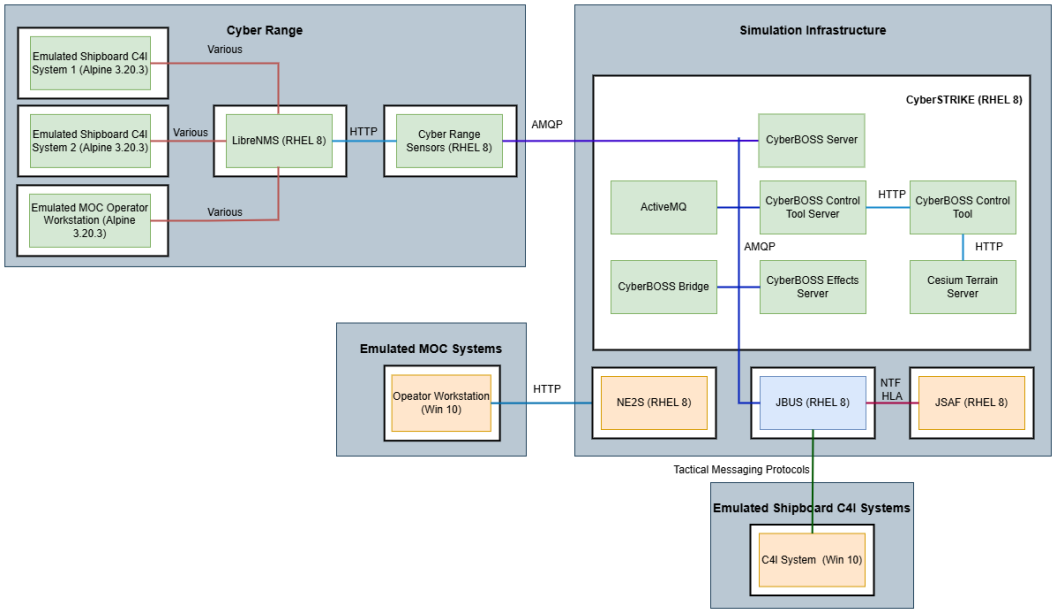

Figure 3. System architecture used for experimentation activities.

The system architecture used for experimentation is shown in Figure 3. This architecture consists of four enclaves: 1.) the cyber range, 2.) the simulation infrastructure, 3.) emulated shipboard C4I systems, and 4.) emulated MOC workstations. These enclaves are described as follows:

1. **Cyber Range.** As described above, the cyber range infrastructure used in our prototyping was comprised of a set of Podman software containers running on a RHEL 8.10 host system. Some Podman containers represented shipboard C4I systems. In our experimentation, we did not use actual VMs of the C4I systems since base Alphine Linux images provided the services needed for simulation of threat cyberattacks during our experimentation. Other Podman containers represented MOC operator workstations. Internal Podman networks were set up to allow communication between the Podman containers within the range as well as communication with the CyberSTRIKE cyberspace brokering architecture. Within the cyber range, we also deployed an instance of the cyber range sensors (LibreNMS), used to monitor the state of the emulated range systems, and an instance of the Cyber Effect Generator application, used to perform cyber BDA and communicate resulting cyberspace effects to the cyberspace brokering architecture (CyberSTRIKE).
2. **Simulation Infrastructure**. Within the simulation infrastructure, the Joint Semi-Automated Forces (JSAF) kinetic simulation was used to provide Constructive models of BLUFOR, threat, and neutral actors and platforms within the scenario. For its simulated naval vessels, JSAF models the communication of self-reporting for those vessels using various tactical messaging protocols. Within the simulation infrastructure, JBUS is used to translate the simulated tactical messages to protocols that are sent to real, shipboard C4I systems. CyberSTRIKE acted as the cyberspace brokering architecture to communicate cyberspace-related objects and effects between the cyber

range and JBUS. The CyberBOSS Control Tool was the cyberspace COP, providing a web interface to control cyberspace effects across the training environment.

3. **Emulated Shipboard C4I Systems.** The simulation infrastructure was connected to Windows 10-based Naval C4I systems. These systems were used to observe changes to the resulting cyber effect caused by cyber range operator actions.
4. **Emulated MOC Workstations**. The CyberSTRIKE cyberspace brokering architecture was connected to the NE2S system. The existing NE2S REST API was used to communicate cyber effect information from CyberSTRIKE. Upon receipt of cyber effect information, NE2S invoked services on an agent running on a Windows 10 workstation to implement the effect. That implementation varied for each effect type, for example displaying the BSOD or changing the memory usage on the targeted system.

Using this prototyping architecture and scenario, we experimented with the coordination of cyberspace effects between the cyber range and JBUS, so that operator actions within the cyber range resulted in visualization of related cyber effects on the connected C4I systems and operator workstations. For C4I system effects, this experimentation involved the following steps, using the example of a DoS cyberspace effect: 1.) Within the cyber range, a simulated threat cyber attack occurred on an emulated Naval C4I system, causing the system to no longer be responsive to ICMP (ping) requests messages. In our experimentation, the action to simulate this attack was performed using scripts, however this could also be performed by threat cyber role player actions within the cyber range. 2.) The LibreNMS monitoring system deployed within the cyber range alerted since the sensor could no longer communicate with the VM representing the emulated C4I system through ICMP (ping). 3.) The cyber effect generator received the alert and its cyber BDA models created and sent a corresponding DoS cyberspace effect to CyberSTRIKE for the emulated C4I system. The DoS cyberspace effect was communicated to JBUS for application of the effect on tactical messaging. 4.) JBUS stopped communication of the tactical messaging corresponding to the target of the DoS effect, causing the associated track to become stale (time late) on the receiving shipboard C4I systems. A similar set of steps was used to communicate and impart cyber effects, such as BSOD and memory/CPU changes, on targeted representative MOC operator workstations. This experimentation provided an initial proof of feasibility of our concept to coordinate cyberspace training between cyber ranges, the simulation environment, C4I systems, and operator workstations.

FUTURE WORK

This work represents a significant improvement to implement cyber-kinetic training since it provides an architecture to automatically communicate cyberspace information across the training environment, so a unified operational picture is provided to all trainees (cyber and non-cyber). This automated coordination minimizes manual methods to communicate and synchronize cyberspace effects between a cyber range and the simulation environment and connected C4I systems. These methods are error-prone and limit training realism. Future work to develop this architecture to support cyber-kinetic training may include:

- Further analysis, in conjunction with Information Warfare (IW) subject matter experts (SME) to determine other cyberspace effects and target systems on which to focus additional development. These effects can be due to both offensive and defensive actions performed within the cyber range by CCMTs and CPTs.
- Development of additional cyber range sensors and cyber BDA models. Our initial work utilized the OSS LibreNMS to provide alerts to our cyber range sensors; however, other OSS and COTS products could be used as a sensor front-end to provide inputs to the cyber BDA models. Additionally, other cyber BDA models can be developed to support additional cyberspace effects, such as data infiltration, data exfiltration, or spear-phishing.
- Use of other communication protocols to communicate the cyberspace effect information between the cyber BDA models and the cyberspace brokering architecture. In our prototyping, the CyberBOSS CDM was used for this communication; however, future work could utilize the emerging SISO Cyber DEM standard (SISO-STD-025-2023) structured Distributed Interactive Simulation (DIS) Protocol Data Units (PDU) could be used to communicate cyberspace effect information.
- Bi-directional effects synchronization, where kinetic events occurring in the LVC training environment could have an impact on the cyber range environment. For example, if physical network nodes are destroyed by a kinetic event in the simulation environment, the effects could be synchronized with the cyber range, impacting the network and connectivity between nodes in the cyber range environment.

CONCLUSION

To maximize their effectiveness during multidomain operations, the Cyber Mission Force teams, such as CPTs and CCMTs, require collective cyber-kinetic training to ensure they work effectively with commanders across the Services and Joint Force to accomplish their assigned missions and achieve information advantage in the battlespace. However, cyber-kinetic training is currently hindered because existing LVC systems used for command staff training are not developed to communicate directly with cyber ranges used for cyber team training, and coordination of cyber effects between these training environments is performed manually. In this paper, we described a novel system architecture developed to automate the communication of cyber effects between a cyber range and Navy LVC systems. This architecture utilizes cyber range sensors for cyber BDA due to operator actions within the range that cause changes to network and system states. The feasibility of this approach was demonstrated though a prototype that coordinated cyberspace effects between the cyber range, LVC environment, C4I systems, and MOC operator workstations. This approach represents a significant improvement for cyber-kinetic training, increasing warfighter readiness for conducting multidomain operations.

ACKNOWLEDGEMENTS

We would like to thank Ms. Natalie Steinhauser and her team at the Office of Naval Research (ONR) for their assistance with this paper and for their support of our research efforts.

REFERENCES

[1] Hasan, O., Crane, D., & Dukstein, G. (2023). *Novel Architecture for Communication of Simulated Cyberspace Effects to Navy Shipboard Systems*. Simulation Interoperability Standards Organization (SISO) 2023 Simulation Innovation Workshop (SIW), Orlando, FL.
[2] Welch, J., Hasan, O., Burch, B., Vey, N., & Geddes, J.A. (2020). *CyberBOSS: An Approach for Control and Interoperation of Cyber for Training*. Simulation Interoperability Standards Organization (SISO) 2020 Simulation Innovation Workshop (SIW), Orlando, FL.
[3] Hasan, O., Welch, J., Burch, B., Vey, N., Geddes, J.A., & Hofstra, K. (2020). *CyberBOSS Common Data Model*. Simulation Interoperability Standards Organization (SISO) 2020 Simulation Innovation Workshop (SIW), Orlando, FL.
[4] Hasan, O., Mendoza, A., Welch, J., Burch, B., & Geddes, J.A. (2023). *Incorporating Navigation Effects into Synthetic Environments for Improved Cyberspace Training*. 2023 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.
[5] Hasan, O., Welch, J., Burch, B., Geddes, J.A., & Vey, N. (2021). *A Cyberspace Effects Server for LVC&G Training Systems*. 2021 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.
[6] Wells, D., & Bryan, D. (2015). *Cyber Operational Architecture Training System Cyber for All*. 2015 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.
[7] Hasan, O., Crane, D., & Dukstein, G. (2024). *Incorporating Simulated Cyberspace Effects on Navy Shipboard Systems during Training Systems*. Simulation Interoperability Standards Organization (SISO) 2024 Simulation Innovation Workshop (SIW), Orlando, FL.
[8] Hasan, O., Crane, D., Welch, J., Geddes, J.A., Strauss, J., Bogler, W. C. (2024). *Development of a Novel Architecture for Improving Cyber-Kinetic Training*. 2024 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.
[9] https://www.librenms.org/
[10] https://github.com/librenms/librenms/blob/master/resources/definitions/alert_rules.json
[11] https://github.com/librenms/librenms/blob/master/resources/definitions/macros.json
[12] Mize, J., Marshall, H., Hooper, M., Wells, R., & Truong, J. (2015). *Cyber Operations Battlefield Web Services (COBWebS) – Concept for a Tactical Cyber Warfare Effect Training Prototype*. 2015 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.
[13] Merritt, M. (2019). *Network Effects Emulation System NE2S / Cyber Emulator.* 2019 Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC), Orlando, FL.