Technical Report
1187

Operational Exercise Integration Recommendations for DoD Cyber Ranges

N.J. Hwang
K.B. Bush

TBD 2015

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS

Prepared for the Assistant Secretary of Defense for Research and Engineering
under Air Force Contract FA8721-05-C-0002.

This report is based on studies performed at Lincoln Laboratory, a federally funded research and development center operated by Massachusetts Institute of Technology. This work was sponsored by the Assistant Secretary of Defense for Research and Engineering under Air Force Contract FA8721-05-C-0002.

The 66th Air Base Group Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

Massachusetts Institute of Technology
Lincoln Laboratory

Operational Exercise Integration Recommendations for DoD Cyber Ranges

N.J. Hwang
K.B. Bush
Group 59

Technical Report 1187

TBD 2015

Lexington Massachusetts

This page intentionally left blank.

EXECUTIVE SUMMARY

Cyber-enabled and cyber-physical systems connect and engage virtually every mission-critical military capability today. And as more warfighting technologies become integrated and connected, both the *risks* and *opportunities* from a cyberwarfare continue to grow—motivating sweeping requirements and investments in cybersecurity assessment capabilities to evaluate technology vulnerabilities, operational impacts, and operator effectiveness.

Operational testing of cyber capabilities, often in conjunction with major military exercises, provides valuable connections to and feedback from the operational warfighter community. These connections can help validate capability impact on the mission and, when necessary, provide course-correcting feedback to the technology development process and its stakeholders. However, these tests are often constrained in scope, duration, and resources and require a thorough and wholistic approach, especially with respect to cyber technology assessments, where additional safety and security constraints are often levied.

This report presents a summary of the state of the art in cyber assessment technologies and methodologies and prescribes an approach to the employment of cyber range operational exercises (OPEXs). Numerous recommendations on general cyber assessment methodologies and cyber range design are included, the most significant of which are summarized below.

- Perform bottom-up and top-down assessment formulation methodologies to robustly link mission and assessment objectives to metrics, success criteria, and system observables.
- Include threat-based assessment formulation methodologies that define risk and security metrics within the context of mission-relevant adversarial threats and mission-critical system assets.
- Follow a set of cyber range design mantras to guide and grade the design of cyber range components.
- Call for future work in live-to-virtual exercise integration and cross-domain modeling and simulation technologies.
- Call for continued integration of developmental and operational cyber assessment events, development of reusable cyber assessment test tools and processes, and integration of a threat-based assessment approach across the cyber technology acquisition cycle.

Finally, this recommendations report was driven by obsevations made by the MIT Lincoln Laboratory (MIT LL) Cyber Measurement Campaign (CMC) team during an operational demonstration event for the DoD Enterprise Cyber Range Environment (DECRE) Command and Control Information Systems (C2IS).[1] This report also incorporates a prior CMC report based on Pacific Command (PACOM) exercise observations, as well as MIT LL's expertise in cyber range development and cyber systems assessment.[2]

[1] CMC is explained in further detail in Appendix A.1.

[2] See References section at the end of the report.

This page intentionally left blank.

ACKNOWLEDGMENTS

The entire DD C5I DoD Enterprise Cyber Range Environment (DECRE) team was a gracious partner in this study. The authors would like to acknowledge Mr. Randy Coonts, Mr. Rod Hallum, Mr. Bert Daniel, LCDR Chris Werber, and Mr. Bob Summit for their hospitality and thoughtful insights provided during the event observations.

The authors would additionally like to acknowledge Dr. Greg Shannon of the Software Engineering Institute (SEI) at Carnegie Mellon University (CMU) for his insightful comments and feedback, and Dr. Stephen King of the Assistant Secretary of Defense for Research and Engineering (ASD(R&E)) for his guidance and support.

This effort was carried out under the auspices of the Cyber Measurement Campaign (CMC), which was funded by ASD(R&E) under Air Force contract FA8721-05-C-0002.

This page intentionally left blank.

TABLE OF CONTENTS

		Page
	Executive Summary	iii
	Acknowledgments	v
	List of Illustrations	ix
	List of Tables	xi
1.	INTRODUCTION	1
2.	CYBER ASSESSMENTS OVERVIEW	3
	2.1 Recommendations	4
3.	THREAT-BASED CYBER ASSESSMENT	7
	3.1 Terminology	8
	3.2 Metric Formulation	9
	3.3 Recommendations	13
4.	CYBER RANGE BACKGROUND	15
	4.1 Purpose and Motivation	15
	4.2 Composition and Architectures	15
	4.3 Use Cases	18
5.	CYBER RANGE DESIGN	21
	5.1 Design Mantras	21
	5.2 Support Tools	22
	5.3 Final Remarks	32
6.	GAPS	33
	6.1 Technology Gaps	33
	6.2 Process and Methodology Gaps	33
7.	SUMMARY	37
Appendix A: Background Material		39
	A.1 Cyber Measurement Campaign Background	39
	A.2 DECRE C2IS Background	39
References		41

This page intentionally left blank.

LIST OF ILLUSTRATIONS

Figure No.		Page
1	Hierarchy of assessment components. Assessments are best formulated with both a top-down focus (i.e., mission-centric) and bottom-up focus (i.e., technology-centric). Interrelating assessment components in this manner creates reuse opportunities across different scenarios and assessment efforts.	5
2	The Defense Science Board's 2013 cyber threat taxonomy [16].	10
3	Mandiant's 2014 threat landscape [17].	11
4	Mandiant's Attack Lifecycle model [18].	11
5	Illustration of core elements of a cyber range relative to core elements of a data center.	16
6	Decomposition of operational spaces within any cyber range.	17
7	Alternative architectures reflecting three fundamentally different approaches: (1) a monolithic range that serves all user bases, (2) a federated model that composes range capabilities on demand, and (3) an ad-hoc distributed approach that tightly couples user communities to specialized ranges that satisfy their needs.	18
8	Cyber range use cases relative to cyber capability maturation model illustrating overlap of technical and operational objectives relative to development, testing, training, and exercise activities.	19
9	Illustration of varying levels of exercise integration between real-world assets, range infrastructure, and modeling and simulation tools.	34
10	Recommended integration between S&T and operational testing [21].	35
11	Test Resource Management Center process.	36

This page intentionally left blank.

LIST OF TABLES

Table No.		Page
1	Summary of MITRE's Common Weakness Enumeration [10].	13
2	Tradeoffs of different types of data generators.	23

This page intentionally left blank.

1. INTRODUCTION

Cyber warfare is here. Computer information systems, networks, and cyber-physical systems (CPSs) already connect and engage virtually every mission-critical military capability, from logistics/supply and communications to kinetic platforms and weapon systems. As more warfighting technologies become integrated and connected, both the *risk* and *opportunities* from a cyber warfighting perspective grow proportionally with Metcalfe's Law [1].

With this potential has come a flood of requirements from the Department of Defense (DoD) to integrate cyber warfighting capabilities into the Department's existing science and technology (S&T) [2], acquisition [3], and operational lines of effort [4]. However, the cyber assessments methodologies—and more broadly the science of cybersecurity—necessary for thorough and accurate evaluations of cyber warfighting capabilities remain in their infancy. Arguably there has never been a greater civilian and military dependence on a domain of which so little is systematically understood and so few capabilities have been rigorously tested.

Drawing upon a great body of prior work in other warfighting domains [5], the cyber technology community has been investing in the development and deployment of cyber ranges [6, 7] as controlled and instrumented environments for in vitro cyber capability testing and training objectives. Most importantly, the introduction of cyber ranges have facilitated technical and operational experimentation under controlled conditions that would otherwise be prohibitive on the real systems in situ, such as when

- Access to the real networks and systems cannot be reliably obtained (e.g., an uncooperative network or mission-sensitive system)
- The risk from collateral effects cannot be tolerated (e.g., critical infrastructure or one-of-a-kind equipment)
- The real systems are too complex for experimentation (e.g., parameter isolation, system instrumentation, data saturation)

Additionally, the cyber domain itself presents special challenges to traditional approaches to technology assessments and operator training events. Intrinsic domain properties such as complexity (nonlinearity), scale, geographic distribution, and logical decentralization compound the challenges of creating repeatable cyber assessments and reliable training scenarios. Accordingly, there is an existing and growing need for high fidelity and configurable environments that meet the challenges of emulating real-world networks and CPSs while still integrating with principal operational exercise events. Ergo, cyber ranges will play an increasingly central role in cyber assessments, especially for developmental and operational testing of system security. Studying the lessons learned and results of cyber assessment events can be informative as to the requirements and qualities of an effective cyber range.

This recommendations report is based upon observations made by the MIT Lincoln Laboratory (MIT LL) Cyber Measurement Campaign (CMC)[3] team during an operational demonstration event for the DoD Enterprise Cyber Range Environment (DECRE) Command and Control Information Systems (C2IS). This report also incorporates a prior CMC report based on Pacific Command (PACOM) exercise observations, as well as MIT LL's expertise in cyber range development and cyber systems assessment.

The remainder of this report is organized as follows: Further background in assessment science is provided in Section 2; discussion of the pinnacle role threat models play in cyber assessments is provided in Section 3; Section 4 explores the roles for cyber ranges within operational exercises; Section 5 provides a survey of the important qualities of cyber ranges as an assessment tool; Section 6 outlines technology gaps discovered during the authors' exercise integration event observations as well as proposed nontechnical advancements to the field; and Section 7 provides some high-level conclusions and remarks.

[3] MIT LL Cyber Measurement Campaign is described in further detail in Appendix A.1.

2. CYBER ASSESSMENTS OVERVIEW

The science of cyber assessment is rooted in traditional systems engineering methodologies [8]. Whether the assessment is for an immature research prototype or an operationally hardened system, designing and executing an effective cyber assessment begins with defining the components of a traditional test and evaluation event. We briefly review what each of these components are below.

Cyber assessment events evaluate one or more *assets* within the context of a *mission*, potentially working in concert to comprise a cyber system or environment (henceforth referred to as a *system under test*, or SUT). A SUT's complexity can range from a single software application to a distributed network of connected mission enclaves. A SUT may include any element of the computational stack, such as computing hardware, software applications, networking facilities, storage technologies, and even the operators and processes that interface with various assets in the SUT.

Assessment objectives must first be defined before continuing with the design of the assessment. The objectives must specify what aspects of the SUT will be characterized and for what purpose.

SUTs are traditionally characterized for

Capabilities: i.e., how effectively does the SUT support the mission?
Performance: i.e., how quickly and efficiently can the SUT perform its capabilities?
Risk: i.e., how severely can the SUT fail, and how likely is it to fail?
Security: i.e., how securely can the SUT perform its capabilities?

These characterizations are traditionally performed for the purposes of

Verification: i.e., does the SUT meet its design requirements?
Validation: i.e., does the SUT fulfill its intended purpose?
Exploitation: i.e., how vulnerable is the SUT?
Mitigation: i.e., how defensible is the SUT?

The assessment event is comprised of a set of *scenarios* that represent different environments and use cases that the SUT would realistically experience and encounter during the mission. Scenarios allow an event to be decomposed into logical sub-events, each with a narrower scope and focus than the entire event. Depending on the assessment's objectives, variations may include different network topologies (e.g., assessing performance when using local computation versus outsourced cloud computation), adversarial models (e.g., assessing security against an internal versus an external adversary), active countermeasures (e.g., assessing system availability with and without a defensive blue team), and much more.

A SUT is assessed based on a set of *metrics* that have been especially designed to characterize the capabilities, performance, risk, or security of the SUT. Metrics are calculated using *measures* that are collected or sampled during the execution of each scenario. Metrics must therefore, by definition, be empirical in nature—both observable and measurable.

Measures are traditionally defined as either *measures of performance* (MOPs) or *measures of effectiveness* (MOEs) [9]. MOPs are quantitative measurements of a SUT's performance and

efficiency, such as operation speed and throughput or system resource consumption. MOEs are measurements of a SUT's capabilities and utility within a mission context, such as the SUT's expected uptime or requisite operator skill level. While MOEs are not always quantitative in nature (e.g., requisite operator skill level), they are always based on observations made during an assessment and therefore never completely subjective.

An assessment may impose *success criteria* for some metrics, which are specific threshold values the SUT is expected to achieve or surpass during the assessment. These are typically used to succinctly inform sponsors and customers on how well a SUT is expected to perform in the field, and are frequently deciding factors for future project funding. Success criteria are commonly defined as a quantitative improvement over a known baseline. This baseline may be another technology comparable to the SUT (e.g., to compare competitors' offerings), an older version of the SUT (e.g., to measure improvements over time), the SUT's design requirements (e.g., to perform quality assurance testing), or a subjectively defined level of achievement (e.g., when there is no comparable technology for novel research).

2.1 RECOMMENDATIONS

If any of the above assessment components are not clearly defined, assessments can become easily unorganized and misguided. Ensuring that an appropriate set of objectives, scenarios, metrics, and measures is defined is critical for aligning the assessment with eventual end users and mission objectives. The following recommendations are provided to assist the formulation and value maximization of a cyber assessment.

Top-down formulation. Formulate the assessment starting with scenario objectives and how the SUT will be used in the field. Use these objectives to identify success criteria that will serve as indicators of mission success in the context of the assessment scenario. From well-defined success criteria, define MOEs that measure the SUT's ability to facilitate mission success. From MOEs, derive MOPs that measure how quickly and efficiently the SUT can perform its functions. Finally, refine MOPs into quantitative metrics can that be observed or measured during the assessment. This approach tends to better align the assessment with the end user and operational use. See Figure 1.

Bottom-up formulation. Formulate the assessment starting with the SUT assets and identifying what metrics can be measured. Use these metrics to identify quantitative MOPs that roll up into operationally oriented MOEs. Then compose MOEs into success criteria thresholds that can serve as indicators of mission success in the context of the assessment's objectives. Finally, build scenarios that will simultaneously permit metric capture while aligning with SUT objectives. This approach tends to better align with more developmental or technical assessments where more comprehensive data capture is necessary. See Figure 1.

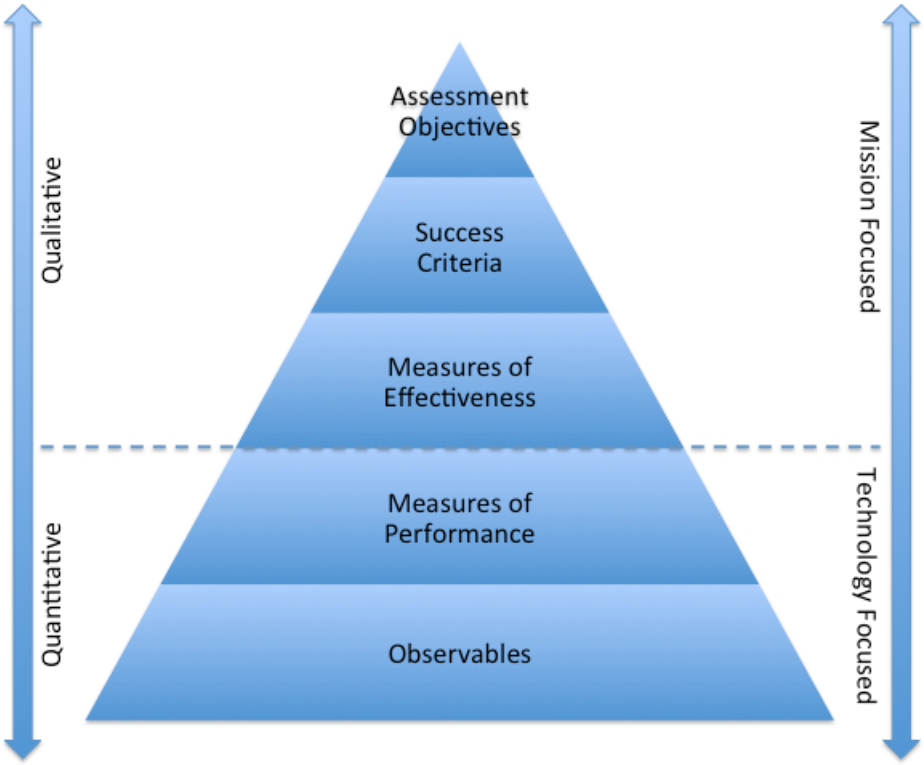


Figure 1. Hierarchy of assessment components. Assessments are best formulated with both a top-down focus (i.e., mission-centric) and bottom-up focus (i.e., technology-centric). Interrelating assessment components in this manner creates reuse opportunities across different scenarios and assessment efforts.

Mapping operational metrics to technical metrics. Metrics that measure the SUT's ability to fulfill a particular mission need should be mapped and based upon lower-level metrics that measure a SUT's capabilities, performance, security, and risk. This is conducive to leveraging developmental testing results when performing operational tests and exercises, and also assists with formulating assessments involving a composition of previously assessed SUTs or assets. Interrelating and reusing metrics to define higher-level metrics can provide better continuity amongst multiple system acquisition efforts.

Allowing metrics and success criteria to evolve. Metrics and success criteria are rarely correctly or robustly defined at their inception, particularly for developmental testing of research prototypes. Metrics and success criteria should enhance, not inhibit, the development and implementation of new systems and technologies. The best assessments result in great systems being transitioned to practice; mediocre assessments result in systems that just meet success criteria. Therefore, it is an acceptable practice to refine metrics and success criteria as an assessment effort matures and evolves to ensure that they are in line with the overarching mission goals and end users.

Metric characterization. Metric quality can be roughly assessed by using some of the criteria below:

- Objectivity: Can the metric be repeatably calculated from a scenario's results, independent of the individual performing the calculation? Does the metric rely on subjective "gut checks" or "eye tests?"
- MOE versus MOP: Are a great majority of the metrics MOPs? MOEs are often much harder to define, but must be a part of any assessment that expects to gain insight into the utility and impact a SUT can have within a mission.
- Relative versus absolute evaluation: Does the metric's value stand on its own, or is it relative to some baseline or success criteria? Are these baseline and success criteria well-defined?
- Quantitative versus qualitative evaluation: Is the metric based on calculations on raw data, or a partially subjective determination? Are qualitative factors well-characterized by a rubric or enumerated set of values (e.g., evaluating a user interface's usability on five-point scale)?
- Monotonicity: Does the metric represent a monotonic (i.e., nonincreasing or nondecreasing) representation of the SUT's properties? Can a decision maker make decisions by optimizing this metric in a particular direction, with all other considerations remaining equal? Metrics should clearly express whether a metric value is "better" or "worse" than another value with respect to mission objectives, and in general should be a monotonic function of its inputs.
- Relevance: Does the metric represent anything relevant to the overall mission objectives and end users? Does the absence of a metric impact a sponsor or customer's ability to make decisions with regards to the SUT?

Threat-derived risk and security metrics. While capabilities and performance characteristics are intrinsic properties of a SUT, risk and security are not. This makes defining metrics that evaluate a SUT's risk of failure and security against a broad spectrum of adversaries a difficult task. It is recommended to define risk and security metrics within the context of a well-defined threat model, and to formulate assessment scenarios based on this threat model. An effective threat model will capture information about the types of relevant adversaries, their resources and intentions, as well as their mechanisms of attack. We delve into greater detail on how to perform threat-based cyber assessments in the next section.

3. THREAT-BASED CYBER ASSESSMENT

While the science of cyber assessment is strongly rooted in traditional systems engineering methodologies, challenges arise in the definition of robust measurements of security and risk. Unlike a SUT's capabilities and performance, a SUT's security and risk are not intrinsic properties of the SUT, as they are very much intertwined with externalities that must be accounted for in any security or risk assessments.

For cyber assessments, the most important externality affecting a SUT's security is the notional adversarial threat that is attacking the SUT.[4] While there are many compilations of possible SUT vulnerabilities [10, 11] that can be used to characterize the security posture of any particular SUT, this fails to account for the likelihood of an adversary's success in exploiting any of these vulnerabilities. Focusing on measuring a SUT's vulnerabilities also tends to draw attention away from the actual assets being protected and the implications of any of those assets being compromised. This is akin to validating the technical specifications of a new car without evaluating the cost of potential repairs and the risk to human life. Moreover, while there may be a set of known vulnerabilities for a particular SUT now, adversaries are continually discovering new vectors of attack that will only cause the set of known vulnerabilities to grow over time. In order to have meaningful security assessment, at least some consideration must be given to the adversaries of concern and risks that they present. Focusing only on the SUT's design and operation is therefore not sufficient for the development of cyber assessment security and risk metrics.

The end user of the SUT is ultimately interested in how well they can accomplish their mission goals, particularly in a contested or compromised environment, and the SUT's assessment must provide useful indicators of this utility to the end user. *Threat modeling*, the practice of systematically characterizing both malicious and unintentional perterbations from operational objectives, is therefore a crucial component in formulating a cyber assessment.

The broad spectrum of adversary types, the resources they have, and their intents can make it difficult to practically scope an assessment. This is all the more reason to have concretely defined assessment objectives and to define relevant and focused threat models for the SUT based on these objectives. Characterizing a SUT's security against "script kiddies" versus an advanced persistent nation-state threat versus an insider threat all require very different assessments. Assessments simply cannot answer the question, "Is the SUT secure?" Assessments must be framed relative to a threat to answer, "Is the SUT secure against this particular category of threat?"

In the remainder of this section, we discuss general methodologies for formulating a threat-based cyber assessment, with a focus on the characterization of a SUT's security (although risk characterization can follow similar processes).

[4] Note that this section specifically addresses assessment objectives involving the security and risk of the SUT itself. In some cases, assessment objectives involve determining how well a SUT can protect a set of assets. The discussion in this section still holds if the notional adversarial threat is attacking the protected assets instead of the SUT itself.

3.1 TERMINOLOGY

We first review some terms that will be used in the remainder of the section.

Security terms. We decompose the "security" of a cyber system into three types of security [12]:

- Confidentiality: Protection against unauthorized or undesired data disclosure.
- Integrity: Protection against unauthorized, undesired, or unauthentic data modification or service behavior.
- Availability: Protection against denial of reliable or timely access to data and services.

Threat model terms. The following is a commonly accepted nomeclature that we use to discuss and define threat models [13]:

- Threat: Potential occurence (malicious or otherwise) that may harm an asset (e.g., using SQL injection to exfiltrate database contents).
- Vulnerability: Weakness that makes a threat possible (e.g., weak SQL command validation).
- Attack: Action taken to exploit vulnerabilities and realize threats (e.g., exfiltration of military unit movement history from a situational awareness tool's records).
- Adversary: Actor conducting attacks (e.g., advanced persistent nation-state performing military espionage).

Defense capability terms. The following is a commonly accepted nomenclature for ways a system can counteract attacks, usually referred to as the "5 D's" [14]:

- Deter: Deter adversaries from conducting attacks. Difficult to measure in assessments, but decreases in attacks could measurably be attributed to the appearance of strong defenses and countermeasures (e.g., indicating a high risk of adversary detection or exposure).
- Detect: Detect attacks and provide notification to responders (e.g., using intrusion detection systems).
- Deny: Prevent attack from materially affecting assets (e.g., using firewalls).
- Delay: Slow down attack progression (e.g., using IP or port hopping to provide a moving target).
- Defend: Mitigate severity of realized threat (e.g., shutting down network capabilities upon intrusion detection).

SUT property terms. We finally define 'SUT security' and 'SUT risk' as follows when discussing security and risk metrics:

- SUT Security: SUT's ability to detect, deter, deny, delay, or defend attacks attempting to compromise its assets' confidentiality, integrity, or availability.
- SUT Risk: Likelihood and severity of a threat realizing one or more SUT vulnerabilities.

3.2 METRIC FORMULATION

Given the terminology above, we recommend the following process for formulating security and risk metrics [15]:

1. Define and characterize the threat by specifying capabilities and goals (i.e., threat characterization).
2. Define possible outcomes resulting from the threat attempting to reach different goals (i.e., asset valuation).
3. Characterize the system being protected, including vulnerabilities and defenses that are relevant to the specific threat (i.e., vulnerability assessment).
4. Analyze the risk using statistical approaches to determine probabilities and the damage for different outcomes and the effect of relevant defenses (i.e., risk assessment).

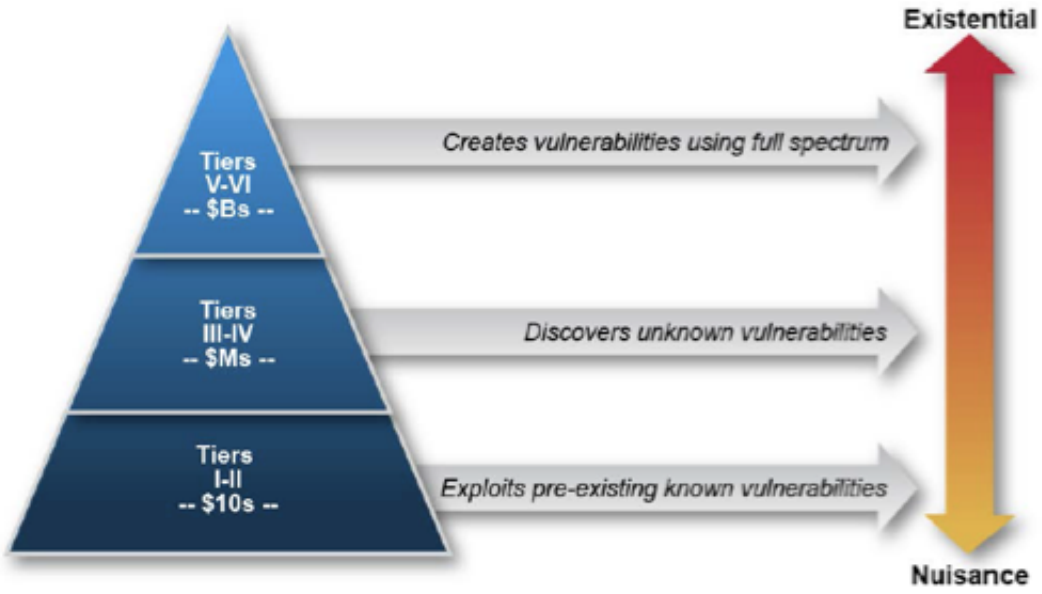
3.2.1 Threat Characterization

Defining and characterizing threat capabilites and goals can be inspired from public works, such as the Defense Science Board's 2013 cyber threat taxonomy [16] and Mandiant's 2014 threat categorization [17] (see Figure 2 and Figure 3).

The above tiers and categories can be used to help first define useful assessment objectives and scenarios, as well as particular assets to focus on during the assessment. Assessing a SUT's security or risk against each of these adversary types will require varying types of synthetic test data, user roles, attack frequencies, background traffic, and other assessment parameters, and can inform the initial design of range infrastructure and support tools.

Threat characterization is particularly important when incorporating "red teams" into an assessment, as the adversaries' capabilities and resources must be appropriately emulated during the assessment. This can be the difference between 1) allowing experienced penetration testers to have months of access to the SUT prior to the exercise (e.g., emulating meditated network attacks from an advanced persistent threat), and 2) providing junior engineers access to the SUT at the start of the assessment (e.g., emulating nuisance attacks from a "script kiddie" threat).

In addition to characterizing the threat's objectives and resources, it is also useful to define their attack patterns as a basis for the assessment's scenarios. This can help narrow the scope of



Tier	Description
I	Practitioners who rely on others to develop the malicious code, delivery mechanisms, and execution strategy (use known exploits).
II	Practitioners with a greater depth of experience, with the ability to develop their own tools (from publically known vulnerabilities).
III	Practitioners who focus on the discovery and use of unknown malicious code, are adept at installing user and kernel mode root kits, frequently use data mining tools, target corporate executives and key users (government and industry) for the purpose of stealing personal and corporate data with the expressed purpose of selling the information to other criminal elements.
IV	Criminal or state actors who are organized, highly technical, proficient, well funded professionals working in teams to discover new vulnerabilities and develop exploits.
V	State actors who create vulnerabilities through an active program to "influence" commercial products and services during design, development or manufacturing, or with the ability to impact products while in the supply chain to enable exploitation of networks and systems of interest.
VI	States with the ability to successfully execute full spectrum (cyber capabilities in combination with all of their military and intelligence capabilities) operations to achieve a specific outcome in political, military, economic, etc. domains and apply at scale.

Figure 2. The Defense Science Board's 2013 cyber threat taxonomy [16].

	NUISANCE	DATA THEFT	CYBER CRIME	HACKTIVISM	NETWORK ATTACK
Objective	Access & Propagation	Economic, Political Advantage	Financial Gain	Defamation, Press & Policy	Escalation, Destruction
Example	Botnets & Spam	Intellectual Property Theft	Credit Card Theft	Website Defacements	Destroy Critical Infrastructure
Targeted	✗	☑	☑	☑	☑
Character	Automated	Persistent	Opportunistic	Conspicuous	Conflict Driven

Figure 3. Mandiant's 2014 threat landscape [17].

the assessment to combating adversarial reconnaissance attempts, or focusing the assessment on how quickly SUT security can be recovered in the midst of an attack, or both. Mandiant's Attack Lifecycle model is included below as an exemplar attack pattern that can be used as the basis for most attack patterns.

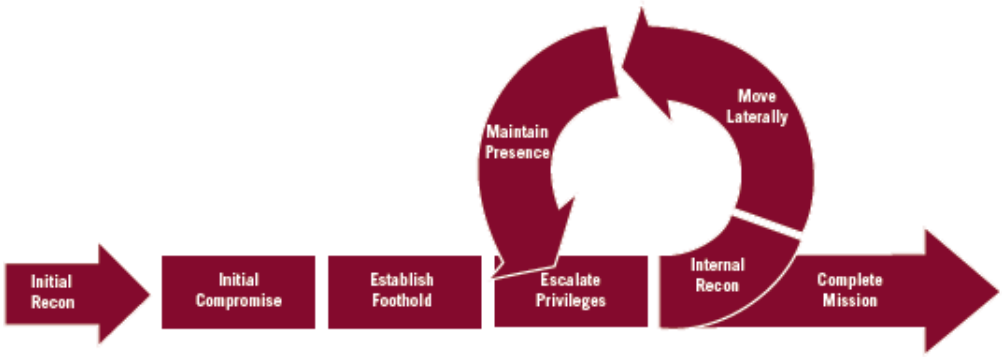


Figure 4. Mandiant's Attack Lifecycle model [18].

Overall, specifying the adversary types of interest is a crucial first step in formulating and planning the assessment.

3.2.2 Asset Valuation

Provided a set of adversary types, one can begin characterizing the possible outcomes of an attack, which entails defining values for the SUT's assets.

Asset valuation should start by considering the three aforementioned aspects of security: confidentiality, integrity, and availability. In the context of the chosen threats to use in the assessment, one can begin identifying critical assets and data and categorizing them by importance. These are the beginnings of the assessment's metrics, as these will eventually be immediate indicators of the damage done to the SUT during assessment scenarios.

For instance, a cyber assessment evaluating the SUT's security against an insider threat attempting to exfiltrate information can rank different confidential data stores or files by their importance to the SUT's mission. This ranking can then be used to rate each emulated exfiltration attempt's severity and success whenever data leaves the SUT's boundaries.

Another example is the assignment of a measure of SUT health based on the availability of critical services, such as connectivity or situational awareness. One more example would be assigning value to individual fields in a network protocol in order to rate the effectiveness of an adversary's attempts to manipulate and corrupt the integrity of packets.

3.2.3 Vulnerability Assessment

After identifying the SUT's critical assets of interest, one can begin characterizing ways a threat can attack and exploit the SUT to achieve the adversary's goals. This involves identifying the SUT's vulnerabilities, as well as identifying appropriate attack patterns for the adversary.

A useful starting point is MITRE's Common Attack Pattern Enumeration and Classification (CAPEC) [19], which consolidates dozens of threat models and attack characertizations into a unified schema. Provided a good understanding of SUT operations and its attack surface, one can select appropriate "mechanisms of attack" based on asset and adversarial capabilities. MITRE's Common Weakness Enumeration (CWE) [10] and Common Vulnerabilities and Exposures [11] databases are also useful references when attempting to characterize a SUT's attack surface.

For the reader's reference, the CWE's high-level mechanisms of different cyber attacks are listed in Table 1. Note that many attacks may fall into multiple categories.

After formalizing the attack vectors that will be observed during the assessment, metrics can be defined that characterize the SUT's security against each attack vector. The "5 D's" (deter, detect, deny, delay, and defend) are recommended starting points for defining these metrics. For example, a measure of "denial and deterrence capabilities" could be the number of unauthorized accesses to a particular directory, while a measure of "delay capabilities" could be the latency required for an adversary to gain access to that directory. A measure of "detection capabilities" could be the precision and recall of a security information and event management (SIEM) system's notifications of unauthorized access to that directory. Finally, a measure of "defense capabilities" could be the total value of files that are opened for reading or are moved to an egress point on the SUT for exfiltration.

3.2.4 Risk Assessment

The final step in developing security and risk metrics involves building probabilistic models of the likelihood and severity of threat realization. Using the results of the threat characterization and asset valuation steps above, one can formalize expressions that attempt to quantify and predict how frequently and how severely assets can be compromised.

For example, one can model the frequency and duration of attack attempts using a Poisson distribution based on empirical data gathered from the field. Based on the behavior of SUT countermeasures during the assessment, metrics can be built that quantify how many attack attempts

Table 1. Summary of MITRE's Common Weakness Enumeration [10]

Mechanism	Example(s)
Gather Information	Sniffing network traffic, port scanning
Deplete Resources	TCP flooding, memory leak exploitation
Injection	Cross-site scripting attacks, SQL injection
Deceptive Interactions	Signature spoofing, phishing attempts
Manipulate Timing and State	Forced deadlock, race condition exploitation
Abuse of Functionality	Disabling security measures, inducing account lockout
Probabilistic Techniques	Brute force cryptanalysis, form input fuzzing
Exploitation of Authentication	Cookie modification, reusing prior authenticated sessions
Exploitation of Authorization	Privilege escalation, environment variable modification
Manipulate Data Structures	Buffer overflows, record corruption
Manipulate Resources	Executable modification, log tampering
Analyze Target	Binary or protocol reverse engineering attempts
Gain Physical Access	Circuit board probing, removable media access
Malicious Code Execution	Worm or trojan execution
Alter System Components	Disabling cooling fans, PCI card installation
Manipulate System Users	On-screen distractions

the SUT will be able to respond to, and the extent or probability of success in deterring, detecting, denying, delaying, or defending the attack. Similar metrics can be used to evaluate the efficacy of "blue teams" assigned to defend a network by measuring the latency required to patch or eradicate vulnerabilities after detection.

3.3 RECOMMENDATIONS

The following recommendations may be useful when formulating cyber assessment metrics for security and risk.

Constrain assessment scope to the identified threat models of interest. An assessment can not and should not cover the gamut of potential threats and attack vectors for the SUT. In today's fiscally constrained environment, effective assessments hinge upon effective scoping. This is especially true for cyber technologies whose attack surfaces are constantly in a state of flux as the cyber warfront evolves. Firmly identifying the relevant threats and attack patterns for a particular assessment allows resources to be efficiently dedicated towards simulating and/or emulating the appropriate threat behaviors and instrumenting the appropriate SUT components. This is far more likely to conclude with valuable insight into the evolution and operational readiness of the SUT. Assessments attempting to demonstrate attacks against a high percentage of the SUT's known attack surface often resort to low-fidelity demonstrations that are compressed within a limited timeframe and yield very small datasets.

Horizontal and vertical brainstorming across attack mechanisms. Significant effort should be devoted to brainstorming various permutations of mission objectives, user scenarios (i.e., proper use of the SUT), active SUT capabilities, and threat objectives (i.e., abuse versus misuse of the SUT) to achieve awareness of the known space of attack mechanisms to incorporate into the assessment. Horizontal brainstorming (e.g., identifying different categories) and vertical brainstorming (e.g., identifying different items within a category) are useful constructs that help ensure coverage of a problem space. Assessment planners should not forget to consider elements of the physical domain as well; prior work on electronic warfare threat modeling provides the option of developing convergent threat models with the cyber threat model methodologies outlined above.

Build metrics that can detect improvements in security and risk posture. How do the metric values change when new security and risk countermeasures are installed or activated? Metrics that do not "move the needle" for SUT security and risk mitigation as the SUT evolves are likely to detract from the cyber assessment's core objectives.

4. CYBER RANGE BACKGROUND

The notion of a *cyber range* is relatively new and warrants focused attention and consideration. This section provides background on the definition, purpose, use cases, and description of cyber ranges.

4.1 PURPOSE AND MOTIVATION

Cyber ranges are synthetic constructed environments designed to emulate real-world digital networks to meet developmental, testing, training, or exercise requirements within the cyber warfighting community. Similar to familiar missile, firing, or flight ranges, cyber ranges provide controlled and instrumented environments for in vitro cyber capability employment, study, and assessment.

In contrast to experimentation done on the real systems (in situ) or in computational simulation (in silico), cyber ranges provide a flexible middle degree of fidelity, allowing for a rich and efficient balance between repeatability and environmental fidelity. Cyber ranges can be further tuned and tailored to the objectives at hand, thereby delivering a breadth of scale, fidelity, and instrumentation for many use cases.

Perhaps most importantly, cyber ranges facilitate technical and operational experimentation under controlled conditions that would otherwise be prohibitive on the real systems/networks in situ, such as when (1) access to the real networks cannot be reliably obtained (e.g., an uncooperative network or mission-sensitive system); (2) the risk from collateral effects can be tolerated (e.g., critical infrastructure or irreplaceable one-of-a-kind equipment); and (3) the real systems are simply too complex for experimentation (e.g., when proper instrumentation cannot be done, or when the instrumentation becomes data saturated).

4.2 COMPOSITION AND ARCHITECTURES

Cyber ranges can be generally described as the composition of four key data center elements, expanded and tailored to serve the specific requirements of a range (outlined below and illustrated in Figure 5).

- **Facilities:** The physical sites that host common data center services such as rack space, power, and cooling equipment. Additional range-specific facilities may include operator training and exercise spaces.
- **Assets:** The collection of computing and networking equipment, such as servers, routers, and load balancers, necessary to satisfy the computational requirements. Additional range-specific assets may include specific target systems, tactical or mission systems, and related infrastructure.
- **Workforce:** The team of core data center service operators, such as system administrators, engineers, and management. Expanded range teams may include test design engineers,

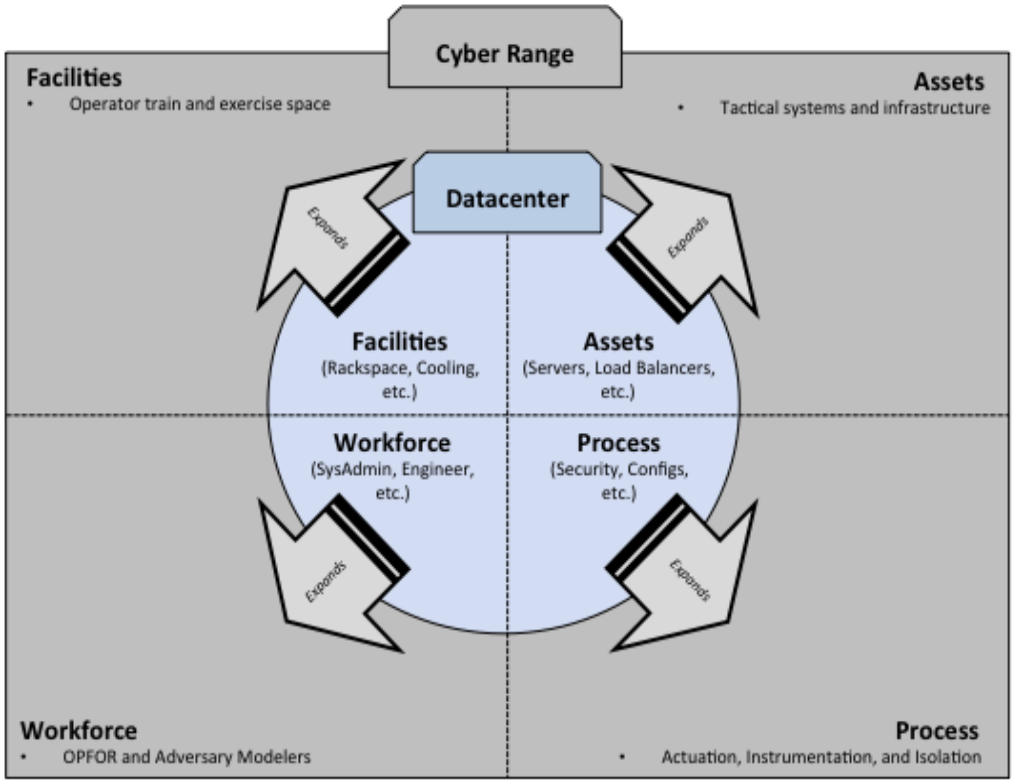


Figure 5. Illustration of core elements of a cyber range relative to core elements of a data center.

analysis and modeling experts, warfighting domain experts, and opposition force (OPFOR) adversary emulators.

- **Process:** The collection of business processes that keep a data center running, such as change configuration management, security procedures and policies, and customer management. Expanded range-specific processes include range actuation (e.g., traffic generation), instrumentation, and isolation assets.

Another valuable way to decompose the elements of a cyber range is by the spaces they occupy. The common cyber range maintains and operates assets in three unique, potentially distributed, spaces. To the degree that cyber ranges are distributed, they necessarily require a secure and high-speed interconnect amongst the spaces to operate effectively (outlined below and illustrated in Figure 6).

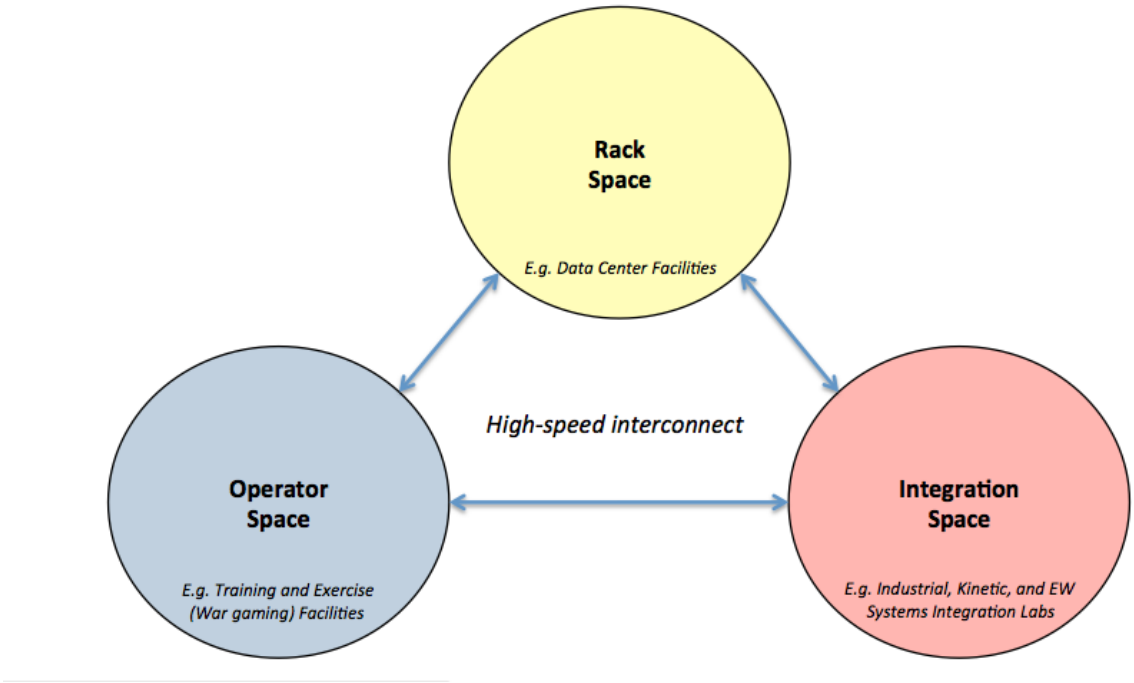


Figure 6. Decomposition of operational spaces within any cyber range.

- **Rack Space:** The rack space comprises the data center facilities, assets, workforce, and processes. In a distributed cyber range, the rack space is often identified as the "site location" because it is where the computational assets are centered.
- **Operator Space:** The operator space provides access to the range as a user. During SUT experimentation involving live OPFOR or blue-team defenders, or during training events, or exercise integration events, the operator space is the location from which those participants engage.
- **Integration Space:** Integration spaces are relatively new elements to cyber ranges and are often at distributed sites—usually co-located with System Integration Laboratories (SILs). Integration spaces provide the means by which cyber ranges are integrated with ranges of other domains to provide true cross-domain experiments. When integrated with major military exercises, the integration spaces are distributed amongst the exercise, connected to the broader live event.

Many design parameters of cyber ranges reflect fundamental trade-offs between generalization and specialization. As the DoD continues to invest in a diverse and geographically distributed array of cyber ranges, thought must be given to the high-level architecture and business processes for their employment in order to maximize their strengths. Figure 7 illustrates three alternative architectures.

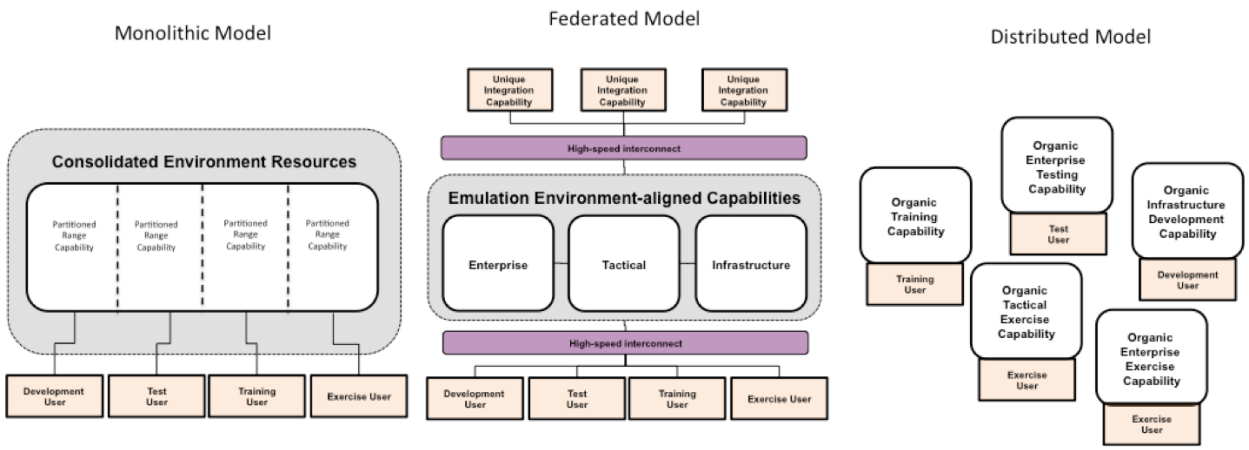


Figure 7. Alternative architectures reflecting three fundamentally different approaches: (1) a monolithic range that serves all user bases, (2) a federated model that composes range capabilities on demand, and (3) an ad-hoc distributed approach that tightly couples user communities to specialized ranges that satisfy their needs.

4.3 USE CASES

When leveraged for test and evaluation objectives, cyber ranges provide an operationally relevant and technically representative test harness in which to conduct cyber security assessments. Ranges can be used to support the following activities:

- **Development and Testing:** The technical and operational activities as part of the production of fieldable cyber or cyber-accessible technologies.
- **Training and Exercises:** The operational activities that support the operator community requirements for personnel development; tactics, techniques, and procedures (TTPs) development; and mission rehearsals.

While unique in their own respects, these use cases have a high affinity for each other as well as similar environment requirements. This offers opportunities for consolidating community resources and developing well-optimized cyber ranges that simultaneously support development, testing, training, and exercise activities. An important distinction can be made between training the operators (e.g., commanders and their staff), where operational impact matters most, relative to training the cyber defenders, where the trail of bits matters most. While these are both training activities, they require fundamentally different range capabilities. Testing support to combatant commands (CCMDs) and testing support to Program Executive Offices (PEOs) have fundamentally different objectives and technical requirements despite sharing a common assessment goal.

Recent trends and events suggest that cyber ranges can be used for more than the testing of SUTs. Increasingly, cyber ranges are being used to create an event environment in which the purpose of that event is external to the range itself. Furthermore, cyber ranges are being used as training tools for cyber warriors and network defense teams—offering enough richness for operators

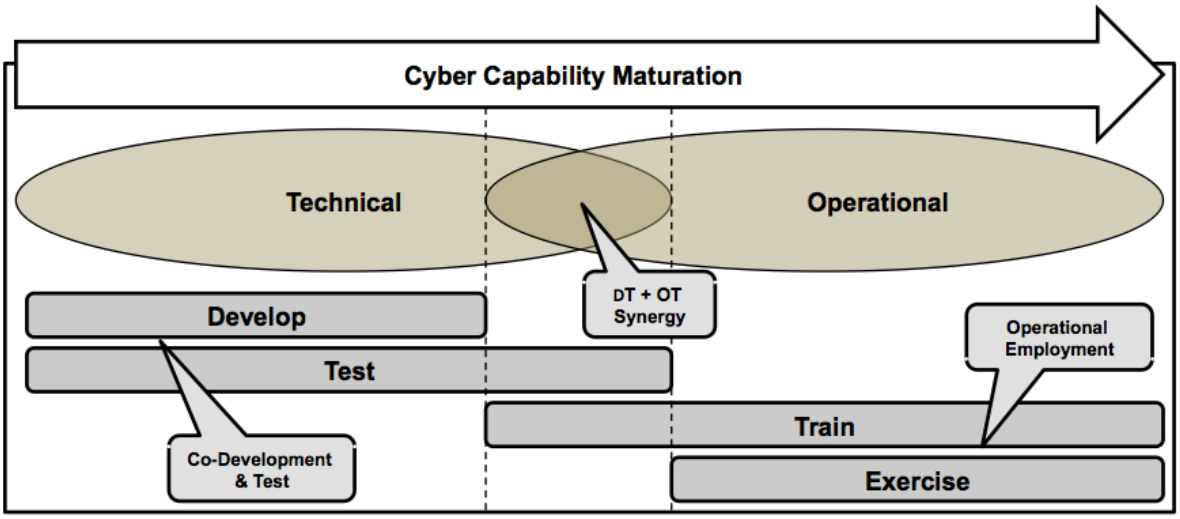


Figure 8. Cyber range use cases relative to cyber capability maturation model illustrating overlap of technical and operational objectives relative to development, testing, training, and exercise activities.

to engage their tools and development skills while still providing the safety margins afforded by a synthetic sandbox. Further, cyber range tests are being used to rapidly acquire experimentally controlled sample points from which cyber defensive strategies and can be generalized.

This page intentionally left blank.

5. CYBER RANGE DESIGN

5.1 DESIGN MANTRAS

The following design mantras are fundamental principles that should be incorporated into all software test and evaluation efforts, first outlined by Varia et al. [20]. We reproduce and slightly amended them here as valuable principles that apply broadly to cybersecurity assessments and the test harnesses that power them. These principles can help guide the design and development of any cyber range test harness.

Black-box treatment of systems. Do not depend on the low-level operation of range systems or systems under test. Instead, expose and document interfaces these systems use to interact with each other.

Event-time agility. Time during formal events is precious. When developing range infrastructure, spend a week of work to save a few hours of event time.

End-to-end automation. Build turn-key interfaces with which an operator can define scenario parameters, push a start button, watch a scenario execute, and push a stop button to end the scenario and collect any necessary data. This allows an operator to focus on a minimal set of user interfaces, as opposed to setting up, monitoring, and tearing down an entire chain of disparate appliances.

Situational awareness. Operators should be able to quickly and easily (1) determine the status and progress of each scenario, and (2) inspect the state of range systems, systems under test, and the range environment.

Minimal infrastructure overhead. Ensure that range systems (e.g., instrumentation, data collectors, system actuators) do not significantly impact any systems under test or influence any metrics being captured.

Repeatability. Scenarios should be fully repeatable. Scenario data, input parameters, and environment configurations should be fully recoverable and quickly reproducible.

Rapid reconfigurability. Build rapidly deployable range systems within a plug-and-play range infrastructure wherein an operator can quickly change range systems, systems under test, scenario parameters, or the range environment (e.g., network topologies, node hardware resources, bandwidth limitations).

Extensibility. To accommodate new scenario and event requirements, enable rapid addition of new range systems, systems under test, scenarios and scenario parameters, and range configurations.

Complete traceability. All scenario inputs, environment/system configurations, events, and results should be fully captured and methodically archived to facilitate robust post-event analysis. All events and measures of interest should be fully traceable to a particular time, set of inputs, and an environment configuration.

Smoke testing. Prepare tests that can quickly prove the stability and interoperability of the range and systems under test. These can rapidly identify simple configuration issues and reduce risk early in the event.

Reusability. To the extent possible, build range systems that enables reuse by future range events.

Tunable fidelity. Test results are always more relevant when test inputs are informed and modeled after real-world use cases. However, high-fidelity test inputs are not cheap to procure or construct, nor are they always necessary to support a given scenario. Ranges should therefore support varying degrees of system fidelity and allow range operators to select the "right fidelity in the right place at the right time."

5.2 SUPPORT TOOLS

Regardless of whether a range is supporting a test or training event, or whether an event consists of pure software systems or involves cyber-physical systems, the design of a harness that event operators use to facilitate event execution within the range can be generally decomposed into these appliance categories:

- Data generation: offline or real-time input data generators
- Instrumentation: sensors that collect measurements relevant to the event that are not otherwise emitted as event artifacts
- Configuration management: appliances that enforce range configurations per a given scenario's prerequisites
- Orchestration: appliances that assure robust scenario execution
- Archivers: appliances that assure all event artifacts are captured and stored
- Analytics: appliances that analyze event artifacts to produce metrics, visualizations, and human-readable reports

Table 2. Tradeoffs of Different Types of Data Generators

	Realism	Dynamism	Complexity	Storage Needs
Stubs	Low	Low	Low	Low
Fuzzers	Medium	Medium	Medium	Low
Simulators	High	High	High	Low
Replayers	High	Low	Low	High

In the sections below, we discuss recommendations for the design and operation of each of these types of appliances within a cyber range harness. Readers are encouraged to reference Section 5.1 for opportunities to incorporate those design mantras into each cyber range harness component.

5.2.1 Data Generation

All scenarios require data to drive their execution. These may be synthetic software-generated files or byte-streams, recorded data sessions that are played back, or even hardware-generated analog or digital signals. This data can range from inputs to systems under test, background network traffic to emulate real-world environments, or test scripts and configuration files.

Scenarios will always include some manual and human-generated input, whether it be a test operator configuring test case parameters and pushing a "start" button, or a team of network defenders that must organically work against an active threat. However, not all scenario data are (or should be) manually generated by a human. Data generators provide the means to *repeatably* generate sets of scenario data with concrete parameters and *tunable fidelity*. Appropriately built data generators allow range operators to quickly and robustly construct datasets that can address the objectives of a range event, both before an event and ideally also during an event should the need arise.

Beyond just being a source for scenario data, data generators provide myriad other tangential benefits. Datasets can be traced back to a parameterized invocation of data generation tools, assuring more complete *traceability* and the ability to perform root-cause analysis during range events. Data generators can also be used to validate other range infrastructure and reduce risk prior to formal range events. Data generators also tend to be much more portable than raw datasets and can be shipped to range event participants so they can independently verify their systems prior to integrating with the range.

While the benefits of having robust data generators are great, these never come without cost. Cyber range developers must navigate a difficult trade space to balance the needs, budget, and schedule of a range event. We summarize some different types of data generators and their applicability to various range objectives and needs.

Stubs. These are low-fidelity, low-cost generators that meet minimal interface requirements for a system to operate. These are not meant to emulate any kind of real-world operation, and should be primarily used for developing new systems and performing preliminary integration with other

systems. Examples include simple client applications that send a fixed response to any request for the sole purpose of allowing the server application to complete data flows.

Fuzzers. These are medium-fidelity, medium-cost generators that stimulate (or attempt to stimulate) all possible types of input to a particular system. The fidelity and cost of these generators are higher because they generally need to adhere to fixed communications protocols and formats, and also tend to require some statefulness to fully traverse all data flow possibilities. These should primarily be used for developing and testing systems, as they are robust in discovering edge cases. They are not appropriate for training or exercises, since they do not attempt to emulate real-world use cases, even though the data is well-formed. Examples include HTTP traffic generators that generate many permutations of randomized header values and request sequences, which would mostly comprise random bytes that conform to the HTTP protocol but have little to no semantic meaning.

Simulators. These are high-fidelity, high-cost generators that model real-world scenarios. These require in-depth understanding of the interfaces, data formats, and objectives of all actors in the scenario, and are therefore very challenging to accurately implement. These are based on models that have gone through significant validation by subject-matter experts, and perhaps have been derived by machine learning on real-world datasets. Simulator models can be developed at the protocol, system, or even user levels with a trade-off of efficiency and fidelity among them. This level of fidelity is commonly needed for training and exercise purposes, where the range must replicate the look and feel of an operational environment. Examples include battle simulators that can synthetically generate battlefield participants and their actions, and subsequently feed this data into situational awareness or command and control appliances.

Replayers. These generators are able to replay captured real-world data within the range. While these are arguably high-fidelity generators, they are disadvantaged due to their static nature. Replayers do indeed emulate real-world environments, but only as far as the number of different scenarios that have relevant data captures to replay. They provide no means to respond to live input, and therefore should have limited use in training and exercises. Furthermore, while these may seem to be low-cost data generation solutions, the storage requirements for live-captured datasets and the mechanics of replaying this data at operational speeds can be very demanding. We recommend these primarily be used for system test purposes and perhaps to inform the design of high-fidelity simulators. Examples include live network capture that can be played back against an intrusion-detection system.

When designing any of these data generators, the mantras of *extensibility* and *reusability* should always be kept in mind. The interfaces, protocols, actors, and behaviors that require data generation in one range event frequently reappear in future range events, so generalizing the design and implementation of these appliances can pay great dividends.

5.2.2 Instrumentation

Range instrumentation is responsible for sensing and capturing data relevant to a range event and is essential for both event objective verification and risk reduction purposes. Event objectives cannot be verified without empirical data collected during the event, and the presence of sensors can greatly improve a range operator's situational awareness during range events.

Selecting and implementing a set of instruments always involves a trade-off between the objective verification and risk reduction requirements of a range event and the hardware capacity needed to accommodate those requirements. More specifically, verification and risk reduction requirements will typically define a required frequency of data collection for a particular set of data. The instruments needed to capture this data will require a certain system overhead that could impact the systems under test, as well as sufficient data storage capacity for the captured data. While the price per unit of data storage and computer power generally decreases over time, the scale and complexity of range events is constantly increasing, putting all these needs at odds.

Range designers must always first ensure that the range's instrumentation sufficiently facilitates event objective verification (i.e., that it is capable of capturing all necessary artifacts for post-event analysis). After these requirements are met, risk reduction requirements can be considered and accommodated as resources allow by trading off the amount of risk that needs to be reduced (e.g., how much state needs to be reported for an operator to detect anomalous behaviors), the frequency of data collection, the instrumentation's system overhead, and the required storage capacity.

To navigate these trade-offs, it helps to decompose range instrumentation needs by the different instrumentation use cases and layers of observability. Range instrumentation is typically installed for two uses:

Situational awareness. This requires high-frequency collection and low storage to provide near real-time status of the range environment and any systems under test. Range events seldomly go smoothly the first time they are performed, and instrumentation greatly increases *event-time agility* by providing range operators immediate indicators of range health and test status. While these instruments will constantly be measuring and reporting data, this data usually does not need to be stored for a significant period of time. These instruments usually comprise sensors such as system resource and network bandwidth monitors and network topology maps.

Artifact capture. This requires medium-frequency collection and medium storage to comprehensively capture the data artifacts necessary to calculate all range event metrics. These instruments help guarantee *complete traceability* of all range events and data during post-event analysis. These instruments tend to not require as high of a collection frequency as situational awareness tools do, so long as the collection frequency meets the needs of the range event's predefined metrics. These instruments usually comprise sensors such as time-stamped software logs and command-line input recordings. They frequently also involve high-level summaries of data captured by situational

awareness and configuration management tools (e.g., average system load during a particular test and the configuration settings of the range environment prior to the test).

Range instrumentation built for any of these three purposes can be used to sense events and state at varying layers of the range's network and compute stack. The layers below comprise a suggested breakdown that is modeled after the OSI network model. The cost of developing and running instrumentation unsurprisingly increases with each additional layer that is instrumented, and range designers must weigh these costs against their range event objectives and the amount of risk they would like to reduce during range events, all while maintaining *minimal system overhead*.

User. At the highest level of abstraction, instrumentation can capture human (or simulated) user actions as they interact with the systems under test and range environment. This is typically used to log when critical inputs are provided by users so that they can be correlated with other observables in the test data. These can be as coarse as time-stamped command-line history logs, or as granular as keyloggers and mouse/screen recorders.

Application. At the next level of abstraction, instrumentation can capture the information communicated amongst applications. This set of data typically involves time-stamped application logs and protocol packet captures (e.g., HTTP or SMTP traffic recordings). Instrumentation at this layer is generally used to observe first-order effects of system stimulus. For example, one can note the exact duration it took for an anti-virus scan to detect a piece of malware after its placement on a system of interest. Detecting higher-order effects, such as the collateral impact of a user's actions on a system's (or a networked system's) resources, requires instrumentation at a different level.

Operating System. Below the application layer, instrumentation at the operating system layer can provide data needed to determine subtle trends in a system's performance. Instrumentation at this layer typically involves system resource monitors or kernel logs that contain information on processes, threads, CPU/RAM/disk usage, interrupts, possible clock drift, etc. Collection frequency is an especially important parameter for these instruments, as the overhead for running and recording this type of data can be prohibitively intrusive and expensive. For example, detecting the CPU and memory usage of an operation that only takes a few milliseconds requires a restrictively high polling frequency that can significantly interfere with the operating system's normal operations.

Transport/Session. For networked systems, it may be required or useful to monitor traffic at the session and transport layer of the network stack. While most range events can extract sufficient data from the application layer to verify event objectives, instrumentation at this layer can provide greater insight into packets as they are received at a network card or router. This can be especially important if the event involves users that can affect network traffic at this level, or if the bandwidth and flow rate of node communications is of interest to the event (e.g., if using the JIOR with many interfacing entities).

Physical/Data link/Network. This low level of instrumentation is typically not needed, as it involves monitoring digital communications at or near the system's hardware level. These certainly have a place, especially when hardware generators are in use, or if cyber-physical systems are being exercised.

In addition to *minimal system overhead*, the mantra of *end-to-end automation* is especially important when designing range infrastructure. Automating the data collection process by frequently scraping logs, archiving them in a centralized location, and backing up this archive on a regular basis greatly reduces the risk of data loss by reducing the burden on range operators to do these manually. Anything that can be done during range events to automatically parse collected data into human-readable formats will do much for situational awareness, expedient troubleshooting, and easy report composition. Automating the end-to-end process of instrumentation and data collection also forces range designers to think about data organization. We advise developing a schema for all range event data well before range events occur and developing the scrapers and parsers to translate raw instrumentation data into well-formed data that can be stored in a central, queryable repository.

One source of data that is frequently overlooked is the event coordinator's log. Range events, whether they be formal tests or operational exercises, are coordinated by a single person who is responsible for event sequencing, allowing for troubleshooting and debugging, authorizing system patches, rerunning exercises that have previously failed, and more. A detailed log of the decisions made during a range event are just as critical as the raw data that will eventually be used for post-event analysis, as analysts will not have the ability to separate valid and invalid datasets without it. Automating the capture of these decisions via a time-stamped chat log or HTML form is advised; in the vein of *minimal system overhead*, the event coordinator's bandwidth is limited just as any other system is, and any instrumentation used to capture his or her actions should minimally impact his or her decision-making capabilities.

Some special remarks are warranted when instrumenting systems that are meant to be exploited, disabled, or otherwise compromised during a range event. In particular, for red teaming exercises, range operators should expect the accuracy and utility of instrumentation to decrease as red teams progress in their tasks. Metrics and the instrumentation built to take measurements for these metrics therefore need to be designed to take this into account. For example, if the efficacy of a red team's ability to deny a particular service is to be measured, a "heartbeat" instrument is more appropriate than a CPU monitor, as CPU monitoring instruments may likely have undefined behavior when its processing capacity is completely used. Whatever metrics and instruments are selected must be able to quantify the differences between an uninhibited system and a compromised system.

Overall, the design of range instrumentation must always flow from the scenarios the range event uses. Scenarios define range event objectives, which in turn define requirements for the range environment and systems under test. These requirements will cover a range of capability, performance, security, and risk requirements for the range environment and systems under test. These requirements will then be used to define metrics that will evaluate how well the range event meets its objectives. These metrics will define measures that need to be extracted from the range envi-

ronment from the test. Finally, instrumentation can be selected based on the measures that need to be extracted.

5.2.3 Configuration Management

Configuration management is essential in range infrastructure for *repeatability*, *test-time agility*, *rapid environment reconfiguration*, and general risk reduction purposes. Similar to the trade-offs involved in designing and implementing range instrumentation, configuration management tools also offer a wide variety of configurations that balance the amount of risk reduced during range events and the overhead required to reduce that much risk.

Configuration management tools generally consist of appliances that can configure systems to a known state, as well as back-up utilities that can "snapshot" a system's current configuration. While it is preferable that configuration management tools have *minimal overhead*, many popular tools require constantly running daemons to maintain system configurations (e.g., Puppet, Chef, Salt). Some tools do not require running daemons (e.g., Ansible), but only offer "on-demand" configuration management as daemons are not constantly polling for configuration differences. Range designers should quantify the overhead incurred by any services used for configuration management and ensure that they do not interfere with any range instrumentation functionality or fidelity. In general, configuration management tools should only be active before and after formal event sequences, and it is therefore usually acceptable for these tools to incur a large amount of system overhead (e.g., network capacity and disk utilization).

These tools should support most of the following nonexhaustive list of items that may need to be verified or configured on each node in a range:

- Installed packages and libraries
- Binary versions for systems under test
- Running services (e.g., clock synchronization utilities, SSH servers)
- Nonrunning services (e.g., server software that is part of the system under test, which should be initialized to a known state and not be left running throughout an event)
- Configuration files (e.g., encryption certificates, hardware-specific application configurations)
- Source code checkout from version control
- Establishing connectivity with file servers and other remote services
- Creating user accounts and groups

Range designers may also find it helpful to consider the following typical use cases when configuration management tools are invoked during range events:

Setting control parameters. Each range event should consist of individual test cases, vignettes, or some denomination of controlled event sequences. Each of these must be carefully configured to a well-documented specification, and the range environment should be meticulously verified to be in a correct state before proceeding. Configuration management tools allow range operators to do this *repeatably* and efficiently without the need to painstakingly log in to each node in the range and manually verify that everything is in order. Using configuration tools in this manner greatly reduces the risk of seemingly minor configuration errors that can waste hours or days of event time, such as incorrect binary versions and unsynchronized system clocks.

Reproducing results. Whether it is to reproduce anomalous behavior during an event or reproduce results months after an event, the ability to reproduce a particular range configuration is an essential need for any range. While traditional configuration management tools might be sufficient to restore the range's state, range operators may want to "snapshot" environments completely in the form of virtual machine images or complete filesystem copies to preserve anomalous range states.

Restoration to baseline states. The ability to "roll back" to a baseline state is also a critical function for a range's configuration management tools. This capability provides range operators confidence that anomalous states can be freely inspected and debugged without the risk of irreversibly corrupting the range environment. It also allows for easy reconfiguration to a baseline comparison configuration that provides a reference for metric evaluation. If disaster strikes (e.g., power outages, accidental deletion of critical files), this capability can be exercised to minimize the amount of data and event time lost.

On the note of recovering from disaster, it is again worth remarking that cyber range activities may require especial consideration when it comes to configuration management. Due to the adversarial nature of some range activities, the importance of preserving valid range environment states is paramount. If part of the range can somehow becomes corrupt during the range event, being able to quickly restore a "clean" environment is worth investing in.

Overall, configuration management tools are necessary to reduce the risk of operator error when configuring and restoring range environments, especially as the environment grows in scale and complexity.

5.2.4 Orchestration

Orchestration range components refer to appliances that partially or fully automated the execution of tests or exercises on the range. Range orchestration can range from completely manual (i.e., range operators manually configuring, launching, and tearing down systems during events) to nearly autonomous (i.e., range operators specify configurations, "push a button," and watch an entire sequence execute). The preference is always towards more automation, as it can vastly increase *event-time agility* and *repeatability*, and also minimize multiple types of risk during a test (e.g., misconfigured test parameters, forgetting to collect certain artifacts). Given the cost of building an *end-to-end automated* orchestration framework and the limited chance for design

iterations, building a turnkey set of orchestration appliances is often a tall order and can take multiple program phases to get right. However, full automation should be envisioned from the beginning of range design and integrated into the range's capability requirements.

When designing orchestration frameworks, the most important objective is to abstract as much of the range's operation away from the range operator as possible. Range operators are ideally only concerned with (1) determining test or exercise parameters, (2) ensuring that a test or exercise sequence is being correctly executed without anomalies, and (3) confirming that all necessary artifacts are collected from a sequence before moving on to the next sequence. If orchestration appliances can be integrated into a range operator's workflow to achieve this level of abstraction, range operators can dedicate their full attention to these primary concerns.

Common orchestration appliances are quite similar to configuration management appliances in form and function. Tools like Puppet, Chef, Salt, and particularly Ansible can be used to great effect for orchestrating test and exercise sequences. However, configuration management tools are usually used for managing and enforcing state rather than executing long sequences of interdependent actions. Additionally, the mantra of *minimal system overhead* needs to be especially accounted for, as any background processes or resource consumption caused by orchestration appliances can significantly impact the event's metrics of interest. Orchestration appliances should incur minimal CPU, RAM, and disk usage on hosts during critical sequences. For interconnected systems under test, network usage should also be minimized. To this end, it is always recommended to have a separate "backchannel" network that is isolated from other nodes involved in a test. Having two network cards on each involved node (one connected to the backchannel for orchestration, and the other open to other nodes) allows orchestration and all its network overhead (e.g., file copies, SSH commands) to occur without interference with the systems under observation.

It is important to note that achieving end-to-end orchestration automation not only requires a robust set of orchestration appliances, but also requires other range components to communicate with one another and be able to individually operate autonomously. Requirements for range orchestration appliances should therefore inform the requirements for other range components. The general sequence of actions an orchestration appliance should be able to support is as follows:

- Reset and/or configure range environment per a specified set of test or exercise parameters
- Spawn instrumentation and background traffic generators and wait for them to achieve steady-state
- Spawn systems under test and feed in test inputs
- Monitor process and range environment health while test executes
- Collect and archive all artifacts after test completes
- Parse artifacts to confirm integrity of captured data and evaluate metrics
- Generate "quick look" visualizations of parsed metrics

We conclude this section with some final remarks on other mantras that should be considered when designing orchestration appliances. *Smoke tests* should be developed alongside the orchestration framework to not only provide a unit test of sorts for the framework, but also to perform quick sanity checks throughout range events. *Black-box treatment of systems* should be incorporated into the design as well, especially so baseline systems can be easily swapped in for systems under test within the orchestration framework. Range operators should be able to plug-and-play different comparison systems with minimal effort. Finally, *complete traceability* is always paramount in range events, and any set of orchestration tools should extensively log the actions they perform with precise time stamping.

5.2.5 Analytics

With the vast diversity of metrics and types of data visualizations possible, it may seem that the suite of analytic tools for a range event must be tailored for each individual event. While this is largely true, this does not preclude building a generalized analytics framework that can be *extended* and *reused* for multiple range events.

As discussed in the instrumentation section, robust analytics begin with the consolidation of data artifacts in a central, queryable repository. Much thought should go into the design of an "artifact database" schema. There is often a trade in database schema design between data redundancy and query speed. For example, if the artifact database is relational, multiple tables for distinct classes of data can be built with joint key indices for later querying. While this takes up less storage space than a monolithic table that stores all the information for each test or exercise sequence, querying these joint tables will likely be slower than querying a monolithic table. Schema decisions can have a surprisingly large impact on report generation downstream, so time spent refining the artifact database schema is time well-spent.

Range designers have a wide variety of data visualization tools that can be used to build plots and tables provided a robust database of artifacts. For a generalized and extensible analytics framework, the primary objective to strive for is to provide an analyst with the ability to compare arbitrary dependent and independent variables present in the dataset. For example, it should be relatively simple for an analyst to configure their analytic tools to plot something like query latency against the number of concurrent users, then alter this configuration to instead plot server resource consumption against the number of concurrent users. The ability to do this in a generalized fashion requires the artifact database to support fairly complex queries, so it is recommended to store artifacts in a well-supported off-the-shelf database system.

If a suite of analytic tools is built in this manner, range operators will have the option of running these analytics during range events to preview what the event results are beginning to look like. This greatly reduces the risk of collecting malformed or uninteresting data, and allows the event execution team to efficiently reprioritize the event's schedule when needed.

5.3 FINAL REMARKS

These guidelines and suggestions for designing various range infrastructure components will hopefully be valuable in future range design efforts. Great ranges are able to gracefully take on new events with unique needs and comprehensively capture all the data needed to complete the event.

While this discussion has focused mostly on completely software-based systems, we believe these remarks largely apply to range events that involve cyber-physical systems. Barring analog-to-digital interfaces for orchestration and instrumentation, the same mantras and general trade-offs apply for any infrastructure built to bring hardware into the loop.

6. GAPS

During observation of an operational exercise integration event, a few capability gaps—some technological and some methodological—were identified. We briefly describe them below as important areas for future work and focus, advancing the state of the art in cyber range capability development and operations.

6.1 TECHNOLOGY GAPS

Beyond Whitecarding: Cyber ranges are playing an increasing role in major military exercises. This is due, in large part, to the flexibility they afford operators to engage freely without worry of destructive or dangerous side effects from their actions. However, military exercises are, by definition, "in situ," events that are driven by real operators on real systems. The seamless integration of cyber ranges into exercises therefore necessarily create a live-to-virtual safety challenge of projecting the **operational effects** into the live domain while sandboxing the **technical perturbations** to within the cyber range. A cross-domain solution that permits this kind of integration has not yet been demonstrated, resulting in a lowest-common-denominator "white carding" activity, where a human exercise operator observes effects in one domain and requests those effects to be simulated in the other. Open research remains to determine the synchrony requirements, degree of fidelity, frequency, and automation of projecting capabilities across the live-to-virtual boundary.

Integrated Modelling and Simulation: A key tenant to efficient and effective cyber range operation is to deliver *the right resolution at the right place and time*. To date, this suggests that the level of fidelity of the entire range is set to the highest fidelity required by any component of the system test. This approach incurs high overhead and reduces operational and experimental agility. An interface that permits range operators to quickly "stub out" low-priority elements of the range environment by connecting those elements to in silico modeling and simulation tools would improve efficiency and agility of ranges. Such a cross-domain solution that permits the bridging from statistical histograms (distributions) to packet traffic emulation does not currently exist.

6.2 PROCESS AND METHODOLOGY GAPS

Integration of Science and Technology (S&T) and Operational Test Events Current cyber technology acquisition life cycles involve the traditional phases of developmental and operational testing to vet and characterize new research prototypes, followed by training and operational exercises to integrate new technology with the intended end users and environment. However, these events are often disjoint in their objectives and results.

S&T events (i.e., development and integration testing) are performed in the early stages of development, usually without significant involvement from potential end users and subject matter experts familiar with mission use cases. Operational events (i.e., operational testing, training events, and operational exercises or rehearsals) are often performed years after S&T events, usually without significant reuse of the methods, metrics, tools, and techniques (MMTTs) or personnel used in S&T

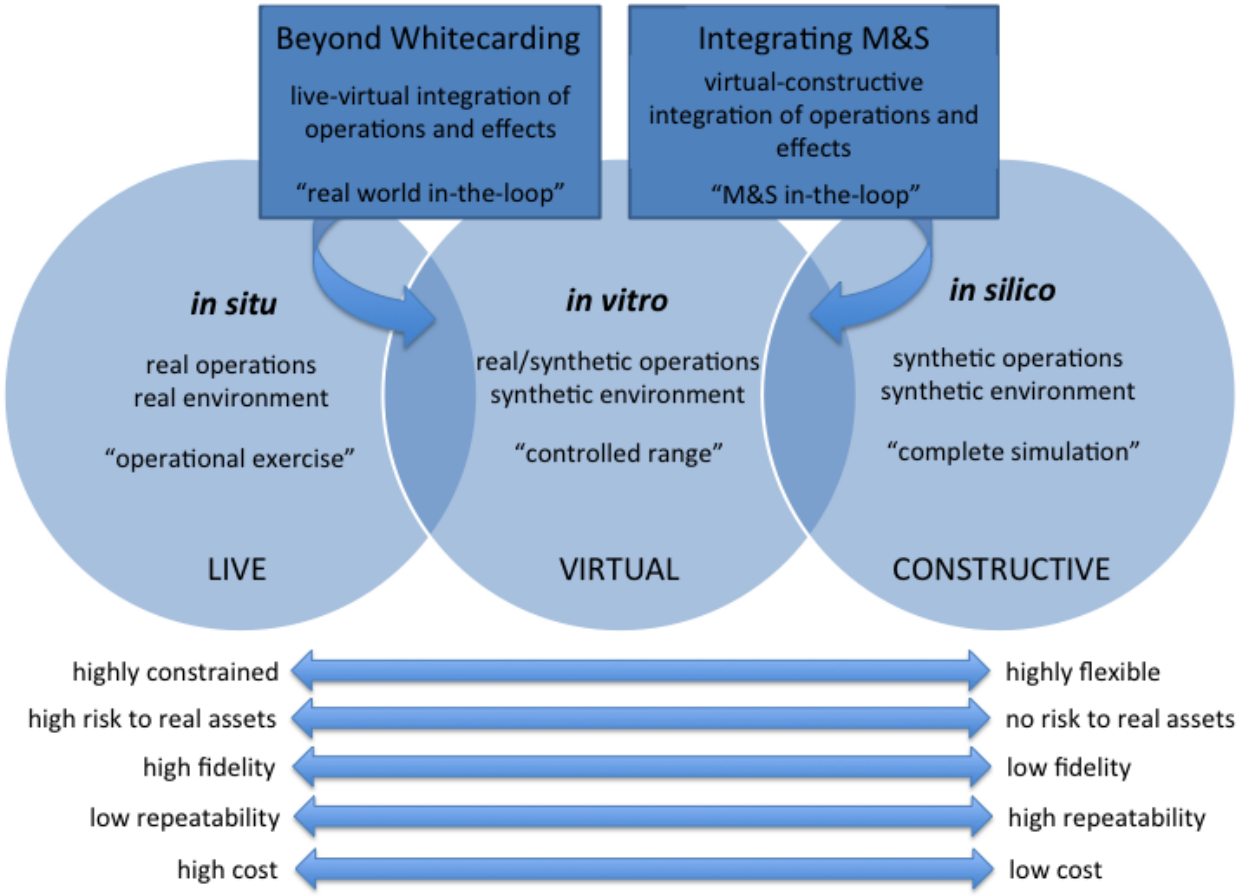


Figure 9. Illustration of varying levels of exercise integration between real-world assets, range infrastructure, and modeling and simulation tools.

events. Tighter integration and feedback between S&T and operational events is needed, such as the life cycle depicted in Figure 10.

Rather than a single pair of S&T and operational events, practices and processes to institute more regular feedback between the two types of events would help remove disconnects between research and operationally focused communities. End users and experts familiar with operational use cases would be encouraged to inform researchers and independent verification and validation (IV&V) teams of mission parameters and objectives. S&T activities would correspondingly be encouraged to report findings in operationally relevant terms, and also explore new or refine existing capabilities based on findings during operational events. This open exchange between entities that can perform controlled experimentation and entities that have access to operationally realistic assets would result in end technologies that are more well-suited to mission objectives and are better characterized in their abilities to support those missions.

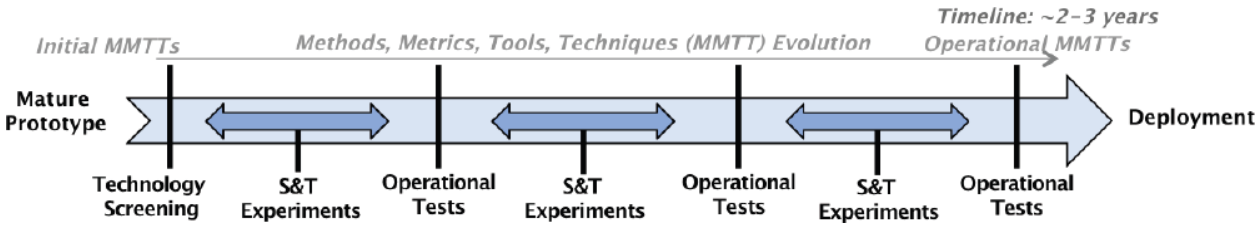


Figure 10. Recommended integration between S&T and operational testing [21].

While there are certainly some current enabling technologies for this frequent integration of S&T and operational events (e.g., the Joint Interoperability Range), new practices are needed to facilitate long-term adoption of an acquisition cycle that more tightly integrates S&T and operational events.

Reusability of MMTTs: Whether a range is used to support the development, test, training, or exercise of a cyber technology, the same types and families of MMTTs must be employed for the preparation, execution, and reporting of cyber assessments and range events (see Figure 11). However, current MMTTs are often uniquely built to support particular events and are therefore difficult to repurpose or reuse. Reusing the findings and MMTTs from prior events can significantly improve the productivity of future events and potentially reduce costs.

In particular, process improvements are needed to enforce the use of consistent metrics throughout the technology acquisition cycle. While there are methodologies for decomposing mission objectives into system observables, these same metrics need to continuously be integrated into the S&T and operational events that follow. Current practices tend to invent new metrics for new events, or worse, not use metrics at all (e.g., confirming that "penetration" of a system was achieved, but not to any quantitative degree).

With regards to methods, tools, and techniques, process improvements are always needed to take better inventory of the cyber range support tools and workflows that are developed for different use cases. Different cyber ranges have all built different methods of traffic generation, data scraping, data visualization, system monitoring, logging, and more, but these appliances are often tailored to a particular environment or event, and not generalized for wider use. As discussed in Section 5, most cyber range support tools perform a repeated set of standard functions, many of which are not different from the needs of standard software development workflows (e.g., configuration management, task orchestration). A better understanding of commercial off-the-shelf options and the robust solutions various cyber ranges supply would do much to reduce the cost of assessment events and facilitate constant improvement of the field's MMTTs over time.

Threat Model Integration Across Acquisition Cycle: As discussed in Section 3, appropriate characterization of relevant threat models is a prerequisite for the formulation of an effective assessment. However, practices and processes must be developed to support an acquisition life cycle that continuously integrates the relevant threat models into each assessment event.

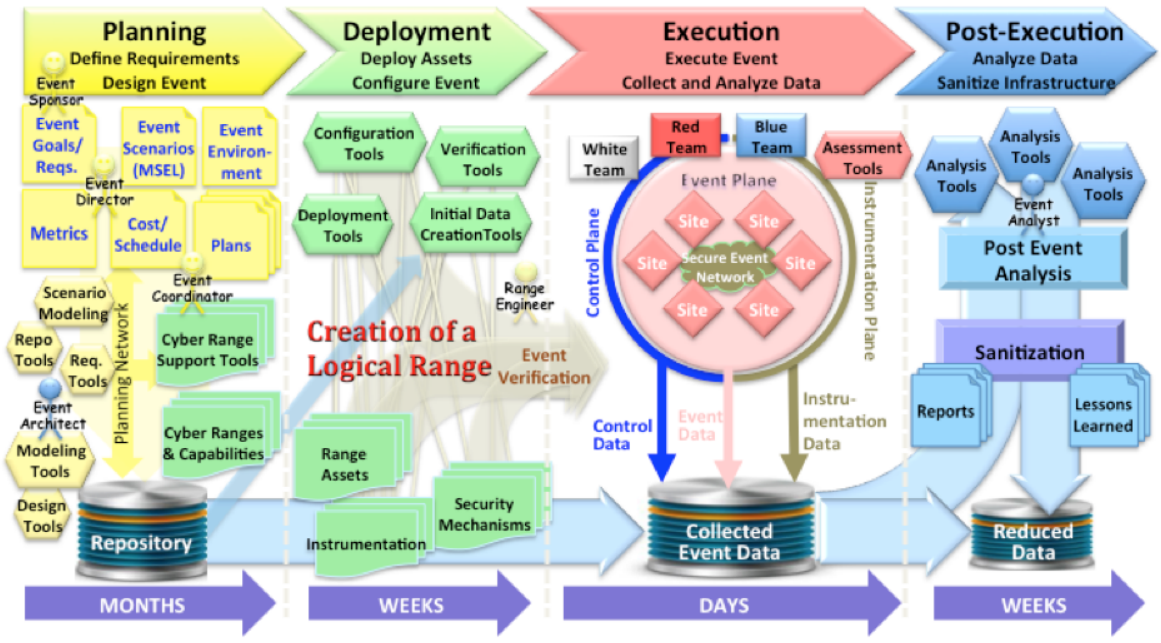


Figure 11. Test Resource Management Center process.

In particular, training and operational exercises should parameterize red teams within the context of these threat models. The level of maturity and resources of the relevant threat should be reflected by the red team's resources and behavior. While this is extremely challenging to accomplish in practice due to staffing restrictions and the difficulty in controlling human parameters (e.g., operator skill level and experience), this is essential if training and operational exercises are to provide meaningful insight into the operational readiness of a particular technology against a particular threat. Opportunities for improvement include more robust provisioning of quantifiable and persistent access to a system or enclave prior to red team exercises, and methods of controlling a system's attack surface to emulate systems that are vulnerable for an extended period of time (e.g., activating and deactivating back doors, beginning vignettes with elevated privileges for certain accounts).

7. SUMMARY

While the science of cyber assessment is still very much in its infancy, the state of the art is rapidly improving. By leveraging the great strides made in system assessment in other warfighting domains and developing unique technologies to support the assessment of cyber technologies, we are better positioned than ever before to effectively design and deploy impactful technologies to the war front.

It has become clear, however, through experience and observation that cyber assessments must not be constrained to the laboratory environment. Operational assessment in a mission context can provide invaluable perspective and feedback to the developmental efforts underway at all stages of maturity. Ergo, balancing the respective benefits of in situ (OPEX), in vitro (range), and in silico (simulation) testing, and matching their utility to the individualized goals of each assessment, are paramount to the success of any future cyber assessment.

Equally as important as developing the technologies and test harnesses is the parallel effort to advance the state of cyber assessment science by developing metrics and processes to conduct cyber tests and evaluations. To that end, we expect that near-term efforts focus work on the technological and methodological gaps identified in Section 6. These constitute areas of research that can dramatically impact our ability to respond to emergent threats and more rapidly transition technologies to practice.

Observations made during the DECRE C2IS OPEX integration capability demonstration reflect the quickly maturing state of cyber ranges and level of interoperability with other entities in our defense grid. The recommendations in this report provide ideas and frameworks with which to further leverage cyber ranges, such as the DECRE, to their maximum utility.

Specific recommendations have been made with respect to the planning and execution of cyber assessments, the framing of those assessments within a threat model, and on the design and employment of in vitro cyber range test harnesses in support of operational assessment activities. The authors sincerely hope that this structured review of knowledge and experience adds value to the continued advancement of the cyber assessment community.

This page intentionally left blank.

APPENDIX A: BACKGROUND MATERIAL

This appendix will provide organizational background and context.

A.1 CYBER MEASUREMENT CAMPAIGN BACKGROUND

The Cyber Measurement Campaign (CMC) is a multiyear research effort sponsored by the Assistant Secretary of Defense for Research and Engineering (ASD(R&E)) to identify and develop metrics and methods for objective and quantitative assessment of cyber capabilities. ASD(R&E) provides S&T leadership throughout the Department of Defense; shaping strategic direction and strengthening the research and engineering coordination efforts to meet tomorrow's challenges. The Cyber System Assessments Group (CSA) at MIT Lincoln Laboratory, a Federally Funded Research and Development Center (FFRDC), provides the U.S. Government with independent assessments of cyber systems and capabilities and serves as the principal performers on CMC.

During its execution, CMC has conducted a study to inventory the cyber test ranges available to the DoD S&T community, developed an assessment framework for measuring the agility and resilience of cyber systems, and conducted proof-of-concept assessments in maturing research technologies. More recently, CMC has developed metrics and methods for use in the evaluation of moving target technologies, which aim to increase the resiliency of systems and networks by introducing dynamism.

In 2014, ASD(R&E) tasked CMC to collaborate with an operational partner to further develop CMC metrics and methods in a realistic cyber environment and gain testing methodology insights from live operational environment exercises and to operationally validate early research findings.

A.2 DECRE C2IS BACKGROUND

In March 2013, Director Operational Test and Evaluation (DOT&E) requested Joint Staff (JS) J6 Deputy Director for Command, Control, Communication, Computers, and Cyber Integration (DDC5I), in concert with mission partners, establish a DoD Enterprise Cyber Range Environment (DECRE) to provide robust, secure, and operationally realistic support to the development, assessment, and training of C4/cyberspace capabilities.

Across three phases of development and test, DDC5I has developed and tested a DECRE Command and Control (C2) Information Systems (IS) range environment to generate, measure, monitor, and assess cyberspace activities on C2IS systems.

This page intentionally left blank.

REFERENCES

[1] G. Gilder, *Telecosm: The World After Bandwidth Abundance* (2002).

[2] S. King, "Defense Cyber S&T Strategies and Initiatives," in *DoD/DHS Small Business Innovation Research Workshop* (2013).

[3] Department of Defense Chief Information Officer, "Department of Defense Instruction 8500.01—Cybersecurity," Department of Defense (2014).

[4] Chairman of the Joint Chiefs of Staff, "Chairman of the Joint Chiefs of Staff Instruction 6510.01F—Information Assurance (IA) and Support to Computer Network Defense (CND)," Joint Chiefs of Staff (2013).

[5] Test Resource Management Center, "Comprehensive Review of Test and Evaluation Infrastructure," Department of Defense, Study Group Final Report (2012).

[6] J. Ranka, "National Cyber Range," in *DARPA Cyber Colloquium* (2011).

[7] C.V. Wright, C. Connelly, T. Braje, J.C. Rabek, L.M. Rossey, and R.K. Cunningham, "Generating Client Workloads and High-Fidelity Network Traffic for Controllable, Repeatable Experiments in Computer Security," in *Recent Advances in Intrusion Detection*, Springer (2010), pp. 218–237.

[8] Office of the Deputy Assistant Secretary of Defense, "Systems Engineering," in *Defense Acquisition Guidebook*, Department of Defense, chap. 4 (2013).

[9] Defense Acquisition University Learning Capabilities Integration Center, "Glossary of Defense Acquisition Acronyms and Terms," Department of Defense (2012).

[10] "Common Weakness Enumeration," The MITRE Corporation, URL cwe.mitre.org.

[11] "Common Vulnerabilities and Exposures," The MITRE Corporation, URL cve.mitre.org.

[12] C. Perrin, "The CIA Triad," *TechRepublic—IT Security* (2008).

[13] J. Meier, A. Mackman, M. Dunner, S. Vasireddy, R. Escamilla, and A. Murukan, "Threats and countermeasures," in *Improving Web Application Security: Threats and Countermeasures Roadmap*, Microsoft Corporation, chap. 2 (2013).

[14] K. Marier, "The 5 D's of Outdoor Perimeter Security," *Security Magazine* (2012).

[15] R. Lippmann, J. Riordan, T. Yu, and K. Watson, "Continuous Security Metrics for Prevalent Network Threats: Introduction and First Four Metrics," Massachusetts Institute of Technology Lincoln Laboratory, Project Report IA-3 (2012).

[16] Defense Science Board, "Resilient Military Systems and the Advanced Cyber Threat," Office of the Under Secretary of Defense for Acquisition, Technology and Logistics, Task Force Report (2013).

[17] Mandiant, "MTrends: Beyonds the Breach," FireEye, Threat Report (2014).

[18] Mandiant, "APT1: Exposing One of China's Cyber Espionage Units," FireEye, Intelligence Center Report (2013).

[19] "Common Attack Pattern Enumeration and Classification," The MITRE Corporation, URL capec.mitre.org.

[20] M. Varia, B. Price, N. Hwang, A. Hamlin, J. Herzog, J. Poland, M. Reschly, S. Yakoubov, and R.K. Cunningham, "Automated Assessment of Secure Search Systems," *Operating Systems Review—Repeatability and Sharing of Experimental Artifacts* (2015).

[21] P. Donovan, W. Herlands, and T. Hobson, "Operational Cyber Testing Recommendations," Massachusetts Institute of Technology Lincoln Laboratory, Technical Report 1179 (2014).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
TBD 2015	Technical Report	

4. TITLE AND SUBTITLE

Operational Exercise Integration Recommendations for DoD Cyber Ranges

5a. CONTRACT NUMBER
FA8721-05-C-0002

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Nicholas J. Hwang and Kevin B. Bush

5d. PROJECT NUMBER
2167

5e. TASK NUMBER
271

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02420-9108

8. PERFORMING ORGANIZATION REPORT NUMBER

TR-1187

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Assistant Secretary of Defense for Research and Engineering

10. SPONSOR/MONITOR'S ACRONYM(S)
ASD(R&E)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Warfighting technologies and systems are being integrated and connected via the cyber domain at unprecedented rates. The cyber warfighting domain is permeating all areas of warfare, and the need to accurately assess the cybersecurity posture of systems has never been more prominent. This report identities key challenges facing the cyber assessment efforts across the science and technology, acquisition, and operational communities and presents a summary of the state of the art in cyber assessment technologies. Specific recommendations are given to balance the benefits of controlled experimentation in synthetic environments with the realism of operational exercises.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified	Same as report	58	

This page intentionally left blank.