Cyber Team Performance Simulation and Validation using Cyber-FIT Version 4.0

Geoffrey B. Dobson and Kathleen M. Carley
July 1, 2024
CMU-S3D-24-113

Software and Societal Systems Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

This work was supported in part by Office of Naval Research (ONR) Minerva-Multi-Level Models of Covert Online Information Campaigns, N00014-21-1-2765, and the Air Force Research Lab FA8650212624 for Cyber-Fit Additional support was provided by the center for Computational Analysis of Social and Organizational Systems (CASOS) and the Center for Informed Democracy and Social Cybersecurity (IDeaS) at Carnegie Mellon University. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the AFRL, or the U.S. government.

Center for the Computational Analysis of Social and Organizational Systems

Keywords: agent-based modeling, simulation, military, cyber warfare, validation

Abstract

This technical report summarizes the research efforts conducted on the Cyber Forces, Interactions, and Terrain (Cyber-FIT) agent-based modeling and simulation framework. Cyber-FIT was developed primarily as an avenue to quantitatively define military cyber team performance measures. Using the tool, realistically scaled cyber conflict simulations are run that programmatically track data that could be collected in real world environments, and then are aggregated and calculated, ultimately informing mission leadership of the cyber team performance. Virtual experiments were run to test hypothesis relevant to cyber mission forces. Cyber-FIT was then used as an engine to develop a new multi-modelling organizational simulation tool called the Organization Simulation in Response to Intrusion Strategies (OSIRIS). This effort introduces human factors behavioral modeling relevant to military research efforts such as role-based workflow modeling, training level, and memory retention. Finally, a systematic methodology was followed to complete a validation of the model.

Table of Contents

List of Figures

List of Tables

1 INTRODUCTION

Commanders of cyber units have a very difficult job. Like others, cyber mission forces must be able to report on their readiness, along with how well they performed on missions. Since cyber is a relatively new domain, and its abstract nature, these tasks are very difficult to quantify and measure. This core operational problem is a key reason for recent calls for advancing the state of modeling and simulation research in cyberspace. For example, understanding how team makeup in terms of knowledge, skill and experience informs cyber mission performance. Also, there is still a gap in the overall understanding of what is mean by cyber mission performance. This work makes progress on those fronts.

The Carnegie Mellon University Center for the Computational Analysis of Social and Organizational Systems' (CASOS) developed an agent-based modeling and simulation framework called Cyber-FIT (Forces, Interactions, Terrain) [1]. The software was used as a computational analysis tool to simulate missions and run virtual experiments on various research questions of interest to cyber researchers. The existing software capability is now able to simulate cyber conflicts at a realistic scale. This version has been presented at several conferences, seminars, and Air Force Research Laboratory Friday technical forums. The current version is being used for an organizational multi-modeling effort and updated to include human behavioral factors.

2 METHODS, ASSUMPTIONS, AND PROCEDURES

Cyber-FIT is composed of two main classes of agents: forces and terrain. The forces represent the military forces in a conflict. The terrain agents represent the computer systems being utilized by the force agents. All agent types are connected by directed link interactions representing communications between agents, conceptually represented in the figure below.

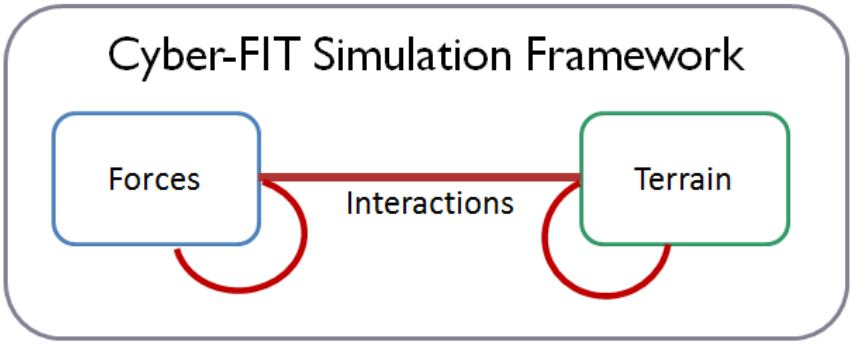


Figure 1: Cyber-FIT conceptual model

Cyber-FIT version four is the most recent release of the framework and the subject of this technical report. Version four is the first version developed in the Java based Repast Simphony tool leveraging recursive agent libraries and object-oriented data structures. Version four added a lot of complexity to agent attributes and behaviors. Friendly force agents were added so that

mission-based information needs for kinetic missions could be simulated and measured. Cyber team defender agents were updated to add the following simulated attributes and behaviors: knowledge, skill, experience, mission, terrain details known, operation types, and operation details. Terrain agents were updated to add the following simulated behaviors: vulnerability complexity, zero-day exploitation, advanced persistent threats, and mission associations. Attacker agents were updated to add the following simulated attributes and behaviors: tier level, kill chain phase, terrain maneuver details, exploitation statistics, terrain command and control, and terrain actions.

The software was run simulating a realistic cyber conflict collecting relevant simulated data for post-processing analysis. This met the goal of simulating all performance measures defined and sourced through discussion with both AFRL researchers and active-duty cyber protection team members. Many of the simulated performance measures are still conceptual due to the difficulty of tracking. All performance measures, simulation results, and statistical analysis are detailed in the published technical report called "Cyber-FIT Version 4" [2].

3 Results and Discussion

Two virtual experiments were run using Cyber-FIT version 4. The model was then validated using a "validation in parts" methodology.

3.1 Virtual Experiment One on Knowledge, Skills, and Experience

The interaction of knowledge, skills, and experience certainly plays a role in how teams perform. When envisioning a perfect cyber team, managers might envision starting from scratch by hiring ideal candidates based on education and experience which are shown on resumes. This assumes that the actual knowledge, skill, and experience of the candidate can be gleaned from the words on the paper resume. This strategy may be ideal but probably unrealistic in both military and industry situations. In the military, commanders are selected for a position and assigned to that location with an already existing cyber team. When the commander arrives, they do not then begin hiring the ideal candidates, the team in place is already assigned to them. Similarly, in the private sector, cyber managers (who may have more flexibility with hiring) come into an already existing organization and then must compete with other firms vying for the same cyber talent. Also, even if cyber leaders do have the ability to add cyber talent, this takes time. Which means that improving the cyber team outcomes might be more likely accomplished through internal team development. So, how should cyber leaders develop the team?

Cyber leaders are usually interested in developing their teams through training. This is especially true with military cyber teams who are almost always "on mission" or "training". Training will usually come in the form of individualized classroom style educational services and products where the primary outcome goal is knowledge acquisition. There is a skill identified that some personnel are lacking, and the training resolves that gap in knowledge. Some of these training courses provide certifications that are recognized by industry by meeting a standard. Returning to the concept of individual knowledge, skills, and experience, training is either meant to increase the knowledge level or the skill level of the individual or both. For example, the Security+ certification provides knowledge only. Whereas a weeklong web development software language training with a capstone project is adding to both knowledge and

skill. A cyber leader sending a group of individuals to a cyber war exercise, or competition is an example of adding to the skill level only. This is what the cyber leader and management must do: decide which training courses are most appropriate given the current environment of the team.

This is the spirit that this virtual experiment exists within. Given that a cyber leader is managing a typical cyber team, which training options should be sought after? Knowledge is likely easiest to add, while skill is more difficult. Also, knowledge is typically not indicative of a "better" team. Skill is far more important, it is the "x" factor. Skill is also far harder to define, measure, and improve. In any event, if a mechanism for identifying and improving skill is available to leadership, how would that difference manifest within a cyber conflict? To test out this concept Cyber-FIT is used to run a virtual experiment varying team skill and adversary tier. If skill is considered an "x" factor, then it should be a key determinant in how likely a cyber team member is to restore compromised terrain. Most cyber subject matter experts believe that experience also plays a factor because over time, being exposed to different tools and techniques, and practicing those, will improve performance. Therefore, skill will act as a multiplicative effect on the amount of experience a cyber team member has in terms of how likely that agent is to restore compromised terrain. While tuning this version of the model, exploratory analysis showed that simply multiplying the square of skill and experience did not make a significant realistic difference in performance. To account for this, a multiplier is added, with each skill and experience combination value being weighted higher in a Fibonacci sequence. The Fibonacci sequence is frequently used in software development relative scoring systems for estimating effort and complexity in project management, so it could be a good candidate for estimating cyber operational capability as well. The defender agent restoral rate (rr) for this experiment will be based on the following equation detailed in the table below where s = skill, e = experience, and m = multiplier and adhering to the following equation.

$$rr = ((s^2 * e) * m)/5{,}000$$

e	S	$(\boldsymbol{s^2 * e})$	Multiplier	Restoral Rate
1	2	4	2	8/5000
2	2	8	3	24/5000
3	2	12	5	60/5000
1	3	9	8	72/5000
2	3	18	13	234/5000
3	3	27	21	567/5000

Table 1: Restoral rate setup

The purpose of this experiment is two-fold: First it will show how skill acquisition can be an extremely important strategy for cyber teams. Second, it will show that Cyber-FIT is reflecting a reasonable expectation of realism when varying team makeup and adversary complexity. For this experiment an average team will engage in cyber conflict with all six tier levels of an adversary vying for a projected cyber terrain. The cyber team will be size nine, where one agent is the team lead, four are on the network squad and four are on the host squad. The cyber team demographics will be one of three settings for the virtual experiment. The first setting will be made up of average knowledge, skill, and experience. The second setting will increase half of the agents' skill level to three, making the average team skill level 2.5. The third setting will increase all the agents' skill level to three, making the average skill level 3.0. In other

words, this simulates swapping four average team members with expert skill team members (setting two). And then swapping the remaining average team members with expert skill team members (setting three). The details of this experiment are detailed in the table below.

Independent Variables		
IV	Variations	Range
Average team skill	5	2, 2.25, 2.5, 2.75, 3
Adversary tier	6	[1 – 6]
Control Variables		
Average team knowledge	1	2.5
Average team experience	1	2
Vulnerability growth rate	1	.001
Defender restoral rate	1	Table 31
Exploit success rate	1	1.0
Dependent Variables		
DV	Type	
Compromise Time	Integer	
This experiment design is 5X6X10 runs = 300 replications		

Table 2: Virtual experiment setup

This virtual experiment, based on the underlying model assumptions and configuration, shows that adding highly skilled team members will have a large effect on cyber team performance. It also shows that an average military cyber team, with average skill, will fare well against adversaries of tier one through four, but not well against adversary tiers five and six. The following figure shows all simulation results in a scatter plot format.

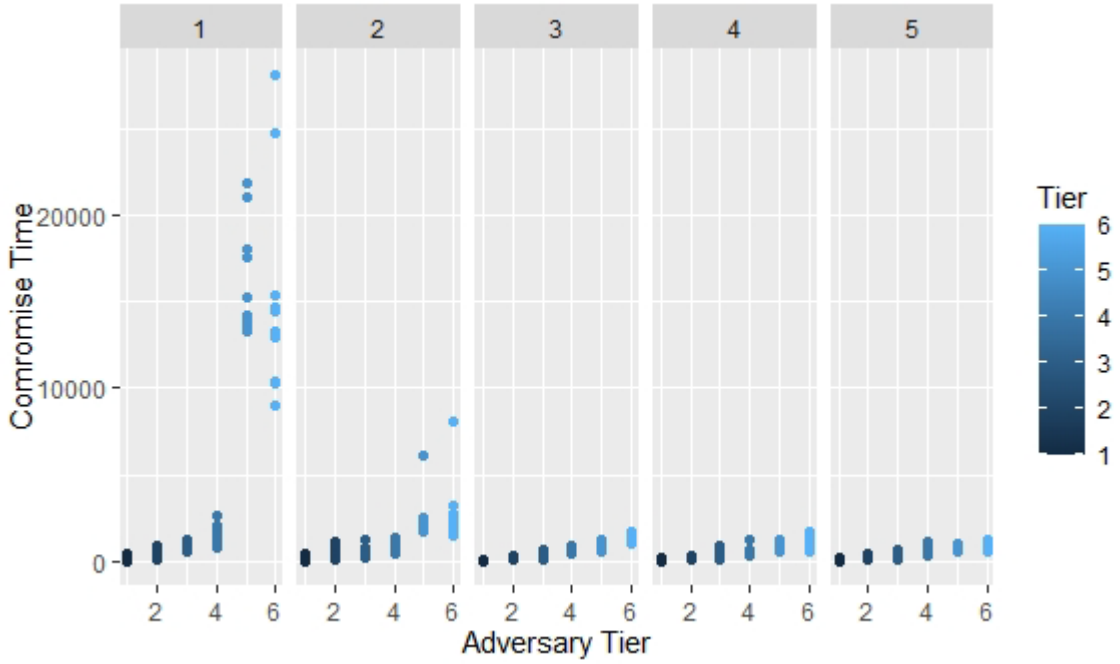


Figure 2: Virtual experiment results

As shown, with an average skilled team, the negative effects (cyber damage) seen within the cyber terrain increases sharply as tier level is increased. The results show that simulated tier five and six adversaries will be able to control the cyber terrain and maneuver within it against an average skilled team. The cyber team will spend a significant amount of time in reactive

mode, finding compromised systems and spending a significant amount of time communicating about and restoring those systems. Once a team is in reactive mode, it is hard to recover, until something significant occurs like the adversary stopping its attack, or systems being pulled, rebooted, etc.

By replacing four (half) of the average skilled agents with expert skilled agents, the setting three cyber team performs significantly better. This team can control the cyber terrain and quickly remediate any compromised systems they come across that have been successfully attacked by all adversaries, even tiers five and six. This means the adversary never tips the scale to the point where the team is in constant reactive mode during the remainder of the conflict. As shown, the mean compromise time from team setting one to team setting three decreases 93.8%. This result represents an approximation of reality based on what is being reported by subject matter experts on the importance of elite cyber professionals. Just adding a couple "sharp shooters" can have enormous effects. Staying with a tier five adversary, the simulation shows that moving from team setting three to team setting five results in another 30.4% decrease in compromise time. Adding more expert team members continues to have a significant effect on performance, but clearly moving from average team skill 2.0 to 2.5 will have a much bigger impact than moving from 2.5 to 3.0. Also, moving from team setting one to team setting two, where one expert skilled agent replaced a medium skilled agent does have a significant impact seeing average compromise time improve to from 2,498.6 to 16,292.7, a decrease of 84.7%. So, this is where the planning decision would come into play. Does the commander accept the risk of a projection of 2,498.6 downtime units with one expert or would they request one more expert to get the projection of 1,011.6? The resulting simulated data from virtual experiment one was run through a regression model with both tier and skill setting as independent variables against the log of compromise time. The regression model is shown in the figure below. As shown both skill setting and tier are significant predictors of the compromise time. Tier level has a higher estimate meaning it has a greater effect on compromise time. The inflection point is clearly shown between tier levels four and five in the table above. This table also shows wide ranges of variance within simulations and virtual experiments that are clearly a combination of both stochasticity and functionality. Consider setting two where the average skill is 2.25. In this case the standard deviation of tier 5 compromise time is 1,212 for a mean of 2,498.6. By adding one more expert to the team, the mean is reduced to 1,011.6 with a standard deviation of 206.3. Another expert decreases the mean by more than half and the standard deviation by more than eighty percent. Variance changes from both randomness and function are embedded into all simulations in this model at different levels which can be studied and improved with more validation.

```
Residuals:
    Min      1Q  Median      3Q     Max
-4.9748 -0.4248  0.0784  0.6311  1.9074

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.58084    0.19564  23.415  < 2e-16 ***
t            0.77812    0.03678  21.155  < 2e-16 ***
s           -0.38419    0.04442  -8.649  3.3e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.088 on 297 degrees of freedom
Multiple R-squared:  0.6375,    Adjusted R-squared:  0.6351
F-statistic: 261.2 on 2 and 297 DF,  p-value: < 2.2e-16
```

Figure 3: Virtual experiment regression analysis

This virtual experiment computationally models what most would agree is occurring in the real world: adding enough experts to a team to make it perform at a high level. For the average team in both military and industry settings, they'll be happy to have an expert or two that can vastly improve the team. Of the thousands of teams operating in the real world, a very small number might be made up of all expert skill level members. Perhaps most elite hacker teams, special cyber operators, and security operations centers handling high value information are made of all or mostly all expert skill. The rest are doing their best to make a marginal increase in skill level through training and practice opportunities. This virtual experiment can help them approximate their gains by doing assessments of their current strength, and what could be improved upon through training or acquisition.

3.2 Virtual Experiment Two on Deployment Time Delay

The exact timing of force deployment in a military conflict is an age-old question. Cyber team deployment is, in essence, sending resources to the cyber terrain prioritized due to mission requirements. The mission requirements are extremely dynamic and change continuously because of changing cyberspace terrain, updated intelligence reporting, availability of forces, changing interests, and geopolitical events to name a few.

Cyber-FIT can be used to simulate the outcomes of importance to inform war-gaming efforts. For this virtual experiment, an aspect of a wargame that is being considered is turn based decision making. In a wargame, typical "COAs" (courses of action) involve where to move forces in the terrains in play. This could be all terrains (air, land, sea, space, and cyberspace) or a subset. Imagine that the commanders, at a turn in the game, must decide whether to deploy cyber forces based on intelligence reporting to an area of terrain. The intelligence report provided in the wargame may indicate that sophisticated cyber adversaries are able to attack cyber assets. The decision (course of action) could be to deploy or delay. Deploy may be the correct move. Perhaps the adversary is already actively engaged, and the forces need to deploy immediately to hunt-forward and take remediating actions. Delay also may be the correct move if the intelligence reports were not accurate, and the enemy is of lower sophistication and therefore easier to deal with. The delay might allow for those cyber forces to be available in the future where they are more useful for other campaign priorities. Thus, wargaming allows the participants to practice moves and then think through the strategic implications. A software tool

that provides cyber force deployment options and simulated results doesn't exist, and this is where Cyber-FIT can be utilized.

For this virtual experiment, a cyber team will deploy to a conflict consisting of five kinetic missions connected to a tactical base infrastructure. The adversary will begin attacking at time t = 0. The cyber team will either: deploy immediately (t = 0), delay one day (t = 1,440), or delay two days (t = 2,880). Also, this experiment will include all adversary tiers, one through six. The outcome measure to consider is compromised systems, which is the number of cyber terrains in a compromised state at any given time. In real-world operations, the primary job of the cyber team is to keep that number to zero, enabling all systems to be always available to kinetic forces. The table below details the virtual experiment design.

Independent Variables		
IV	Variations	Range
Deployment delay time	3	0, 1, 2
Adversary tier	6	[1 – 6]
Control Variables		
Average team knowledge	1	2.5
Average team experience	1	2
Average team skill	1	2
Vulnerability growth rate	1	.001
Exploit success rate	1	1.0
Dependent Variables		
DV	Type	
Compromise Time	Integer	
This experiment design is 3X6X10 runs = 180 replications		

Table 3: Virtual experiment setup

This virtual experiment, based on the underlying model assumptions and configuration, shows that delaying the deployment of cyber forces will have an increasingly greater impact as the sophistication of the adversary is increased. This is an expected general outcome. This simulation shows that for adversary tiers one through four, the cyber team would be able to overcome the adversary quickly and return to baseline cyber terrain performance where very few compromises are being accomplished. All tiers one through four will be at baseline by day three (t = 4,3200) of the simulated conflict even if the cyber team delays its deployment by two days. Similarly, for tiers one through four, the cyber team will return the cyber terrain to baseline deployment by day two if the team delays deployment by one day. This means that whether the team delays for one day or two days, it will still take the team one day to return to baseline. This is different than the response simulated for tiers five and six. For tiers five and six, a one-day delay will take more than two days to recover from. For tiers five and six, a two-day delay could be very difficult to recover from as shown in the below figure.

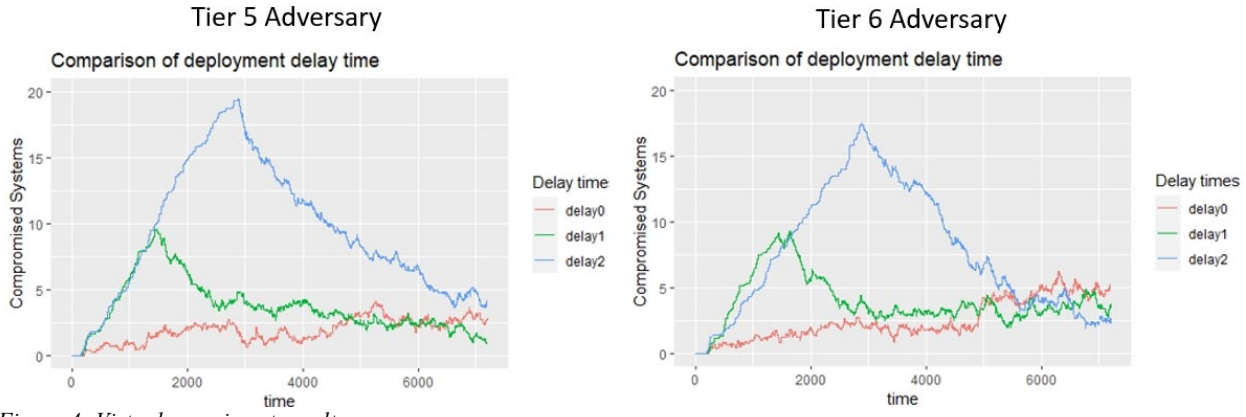


Figure 4: Virtual experiment results

The utility of a simulation tool for wargaming is apparent here. The details of the game will determine how granular a simulation tool must be, for simulated outcomes. If the wargame is taking turns in a one-day time horizon, then this type of simulation would work perfectly. For example, if the participant chose to delay for two days, and the adversary ended up being tier three, and the cyber terrain was needed on day four, then then the cyber team would have been very likely to restore the terrain by the time it was needed. However, under the same circumstances, if the adversary turned out to be tier five, then the systems would be much more likely to not be available on day four (t = 5,760). The wargame might take a statistical sampling of simulation turns and then provide a key terrain cyber availability value based on the average available. The wargame might also simulate all systems in question each turn. In the former, there would have to be data processing capabilities built into the wargame software. In the latter, higher variance simulation software would have a greater effect in terms of randomization presented to the participants. In any event, this virtual experiment is a proof of concept for how Cyber-FIT, and more generally, agent-based systems should be used for higher fidelity cyber informed wargames.

3.3 Cyber-FIT Validation

North and Macal defined seven different types of validation tailored towards agent-based modeling: requirements validation, data validation, face validation, process validation, model output validation, agent validation and theory validation [3]. The next sections apply each validation methodology to Cyber-FIT, validating the model in parts.

3.3.1 Requirements Validation

The guiding question [3] for requirements validation is: “Is the model solving the right problem”? The requirements for this software research effort were published in a series of government documents which all call for a simulation tool such as Cyber-FIT. The first of the United States government documents calling out the problem to solve was the Department of Defense Science Board report of 2013 titled “Resilient Military Systems and the Advanced Cyber Threat” [4]. The report was the result of a task force examining the state of cyber security for DoD and then making a series of recommendations. For example, the report states “The Task Force could not find a set of metrics employed by DoD or industry that would help DoD shape

its investment decisions. A qualitative comparison of resources and DoD level of effort in relation to the success rate of red teams is clear evidence of the lack of useful metrics". Not having meaningful metrics is a glaring problem. Without meaningful metrics, it is very difficult to reason about tradeoffs on decisions affecting the cyber force. Further, the report recommends the need to "increase feedback from testing, red teaming, intelligence community, and modeling and simulation as a development mechanism to build out DoD's cyber resilient force". In other words, the task force is beginning to define the requirement for a modeling and simulation tool.

The next document, published in 2015, is the Department of Defense Cyber Strategy [5]. This document had similar recommendations in terms of using modeling and simulation for assessing cyber forces. The first strategic goal of the Cyber Strategy is to "build and maintain ready forces and capabilities to conduct cyberspace operations". Several modeling and simulation points are called out as requirements supporting the strategic goal. The first is to "establish an enterprise-wide cyber modeling and simulation capability" and "develop the data schema, databases, algorithms, and modeling and simulation capabilities necessary to assess the effectiveness of cyber operations." This means that software must be designed (most likely an object-oriented software) that has integrated functions programmatically enforcing algorithms which ingest data, make changes, and then output data in schemas. The output data can be stored in databases or file systems. Next, the strategy calls for a need to "assess cyber mission force capacity" to "achieve its mission objectives when confronted with multiple contingencies". This language is defining a requirement that the software output should be associated with mission metrics. Also, it must be able to handle multiple scenarios of varying situations that the cyber force could be confronted with. Lastly, in this document, the "Joint Staff…will propose, collect, analyze, and report a set of appropriate metrics … to measure the operational capacity of the CMF". This language is defining requirements for the modeling software around how the output data should look. It should either define the metrics of interest or define the output data such that post-processing software would be able to show desired metrics. Ideally, a multitude of data would be available so any number of software applications could ingest and utilize the data in novel ways.

The next document, published in 2019, is the White House Executive Order on America's Cybersecurity Workforce [6]. The order states: "The Secretary of Homeland Security, in consultation with the Secretary of Defense, the Director of the Office of Science and Technology Policy, the Director of OMB, and the heads of other appropriate agencies, shall develop a plan for an annual cybersecurity competition (President's Cup Cybersecurity Competition) for Federal civilian and military employees. The goal of the competition shall be to identify, challenge, and reward the United States Government's best cybersecurity practitioners and teams across offensive and defensive cybersecurity disciplines". The competition is now very popular and has been held several times, with some lawmakers proposing bills to codify it as a yearly budget item [7]. The competition challenges participants with cyber skill games where they gain points and move through rounds ultimately vying for the championship trophy. This means that there is a scoring aperture that defines who has the best skill and differentiates individual and team, adding another requirement to the simulation software.

Software development requirements writing typically begins with customer, or external user desires, and then moves to the internal development necessary to set up the appropriate objects and data structures to deliver the higher-level capabilities. For Cyber-FIT version 4, this is described in the table below. The Observer class of Cyber-FIT software is an abstraction of what the components are doing collectively to meet the overarching goals of the framework.

Following the Observer class are the main component classes making up the agents, interactions, and mission objects that must operate independently to achieve the collective system functionality. The tables below are written in the Agile Development User Story method, by software class, completing the requirements validation exercise for Cyber-FIT.

Observer Class	
No.	User Story
1	As a model user, I want to simulate effects of cyber forces as a development mechanism in building a resilient cyber force
2	As a model user, I want to have access to a well-documented software that defines algorithms, models, and database objects for cyber modeling and simulation
3	As a model user, I want to simulate various cyber mission types against multiple types of adversaries
4	As a model user, I want to simulate various cyber mission types against multiple types of adversaries
5	As a model user, I want to export the simulation data in various industry standard formats such as JSON, csv, XML, etc.
6	As a model user, I want the formulas which describe mission success embedded into the software, or the data output in a way that post-processing software can define the formulas
7	As a model user, I want the data that is output to be in a format that can define individual metrics, team metrics, or a combination of both
8	As a model user, I want a mechanism in the form of configurations that can incorporate theoretical models such as cyber situation awareness and organization theory

Table 4: Observer class requirements

Terrain Class	
No.	User Story
1	As a researcher, I want to model the effect of computing systems incurring vulnerabilities,
2	As a researcher, I want to model different behaviors, as a result of external stimuli, of different types of computing systems in terms of network architecture, routing, and information sharing
3	As a researcher, I want the computing systems within the model to track vulnerability level, malicious code present, and compromise status
4	As a researcher, I want to track computing systems to missions for mission level analysis

Table 5: Terrain class requirements

Defender Class	
No.	User Story
1	As a researcher, I want to differentiate the team members by squad for modeling of differentiable sub-element behaviors
2	As a researcher, I to model the basic operations of a cyber defender in terms of survey, securing, and protecting terrain
3	As a researcher, I want to change behaviors of individual agents based on frameworks such as NICE [8], along with emerging policy and doctrine

4	As a researcher, I want individual team members to track demographic data based on typical military and industry reporting requirements
5	As a researcher, I want to model cyber incident response procedures
6	As a researcher, I want the cyber team members to interact with each other in the form of information sharing, reporting, and directives
7	As a researcher, I want the cyber team members to interact with computing systems differentiated based on the purpose of the cyber operations they are working
8	As a researcher, I want the cyber team members to behave differently based on their level of knowledge, skill, and experience
9	As a researcher, I want the cyber team members to have functionality representing a cognitive model of the situation that changes over time

Table 6: Defender class requirements

Attacker Class	
No.	User Story
1	As a researcher, I want to model the attacking agent behavior moving through steps such as the cyber kill chain, or other similar intrusion chain methodologies
2	As a researcher I want to control behavior within the steps or phases of the intrusion/kill chain methodologies
3	As a researcher, I want the attackers to have varying complexities leading to differential behavior
4	As a researcher, I want the attackers to track summary statistics about success and failure per attack attempt
5	As a researcher, I want the attackers to have to be dependent on vulnerabilities existing on terrain they are attacking for success criteria
6	As a researcher, I want some high-level attackers to be able to exploit zero-day vulnerabilities

Table 7: Attacker class requirements

Friendly Class	
No.	User Story
1	As a researcher, I want the non-cyber agents to be modeled as using the computer systems for the purpose of their missions, tracked by mission
2	As a researcher, I want the friendly agents to request information and track whether or not the information was received, how quickly, and of what quality
3	As a researcher, I want the friendly agents to share information that is received with other team members

Table 8: Friendly class requirements

Interaction Class	
No.	User Story
1	As a researcher, I want interactions to be tracked by which classes are being connected, with differentiating behavior basis
2	As a researcher, I want to model interactions by type which determines how long they persist

Table 9: Interaction class requirements

3.3.2 Data validation

The guiding question [3] for data validation is: "Have the data used in the model been validated"? This refers to the input data the model ingests along with data that controls agent behaviors, typically called control variables. The data used for Cyber-FIT come from a mixture of sources such as official government websites, sampling, interviews with experts, policy-based literature, and empirical studies. With any novel model, there will be some aspect that doesn't have data or literature backing it, which is typically why it is new and of interest to emerging research. For Cyber-FIT there are several examples of this and can be outlined with respect to the three most important agent classes: terrain, defender, and attacker. The table below details the most pressing model behaviors per agent class that would make the system more realistic if data were available.

Class	Behavior	Data needed
Terrain	Connecting computing systems, enforcing networking rules, reading, and writing information	Routing protocols, typical network architectures, empirical network data
Defender	Operational behavior detailing type, resources needed, frequency, and reporting	Team makeup demographic data, self-assessment, types of operations, typical communications
Attacker	Cyber attack campaigns based on motivation and group affiliations	Attack patterns affiliated with organization and artifacts associated with specific techniques

Table 10: Data needed per class

When building a model from scratch, designers must be selective in what behaviors to add. Too many additions at once introduces the risk of too much complexity too quickly, resulting in outcome variables which are hard to disentangle from behaviors. It's typically advised to add data and behaviors one at a time, so that those behaviors can be validated along the way. This is the tension between transparency and veridicality which can be analogous to simplicity and complexity [9]. Cyber-FIT was built using a spiral development methodology, each version adding a minimum amount of complexity, while adding research functionality as shown in the figure below.

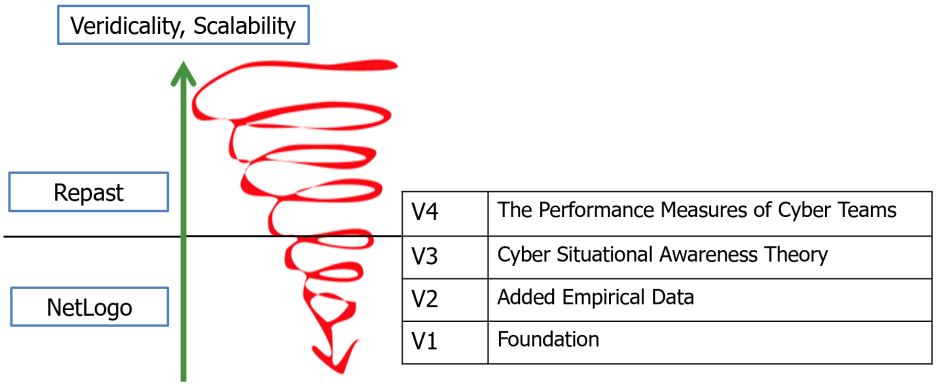


Figure 5: Spiral development methodology

Earlier version model defender agents only looked for vulnerable and compromised terrain agents and attempted to fix on the spot. If they were unsuccessful, they moved on to others. In Cyber-FIT version 4, a more complex defender agent behavior was designed. This was in the form of 1) keeping track of the known system vulnerabilities, 2) continuing to attempt to restore known compromised machines at a restoral work selection rate, 3) communicating with other team members about known vulnerabilities and compromises, 4) selecting terrains to interact with based on squad assignment, and 5) selecting operations from a list. This level of veridicality was a sufficient change to observe outcome variables in the form of virtual experiments, sensitivity analysis, and validation efforts. From the table above, this is the defender agent behavior that would be updated. Future versions will address more complexity in terms of terrain and attacker behaviors.

While considering the behavior changes for defender agents, many questions come up such as: How many team members are on a typical team? What is their makeup of knowledge, skill, and experience? How often should team members send information to each other? How long should operations take? How much time transpires between operations? Does it take longer to conduct operations based on knowledge, skill, or experience? The list of questions can go on and on. With the primary design change in this version being the behaviors that could be influenced by knowledge, skill, and experience, a survey could help answer some of these questions. A Qualtrics survey was designed to collect data for this purpose. The questions were presented in three parts: demographics, interactions, and performance. The sixteen-question survey was taken anonymously by a random assortment of cyber security professionals advertised through several cyber security email distribution lists.

The first questions in the demographics category asked the respondents to report on information regarding experience type, experience length, self-assessment of skill, and level of education. The answers to these questions provided data with which an approximation of the typical team can be made. The typical team has 6 – 10 personnel, has a wide variance of experience, and is highly educated. The next questions in the interactions category asked about the frequency of types of operations and communications during normal operations and during incident operations. The data shows that cyber team members conduct a variety of operations and during incidents, the frequency of communications and interactions goes up. Finally, the questions in the performance category asked questions about how the respondents assessed their own performance, and how their performance was assessed by leadership and management. The data show that cyber team members don't typically understand how well their team is performing, and there is a wide variety of ways that teams are assessed. All survey data and summary statistics are provided in the Appendix.

With the completion of the cyber team survey, the data validation goal for this version of Cyber-FIT was reached. In version 4, there are several sources of validated input data affecting model behaviors coded in the software. This adds to the overall validity of the system processes and the realistic behaviors and output data, which is described in the next section in the form of face validation. The table below details all the input data and control variable data that is based on either empirical data or system behavior from literature.

Input Name	Description	Data Source
Terrain Vulnerability Growth Rate	The terrain vulnerability growth rate (VGR) is based on the MITRE CVE database which tracks all known software vulnerabilities per	MITRE Common Vulnerabilities

	operating system type and version. The VGR can be controlled by OS type, patch level, environment, or timing within the mission.	Enumeration database [10]
Defender Knowledge	Defender knowledge level is a quantification of the sum total knowledge acquired over the individual's cyber security career which can be made up of formal education and certifications.	Cyber Team Survey
Defender Skill	Defender skill level is a quantification of the inherent level of skill the individual possesses. This agent trait is most difficult to quantify due to its nebulous nature.	Cyber Team Survey
Defender Experience	Defender experience is the easiest trait to quantify as it is simply the amount of time the individual has spent working in the cyber security industry	Cyber Team Survey
Defender Cyber Operations	The defender agents, when conducting normal operations pull from a list of operation types based on the CISA NICE Cyber Security Framework	CISA cyber operation types [8]
Defender Interaction Rate	The defender agents will choose to interact with other defender agents or terrain agents and this will be increased when cyber incidents are occurring modeling real world bustiness	Cyber Team Survey
Attacker Kill Chain Phase Time	The amount of time an attacker spends in each of the cyber kill chain phases	Empirical Data [11]
Attacker Zero-Day Development	The chances that a tier six attacker agent can develop a zero-day attack	Empirical Data [12]
Attacker Tier Level	The tier level of the attacker agent from one to six based on Defense Science Board Report	Defense Science Board Report [4]
Mission Terrain Configuration	The number of networking devices, servers, and host terrain agents per mission	Empirical Data [13]
Mission Cyber Operations	The cyber mission to conduct which are one of three types: survey, secure, and protect	Gaining Cyber Dominance Technical Report [14]

Table 11: Data validation summary

3.3.3 Face Validation

There are two guiding questions [3] for face validation. The first is: "when looked at in a systematic way, do the assumptions upon which the model is based seem plausible"? The second is: "do the model results look right"? The face validation of Cyber-FIT was accomplished by holding a focus group of three experienced cyber security subject matter experts. All three participants have more than twenty years of cyber security experience. One has active-duty military experience only, another has active-duty military experience, is retired, and now has six years of industry experience, and the third has over twenty years of industry experience. This mixture was sought so that active-duty only, active-duty/industry mixture, and industry only perspectives would be included. Cyber-FIT is a military style software simulation tool, but the concepts are general enough that someone without military experience would still understand the underlying cyber security concepts. In fact, most industry security operations centers operate in similar fashions to military cyber protection teams, so the carryover is apparent.

The focus group was held in January 2022 over zoom for one hour. There were two parts to the focus group. In part 1, the previously described cyber team survey was reviewed to face validate the responses. All three individuals validated that their experience was in line with the survey responses. Two questions were of most interest to the focus group. The first was the question where 19 out of 20 respondents said that interactions amongst team members increases

during cyber incidents. According to the focus group there are three important dimensions of the burstiness in communications and activity associated with this team behavior, all based on stress. The first is stress around environment familiarity. Stress will manifest itself in different ways, most usually associated with how well the team knows the environment (the cyber terrain). Teams that are unfamiliar with the computer network they are protecting will have a much higher amount of stress. This will lead to looking for things in a myriad of places because they might not understand exactly what certain security tools are reporting on, or the details of the configuration. Teams with more experience and knowledge of the environment will be able to dial into the tools that are most useful for that specific problem they are seeing. Cyber-FIT does model uncertainty with the agents, some percentage of the time, doing nothing, because of confusion. Also, Cyber-FIT will increase the interactions between cyber team machines and network machines, increasing the computer-computer connections. The experts agreed this was a potentially useful behavior and could be extended in many ways based on team member variables. The differences that the experts know happen in real operations could be experimented with. Second, stress will increase as time goes on if the problem is not identified. Teams would typically become hastier in their searches over time, and the searches (connections to machines and observing dashboards) would go deeper into the network. Also, interactions amongst team members would increase and would be apparent through a variety of tools. Finally, stress is highly dependent on reporting requirements. Most security operation center, and certainly all military cyber teams, will have specific reporting instructions that must be followed based on the severity of the incident. The higher the severity, the higher the stress. If there is a report due every hour, with updated details about what is being done and what has been found so far, then that will drive the activities of the team. The higher up the reporting chain in the organization, the higher the stress and the higher number of managers involved. All three of these stress responses could be modeled, simulated, and experimented with in Cyber-FIT.

The other cyber team survey question of particular interest to the focus group was about perceptions of how well cyber teams perform. The question was “On cyber teams you’ve been on, do you typically have a good understanding of how well the team is performing?”. Six respondents said no, two respondents said unsure, and eleven said yes. Overall, 42% of respondents either didn’t know or were unsure. If this is representative of cyber teams at large, this is an enormous number of cyber security professionals not understanding if their operations are positively affecting the organization. The focus group participants were not surprised by this result and thought it tracked well with their experience. As experienced cyber security professionals, senior among their peers and typically in leadership positions, they did have a sense of how their teams perform, but it would be hard to quantify, which is one of the primary drivers of Cyber-FIT development. The focus group brought up several reasons for this overall misunderstanding. First is attrition amongst the information technology professionals in both military and industry environments. Measuring performance successfully is a long process of baselining, setting improvement targets and then reassessing. All those activities are measuring the skills of the people involved. If the team experiences turnover, then the people are different, and some sense of measurement is disrupted. Another issue like attrition is the changing technological environment that the team works in. If new cyber tools are introduced, or there are major changes to the network being protected, this changes how the team works and throws off previous measurements of performance. This means that the best measurement methodologies should be tool and environment agnostic, instead focusing on the team behaviors and processes.

The focus group gave feedback about how teams are typically assessed. “Purple” teams are an industry standard where the team will exercise how an attacker (red) and defender (blue) might engage in the operational environment. This can be a tabletop exercise where documentation is examined, and a leader works through a set of questions. Another way that teams are frequently assessed is through external audits like a consulting company testing the team, or a penetration testing team attempting to hack into the network. The focus group also identified cyber competitions and exercises as a way that teams are currently assessed. Competitions can show how teams stack up against other teams. The problem is that the competitions are almost never a realistic match compared to what the teams do in normal operations.

Next, the focus group discussed ways that team performance could be assessed given no financial or resource constraints. They all agreed that the ultimate mechanism would be a high-fidelity cyber range in a virtual environment where any possible cyber incident that might come their way could be simulated, appearing just like how it would manifest in their own network. Essentially, a practice field that is a replication of their real field, just like sports teams practice on. There is existing literature [15] [16] on this line of research and clearly an active need for military applications. If the cyber range is up and operational, the organization can run two teams through the same scenario and see who does better. Since most organizations cannot afford to create a realistic practice range, the next best option is using the organizational data already present and defining business metrics that are tailored to the organization. This is hard in practice, but good cyber leadership can make things like this happen. For example, most cyber teams use an incident tracking system of some sort with “cases” that occur. The case data is a good source of information for what was discovered, how quickly, who worked on it, was the appropriate attributions made, etc., etc. By creating a standardized reporting process, analytics can be developed showing how well the team is performing. In a similar mindset, data about what the users are doing on the network can also be used to approximate team performance. If users need to access certain assets, then how well the information technology is providing that access is a performance measure. Server logs, internet traffic, and database logs can provide that information.

The focus group then received a demonstration of Cyber-FIT version four with a realistic simulation and walked through the output measures (cyber team performance measures). The focus group agreed that time to compromise (from attacker perspective) is one of the most used within training and exercises in controlled environments. Cyber teams will defend a network and the attacker will try to compromise machines, with the best teams able to maximize time to compromise. The others that are most prevalent, according to the focus group, are: time to detect and time to restore. Finding and fixing problems is the primary purpose of the cyber defense team. Finally, compromise time is also one of the most important metrics because it is essentially combining time to detect and time to restore. All of the other measures make sense from a cyber leadership perspective but aren’t currently being tracked because of the difficulty in gaining the relevant data.

The final portion of the focus group was discussing the applicability of Cyber-FIT to real world problems. Each member of the focus group was able to provide a different use case that would be of interest to parties in positions of cyber leadership. The first would be to use it for war-gaming as a course of action (COA) analysis. COA analysis is used at high levels of military analysis and there is little in the way of war-gamin for cyber currently available. Secondly, the model could be used for policy analysis. Many cyber policies use language that is vague. Running a Cyber-FIT simulation with the definitions laid out by NIST and the DoD would

provide a computational analysis of what the policy is prescribing in the form of cyber work roles. Last, the model can be used for virtual experimentation when it comes to assisting in decision analysis. How should a cyber team be trained and when should it deploy? This is done by altering configurations of input and control variables and observing differences in outcomes. Overall, the focus group was extremely positive in the usefulness of the Cyber-FIT framework. There is certainly a gap in the science that this model is addressing since none of the focus group members have ever come across a tool that is addressing an extremely clear need. The two questions of face validation are affirmed to be positive.

3.3.4 Process and agent validation

North and Macal, when listing [3] out the validation types for agent-based modeling list "process validation" as a different type than "agent validation". The guiding research question for process validation is: "Do the steps in the model and the internal flows of what is being modeled correspond to the real-world process"? Separately, the guiding research question for agent validation is: "Do agent behaviors and interaction mechanisms correspond to agents in the real world"? For Cyber-FIT, it would be too difficult to separate those two questions. The agent behaviors and interaction mechanisms are the internal flows. For other agent-based models, that depend on more business rules, or already existing activities and processes that are independent of the agent behaviors, this separation would make sense. But for Cyber-FIT both questions will be addressed with the same analysis.

To begin with process validation, consider the workings of all the agents together in the model. The figure below shows a screenshot of the Cyber-FIT version four user interface along with pictorial representations of the agent types. As shown, there are four agents working together: terrain agents, friendly agents, defender agents, and attacker agents. Each agent process will be described next.

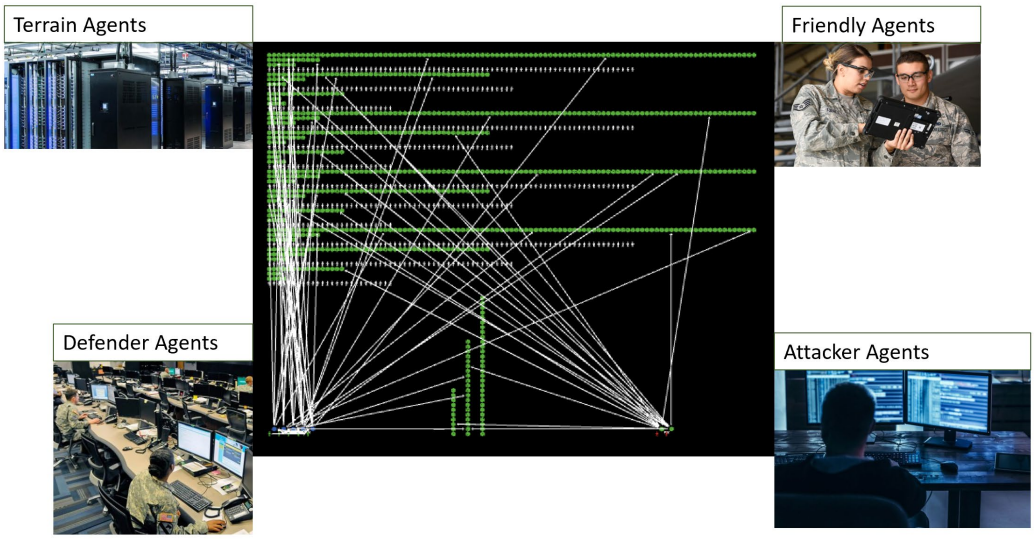


Figure 6: Cyber-FIT representation

Each time step all terrain agents stochastically generate new vulnerabilities that are added to its vulnerability array variable. This is based on the terrain vulnerability rate control variable

and vulnerability generation method. Next terrain status values are updated based on if other interactions have changed the terrain's status from one possible state to working, degraded, or compromised. Finally, terrain statistics are updated collecting temporal data on vulnerabilities and terrain status. The terrain agent step algorithm is in the next table below.

Algorithm: Terrain Agent Step	
1:	if (generate_vulnerabilities == true)
2:	then add vulnerability ID number to *vulnerability[]* array
3:	Update *terrain_status* values
4:	Generate terrain_statistic values

Table 12: Terrain agent step algorithm

Each time step all defender agents either complete restoral operation, continue their current operation, or get a new operation. During restoral operations the defender agent will connect to its workstation and then connect to a known compromised terrain agent to attempt to restore it. If the defender agent is not aware of any compromised terrain, it will either continue its current operation, or select a new operation to conduct. Finally, the defender agent updates values associated with team performance and mission information. The defender agent step algorithm is in the next table below.

Algorithm: Defender Agent Step	
1:	if (compromised_terrain == true)
2:	then restoral operations AND message_team_lead
3:	else if (*operation_complete* == false)
4:	then continue_operation AND message_team
5:	else get_operation
6:	Update *cyber_mission* values
7:	Update situation_awareness values
8:	Update *performance* values

Table 13: Defender agent step algorithm

Each time step all friendly agents stochastically interact with cyber terrain agents to read information from that cyber terrain. The terrain agent will send back information if it is normally operating (not compromised) at a speed based on the vulnerability level of the cyber terrain agents associated with that mission. This is simulating the primary usage of the computer network: reading information necessary for doing their job. The friendly agent step algorithm is in the next table below.

Algorithm: Friendly Agent Step	
1:	if (get_information == true)
2:	then read_mission_terrain_agent
3:	Update *cyber_mission* values

Table 14: Friendly agent step algorithm

Each time step all attacker agents work through the cyber kill chain which is: reconnaissance, weaponization, delivery, exploitation, command and control, and actions on objectives. Each phase has stochastic elements regarding time spent in the phase along with behavior and success criteria. The attacker agents will update performance values along the way in certain phases. The attacker agent step algorithm is in the next table below.

```
Algorithm: Attacker Agent Step
1:    switch (phase)
2:        case 0: if (phase_complete == false)
                      initialize_attack_resources
                      else set phase = 1
3:        case 1: if (phase_complete == false)
                      read_terrain_vulnerabilities
                  else set phase = 2
4:        case 2: if (phase_complete == false)
                      weaponize_attacks
                          if (weaponize_fails) set phase = 1 || 0
                  else set phase = 3
5:        case 3: if (phase_complete == false)
                      deliver_payload
                  else set phase = 4
6:        case 4: if (phase_complete == false)
                      if (attack_success) set phase = 5
                  else set phase = 0
7:        case 5: if (phase_complete == false)
                      command_and_control
                  else set phase = 6
8:        case 6: if (phase_complete == false)
                      actions_on_objectives
                  else set phase = 7
9:    Update attacker_statistics values
```

Table 15: Attacker agent step algorithm

The process and agent validation of Cyber-FIT was completed first through many conversations with subject matter experts, cyber military personnel, and other researchers. Then it was finalized with the focus group described in the previous section. Ultimately, each agent is following internal flows and processes that map to real world behaviors. One way this has been designed is by imagining any behavior that could possibly be added to the model and ensuring that there is a place for that behavior. If there is, then the underlying mechanism corresponds to real world behavior, at a basic level. For example, if a use case emerged to study how cyber operation error rates would affect the team performance, there is a place in the continue_operation() method in the table above to add an error rate that would affect the cyber operational behavior of the defender agent.

3.3.5 Model Output Validation

The guiding question [3] for model output validation is: "if the real-world system is available for study, do the model outputs match the outputs of the real-world system?" This validation type is traditionally what people think about when they think of "validation" in a general sense. Does the simulation match reality? Obviously, this is very difficult. Difficulties arise for many reasons including scope of simulation, complexity of the model, difficulty with noise and perturbations, and availability of empirical data. In many instances, the goal of a simulation model is to go from idea to grounded theory. An example is using the Construct simulation tool to simulate organizational behavior, validated by empirical communication data resulting in a grounded theory of referential data knowledge transfer [17]. Cyber-FIT could

potentially lead to grounded theory on cyber team performance. Cyber-FIT model output was validated using empirical network data.

In previous work, a cyber situation awareness dashboard was improved with network science data [18]. That research was asking the following questions: "does binning data, and then calculating graph level measures, provide a more granular picture of the network so that anomaly detection is easier to accomplish? If so, can we create "normally operating" network science-based signatures? How can organizations use these insights, incorporating their known patterns of life, to achieve enhanced cyber situational awareness?" In summary, that research was able to show three key findings. First, binning the data did result in different distributions of network measures. Second, the network data showed strong signs of periodicity. Third, patterns of life incorporated into dynamic network analysis improved cyber situation awareness.

That work went through a systematic three step process to gather and analyze the data. The first step retrieves flow records into four bins using criteria to separate human and autonomic in and out traffic. These records are exported in comma-separated-value format. Step two imports the saved files into the CASOS tool ORA and converts them to DyNetML files, allowing for network data inspection. Step three conducts a dynamic network analysis on the four datasets measuring various network measures on an hourly basis. Autonomic in traffic is traffic flowing into the network and likely generated by computer software (no human engagement). Analysis of the autonomic in traffic showed that the average density over the entire data set was .0002 and average network centralization total-degree was .0004. These measures were found on NetFlow covering approximately 25,000 nodes (computer hosts). The table below shows the binned network measures results from the study.

NetFlow Type	Avg. Density	Std. Dev	Avg. Centralization, Total Degree	Std. Dev
Autonomic In	0.000243	0.000081	0.000405	0.000621
Autonomic Out	0.000189	0.000080	0.000981	0.000265
Human In	0.000096	0.000034	0.000872	0.000321
Human Out	0.000130	0.000049	0.001075	0.000299

Table 16: Netflow empirical data

So, will Cyber-FIT output similar network measures? This will be an excellent test of the framework. To test this Cyber-FIT is set up to support 20 kinetic missions consisting of 2,500 computer nodes. For this simulation, both the cyber team (defender agents) and the adversary (attacker agents) are turned off. This simulates removal of human traffic within the network. Also, since Cyber-FIT only simulates the internal network traffic, this is akin to the bin of inflow traffic only. Essentially, this simulation is set up to only track the autonomic inflow, like the empirical data set it will be compared to. The simulation is run for five simulated days (7,200 ticks) and the terrain agent to terrain agent interactions are collected. This data is ingested into ORA and analyzed as a dynamic network over five simulated days. Next, a sample of the NetFlow data (one hour of each day of the full set) is ingested into ORA and key framed by day (like the Cyber-FIT simulation data) in order to do a side-by-side comparison. This sampling of empirical data includes all bins, but as noted in the table above the different bins don't display

huge differences, they are all on the scale of 10^{-3}. We are hoping to see that Cyber-FIT can output on the same scale. The figure below shows the results of a dynamic network analysis using ORA for both the empirical data (on the left) and the simulation data (on the right).

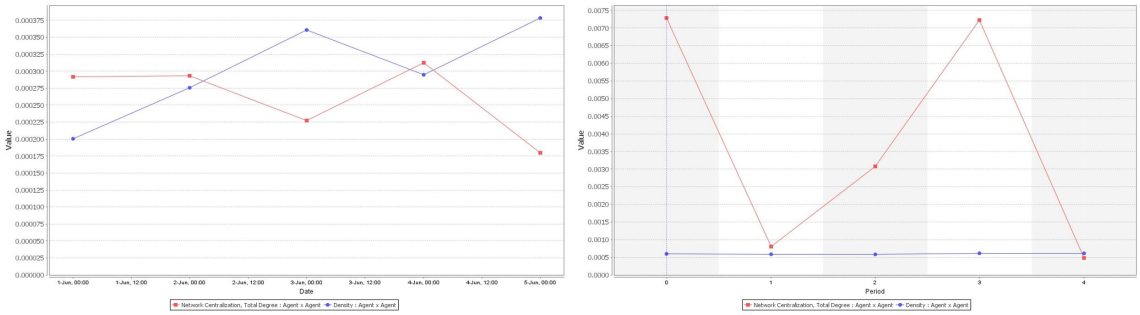


Figure 7: Model output validation results

The results show that Cyber-FIT can simulate similarly scaled network measures. On the left, the empirical data sample, over five days, results in a range of density values of [0.000200 – 0.000378]. The empirical data results in a range of network centralization total degree values of [0.00018 – 0.000313]. The simulated data results in a range of density values of [0.000590 – 0.000611] and a range of network centralization total degree values of [0.000473 – 0.007282]. In summary, the model is outputting network data that is measured at the graph level on the same order as empirical data. This shows that the framework overall is doing its primary job of being a test bed with which experimentation can lead to very realistic simulation outcomes. The simulated data shows a much higher variance in network centralization total degree than density, as expected. The software simulates connections amongst networked computers at a low rate and this means that over time approximately the same number of connections will occur during time periods. However, the randomization of the structure of the networked computers will be different in different time periods. This is due to the current version of Cyber-FIT not enforcing routing protocols. Any terrain agent can connect to any other terrain agent. In the empirical data this is different. The empirical data shows low variance in network centralization total degree because the structure of the network will be similar between time periods because routing protocols are enforced.

3.3.6 Theory validation

The guiding question [3] for theory validation is: "Does the model make a valid use of the theory?" This validation type is arguably the most difficult to present a clear case for. The previous validation types of agent-based models are much clearer in what is being described and claimed. One can easily observe a flow chart and understand the totality of what is occurring. This can then be compared with natural phenomena and the parts that are not being modeled or abstracted away can be discussed. This ends with either agreement or disagreement on the validity of a concept or behavior. There may be disagreement, but at least the disagreement can be pointed to. The same could be said about input, data, and output validation. Theory validation is much more abstract. The main theoretical research area of Cyber-FIT is computational and mathematical organization theory. A definition of this theory is: "Computational and mathematical organization theory is an interdisciplinary scientific area whose research members

focus on developing and testing organizational theory using formal models. The community shares a theoretical view of organizations as collections of processes and intelligent adaptive agents that are task oriented, socially situated, technologically bound, and continuously changing" [19]. This is exactly what Cyber-FIT sets out to do, most specifically modeling the cyber team as adaptive agents. Each part of that theory definition can provide clarity on how Cyber-FIT does make a valid use of computational and mathematical organization theory. This is the overarching theory that this model aims to extend and contribute to. When moving down a level to its many sub-components, other theories can be integrated and validated as well. The table below provides a description of each of the theories that are validated within Cyber-FIT.

Overarching	
Theory	Agent Classes
Computational and Mathematical Organization Theory	Defender, Terrain, Attacker, Friendly, Interactions
Sub-component Theories	
Theory	Agent Classes
Cybercrime	Attacker
Cyber Situation Awareness	Defender
Performance Theory	Defender
Network Science	Defender, Terrain, Interactions

Table 17: Theory validation summary

Speaking from a cyber team perspective, the team is made up of intelligent adaptive agents. They keep track of what is occurring on the cyber terrain they are interacting with, providing a simulation of intelligence. The cyber team agents are task oriented. They are always working on a specific defensive cyber operation with a goal. Once the goal is reached, they move on to a new task. They are socially situated, that is they will communicate with other team members to pass informational messages. They are technologically bound because of the agent rulesets that determine what they will or will not do. Finally, the cyber team agents are continuously changing. On every tick they either continue their current operation (updating their own cognitive model of the terrain agents), getting a new operation (based on what is occurring in the environment), working to restore compromised terrain agents, or doing nothing (simulating and tracking stuck time). All in all, this model, holistically, is primarily planted in computational and mathematical organization theory.

Moving on to the sub-component theories, the first is cybercrime, which was the driver of Cyber-FIT version two. In version two, the attacker agents were upgraded to force their behavior through the cyber kill chain. This is a simulation model of an aspect of cybercrime. Gordon and Ford define cybercrime as "any crime that is facilitated or committed using a computer, network, or hardware device" [20]. They go on to differentiate two types of cybercrime where type I is associated with technology and type II is associated with the human element. Cyber-FIT version two could be described as modeling this type II aspect of cybercrime. The cyber kill chain itself, is an attempt at merging the technological and human aspects of what nearly always must occur to compromise a computer system. To better understand the human aspects of cybercrime, many have contributed to adversarial modeling, tangentially related to this work. For example, states of adversary behavior can be simulated, and or observed which creates graphs. This can be used to build simulation models for security system evaluation [21]. Cyber-FIT can be extended to

simulate increasingly complex attacker behaviors and then output many different data temporally.

Cyber situation awareness theory working definition used by Cyber-FIT is given by Onwubiko [22] as "processes and technology required to gain awareness of historic, current, and impending (future) situations in cyber". This drives the agent and team level computations of cyber situation awareness. Cyber situation awareness is measured as a function based on knowledge (past), comprehension (current) and projection (future), of the cyber operations. As of this writing, situation awareness is not a measurement that cyber teams discuss quantitatively, it is an abstract concept. This work provides a simple mechanism for simulating cyber conflict, and specifically defining the data that could compute cyber situation awareness. Situation awareness has always been difficult to measure, even in situations where relevant knowledge is much clearer. A famous example is the work of Endsley to query fighter pilots and determine at different points in a simulated mission if they could pinpoint enemy locations [23]. Cyber-FIT can be used in a similar fashion with military cyber teams. This work contributes to the field of cyber situation awareness theory by 1) creating a novel metric computationally defining cyber situation awareness and 2) providing a software framework to experiment with that definition or extend it.

Performance theory is also incorporated into this model as the current version output measures define the performance of the simulated cyber team. Measuring performance is typically situational [24] and teams with a specific common goal are usually easiest to measure. Work has been done using surveys for teams with tasks more difficult to define [25]. Cyber team performance measurement has been studied recently by various works. One study used a combination of self-assessments, exercise data, and observer data to compare teams. This study identified several performance measures that are very similar to those simulated in Cyber-FIT including "attack discovery", "vulnerability removal", and "DMZ attack success rate" [26]. Cyber-FIT as constructed can support all the measurement types and styles proposed in each of these works. Due to the object-oriented nature of the software, each agent can be instrumented, so to speak, with any imaginable data structure. For example, if the performance metric depends on a new data definition, it can be added to the team object, individual agent object, or some number of the terrain objects.

The last sub-component theory is network science. A key design decision for the architecture of the framework in version one was to link the agents. This proved very useful, especially in version four where the object-oriented nature of the model made it easy to collect link information temporally. Network science measures can be computed for the entirety of the simulated conflict. A primary contribution of this work is that based on the current version, any network science measure could be analyzed as it relates to the links connecting agents. This could be communication networks, computer architecture networks, attack graphs, etc. Cyber-FIT is now well suited to carrying out simulations of organizational change. An illustrative example is a simulation where the trend in stability changed as number of employees was increased for an organization [27]. Cyber-FIT could be used to determine if this behavior can generalize to cyber operations communications and learning within the organization.

3.3.7 Validation Conclusion

In summary, this model has been validated, in some way, using all validation strategies described by North and Macal [3]. The art, rather than science, of model development is the key

driver for validation decisions. Cyber-FIT began as an abstract concept drawn on a whiteboard. Once version one was completed, it was clear that there was something interesting and novel in place. Version two was the first validation attempt as the model was tuned to simulate ranges of outcome variables that matched the empirical data provided by the Alphaville exercise [28]. Version three incorporated the first specific theoretical validation by incorporating cyber situational awareness. Version four was a large overhaul and re-architecture of the software and included all remaining validation types. It is difficult to clearly articulate precisely what parts of a model are validated, in what way, and by how much. This is a difficulty encountered by nearly all software developers – the conceptual model differs from person to person. This is precisely why the validation in parts was described systematically, type by type. A summary of this can be shown, visually, in different ways, depending on how the parts of the model are categorized. The next two figures are representations of the totality of validation in parts for the entire model.

Behavior Data	Requirements Validation	Data Validation	Face Validation	Process Validation	Output Validation	Agent Validation	Theory Validation
Terrain Vuln. Growth Rate	☑	☑	☑			☑	
Defender K, S, E	☑	☑	☑			☑	☑
Defender Cyber Ops	☑	☑	☑	☑		☑	☑
Defender Interaction Rate	☑	☑	☑	☑		☑	
Attacker Phase Time	☑	☑	☑	☑	☑	☑	☑
Attacker Zero-Day Rate	☑	☑					
Attacker Tier Level	☑	☑	☑				
Terrain Configuration	☑	☑	☑				
Mission Cyber Ops	☑	☑	☑	☑		☑	

Figure 8: Validation in parts by behavior

Output Data	Requirements Validation	Data Validation	Face Validation	Process Validation	Output Validation	Agent Validation	Theory Validation
Terrain Vulnerability Rate	☑		☑	☑		☑	
Terrain Compromise Rate	☑		☑	☑		☑	
Time to Detect	☑		☑	☑		☑	
Time to Restore	☑		☑	☑		☑	
Operational Efficiency	☑		☑				
Cyber Situation Awareness	☑		☑				☑
Cyber Mission Capability Rate	☑		☑	☑			
Time to Compromise	☑		☑	☑		☑	
Compromise Rate	☑		☑	☑		☑	
Network Measures	☑		☑		☑		
Mission Measures	☑		☑				

Figure 9: Validation in parts by output

3.4 Organizational Multi-Modeling

A new agent-based simulation model that can be used to analyze various organizational policies has been created at CASOS called the Organization Simulation in Response to Intrusion

Strategies (OSIRIS). This model [29] uses Cyber-FIT as the engine to drive defender and attacker actions and overlay an organizational system where different agents simulate typical user behaviors such as using desktop applications, accessing the internet, and checking social media sites. The interface affords researchers the ability to design an organization from scratch and assign computing resources to each simulated user agent along with social network connections and reporting structures as shown in the following figure.

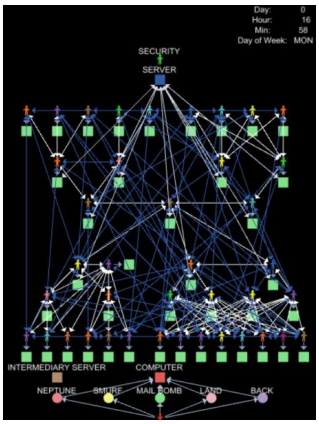


Figure 10: OSIRIS interface

OSIRIS has been used for several simulations and virtual experiments such as assessing the efficacy of the human firewall with improved cyber security awareness training [30]. OSIRIS and Cyber-FIT are under active development and have added the following human behavioral models: socio-physical: tiredness, socio-physical: stress, socio-physical: distracted, social network: social influence, social network: in the know (high degree centrality), social network: likelihood they will infect others (high outdegree), cognitive bias: escalation of commitment, cognitive bias: confirmation bias, cognitive bias: optimism bias, social cognitive: generalization, social cognitive: stereotyping, social cognitive: generalized other. For each behavior, a stochastic variable is added to agent software that changes the rates at which they respond and communicate along with how likely they are to successfully take corrective actions when responding to cyber incidents. Terrain agents (computing devices) for both OSIRIS and Cyber-FIT have added realistic vulnerabilities based on recent work [31].

4 CONCLUSIONS

Cyber-FIT agent-based modeling and simulation framework version four has been developed, used, analyzed, and validated in several ways. The current model is a realistic computational model of cyber team performance and can be used to run a wide range of virtual experiments and generate hypotheses valuable to cyber researchers. Cyber-FIT was leveraged to overlay a new model (OSIRIS) which has been used to simulate organizational responses to cyber attacks.

5 References

[1] G. B. Dobson and K. M. Carley, "Cyber-FIT: An Agent-Based Modelling Approach to Simulating Cyber Warfare," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, 2017.

[2] G. B. Dobson and K. M. Carley, "Cyber-fit agent-based simulation framework version 4," Carnegie Mellon University, Pittsburgh, PA, 2021.

[3] M. J. North and C. M. Macal, Managing Business Complexity: discovering strategic solutions with agent-based modeling and simulation, New York: Oxford University Press, 2007.

[4] Defense Science Board, "Resilient Military Systems and the Advanced Cyber Threat," Department of Defense, 2013.

[5] Department of Defense, "The DoD Cyber Strategy," Washington D.C., 2015.

[6] The White House, "Executive Order on America's Cybesecurity Workforce," 2 May 2019. [Online]. Available: https://www.whitehouse.gov/presidential-actions/executive-order-americas-cybersecurity-workforce/. [Accessed 3 May 2019].

[7] D. B. Johnson, "House set to debate bills on cyber education, President's Cup and TikTok," *SC Magazine,* 28 February 2022.

[8] W. Newhouse, S. Keith, B. Scribner and G. Witte, "National initiative for cybersecurity education (NICE) cybersecurity workforce framework," Cybersecurity and Infrastrcuture Security Agency, 2017.

[9] K. M. Carley, "Simulating society: The tension between transparency and veridicality," in *Agents*, Chicago, IL, 2002.

[10] MITRE, "Common Vulnerabilities Enumeration," MITRE, [Online]. Available: https://cve.mitre.org/. [Accessed 29 March 2022].

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

[12] L. Bilge and T. Dumitras, "Before we knew it: an empirical study of zero-day attacks in the real world," in *ACM conference on Computer and communications security*, Raleigh, NC, 2012.

[13] Applied Computer Research Inc., "Identifying IT Markets and Market Size by Number of Servers," 2011. [Online]. Available: https://www.missioncriticalmagazine.com/ext/resources/MC/Home/Files/PDFs/WP_ACR-IT-Server-Market.pdf. [Accessed 29 March 2022].

[14] G. Longo, "Gaining Cyber Dominance," Carnegie Mellon University Software Engineering Insititue, Pittsburgh, 2015.

[15] G. B. Dobson, T. G. Podnar, A. D. Cerini and L. J. Osterritter, "R-EACTR: A Framework for Designing Realistic Cyber Warfare Exercises," Software Engineering Institute, Pittsburgh, PA, 2017.

[16] T. G. Podnar, G. B. Dobson, D. D. Updyke and W. E. Reed, "Foundation of Cyber Ranges.," Software Engineering Institute, Pittsburgh, PA, 2021.

[17] C. Schreiber and K. Carley, "Going beyond the data: Empirical validation leading to grounded theory," *Computational & Mathematical Organization Theory,* vol. 10, no. 2, pp. 155-164, 2004.

[18] G. B. Dobson, T. J. Shimeall and K. M. Carley, "Towards Network Science Enhanced Cyber Situational Awareness," *International Journal of Cyber Situational Awareness,* vol. 2, no. 1, pp. 11-30, 2017.

[19] K. M. Carley, "Computational and mathematical organization theory: Perspective and directions," *Computational & mathematical organization theory,* vol. 1, no. 1, pp. 39-56, 1995.

[20] S. Gordon and R. Ford, "On the definition and classification of cybercrime," *Journal in computer virology,* vol. 2, no. 1, pp. 13-20, 2006.

[21] E. LeMay, W. Unkenholz, D. Parks, C. Muehrcke, K. Keefe and W. H. Sanders, "Adversary-driven state-based system security evaluation," in *6th International Workshop on Security Measurements and Metrics*, 2010.

[22] C. Onwubiko, "Understanding Cyber Situation Awareness," *International Journal on Cyber Situational Awareness,* vol. 1, no. 1, pp. 11-30, 2016.

[23] M. R. Endsley, "Measurement of situation awareness in dynamic systems," *Human factors,* vol. 37, no. 1, pp. 65-84, 1995.

[24] M. T. Brannick and C. Prince, "An overview of team performance measurement," in *Team performance assessment and measurement*, New York, Psychology Press, 1997, pp. 15-28.

[25] G. Hallam and D. Campbell, "The measurement of team performance with a standardized survey," in *Team performance assessment and measurement*, New York, Psychology Press, 1997, pp. 167-184.

[26] M. Granåsen and D. Andersson, "Measuring team effectiveness in cyber-defense exercises: a cross-disciplinary case study," *Cognition, Technology & Work,* vol. 18, no. 1, pp. 121-143, 2016.

[27] K. M. Carley and V. Hill, "Structural change and learning within organizations," *Dynamics of organizations: Computational modeling and organizational theories,* pp. 63-92, 2001.

[28] A. Rege, E. Parker , B. Singer and N. Masceri, "A qualitative exploration of adversarial adaptability, group dynamics, and cyber intrusion chains," *Journal of Information Warfare,* vol. 16, no. 3, pp. 1-16, 2018.

[29] J. Shin, G. B. Dobson, K. M. Carley and L. R. Carley, "OSIRIS: organization simulation in response to intrusion strategies," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, Pittsburgh, PA, 2022.

[30] J. Shin, L. R. Carley, G. B. Dobson and K. M. Carley, "Modeling and simulation of the human firewall against phishing attacks in small and medium-sized businesses," in *2023 Annual Modeling and Simulation Conference (ANNSIM)*, 2023.

[31] J. Shin, G. B. Dobson, L. R. Carley and K. M. Carley, "Revelation of System and Human Vulnerabilities Across MITRE ATT&CK Techniques with Insights from ChatGPT," Carnegie Mellon University, Pittsburgh, PA, 2023.

[32] G. B. Dobson and K. M. Carley, "Cyber-FIT: An Agent-Based Modelling Approach to Simulating Cyber Warfare," in *International Conference on Social Computing, Behavioral-*

Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation, Washington D.C., 2017.

[33] G. B. Dobson, A. Rege and K. M. Carley, "Virtual Cyber Warfare Experiments Based on Empirically Observed Adversarial Intrusion Chain Behavior," in *13th International Conference on Cyber Warfare and Security*, 2018.

[34] G. B. Dobson and K. M. Carley, "A Computational Model of Cyber Situational Awareness," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation* , Washinton D.C., 2018.

[35] G. B. Dobson and K. M. Carley, "Cyber-FIT Agent-Based Simulation Framework Version 4," Center for the Computational Analysis of Social and Organizational Systems, Pittsburgh, PA, 2021.

[36] J. Shin, G. B. Dobson, K. L. Carley and L. R. Carely, "OSIRIS: Organization Simulation in Response to Intrusion Strategies," in *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, Pittsburgh, PA, 2022.

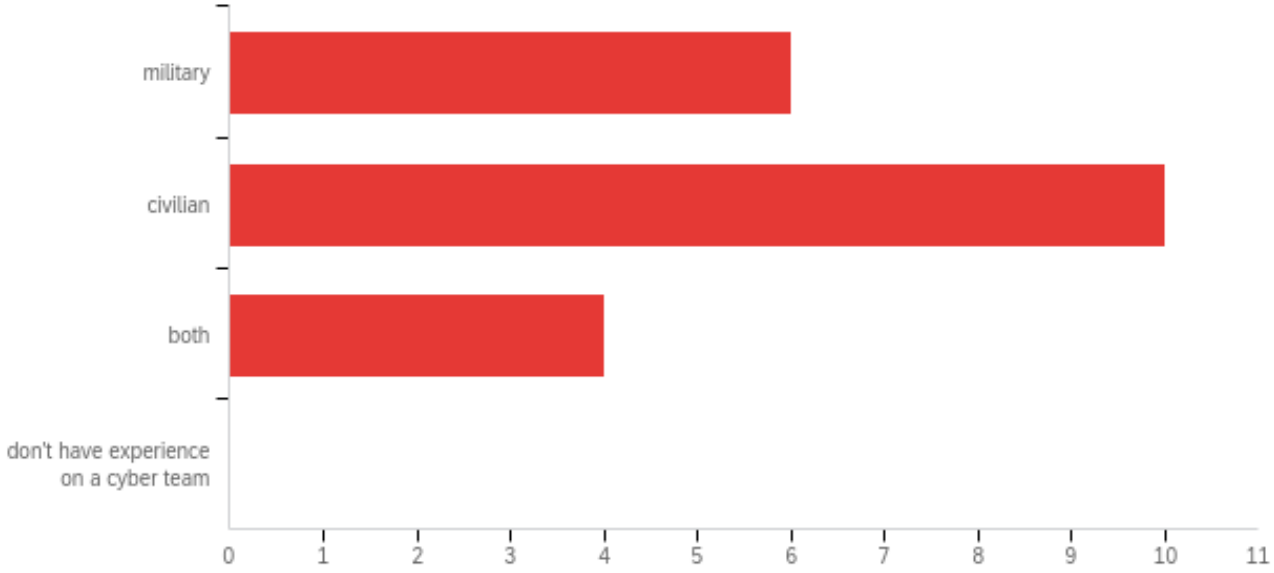
[37] MITRE. [Online]. Available: https://attack.mitre.org/.

[38] G. E. Box, "Robustness in the Strategy of Scientific Model Building," in *Robustness in Statistics*, G. N. W. ROBERT L. LAUNER, Ed., Research Triangle Park, NC, Academic Press, 1979, pp. 201-236.

[39] US Air Force, "Research Methods and Technologies for Blended Live and Synthetic Personalized Learning, Modeling and Assessment Open Broad Agency Announcement," 2020.

6 APPENDIX A – SURVEY RESULTS

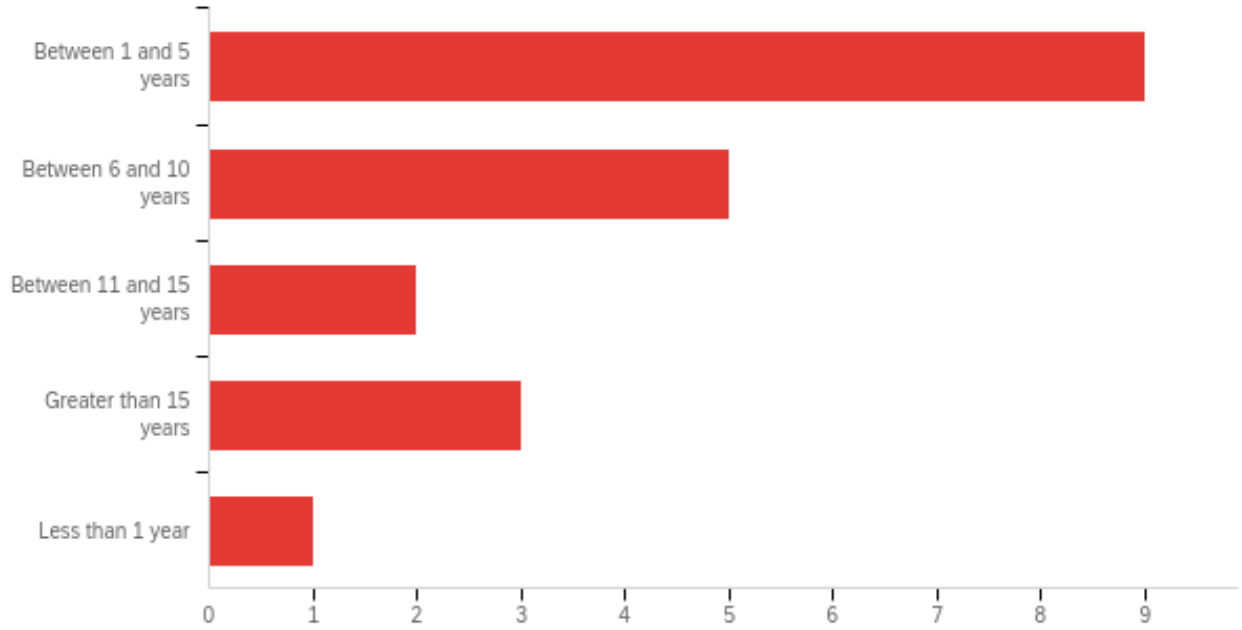
Q1 - Do you have experience on a military or civilian cyber team?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Do you have experience on a military or civilian cyber team?	1.00	3.00	1.90	0.70	0.49	20

#	Answer	%	Count
1	military	30.00%	6
2	civilian	50.00%	10
3	both	20.00%	4
4	I don't have experience on a cyber team	0.00%	0
	Total	100%	20

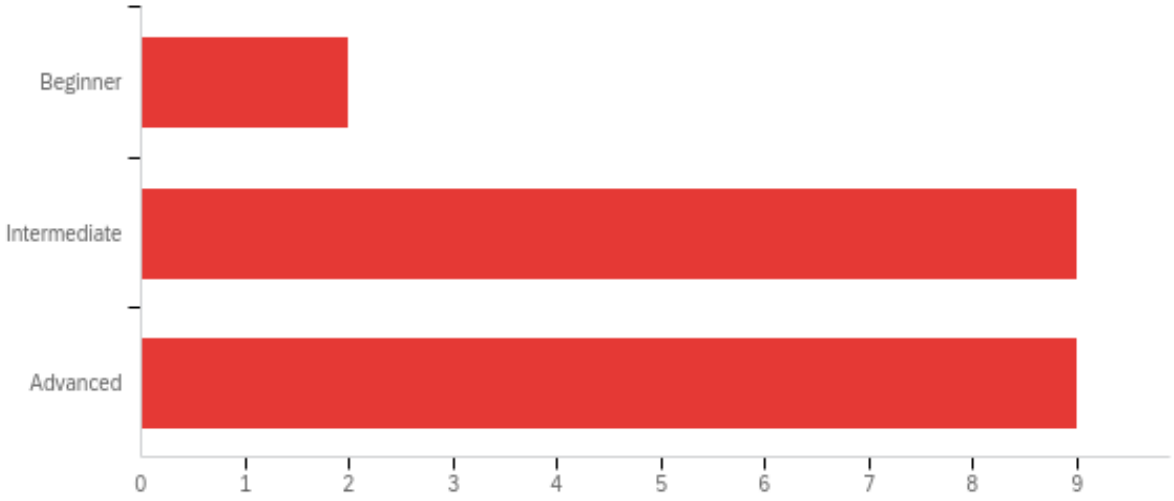
Q2 - How many years of experience do you have on a cyber security team?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How many years of experience do you have on a cyber security team?	1.00	5.00	2.10	1.26	1.59	20

#	Answer	%	Count
1	Between 1 and 5 years	45.00%	9
2	Between 6 and 10 years	25.00%	5
3	Between 11 and 15 years	10.00%	2
4	Greater than 15 years	15.00%	3
5	Less than 1 year	5.00%	1
	Total	100%	20

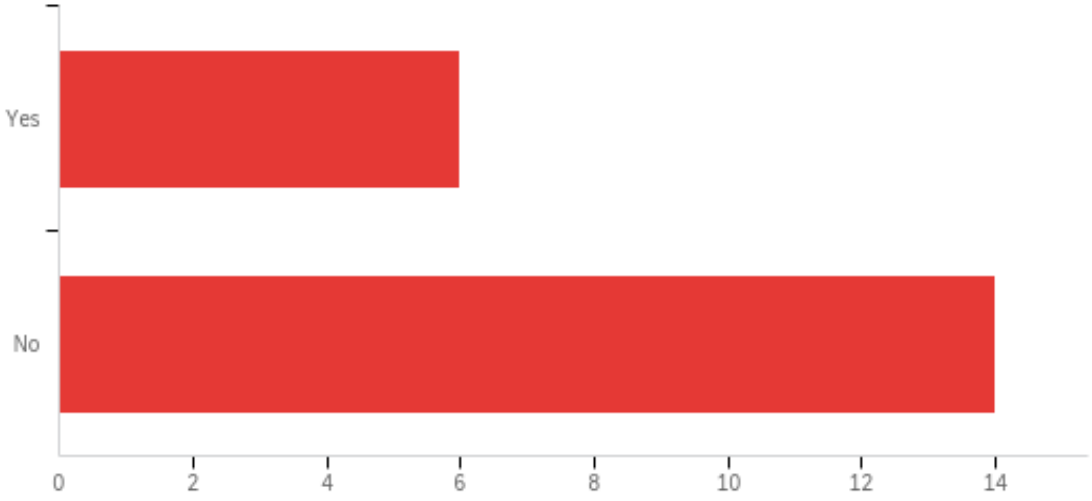
Q3 - What is your assessment of your current cyber security skill level? (This is your assessment of the technical cyber security skills needed to complete tasks associated with your job)



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	What is your assessment of your current cyber security skill level? (This is your assessment of the technical cyber security skills needed to complete tasks associated with your job)	1.00	3.00	2.35	0.65	0.43	20

#	Answer	%	Count
1	Beginner	10.00%	2
2	Intermediate	45.00%	9
3	Advanced	45.00%	9
	Total	100%	20

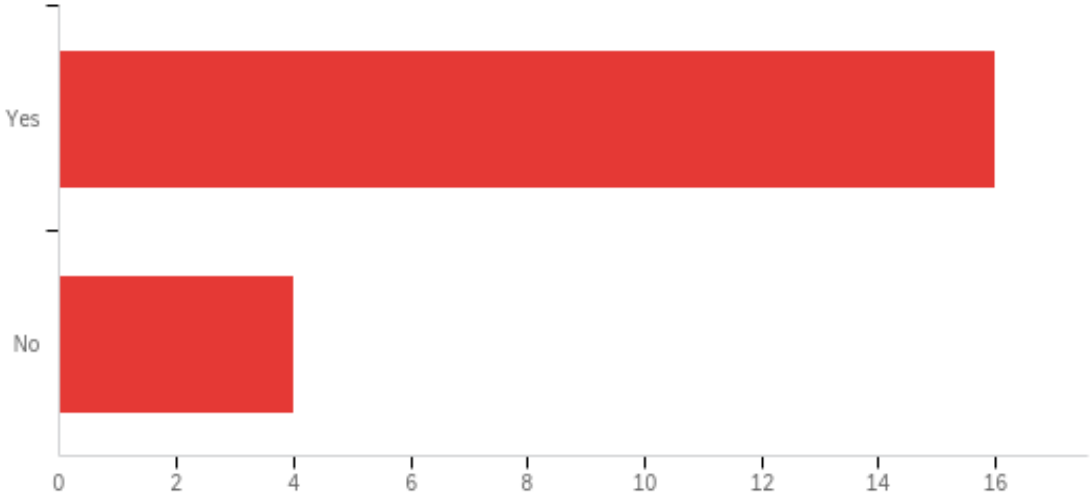
Q4 - Do you have an associate's degree in information technology, computers, or cyber security?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Do you have an Associate's degree in information technology, computers, or cyber security?	1.00	2.00	1.70	0.46	0.21	20

#	Answer	%	Count
1	Yes	30.00%	6
2	No	70.00%	14
	Total	100%	20

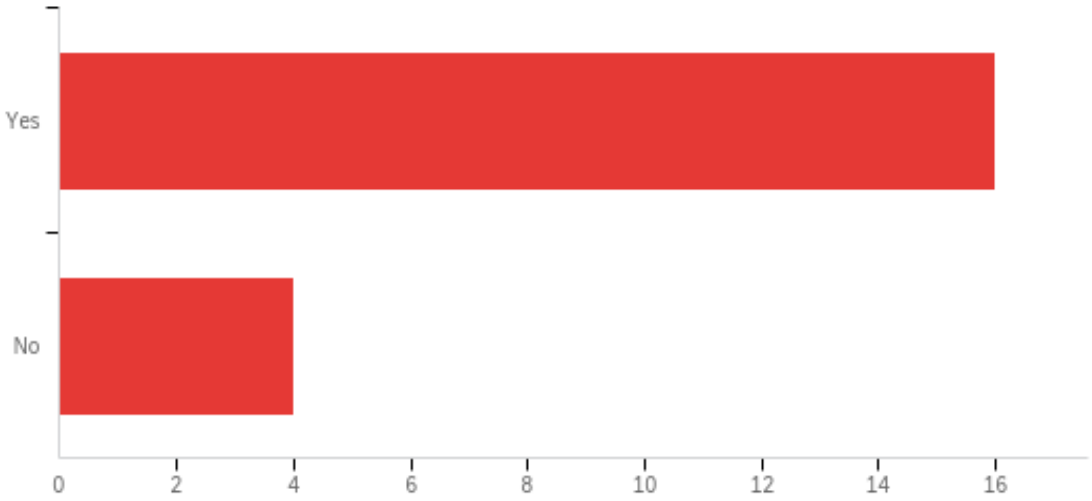
Q5 - Do you have a bachelor's degree in information technology, computers, or cyber security?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Do you have a Bachelor's degree in information technology, computers, or cyber security?	1.00	2.00	1.20	0.40	0.16	20

#	Answer	%	Count
1	Yes	80.00%	16
2	No	20.00%	4
	Total	100%	20

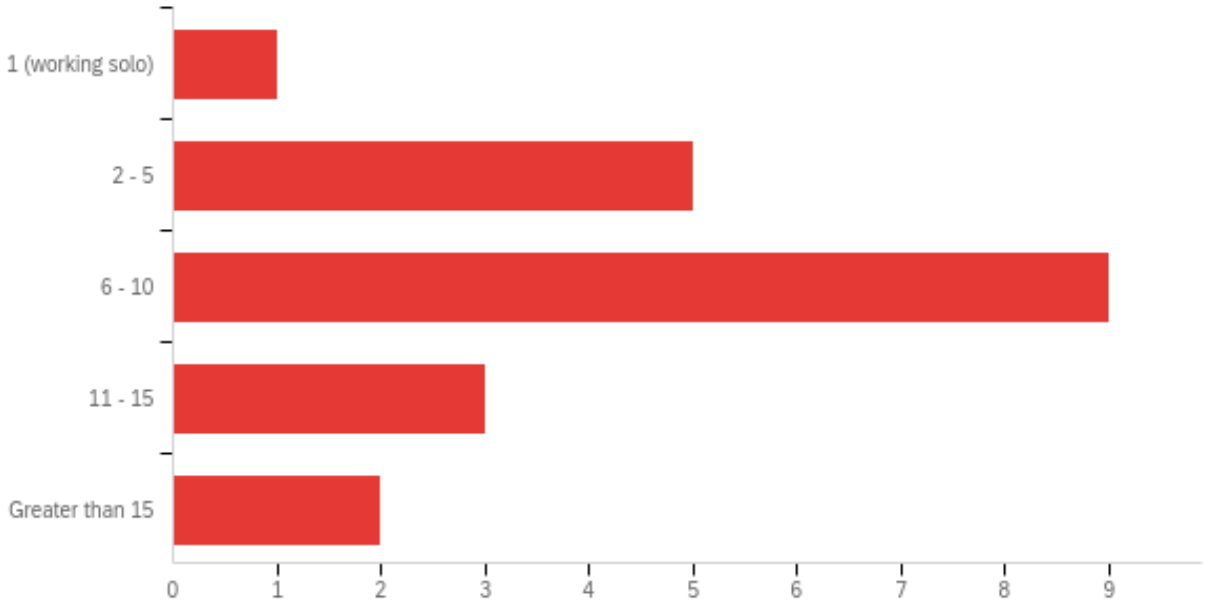
Q6 - Do you have an industry recognized certification such as Security+, CISSP, or other?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Do you have an industry recognized certification such as Security+, CISSP, or other?	1.00	2.00	1.20	0.40	0.16	20

#	Answer	%	Count
1	Yes	80.00%	16
2	No	20.00%	4
	Total	100%	20

Q7 - How many personnel are typically on cyber teams that you've worked on?



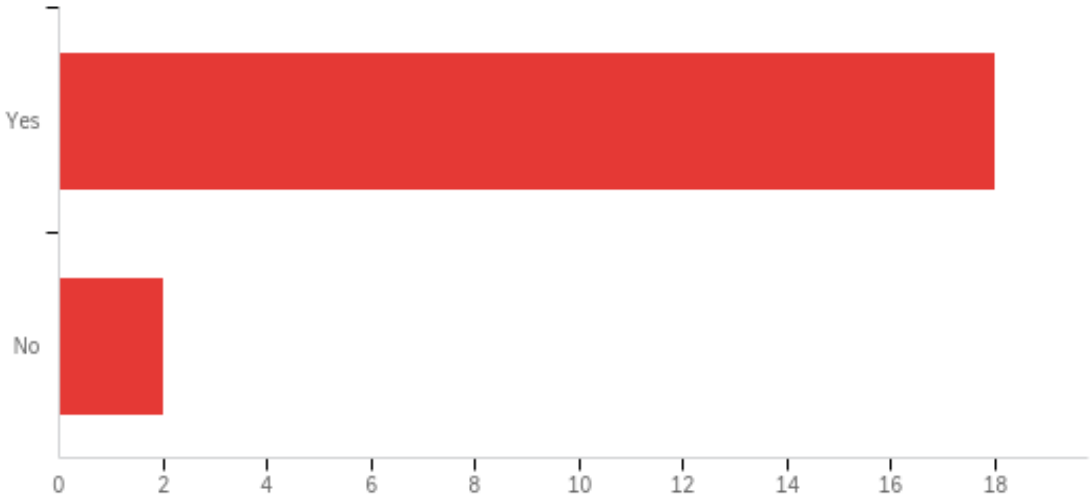
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	How many personnel are typically on cyber teams that you've worked on?	1.00	5.00	3.00	1.00	1.00	20

#	Answer	%	Count
1	1 (working solo)	5.00%	1
2	2 - 5	25.00%	5
3	6 - 10	45.00%	9
4	11 - 15	15.00%	3
5	Greater than 15	10.00%	2
	Total	100%	20

Q8 - During normal team operations, what percentage of the time are you doing the following types of tasks? (Your answers must add up to 100, shown in the total)

#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Surveying systems	0.00	90.00	20.25	18.47	341.19	20
2	Updating and working on systems	0.00	60.00	19.25	14.08	198.19	20
3	Interacting with other team members	0.00	60.00	29.00	14.46	209.00	20
4	Reporting about systems	0.00	50.00	20.25	11.67	136.19	20
5	Other	0.00	45.00	11.25	12.54	157.19	20

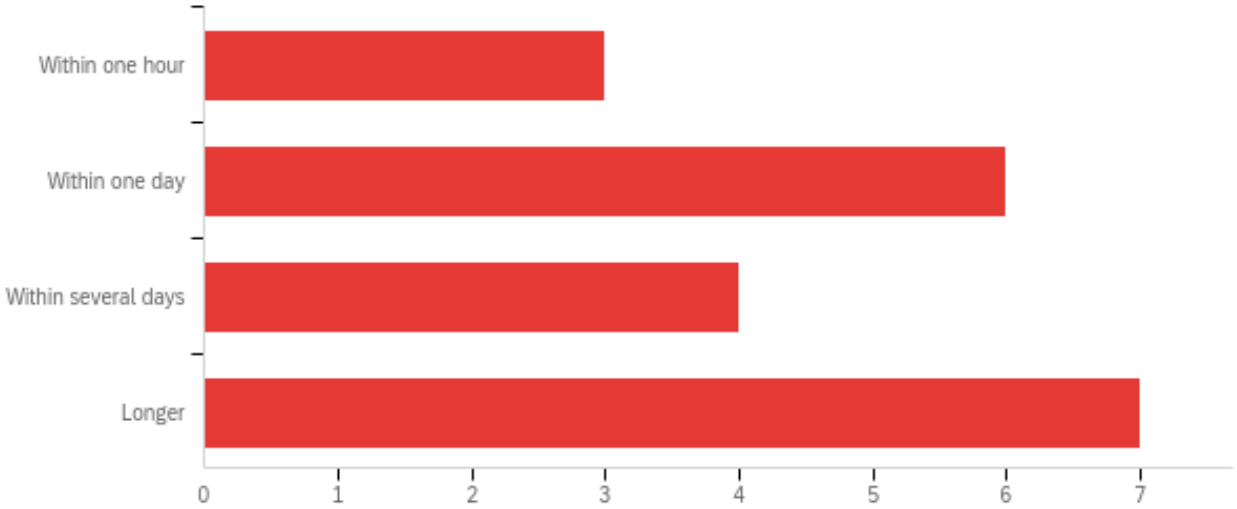
Q9 - Have you experienced cyber incidents in an operational environment?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Have you experienced cyber incidents in an operational environment?	1.00	2.00	1.10	0.30	0.09	20

#	Answer	%	Count
1	Yes	90.00%	18
2	No	10.00%	2
	Total	100%	20

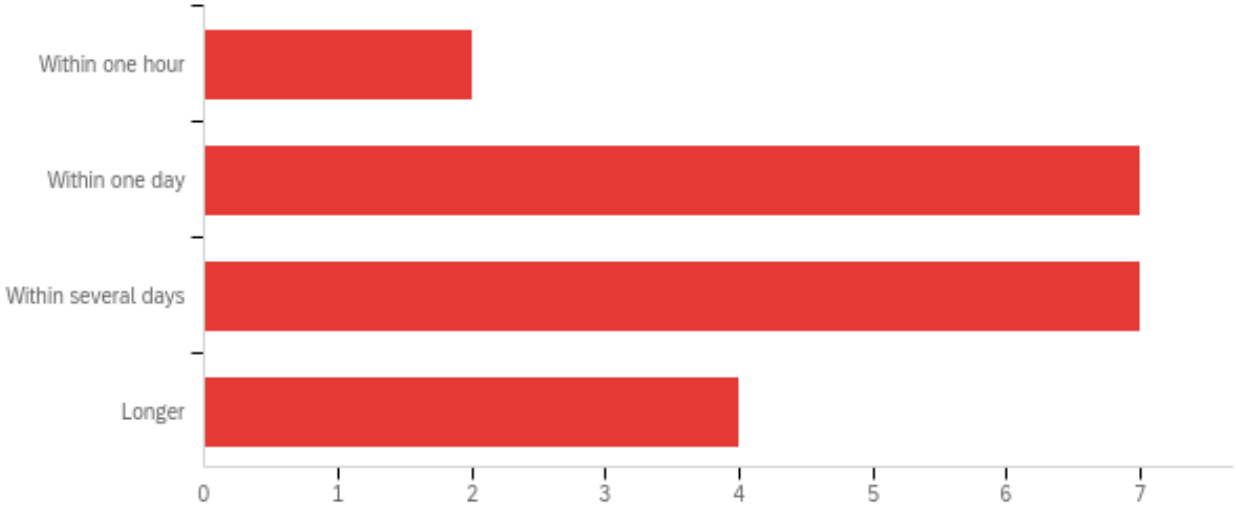
Q10 - When experiencing cyber incidents, how long after the incident actually began, on average, are you alerted (through a security system or human investigation)



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	When experiencing cyber incidents, how long after the incident actually began, on average, are you alerted (through a security system or human investigation)	1.00	4.00	2.75	1.09	1.19	20

#	Answer	%	Count
1	Within one hour	15.00%	3
2	Within one day	30.00%	6
3	Within several days	20.00%	4
4	Longer	35.00%	7
	Total	100%	20

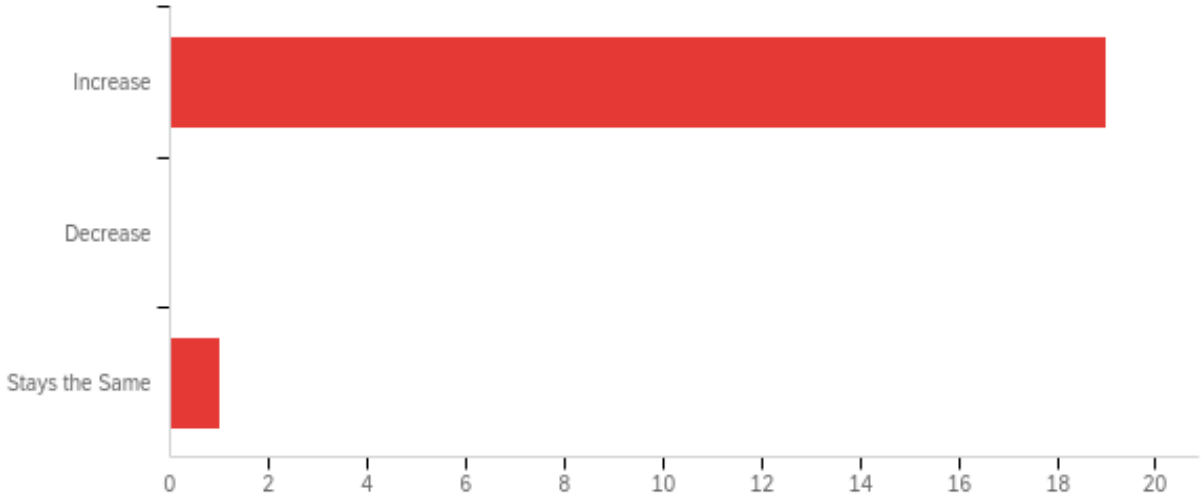
Q11 - When experiencing cyber incidents, how long after the incident is identified, on average, does it take to mitigate?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	When experiencing cyber incidents, how long after the incident is identified, on average, does it take to mitigate?	1.00	4.00	2.65	0.91	0.83	20

#	Answer	%	Count
1	Within one hour	10.00%	2
2	Within one day	35.00%	7
3	Within several days	35.00%	7
4	Longer	20.00%	4
	Total	100%	20

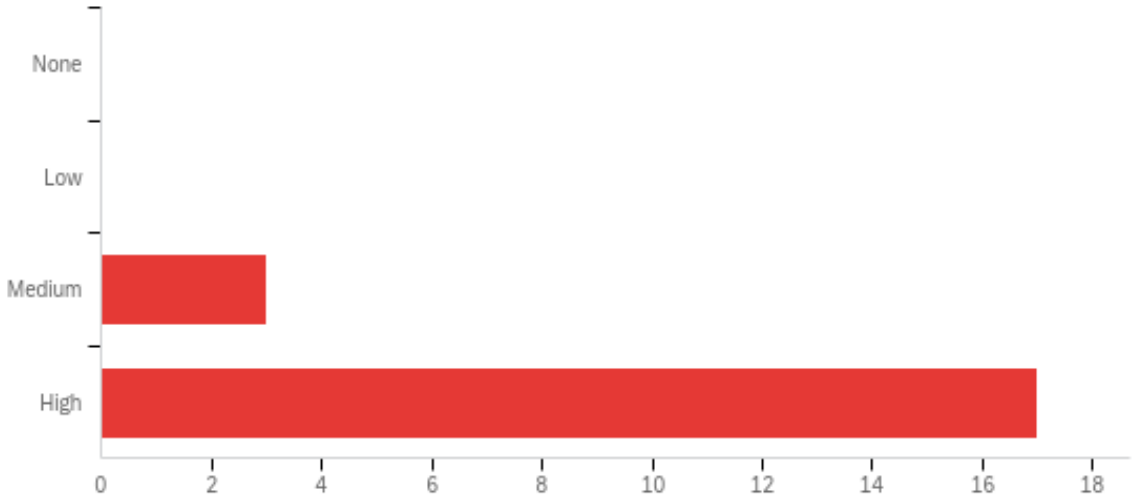
Q12 - Think about how much you interact with other members of your cyber security team. When comparing the amount of interaction you have, does the level of interaction during an incident, as compared to normal operations:



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Think about how much you interact with other members of your cyber security team. When comparing the amount of interaction you have, does the level of interaction during an incident, as compared to normal operations:	1.00	3.00	1.10	0.44	0.19	20

#	Answer	%	Count
1	Increase	95.00%	19
2	Decrease	0.00%	0
3	Stays the Same	5.00%	1
	Total	100%	20

Q13 - During a cyber incident, is the level of interaction within your cyber team



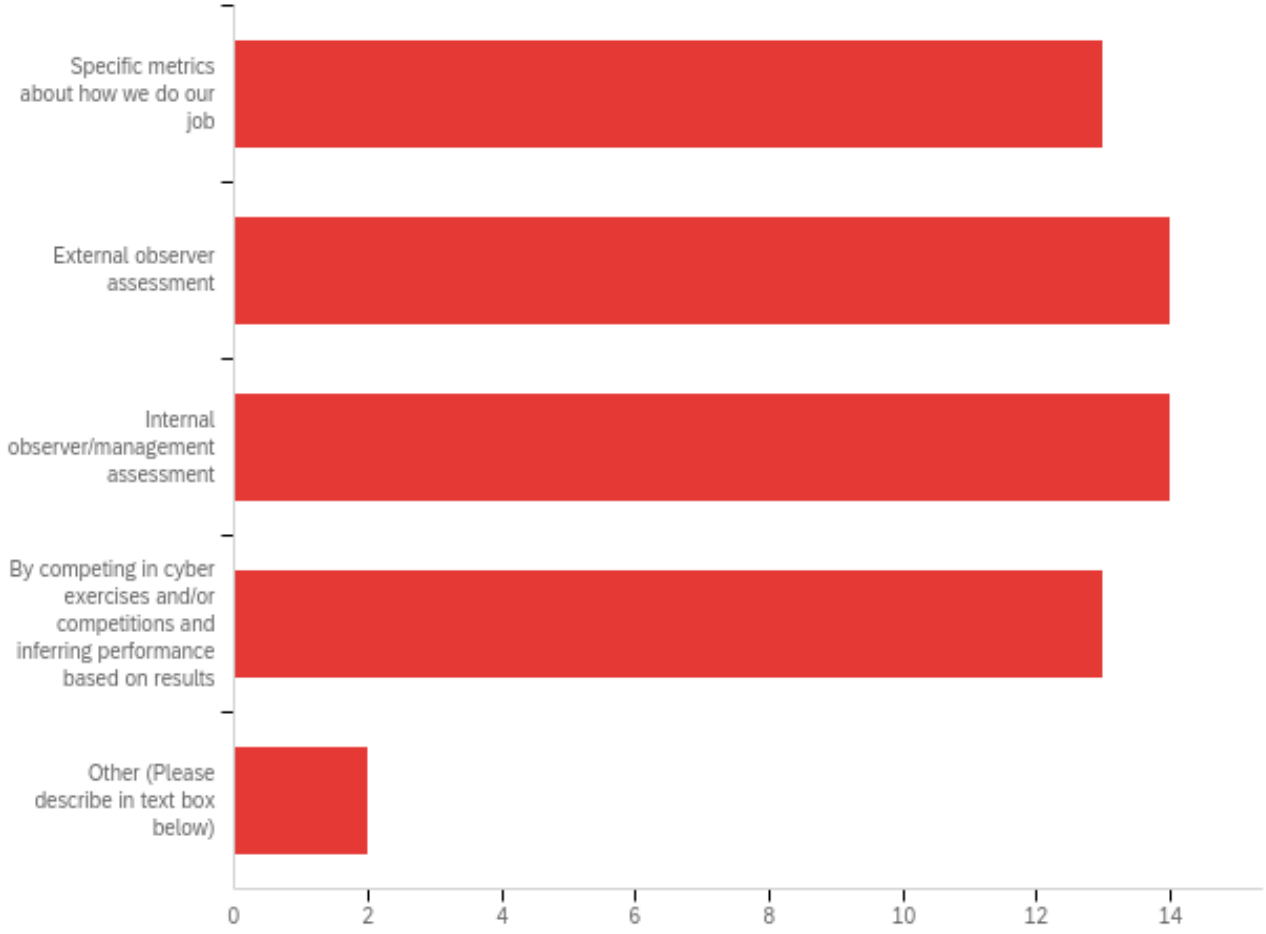
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	During a cyber incident, is the level of interaction within your cyber team	3.00	4.00	3.85	0.36	0.13	20

#	Answer	%	Count
1	None	0.00%	0
2	Low	0.00%	0
3	Medium	15.00%	3
4	High	85.00%	17
	Total	100%	20

Q14 - During operations where an incident has been recognized, what percentage of the time are you doing the following types of tasks? (Your answers must add up to 100, shown in the total)

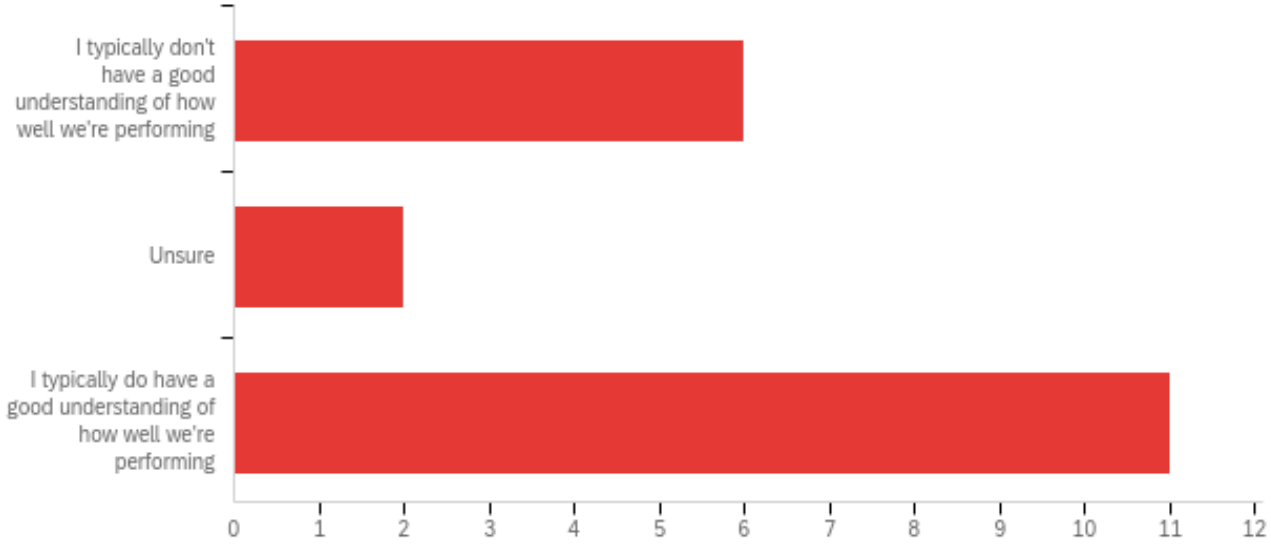
#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	Surveying systems	0.00	50.00	19.00	14.20	201.50	20
2	Updating systems	0.00	40.00	10.50	12.34	152.25	20
3	Interacting with team members	10.00	50.00	36.00	11.58	134.00	20
4	Reporting about systems	0.00	50.00	17.75	14.18	201.19	20
5	Restoring systems that have been compromised	0.00	50.00	11.75	12.07	145.69	20
6	Other	0.00	30.00	5.56	8.48	71.91	18

Q15 - On cyber teams I've been on, our performance has been assessed in the following ways: (select all that apply)



#	Answer	%	Count
1	Specific metrics about how we do our job	23.21%	13
2	External observer assessment	25.00%	14
3	Internal observer/management assessment	25.00%	14
4	By competing in cyber exercises and/or competitions and inferring performance based on results	23.21%	13
5	Other (Please describe in text box below)	3.57%	2
	Total	100%	56

Q16 - On cyber teams you've been on, do you typically have an understanding of how well the team is performing?



#	Field	Minimum	Maximum	Mean	Std Deviation	Variance	Count
1	On cyber teams you've been on, do you typically have an understanding of how well the team is performing?	1.00	4.00	2.95	1.36	1.84	19

#	Answer	%	Count
1	I typically don't have a good understanding of how well we're performing	31.58%	6
3	Unsure	10.53%	2
4	I typically do have a good understanding of how well we're performing	57.89%	11
	Total	100%	19