Data Farming and Quantitative Analysis of Cyber Defense Technologies and Measures

Gary Edward Horne
Blue Canopy Group
Reston, Virginia
ghorne@bluecanopy.com

Santiago Balestrini Robinson
Georgia Tech Research Institute
Atlanta, Georgia
sanbales@gatech.edu

ABSTRACT

Data Farming is a quantified approach that examines questions in large possibility spaces using modeling and simulation. It evaluates whole landscapes of outcomes to draw insights from outcome distributions and outliers. The Data Farming Support to NATO task group has codified the data farming methodology and a follow-on task group is now applying data farming to important NATO question areas. The development of data mining and data visualization techniques also continues to help us understand the huge amount of simulation data resulting from data farming.

This paper outlines the documented data farming techniques, illustrates data farming in the context of quantitative analysis of cyber defense technologies and measures, and describes the links to predictive analyses made possible by advancing the connection of data farming to data mining and data visualization. The paper describes a prototype simulation using an agent-based model (NetLogo) that the NATO task group team has developed with feedback from subject-matter experts. The paper also describes the data farming the team has performed on key model parameters to begin to get insight into cyber security "what-if" questions.

ABOUT THE AUTHORS

Gary Edward Horne is a Research Analyst at Blue Canopy Group with a doctorate in Operations Research from The George Washington University. During his career in defense analysis, he has led data farming efforts examining questions in areas such as humanitarian assistance, convoy protection, and anti-terrorist response. He chaired the NATO Modeling and Simulation Task Group MSG-088, "Data Farming Support to NATO," which completed documentation of the data farming process in 2014. He continues work with NATO as co-chair of the follow-on task group "Developing Actionable Data Farming Decision Support for NATO."

Santiago Balestrini Robinson is a Research Engineer II at the Electronic Systems Laboratory in the Georgia Tech Research Institute. His interests in quantitative modeling and simulation span multiple domains, ranging from large-scale military campaign level analyses to the development of strategic-level cyber defense models, as well as humanitarian aid and disaster relief analyses. In conjunction with quantitative modeling and simulation-based analysis, he also has developed novel opinion-based decision support tools. He earned a B.S., an M.S., and Ph.D. in aerospace engineering from the Georgia Institute of Technology in 2003, 2006 and 2009, respectively.

Data Farming and Quantitative Analysis of Cyber Defense Technologies and Measures

Gary Edward Horne
Blue Canopy Group
Reston, Virginia
ghorne@bluecanopy.com

Santiago Balestrini Robinson
Georgia Tech Research Institute
Atlanta, Georgia
sanbales@gatech.edu

INTRODUCTION

Data Farming is a quantified approach that examines questions in large possibility spaces using modeling and simulation. It evaluates whole landscapes of outcomes to draw insights from outcome distributions and outliers. This evaluation is made possible by "growing" massive amounts of data through the execution of many simulation runs. The name *Data Farming* was initially coined in 1997 (Horne, 1997). Since that time the data farming community has grown to include people from over a dozen nations. Data farming continues to evolve from initial work in a United States Marine Corps effort called Project Albert (Hoffman and Horne, 1998) to the work documented in the latest edition of the *Scythe* (Horne and Meyer, 2015) documenting Workshop 29 held in March 2015 in Finland. The *Scythe* is the publication of the International Data Farming Community that contains proceedings of workshops that have been held over the years. The 30th Workshop just took place in February 2016 in Catania, Italy and the *Scythe* for that event is in processing at this time and will be available soon at www.datafarming.org.

Data farming uses simulation in a collaborative and iterative team process (Horne and Meyer, 2004) that has been used primarily in defense applications (Horne and Meyer, 2010). This process normally requires input and participation by subject matter experts, modellers, analysts, and decision-makers. Data farming is a process that has been developed to support decision-makers by answering questions that are not currently addressed. Data farming uses an inter-disciplinary approach that includes modeling and simulation, high performance computing, and statistical analysis to examine questions of interest with large number of alternatives. Data farming allows for the examination of uncertain events with numerous possible outcomes and provides the capability of executing enough experiments so that both overall and unexpected results may be captured and examined for insights.

In 2010, the NATO Research and Technology Organization started the three-year Modeling and Simulation Task Group "Data Farming in Support of NATO" to assess and document the data farming methodology to be used for decision support. This group was called MSG-088 and this paper includes a summary of the six realms of data farming as outlined during the course of MSG-088 (Horne et al., 2014). Upon completion of MSG-088, a follow-on task group called "Developing Actionable Data Farming Decision Support" was initiated by NATO and was designated MSG-124. This new three-year task group is performing work in selected application areas important to NATO, one of which is cyber defense.

This paper outlines data farming techniques and illustrates data farming in the context of quantitative analysis of cyber defense technologies and measures. The paper describes a prototype simulation using an agent-based model (NetLogo) that the NATO task group team has developed and has vetted with experts, incorporating their recommendations. The paper also describes some of the data farming efforts the team has performed as the work continues with the goal of getting insight into cyber security what-if? questions.

DATA FARMING LOOP OF LOOPS

Data farming uses an iterative approach that is illustrated by the loop of loops in Figure 1 (www.datafarming.org, 2016). The first realm, rapid prototyping, works with the second realm, model development, iteratively in an experiment definition loop. A rapidly prototyped model provides a starting point in examining the initial questions and the model development regimen supports the model implementation, defining the resolution, scope, and data

requirements. The third realm, design of experiments, enables the execution of a broad input factor space while keeping the computational requirements within feasible limits. High performance computing, realm four, allows for the execution of the many simulation runs that is both a necessity and a major advantage of data farming. The fifth realm, analysis and visualization, involves techniques and tools for examining the large output of data resulting from data farming efforts. The final realm, collaborative processes, underlies the entire data farming process and these processes will be described in detail in this section. These realms are described in detail in Horne et al., 2014, but will be summarized below.

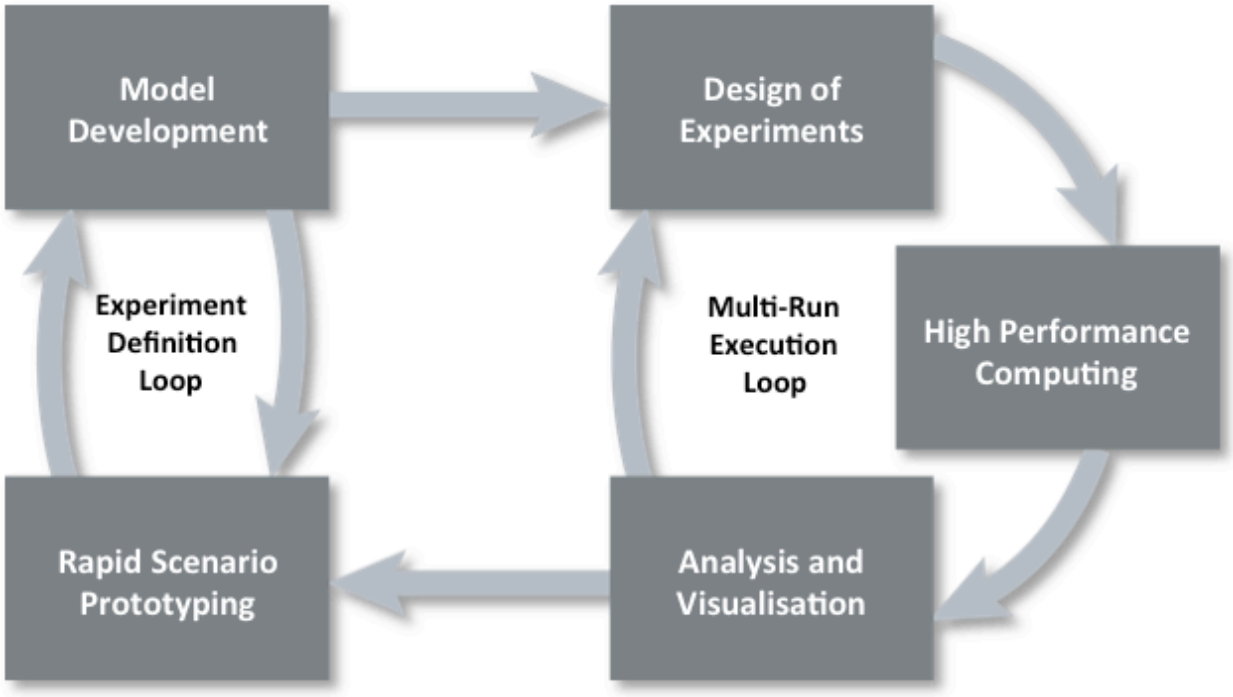

Figure 1. Data Farming Loop of Loops

Rapid Scenario Prototyping

The model development and the rapid prototyping realms of data farming together make up the experiment definition loop in Figure 1. As such, they work hand-in-hand with each other and we could choose either realm to begin our detailed description of data farming. Thus the rapid scenario prototyping process is a good place to start our discussion, although starting with model development realm would also be appropriate.

As with the data farming process in general, the rapid scenario prototyping should always be within the context of the questions to be answered. These questions have to be prepared in such a way that simulation can help to find answers and to get insights. The most important step here is to define measurements to be collected by means of simulation together with required input and output data for the simulation. In most cases this step already requires some rough ideas about the scenario settings. Thus, this realm simply represents the initial formation of the basics of a scenario to be simulated.

Model Development

As stated in the previous subsection, the model development realm works hand-in-hand with the rapid scenario prototyping realm in the experiment definition loop on the left side of Figure 1. The fundamental output of this loop is a scenario instantiated in a working simulation model that captures the essence of a question and that can be sent to the multi-run execution loop of the data farming process. Of course, more insight into the question, refinement of the question, and/or deeper examination of the question may be enabled later through a return to the experiment definition loop later in the process.

When developing models, both modeling and subject matter experts should be present. Rapid scenario prototyping provides model requirements for model development. For example, it is important to do one thing well, such as creating aggregated models that combine simple models instead of building single monolithic models, whenever possible. The more independent models are from each other, the better the potential results. Thus, one needs to encourage modularization and clear separation of different models, including development practices for using models of different aggregation level and scope. Other important characteristics of models as they are developed include reusability, interoperability, repeatability, and thorough documentation. And, finally, openness should be encouraged, including the sharing of source code when possible given other constraints.

Design of Experiments

Design of experiments is one of the three realms of data farming in the multi-run execution loop. Along with the realms of high performance computing and analysis and visualization, the realm of design of experiments allow us to perform multiple runs to gain simulation results over a wide landscape of possibilities. Simulation models have many inputs or parameters (factors) that can be changed to explore alternatives. A designed experiment is a carefully chosen set of combinations of these inputs, called design points, at which the simulation model will be run. Design of experiments provides smarter ways of setting up the experiment that facilitate follow-on analysis and visualization of results in a reasonable amount of time.

Changing the factors in a brute force way, by looking at all possible combinations, is impractical or impossible, except for extremely simplistic simulations with only a handful of factors. Changing the factors all at once limits your insights. It will allow you to see whether or not this changes the responses, but you will not be able to tell why the changes occur. For example, if mission effectiveness improves when you equip a squad with better sensors and better weapons, you will not know whether it is the weapon or the sensor that has the most impact. Changing the factors one at a time also limits your insights. If the squad gets a very small improvement from a better weapon, a very small improvement from a better sensor, but a large improvement from both, you will not be able to identify this interaction (or synergistic effect) if the experimental design does not involve factors for both the weapon and the sensor.

High Performance Computing

HPC consists of both hardware and software resources. HPC systems can be configured as a single supercomputer with thousands of processors, as a network of clustered computers, or even as a single powerful desktop computer with multi-core processors. The hardware on these systems includes such things as processors, memory, networking hardware, and disk storage. HPC software includes, among other things: the operating system; underlying or supporting software which provide the environment to execute the model; and the data farming software, which enables running instances of the model across the HPC systems' "compute units". By generating and managing each of the model runs over a set of design points or input sets, the data farming software provides the infrastructure "glue" that "sticks together" the model, its set of inputs, the design, and the HPC resources.

The main purpose of HPC in the context of data farming is to provide the means to execute a data farming experiment. Other purposes of HPC are for use in analysis and visualization of the output and for generating designs used in future data farming experiments. Given the large number of model runs made in a typical data farming experiment, HPC facilitates conducting the experiment in a timely manner as well as supporting the storage and analysis of huge volumes of output.

Analysis and Visualization

Analysis in the data farming context is the process of examining data that is produced by data farming processes using statistical, summarization and presentation techniques to highlight useful information, extract conclusions, and support decision-making. Visualisation is a collection of graphical and visual analysis techniques used to optimize and speed the process of exploring data, conveying understanding, and presenting in data farming processes. Much of the current usage of analysis and visualization in the data farming process has been the analytic examination of multiple replicate and excursion model output.

In order to exploit the potentially huge data output from the high performance computing execution of the design of experiments, highly effective analysis techniques must be employed. Statistical analysis and visualisation can be used to discern whether data may has useful meaningful value and aid in the translation of data into information that is useful in making progress in understanding possible answers to the questions at hand. The ability to use multiple techniques provides the ability to explore, investigate, and answer the questions posed. Every technique has strengths and limitations, therefore the use of a family of techniques is preferable to the use of a single technique, especially for high-dimensional data sets. As stated earlier, data farming gives us the ability to map the landscape of possibilities and in the process discover outliers. These outliers should always be considered and only be eliminated for appropriate reasons and can be investigated as a separate cohort of the data using various analysis and visualisation techniques.

Collaboration

The spirit of collaboration is the key tenet of data farming. It underlies the loop of loops in Figure 1 and holds within it much of the power of data farming. Throughout the development of data farming and the formation of the data farming community, people have openly shared experiences and expertise. One focus for collaborative efforts has been and continues to be the international workshops. The first international workshop took place in 1999 at the Maui High Performance Computing Center. The first four workshops were methodology driven, dealing with complex adaptive systems modeling and agent based representation, with statistical experiment design and experiment evaluation. The subsequent workshops were and continue to be application driven and contributions to the overall advancement of data farming takes place in the development of simulation models, scenarios within the models, and computer clusters to run the models audacious numbers of times.

The real work is in making progress on important questions and the real secret is the combination of military subject matter experts and highly knowledgeable and multi-disciplinary scientists. This special mix of personnel has been the hallmark of the international workshops and this mix has promoted much networking opportunity. It has been a dynamic combination to have data farming work teams headed up by a person who really knows and cares about the question (e.g. a military officer who knows that the answers may have an impact on both mission success and lowering casualties) and supported by men and women with technical prowess who can leverage the tools available.

MSG-088 documented the following aspects of the collaborative processes in data farming: defining the characteristics and dimensions of collaboration in data farming, collaboration within and between the realms in data farming, collaboration of the people, collaboration of the data farming results, and the application of collaboration tools. This information can be found in the full report as well as information on the current status of data farming in the attending nations and ideas about the future development of data farming (Horne et al., 2014).

CYBER QUESTIONS

The Cyber Defense Syndicate of MSG-124 has been using data farming techniques to explore solutions to improve NATO's resilience to cyber-attacks. The scenarios considered so far have spanned the threat spectrum, ranging from lone hackers to cyber espionage organizations. The team is leveraging a NetLogo model developed by the team, and is evolving the model as its behaviors and the needs of the stakeholders are better understood. Initial analyses have focused on exploring the value of various network topologies and organizations, firewall policies and intrusion detection systems.

The overall goal of the team is to leverage the current research, develop a suitable simulation, and explore possible scenarios through data farming that could facilitate the understanding of aspects of cyber defense important to NATO. The group has begun to 1) define questions within the cyber defense area in conjunction with cyber defense experts of NATO and the participating nations, 2) provide modeling and simulation support for various cyber defense questions, and 3) perform analysis and iterative exploration of “What-if?” questions to reveal the landscape of possibilities inherent in the scenarios and enable the study of any “outliers” that are discovered.

Data farming techniques have proven to be useful thus far in MSG-124 efforts as evidenced by the accomplishment of the three tasks above as well as making progress toward the overall goal.

NETLOGO MODEL

The approach to answering these questions started by developing prototype scenarios in an extensible agent-based model and using them to conduct stochastic simulations. It was considered paramount that the model should be easy to distribute and share with the nations, allowing the data and processes modeled to remain unclassified and the framework to be freely distributable. NetLogo was selected, because it is a free agent-based modeling framework with a wide community of users and an ever-growing list of extensions and features (Wilensky, 1999). The selection of the elements to be included in the initial scenarios was informed by NATO sources and the concepts and ideas communicated by the subject matter experts from the nations. Table 1 presents the list of major model parameters.

Table 1. Major model parameters

Name	Unit	Description
number-of-servers	int	The number of servers in the network.
number-of-servers-in-dmz	int	The number of servers that are in the demilitarized zone. This value can range between 0 and the number-of-servers.
number-of-subnets	int	The number of subnets in each of the networks.
number-of-clients-per-subnet	int	The average number of clients in each of the sub networks.
server-vulnerabilities	int	Number of vulnerabilities that exist for servers.
server-percent-vulnerabilities	%	Fraction of vulnerabilities present in any one server.
router-vulnerabilities	int	Number of vulnerabilities that exist for the routers/switches.
router-percent-vulnerabilities	%	Fraction of vulnerabilities present in any one router/switch.
pc-vulnerabilities	int	Number of vulnerabilities that exist for the clients in the subnets.
pc-percent-vulnerabilities	%	Fraction of vulnerabilities present in the clients inside a subnet.
mean-time-to-update	days	Average time it takes for the network to issue an update.
mean-vulnerabilities-removed	int	Average number of vulnerabilities removed by any update.
mean-vulnerabilities-added	int	Average number of vulnerabilities added by any update.
shut-down-threshold	%	Percentage of sensors that have to issue an alarm before the system administrator will shut it down.
shut-subnet-threshold	%	Percentage of sensors that have to issue an alarm before the system administrator will shut down the affected subnets.
sensor-p-detect	%	Sensor probability of detecting an attack.
susceptibility-to-phishing	%	Probability that any use of the subnet is susceptible to a phishing attack.
mean-time-to-restart	hrs	Average time it takes for the system administrators to restart the elements of the network after the shut down.
number-of-attackers	int	The total number of attackers (hackers)
mean-attack-time	hrs	Average time it takes a hacker to perform an attack. This is the nominal time that is extended/contracted by the different tasks the hacker performs.
min-competency	%	Minimum competency that the hackers possess. This is a non-dimensional factor between 0 and max-competency.
max-competency	%	Maximum competency that the hackers possess. This is a non-dimensional factor between min-competency and 1.
hacker-learning-time	days	Average time it takes for the hackers to learn new vulnerabilities. This is the mean value from an exponential distribution.
mean-vulnerabilities-known	int	Average number of vulnerabilities the hackers may know at time zero.
mean-vulnerabilities-learned	int	Average number of vulnerabilities the attackers learn each time they elapse their randomly generated time to learn.

The simulation and scenarios are continuing to evolve to provide capability to answer cyber questions important to NATO, but not to model every possible cyber threat. Currently the efforts do not explicitly address natural or inadvertent user errors, but are focusing on intentional attacks, in particular, penetration attacks. The simulation is composed of three primary elements: the network, the system administrator and the attackers. The subsections below will describe the three elements in more detail.

The Network

The network is composed of three primary elements: routers/switches, servers, and subnets with terminals. All the networks have systems in the demilitarized zone (DMZ), with at least one router that serves as the portal to the wide area network, and a user-specified number of servers. The final element is the sensors to detect the cyber-attacks. The sensor model is general and does not differentiate between the different types of cyber-attack sensors, e.g., NetFlow, honeypots, and Samhain. The sensors are associated with the other elements of the network and can detect attacks based on their probability of detection.

The routers/switches element connects the elements of the network, and even though firewalls are not modeled explicitly, when attackers attempt to penetrate the network, they must be able to exploit a vulnerability before they can compromise other parts of the network.

The network generation algorithm currently creates a bus network, but it can be extended to create tree, star, ring, fully connected, mesh or line networks. Bus networks were deemed to be the most representative for the applications of interest by the groups of subject-matter experts consulted. The network is currently generated by creating the world facing router first, and then sequentially adding routers for each subnet specified in series and finally the servers are randomly associated with the routers except for the number of routers that are placed in the DMZ.

The System Administrator

The system administrator is currently modeled using a simple algorithm using the shutdown thresholds specified and the alarms communicated by the sensors. The system administrator monitors the sensor alarms and either shuts down affected subnets, or the entire network depending on the number of alarms and the threshold parameters (i.e., shut-down-threshold and shut-subnet-threshold).

The following example illustrates the activity of the system administrator (SA). The SA monitors a network of 4 subnets with a total of 5 sensors, with a shut-subnet-down threshold of 18% and a shut-down-threshold of 28%. As an attack is detected, the sensors will trigger an alarm. If one sensor issues an alarm, that represents 20% of the sensors and will force the SA to shut down the affected subnets. If two sensors issue an alarm, that will trigger a total network shutdown as that represents 40% of the network. The logic of the SA is simple, but provides a first iteration for the logic that a reactive administrator may follow. It could clearly be improved if the sensors were specialized and the risk of the different activities that the specialized sensors could detect was defined. This system would produce more accurate reactions. Ideally, sensor fusion algorithms could be evaluated, potentially defining requirements for data fusion algorithms, such as accuracy, false positive and negative rates, etc.

The Attackers

There are multitudes of ways that the actions of cyber attackers can be modeled. The key concept is to do so in the simplest manner possible while still capturing the primary behaviors and traits. A model of attacker tasks provides a series of tasks that hackers follow and all their potential sequences (de Souza et al., 2006). Figure 2 reproduces the task model where the blocks represent the tasks hackers perform and the arrows the transitions. For the model, each hacker follows a different strategy by having different probabilities for transitioning between states. A multitude of cyber-attack models were reviewed, including the Hacker Attack Representation Model (Karpati et al., 2013), generic attack graphs (e.g. Eom et al., 2008), agent-based models (e.g. Kotenko, 2005), and other procedural models (e.g. Tidwell, 2001). The main drawback of these approaches for this particular application is the level of detail and complexity required to represent cyber-attacks. The de Souza model provides a simple framework on which more complex representations for cyber attackers' activities can be modeled. This framework aligns with the underlying methodology of modeling by which we start with the simplest model possible and add complexity as needed.

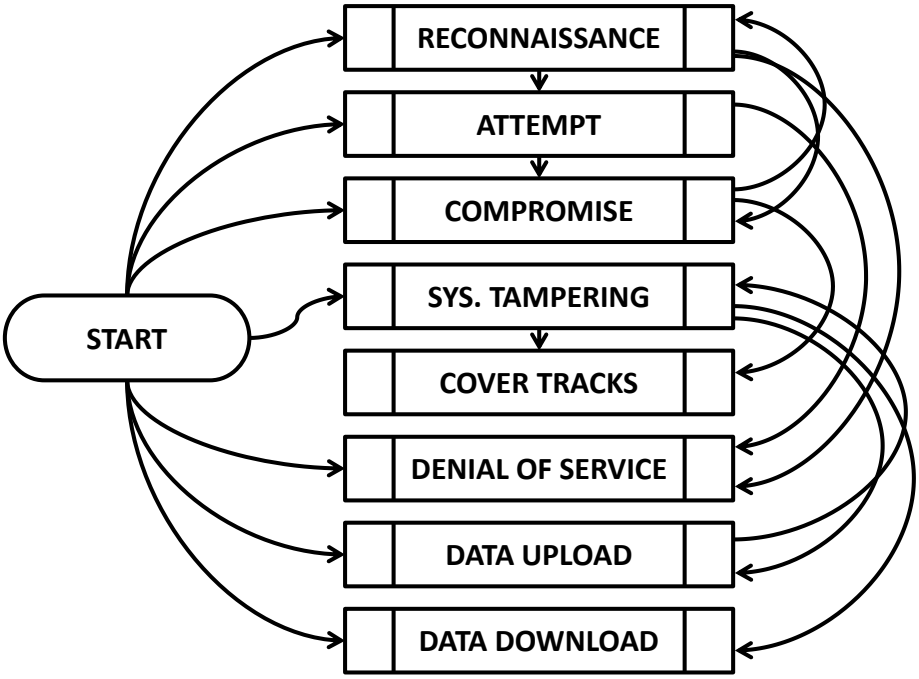

Figure 2. Attacker Task Model (Based on de Souza *et al.*, 2006)

The NetLogo model we are using continues to be developed and the current model is shown in Figure 3. This model has been used in the data farming process and has shown promise in initial efforts.

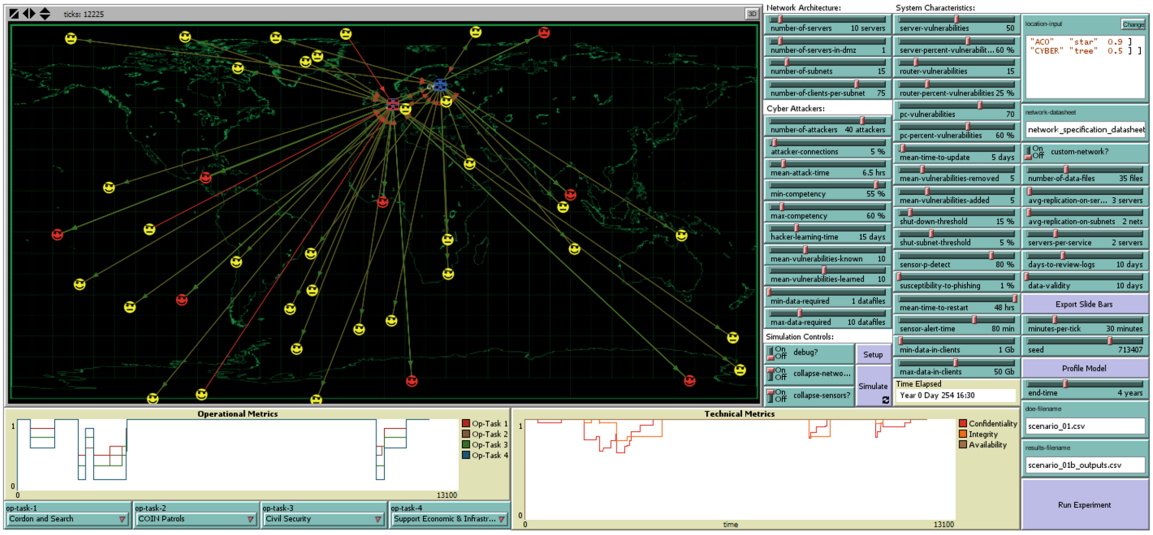

Figure 3. Screen Capture of the NetLogo model.

METRICS

The model produces two sets of metrics. The first is the C-I-A set of metrics, which are the traditional cyber defense metrics, i.e., Confidentiality, Integrity and Availability. The second set is a set of operational metrics derived from the protection and availability of operationally relevant services.

Confidentiality – Integrity – Availability

The information security C-I-A, or CIA (Confidentiality, Integrity and Availability), paradigm serves as a well-established and comprehensive reference from which to derive the set of measures (Canal, 2005). The CIA paradigm is based on the three concepts described below.

- Confidentiality (C) is the ability to grant access to authorized users and deny access to unauthorized users.
- Integrity (I) is the ability to guarantee that some information or message hasn't been manipulated.
- Availability (A) is the ability to access information or use services at any moment we demand it, with appropriate performance.

Confidentiality and integrity are concerned with protecting the information, while the availability ensures that the systems provide the necessary services for the users. These two key concepts form the basis for the metrics of the model. These two concepts must be traded off, as maximizing confidentiality and integrity negatively impacts availability. To illustrate this tradeoff, one can think of an extremely secure system where the information is so protected, locked down to such a degree, that its availability is extremely limited, as in only a few users can physically access it, and only after extensive efforts and commitment of time.

The CIA paradigm is nonetheless highly abstract and for the purposes of this paper just serves as a means to classify the types of goals for maintaining a secure network. The authors recognize that the focus on effects rather than causes makes the CIA paradigm not scientific for purposes of analyzing attacks, but this focus is not the purpose of using this framework. The paradigm can also be used to categorize the elements of a network, namely the hardware, software and communications systems.

Operational

Operational metrics have been part of the work of the MSG-124 task group (Horne and Meyer, October 2014) and in this paper we provide an overview. The operational domain is characterized by a series of operational tasks that can be mapped to types of operations. These tasks consist of activities like "Movement to Contact", "Area Defense", "Cordon and Search", etc. The team used various military doctrinal references and the experience of the uniformed members to identify 20 operational tasks that were then mapped to 4 types of operations, namely: "Offense", "Defense", "Stability", and "Irregular Warfare". The benefit of having a mapping to fewer and higher level operational concepts facilitates the comparisons of alternatives later on.

With a conceptual framework for mapping operational concepts to each other, and their impact if one cannot be achieved, the next task consisted of mapping the impact of having various networked services (those things that would be denied or compromised by cyber attackers) to the operational tasks. This mapping was captured with two matrices, one for the impact of having the service denied, and the other having the service compromised (with the implication that the enemy could not only intercept, but also modify the information to confuse the friendly forces).

With a mapping to the services, all that remains is to add the concept of services to the cyber defense agent-based model. The idea of a service is represented as an agent that is connected to one or more servers (that host the service) and data files (that represent the data in the service). As attackers penetrate a network, they eventually reach a server or data file that is associated with a service. Depending on the action the attacker is performing, the data associated with these services can be considered to be compromised (in which case the service is considered compromised) or the server is taken down and the service is denied. This result can then be propagated up the operational concepts to any level of abstraction that has been included in the mappings described previously.

RECENT ANALYSIS EFFORTS OF MSG-124

As mentioned earlier, MSG-124 is the NATO task group called Developing Actionable Data Farming Decision Support, and is performing work in selected application areas important to NATO, one of which is cyber defense. The cyber team of MSG-124 decided to focus on one question in their recent analysis efforts: What factors are the most crucial to each of the metrics. These metrics include four operational metrics and the CIA metrics of Confidentiality, Integrity, and Availability.

Due to the limited amount of time to run the model, the team decided to use a screening design to assess the impact 14 factors. The factors and their units and ranges are shown in Table 2 below. The experimental design consisted of 29 cases. Each case was repeated 150 times, for a total of 4,350 cases. The metrics assessed were: (1) Cordon and Search, (2) Counter-Insurgency (COIN) Patrols, (3) Civil Security, and (4) Support and Economic Infrastructure. These were deemed different enough to capture the diversity in the types of operations analyzed and ensure the results would be more universally applicable.

Table 2. Robust Screening Design (14 factors, 29 cases)

Case #	mean-time-to-update	sensor-p-detect	susceptibility-to-phishing	hacker-learning-time	min-competency	max-competency	mean-vulnerabilities-known	mean-vulnerabilities-learned	number-of-attackers	shut-down-threshold	shut-subnet-threshold	pc-percent-vulnerabilities	pc-vulnerabilities	avg-replication-on-servers
	days	%	%	days	%	%	int	int	int	%	%	%	int	int
1	45	85	5	45	10	50	20	1	22	4	4	30	10	9
2	45	15	2.55	10	45	50	2	5	40	15	4	30	10	9
3	45	85	0.1	27.5	10	95	2	1	40	15	4	80	10	1
4	45	85	5	10	10	95	2	3	4	15	1	30	80	9
5	45	15	0.1	45	10	72.5	2	5	4	4	4	80	80	9
6	5	85	0.1	45	27.5	95	2	5	40	4	1	30	10	9
7	5	15	5	27.5	45	50	20	5	4	4	1	30	80	9
8	5	85	0.1	45	45	50	2	1	4	15	4	30	80	5
9	5	15	5	45	10	95	11	5	4	15	4	30	10	1
10	45	85	5	45	45	50	2	5	4	9.5	1	80	10	1
11	45	15	5	10	27.5	50	20	1	4	15	4	80	80	1
12	5	85	5	10	10	50	2	5	40	4	4	55	80	1
13	5	15	0.1	45	45	50	20	3	40	4	4	80	10	1
14	5	15	5	45	10	50	2	1	40	15	1	80	45	9
15	45	50	0.1	45	10	50	20	5	40	15	1	30	80	1
16	45	85	0.1	10	45	50	11	1	40	4	1	80	80	9
17	45	15	0.1	45	45	95	20	1	4	15	1	55	10	9
18	5	15	0.1	10	45	95	2	5	22	15	1	80	80	1
19	45	15	5	10	10	95	20	5	40	4	1	80	10	5
20	25	85	5	45	45	95	20	5	40	15	4	80	80	9
21	45	15	5	45	45	95	2	1	40	4	2.5	30	80	1
22	5	15	0.1	10	10	95	20	1	40	9.5	4	30	80	9
23	5	85	2.55	45	10	95	20	1	4	4	1	80	80	1
24	25	50	2.55	27.5	27.5	72.5	11	3	22	9.5	2.5	55	45	5
25	5	50	5	10	45	95	2	1	4	4	4	80	10	9
26	5	85	0.1	10	10	50	20	5	4	15	2.5	80	10	9
27	5	85	5	10	45	72.5	20	1	40	15	1	30	10	1
28	45	85	0.1	10	45	95	20	5	4	4	4	30	45	1
29	25	15	0.1	10	10	50	2	1	4	4	1	30	10	1

RESULTS

The results from the experimental design were analyzed using SAS's JMP statistical analysis tool. The authors were able to execute sufficient runs to estimate the mean value and standard deviation for each of the 7 metrics. This work enabled the analysis of the impact of each factor and the interactions on the estimated value of each of the metrics and the assessment of impact in affecting variability. Table 3 depicts the p-values of the most significant parameters for each of the four types of operations analyzed. For each of the four operations, the impact of each factor was assessed in terms of operational metrics' average value (i.e., mean) and variability (i.e., standard deviation).

Table 3. Individual p-Values of most significant factors for each of the four types operations analyzed.

Individual p-values for most signficant factors and interactions	Cordon and Search		COIN Patrols		Civil Security Patrols		Sup. Econ. & Infra. Ops	
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
Number of Cyber Attackers	0.0027	0.0018	0.0021	0.0010	0.0030	0.0025	0.0025	0.0030
Sensor Probability of Detect	0.1762	0.0766	0.1751	0.0693	0.1924	0.0834	0.1883	0.1018
Avg Replication on Servers	0.1441	0.1270	0.1585	0.1194	0.1737	0.1334	0.1521	0.1530
Mean Vulnerabilities Learned	0.1645	0.1490	0.1657	0.1492	0.1842	0.1588	0.1711	0.1506
Shutdown Threshold	0.2459	0.1133	0.2418	0.1132	0.2494	0.1107	0.2473	0.1264
Number of Cyber Attackers * Mean Vulnerabilities Learned	0.1184	0.3992	0.1231	0.4077	0.1381	0.3907	0.1293	0.1840
Number of Cyber Attackers * Sensor Probability of Detect	0.2360	0.0720	0.2373	0.0662	0.2535	0.0822	0.2519	0.0975

For Cordon and Search, the number of attackers is the most important factor affecting the expected value of performing this task, and the only one that is statistically significant. The relationship is intuitive, in that increasing the number of cyber attackers decreases the ability to perform the mission. Nonetheless, the variability of this metric is also affected by the sensor's probability of detecting an attack (sensor-p-detect) and its interaction with the number of attackers. The results indicate that increasing the sensor-p-detect increases the variability in the metric, as does increasing the number of attackers. If both increase the variability is further increased due to interactions between the two factors. If one increases but the other decreases, the interaction factor reduces the variability, if both decrease, the interaction factor has a diminishing returns effect on the reduction in the variance. Counter Insurgency (COIN) Patrols, are affected almost identically to Cordon and Search. This outcome is to be expected, as the two types of operations are similar in the types of services they require and the threats they face. Civil Security and Support Economic and Infrastructure Operations both displayed very similar results. This outcome is an indication that the ranges chosen were not comparable, as the number of attackers dominates the behavior of the model.

If this model were verified, validated and accredited, it would be important to re-assess the ranges for the factors. If the ranges were deemed to be correct, two primary options would be available to the analysts: (1) spend considerable effort attempting to quantify the expected number of attackers to narrow its variability, (2) create a number of scenarios, e.g., select a worse-case or conservative number and a nominal number of attackers and repeat the analysis to identify the factors that are most critical in each case.

Some factors were marginally not significant and deserve to be mentioned. These are listed below with a short explanation and potential implication.

- Average replication of files on servers: This factor appears to be almost significant in all cases, and it is inversely correlated with the metrics, indicating that if a file is replicated more on different servers, the ability to perform the mission is jeopardized. This result is an indication that confidentiality and data integrity are more important than availability.
- Mean vulnerabilities learned vs. mean vulnerabilities known: Due to the long term over which the simulation is executed, the initial number of vulnerabilities the attackers know is not as critical as to how many vulnerabilities they learn. As expected, the more vulnerabilities they learn during each period, the lower the ability to perform the mission due to the cyber-attacks.
- Sensor Detection Probability: This factor is non-intuitively correlated with the expected value for the ability to perform the operations. The statistical analyses indicate that higher probabilities of detecting an attack reduce the ability to perform the mission. This finding is an indication that availability is decreasing, but it merits further study.
- Shutdown Threshold is one factor that seems to provide improvement in both increasing the expected value and reducing the variability. Increasing the factor implies neglecting more alarms and only shutting down the portions of the network when a sufficient number of sensors are activated. This finding again implies that the model is operating in a scenario where the availability of the services is driving the ability of the operational forces to conduct their respective missions.

The operational metrics can be contrasted with the more technical CIA metrics. It is important to note that the CIA metrics are not a well established and agreed upon concept, but the authors used the concepts described in the CIA paradigm to develop metrics that are aligned with the concerns of each of the elements of the CIA paradigm. Table 4 depicts the p-values for the most significant factors on the CIA metrics.

Table 4. Individual p-Values of most significant factors for each of the CIA metrics.

Individual p-values for most signficant factors and interactions	Confidentiality		Integrity		Availability	
	Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
Number of Cyber Attackers	0.0067	0.0019	0.0100	1.0000	0.0277	0.0049
Sensor Probability of Detect	0.9582	0.3071	0.9681	0.8305	0.0293	0.0037
Mean Time to Update	0.1917	0.1373	0.1224	0.0473	0.4663	0.4005
Avg Replication on Servers	0.0315	0.0019	0.0329	0.3847	0.8985	0.4332
Min Attacker Competency	0.2513	0.2404	0.1619	0.0482	0.5002	0.3482
Max Attacker Competency	0.3008	0.2065	0.2013	0.0496	0.1927	0.0409
Mean Vulnerabilities Known	0.3599	0.3040	0.2211	0.0532	0.2163	0.1694
Susceptibility to Phishing	0.6905	0.9233	0.8199	0.0999	0.7898	0.9497
Shutdown Threshold	0.4612	0.9081	0.4595	0.9981	0.0104	0.7886
Shutdown Threshold * Number of Cyber Attackers	1.0000	1.0000	1.0000	1.0000	0.0226	0.0083
Shutdown Threshold * Sensor Probability of Detect	1.0000	1.0000	1.0000	1.0000	0.0322	0.0028
Number of Cyber Attackers * Avg Replication on Servers	0.0946	0.0322	0.1033	1.0000	1.0000	1.0000
Mean Time to Update * Mean Time to Update	0.0832	0.1087	0.0907	0.9753	1.0000	1.0000
Mean Time to Update * Min Attacker Competency	0.8468	1.0000	0.8931	0.0700	1.0000	1.0000
Mean Time to Update * Max Attacker Competency	1.0000	1.0000	1.0000	0.1104	1.0000	1.0000
Mean Time to Update * Mean Vulnerabilities Known	1.0000	1.0000	1.0000	0.0694	1.0000	1.0000
Min Attacker Competency * Max Attacker Competency	1.0000	1.0000	1.0000	0.1157	1.0000	1.0000
Number of Cyber Attackers * Sensor Probability of Detect	1.0000	1.0000	1.0000	1.0000	0.0674	0.0162

The first metric to be considered is Confidentiality. Confidentiality, as described earlier, is concerned with maintaining the secrecy of the information. As shown in Table 4, the ability to maintain confidentiality is mostly impacted by the number of attackers and the amount of replication of the files. These results are logical and agree with that is expected. The insightful result is how much they impact confidentiality. In addition to these two factors and their linear interaction, which is reinforcing, the mean time to update the systems is also important in

maintaining confidentiality. The effect has a quadratic form, which is indicative that there is a minimum amount of confidentiality for a nominal frequency of update, but updating more or less produces higher levels of confidentiality. This result can be explained by the number of vulnerabilities added during the updating of systems, which was not varied in this study. The variance of the confidentiality metric is dominated by the number of attackers and the average replication of the data files in the servers. The time to update the system is marginally under the statistical significance threshold. While the number of attackers and the average replication increase the variability, the longer times to update decreases the variability of the metric.

While Confidentiality is interested in maintaining the secrecy of the data, Integrity is concerned with maintaining accuracy or veracity, i.e., avoid having attackers manipulate the data. Table 4 shows that the expected value of Integrity is driven by the same factors as Confidentiality. The variance in the value of Integrity on the other hand is impacted by other factors. In particular, the competency of the attackers increases the variability. This is an indication of that the model for the penetration attacks that exploits manipulation of data is more heavily impacted by how competent the attackers are.

The number of vulnerabilities known by the hackers at the beginning of the simulation also increases the variability. The more vulnerabilities the attackers know about at the beginning of the simulation, the faster they may penetrate the network. As they penetrate the network, it may be easier for them to manipulate the configuration data files and obtain deeper footholds in the network. Lastly, increasing the susceptibility to phishing decreases the variability in the integrity. This finding is interesting, because this factor, as the other factors mentioned in this paragraph and the prior, is not shown to be statistically significant to the expected value of Integrity. What can explain this phenomenon is that as more users fall for phishing attacks, the attackers gain deeper access to the network more easily, making it easier for them to hop from one part of the network that contains Integrity information to the other. The model is apparently indicating that attackers will penetrate the network modeled regardless of the propensity of users to fall for phishing attacks, but the lower the phishing, the more variability in the impact they will have on the Integrity of the data in the network.

The last of the CIA metrics is Availability, and the one that must traditionally be traded off against the first two. Table 4 shows that the shutdown threshold, the number of attackers, and the probability of a sensor detecting an attack are the primary parameters that drive the availability of the network. The higher the threshold, i.e., the less sensitive to alarms, the higher the availability. Conversely, the more attackers, or the more sensitive the sensors, the lower the availability. There are interesting interactions between these factors, e.g., if both number of attackers and the shut-down threshold increase, the availability goes up, as it does if shut-down threshold and the sensors' sensitivity increases. Conversely if one increases and the other decreases the availability is reduced. If both decrease, the availability increases. These interactions indicate that there are effects from these factors that put the model in different states and are critical when trying to find good balances between confidentiality, integrity and availability. Furthermore, these factors are doctrinal, materiel and noise parameters, once again, highlighting the complexity of the cyber problem and the need to assess them jointly.

The variance in availability is also impacted by the maximum competency of the attackers as shown in Table 4. The variance in availability decreases when the maximum competency of the attackers increases. This finding can be attributed to the fact that more competent attackers may penetrate the network more easily and spend less time trying to enter, which reduces the likelihood that they will be detected by one of the sensors. The variance in availability is reduced when the shut-down threshold is increased, but it increases with higher number of attackers and more sensitive sensors. The interaction factors are also statistically significant in explaining the variability in the availability of the systems.

FUTURE WORK

The work of the Cyber Team within MSG-124 is continuing, with the final report to be finished in 2017. Part of the plan for future work is to continue exploring the non-intuitive results and use the data farming methodology with high-performance computing resources to execute more explorations with smaller ranges for the number of attackers. The plan for MSG-124 and beyond also includes initiating the examination of additional cyber questions using data farming.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the efforts of our fellow members of the Cyber Defense Syndicate of NATO Modeling and Simulation Group 124.

REFERENCES

[1] Canal, V.A., "On Information Security Paradigms," ISSA Journal, September 2005

[2] de Souza, I.G., Berk, V.H., Giani, A., Bakos, G., Bates, M., Cybenko, G., and D. Madory, "Detection of Complex Cyber Attacks," Proc. SPIE 6201, Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security and Homeland Defence V, 620106, May 10, 2006.

[3] Eom, J.H., Han, Y.J., Park, S.H., and T.M. Chung, "Active Cyber Attack Model for Network System's Vulnerability Assessment," in proceedings of the International Conference on Information Science and Security, pp. 153-158, 2008.

[4] Hoffman F. G., and Horne, G. E., *Maneuver Warfare Science 1998*. Quantico, Virginia: Marine Corps Combat Development Command, 1998.

[5] Horne, G., "Data Farming: A Meta-Technique for Research in the 21st Century," briefing presented at the Naval War College. Newport, RI, November 1997.

[6] Horne, G. and Meyer, T., *Scythe, the Proceedings and Bulletin of the International Data Farming Community.* Issue 16 – Workshop 28, Jefferson, Maryland, USA, October 2014

[7] Horne, G. and Meyer, T., *Scythe, the Proceedings and Bulletin of the International Data Farming Community.* Issue 17 – Workshop 29, Riihimaki, Finland, March 2015

[8] Horne, G. and Meyer, T., "Data Farming: Discovering Surprise," Proceedings of the 2004 Winter Simulation Conference, eds. R. Ingalls, M. D. Rossetti, J. S. Smith, and B. A. Peters, 171-180. Washington, DC, December 2004.

[9] Horne, G. and Meyer, T., "Data Farming and Defense Applications," MODSIM World Conference and Expo, Hampton Roads Convention Center, Hampton, VA, USA 13-15 October 2010.

[10] Horne, G. et al., *MSG-088 Data Farming in Support of NATO, Final Report*, NATO Science and Technology Office (STO), Paris, France, 2014.

[11] Karpati, P., Opdahl, A. L., and G. Sindre, "HARM: Hacker Attack Representation Method," Software and Data Technologies, vol. 170, pp. 156-175, 2013.

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

[13] Tidwell, T., Larson, R., Fitch, K., and J. Hale, "Modeling Internet Attacks," in proceedings of the 2001 IEEE Workshop on Information Assurance and security, vol. 59, 2001.

[14] Wilensky, U., "NetLogo," Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, 1999. http://ccl.northwestern.edu/netlogo/.

[15] www.datafarming.org, accessed January 2016.