Cyber Attack Forecasting System (CAFS)

C Thomas Savell
GCAS, Inc.
San Marcos, CA
ctsavell@gcas.net

Ambrose Kam
Lockheed Martin Corp
Moorestown, NJ
ambrose.kam@lmco.com

Nataliya Shevchenko, Brett Tucker
Carnegie Mellon University
Pittsburgh PA
nataliya@andrew.cmu.edu
brettt@andrew.cmu.edu

ABSTRACT

Under USAF contract, researchers at Carnegie Mellon University, GCAS, and Lockheed Martin have partnered to research how to model and forecast a cyberthreat's maneuvers in a compromised network. Historically, the United States Department of Defense (US DoD) has studied and developed methods and models for attack warning and attack assessment of ballistic missile defense. These methods included probabilistic multi-model filters and multi-hypotheses tracking. This proven technology will now be leveraged to track and forecast cyberthreat attack vectors to effectively defend organizational high value assets and neutralize those threats. This is different from today's technology in that it adds the ability to predict the potential next move in the attack vector using tools like Multi Hypothesis Method (MHM) within a Bayesian framework. To that end, the 4-month Phase-I research effort delivered a simplified simulation of a cyberattack of a single intruder with a limited number of maneuver tactics.

ABOUT THE AUTHORS

C Thomas Savell is the founder and current CEO of GCAS, with over 50-years of experience in the DoD industry. He is designer of GCAS' suite of AI software products used by MDA, US Navy, US Army, Oshkosh Corporation and General Motors, and the ***GCAS***™ accounting software suite used by numerous DoD contractors. Prior to founding GCAS, he was the Technical Director for SDRC, Head of Advanced Products for Solar Turbines International, and Department Head for General Electric AEG's Dynamic Analysis and Testing Group, where he received GE's Young Engineer of the Year award. He is a Guggenheim, Ford Foundation & NASA Fellow with a PhD from the University of Cincinnati, and a BS and MS from Ga Tech.

Ambrose Kam is Lockheed Martin Fellow and Cyber Threat Modeling Expert, with over 25-years of experience in the Department of Defense (DoD) industry. He is one of the earliest pioneers at applying modeling, simulation and operations analysis techniques to threat modeling and cyber resiliency assessment. He is a lecturer at MIT, Georgia Tech, and industry consortium (e.g., MORS and NDIA). His most recent efforts are in wargaming, machine learning/deep learning, Cyber Digital Twin and blockchain which earned him patents and trade secret awards. In 2017, Ambrose won the prestigious Asian American Engineer of the Year (AAEOY) award for his technical leadership and community services. He holds several advanced degrees from MIT and Cornell University.

Nataliya Shevchenko is a Senior Member of Technical Staff at CMU. Her expertise includes in Model-Based System Engineering (MBSE), Cyber threat modeling methods, software development lifecycle & processes, software architecture principles & practices, cyber/internet security, with key contributions, SysML and Threat Modeling. She has authored white papers on Threat Modeling Methods that was added to *Defense Technical Information Center* (dtic.mil) and published on *SEI Insights*, and collaborates with Security Engineering Risk Analysis (SERA) team on integration of SERA and threat modeling methods. Nataliya has a MS (Mathematics) from Ukraine's Donetsk State University and an MS (Software Engineering) from CMU..

Brett Tucker the Technical Manager of Cybersecurity Risk at Carnegie Mellon University's SEI. He has 19 years of experience in the public and private sectors. Prior to joining the SEI, He was the Global Risk Manager for Westinghouse where he managed the corporate enterprise risk portfolio and global insurance programs. He also served as an Intelligence Officer at the CIA and is a veteran of the US Navy Nuclear Propulsion Program. Tucker holds a degree in chemical engineering from the University of Notre Dame, a master's degree in engineering management from Old Dominion University and an MBA from Penn State University.

Cyber Attack Forecasting System (CAFS)

C Thomas Savell
GCAS, Inc.
San Marcos, CA
ctsavell@gcas.net

Ambrose Kam
Lockheed Martin Corp
Moorestown, NJ
ambrose.kam@lmco.com

Nataliya Shevchenko, Brett Tucker
Carnegie Mellon University
Pittsburgh PA
nataliya@andrew.cmu.edu
brettt@andrew.cmu.edu

FORECASTING THE MOVEMENT OF CYBERTHREATS

Information derived from data is the most important asset in any organization. Enhancing cybersecurity and protecting critical information infrastructures are essential to the US national security and economic wellbeing. Cyberthreats are becoming more technically sophisticated and pervasive such that, despite significant advances in cybersecurity, IT/OT professionals charged with protecting digital assets continue to perform at a significant disadvantage from a defensive posture. To tighten the security, various sensors and controls are implemented in the network for monitoring network traffic and limiting access to assets, but these are limited as to their functionality and performance. Hence, there is a need to model the behaviors of the attackers and understand how threat actors would attack their targeted system. In cyber, the phrase "lateral movement" typically refers to the techniques that a cyber threat actor uses to traverse their attack deeper into a network as suggested schematically in Figure 1. Note that our modeling algorithms are for predicting the movement of a cyberthreat who has already compromised and gained access to the network.

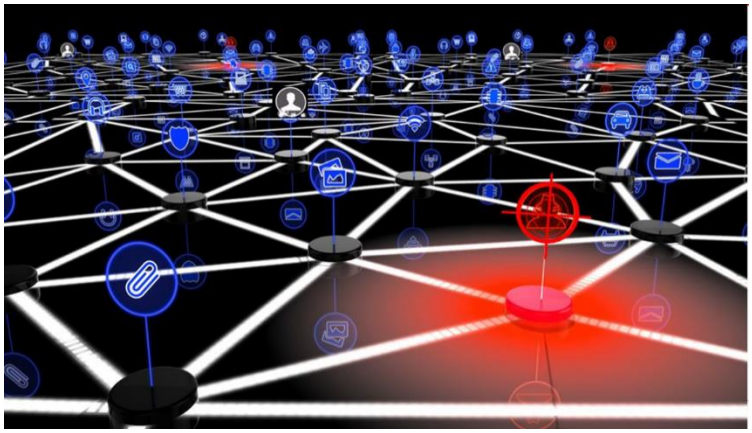

Figure 1: Predicting the Future Vector of a Cyberthreat in a Compromised Network

Once a foothold is gained within the intended network, the attacker maintains access by moving through the environment and obtaining increased privileges using various tools. Lateral movement is one of the necessary threat actor tactics that distinguishes today's "Advanced Persistent Threat (APT)" from simplistic cyberattacks of the past. After gaining initial access to an endpoint, such as through an email phishing attack or malware attack, the cyber attacker pursues escalation of privileges, which enables true control of the system and its assets. For example, an APT might impersonate a legitimate user and move through multiple systems in the network until the end goal is reached. Attaining that objective involves gathering information about multiple systems and accounts, obtaining credentials, escalating privileges, and ultimately gaining access to the identified payload.

The Cyber Kill Chain® (Yadav & Rao, 2015) (Lee, et a, 2016), (LMCO1, n.d.), (LMCO2, n.d.) shown in Figure 2 is a framework developed by Lockheed Martin to illustrate how a given cyber incident would evolve. There are seven phases within Cyber Kill Chain. The initial phase is the Reconnaissance Phase where basic information about the targeted network is collected. During reconnaissance, the attacker observes, explores, and maps the network, its users, and devices. This mapping allows the attacker to understand host or IP address naming conventions and network hierarchies, identify operating systems, locate potential payloads, and acquire intelligence to make informed moves. Threat actors typically deploy a variety of tools to find out where they are located in the network, what they can get access to and what firewalls or other deterrents are in place. An attacker can leverage many external custom and open-source tools for port scanning, proxy connections and other techniques, but employing built-in OS or support tools offer the advantage of being harder to detect. To move through a network, an attacker needs valid login credentials. The term used for illegally obtaining

credentials is called “credential dumping” (RedCanary, 2022). Login credentials could be stolen from users by tricking them into sharing them by social engineering tactics, keylogging, and phishing attacks.

The process of performing internal reconnaissance and then bypassing security controls to compromise successive hosts can be repeated until the targeted node has been exploited. As cyberattacks become more sophisticated, they often contain a strong human element. This is particularly true for lateral movement when an organization might be faced with moves and countermoves from an adversary. But threat actor behavior can be potentially detected and identified by a robust security solution. This Analytic Monitoring type of solution is highlighted as one of many cyber resiliency techniques in NIST 800-160 Volume 2 (NIST, 2019). Effective monitoring might require operator training, anomaly analysis, threat modeling, etc. This is where multi-hypothesis Bayesian cyberattack modeling could be thrusted into good use.

Figure 2: Cyber Kill Chain® (LMCO2, n.d.)

In addition to modeling the threat actor behavior, it would be prudent to forecast their subsequent steps in the attack, so that appropriate defense strategies could be employed to impede the movement or to stop the attack. To do that successfully, cyber defense must react appropriately given a certain time frame. This is what robust active defense is about (and why attack forecasting is so critical). CrowdStrike routinely publishes cyberattack metrics based on real world incidents. One of the metrics that CrowdStrike attempts to measure is called “break out time.” Breakout time is defined as the time it takes for an attack to begin moving laterally into other systems in the network after initially compromising a machine. In 2022, CrowdStrike reported the average breakout time to be 1 hour and 58 minutes (Crowdstrike, 2022). Therefore, an organization has less than roughly two hours to detect, investigate and remediate or contain the threat. Obviously, this is a very stressful timeline to the Cyber Protection Team (CPT), especially when threat actors can launch their attacks from anywhere around the world and at any time. The longer it takes for a CPT to identify and respond, the higher the risk of a data breach. This is the reason why the cyber industry is pushing adherence to the 1-10-60 rule (Crowdstrike, n.d.). This means detecting an intrusion within 1 minute, investigating within 10 minutes and isolating or remediating the problem within 60 minutes. The longer an adversary is allowed to engage in lateral movement over a protracted dwell time, the more likely an attack will eventually succeed and with greater damage.

SUMMARY OF TECHNICAL APPROACH

This technical paper describes the development of statistical methods for predicting the future movement of a cyberthreat who has breached an IT or OT network. Specifically, the approach uses Bayesian, Utility, Prospect, Game

Theory, and Multi-Hypothesis Methods (MHMs). These methods are employed to determine the most likely prior movement threat vector and based on this history, to provide a prioritized list of most likely future maneuvers by the threat agent. For this body of research, two competing probabilistic architectures have been developed for predicting the cyberthreat's future movement in the network:

1. A pure Bayesian approach using Bayesian Networks (BNs) (Pearl, 1997), (Pearl, 2000), and
2. A Decision Network (DN) (Jensen, 2001) which extended the BNs to include decision and utility theory.

The Probabilistic Relation Models (PRMs) object-oriented architecture was used to modularize the design. Separate modules were constructed to represent the various threat actor types, and the various network node types categorized as assets, sensors, and controls.

Probabilistic Tracking Methods

The long term goal of this research is the development of a novel Intrusion Detection System (IDS) that utilizes Multiple Hypothesis Methods (Blackman, 1986), (Streith & Luginbuhl, 1995), (Blackman & Popoli, 1999), (Blackman, 2004), (Kim, et al, 2015) in conjunction with the following advanced statistical technique:

- Kalman Filters (KF) (KF, n.d.),
- Markov Chains (MC) (MC, n.d.),
- Dynamic Bayesian Networks (DBN) (DBN, n.d.),
- Dynamic Decision Networks (DDN) (Howard & Matheson, 1984), (Zhang, 1998),
- Second Order Uncertainty (SOU) (Borsotto & Savell, 2006), (Woodson, & Savell, 2018) and
- Probabilistic Relational Models (PRMs) (Koller & Pfeffer, 1998), (Pfeffer, 2000), (Getoor, et. al, n.d.)

to dynamically track the progress of a cyber intruder in a compromised network over time.

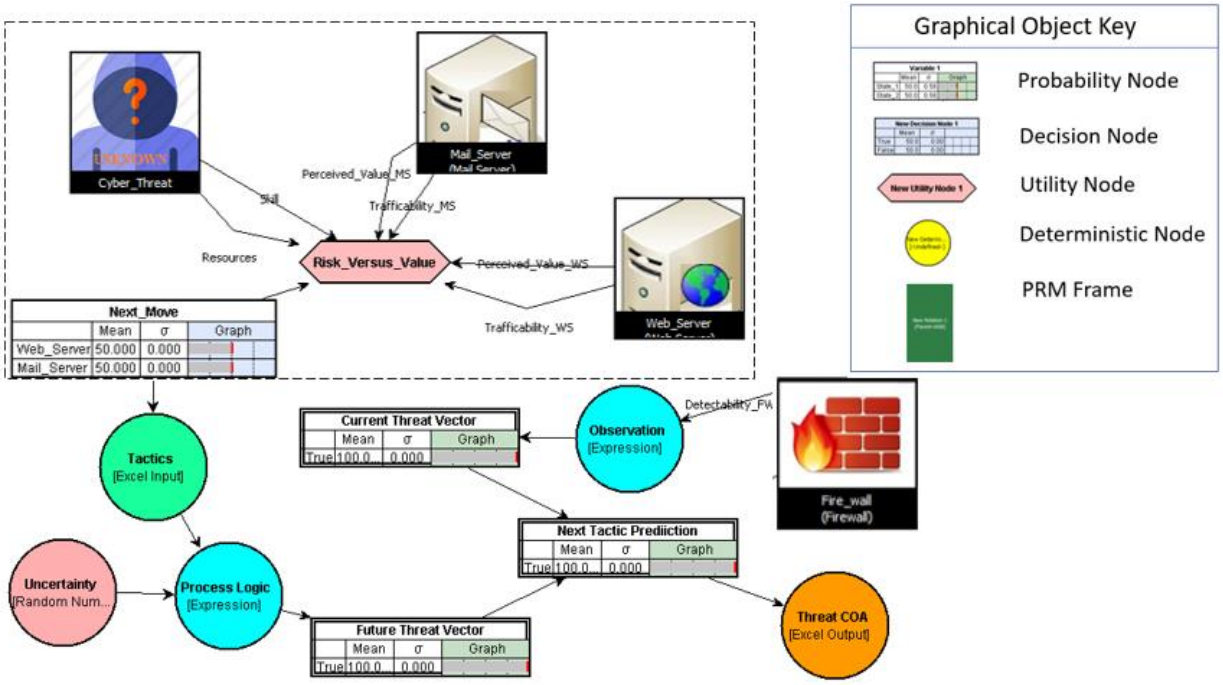

Figure 3: Recursive Dynamic Decision Network using PRMs

Figure 3 gives a high-level schematic representation of the proposed approach using interacting PRMs. There are five graphical objects used to construct prediction models containing underlying code, which when the model is compiled, produces executable Java code that can be inserted into other software.

1. **Probabilistic Nodes** that execute Bayes theorem in a Bayesian Network (BN),
2. **Decision Nodes** that establish alternative options for subsequent Course of Action (COA),
3. **Utility Nodes** used to assign value and assess risk associated with a decision,

4. **Deterministic Nodes** used for performing various computational function such as reading a database or executing a formula, colorized to indicate the type of function they perform, and
5. **Probabilistic Relational Model (PRM) Frame** which are object-oriented constructs holding other graphical models.

All these objects that provide inference and data communication between the objects are connected by arrows.

Figure 3 is a composite of two models. The model shown in the dashed box is a single time frame snapshot of the decision network, while the model below it, is a representation of the Dynamic BN (DBN) used to calculate the threat vector movement over time. However, the DBN shown at the bottom of Figure 3 is misleading. It is using Continuous Linear Gaussian (CLG) Probability nodes as designated by the double-lined border on the node versus the single line "Next Move" discrete node in the upper portion of the figure. CLG nodes imply that the cyberthreat's movement in the network can be represented by a closed form equation or process, such as the Kalman Filter representation used in missile tracking. In fact, this movement should be framed as "discrete movements" or specific "snapshots" in time. Our research indicated that time should not be considered a key parameter, as APT's typically do not behave like a "berserker". Rather, the attackers sneak around the system with discrete changes discontinuously over time. These attackers may even remain resident in the system for long periods of time without any action to determine if there is any danger of detection, or they may also be just watching and waiting for an opportunity to move forward.

Nevertheless, tracking of a cyberthreat is indeed analogous to tracking a target in missile defense (Stotz & Sudit 2007), (Schwoegler1, 2011), (Schwoegler2, 2011). In general, the difference is that target tracking of a threat in missile defense is governed by continuous kinematics with a limited number of possible maneuvers that the threat can execute, such as a ballistic trajectory with acceleration limitations. Whereas a cyberthreat can maneuver between any possible finite states found in the IDS. Other differences include the observation/sensor-type, track formation, prediction and filtering, hypothesis formation, resource management, and data type. Data type is another significant difference in that with cyber tracking, the sensor data is represented by discrete Indicators of Compromise (IOC) or Attack (IOA), typically an IP or port addressed per IODEF (datatracker.ietf, 2016) alert format, which are discrete states. Observation data from a missile target tracking sensor, such as a radar, is continuous. In this context, if a variable can take any value between its minimum and maximum value, then it is called a continuous variable. Whereas a discrete state variable has a finite set of unique values (albeit, possibly a large number of discrete values). Continuous versus discrete is important in deciding on the appropriate analytical methods to be deployed in the tracking.

Cyberthreat Agent Model

The threat model is multi-dimensional and provides some relative quantification of factors as they relate to each other. Specifically, threat actors were decomposed into six basic types (REDLEGG, 2020), (CCCS 2021) include: Thrill Seeker, Cyber Criminal, Nation State, Insider Threat, Hacktivist and Terrorist Group, as shown in Figure 4.

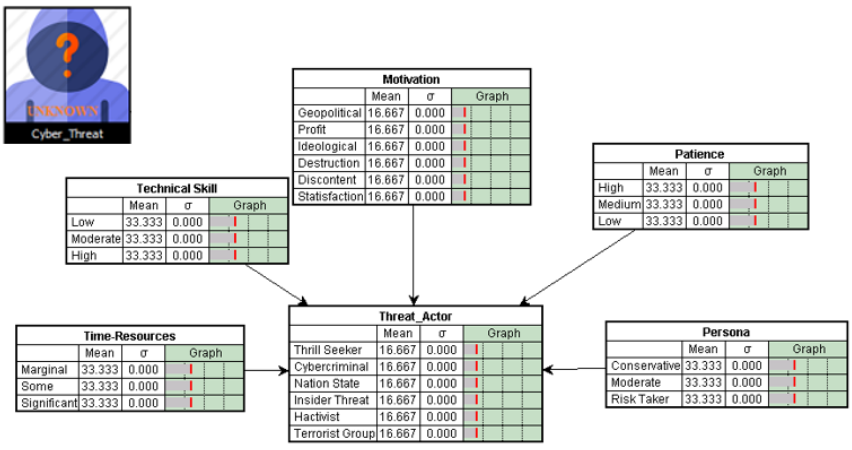

Figure 4: Cyberthreat Agent PRM

Note that the PRM in Figure 4 is uncompiled with no imposed external evidence. Therefore, all the mean and standard deviation probabilities numerically expressed in the blocks are uniformly distributed across all the states for that node.

Each of these threat actor categories were further characterized by their source of motivation, persona, patience, technical skill, and resources as individual attribute nodes shown in Figure 4. Table 1 provides the discrete states characterize the criteria classification for each of these attribute nodes.

Table 1: Discrete States for Each Attribute Node defining a Threat Actor

Motivation	Persona	Patience	Technical Skill	Time-Resources
Geopolitical	Conservative	High	Low	Marginal
Profit	Moderate	Medium	Moderate	Some
Ideological	Risk Taker	Low	High	Significant
Destruction				
Discontent				
Satisfaction				

By pivoting these characteristics with the six threat actor types, 486 scenarios were developed. Each characteristic in Table 1 was assigned a value ranging from 1 to 100 to represent a mean and a standard deviation. Once assigned, these values provided relative means of weighting each characteristic. Additionally, the values provided were given a ceiling such that no scenario total was greater than another. This methodology enabled a better one to one comparison of each threat actor scenario. The values of the mean multiplied by the standard provided a basis of calculation for the model.

The threat modeling, to this point was developed largely using expert judgement from multiple SMEs. Future effort would seek to adopt and incorporate publicly known data sets that better characterize APT behaviors. This will be challenging, since many data sets lack the detail needed to achieve statistically relevant conclusions. However, this research effort provides a basis for proper collection as organizations harvest more forensic data on their incidents. Current state of incident data collection and analysis does not allow to track APT behavior in real-time or near-real-time automatically. Even though we can harvest indicators close to near-real-time and perform basic analysis on them, human analysts should categorize and recognize cohesive patterns. Furthermore, this research suggests that a standard ontology for data collection on known compromises would greatly enhance model development in the future.

Data For Threat Behavior Forecasting

Addressing a question about the data is essential for the modeling. We need to look at three areas: (1) the data to model the network under investigation, (2) Indicators of Compromise (IOCs), and (3) the Indicators of Attack (IOAs).

To model the network under investigation, we need to know the network architecture, which includes network branches and enclaves, network nodes, their business or network function, their status as asset/control/sensor (see "Network Component Model"), status of the configuration (standard/non-standard), status of vulnerabilities (known vulnerabilities), connections between network nodes, protocols, ports, constraints, and connections between branches and enclaves.

An IOC is "a piece of digital forensics that suggests that an endpoint or network may have been breached" (Crowdstrike, 2021). Those indicators provide us with a view to the past, to what has already happened. If identified and analyzed fast enough, they can minimize the effect of the attack by employing remediation tactics immediately, and possibly stop the attack before it successfully finished, thereby utilizing features of both Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS).

Examples of IOCs are:

- Unusual inbound and outbound network traffic,
- Anomalies in privileged user account activity,
- Unknown application within the system,
- Large numbers of requests for the same file, and
- Unusual DNS requests.

IOAs, even though in some cases similar to IOCs, are the actions that the attacker performs during the attack, before IOCs become detectible (Crowdstrike, 2021). Looking at IOAs allows us to concentrate on the final goal of the attacker, not his/her methods, or tools. Identifying IOAs in real- or near-real-time helps the analysts to recognize an attack while it's in progress and possibly stop it.

Examples of IOAs are:

- Internal host communicates with known bad destination or external host that you don't conduct business,
- Internal host uses non-standard (for your company) ports,
- Public server/DMZ communication with internal host,
- Network scan by internal host, and
- Reinfection by same malware in a few minutes after clean-up.

To collect and analyze both IOCs and IOAs, an organization needs to have mature digital forensics, Endpoint Detection and Response (EDR), Security Information and Event Management (SIEM). Constriction of a collection of attack threads from known IOCs and IOAs, by putting them in a sequence of events that constitute an attack thread will facilitate both the forecasting of the attacker behavior and modeling of an attack on a given network. The database of attack threads can be enhanced by (1) information about level of technical skills and resources that the attacker should have to implement specific attack threat, (2) possible motivation behind, as well as (3) time aspect. Comparing IOCs and IOAs collected from the network under investigation to the attack threats from this database will provide this model with the data needed to predict the next move by the attacker.

Network Component Model

To model the network components, we define three basic types of network nodes as illustrated in Figure 5:

- **Asset** – a node that provides a process that leads to the delivery of a critical service or has other type of critical business value (e.g., database server, mail server),
- **Control** – a node that perform control functions within a network to reduce exposure (e.g., firewall),
- **Sensor** – a node that enables identification of threats (e.g., IDS).

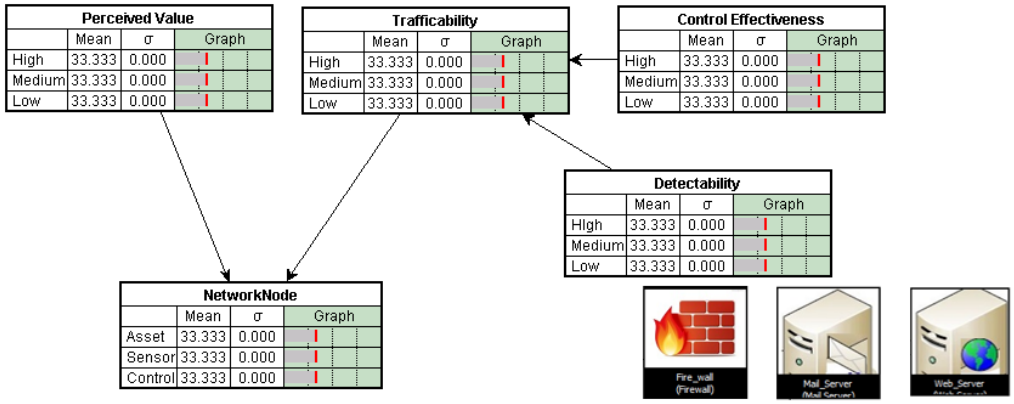

Figure 5: Network Component PRM

Accompanying these types are the following attributes – Perceived Value, Trafficability, Control Effectiveness and Detectability, which allows us to model any network node by dialing up or down values, assigned to these attributes.

Perceived Value is an attribute to describe how valuable this specific node from the viewpoint of the threat actor. Dialing up or down the assigned value, we can model/assume how this node can be desirable to the threat actor. Thus, together with motivation this attribute can affect the threat actor's maneuvers regarding this node.

Trafficability is an attribute to represent the relative ease the threat actor has in navigating through the component. This attribute, taken together with the threat actor's skill level and patience, can make the node suitable for a role as a steppingstone in the vector of the attack. It is further defined by **Control Effectiveness**

and **Detectability**. **Control Effectiveness** captures the inverse of vulnerability level of the node. And **Detectability** represents the ability of the node to sense the presence of the threat.

Note that Figure 5 is uncompiled. Therefore, all the probabilities are uniformly distributed across the states for each attribute node in the network. As will be demonstrated later, unique probability distributions result when external evidence is imposed, and the network is compiled. The two PRMs depicted for the Threat Agent and Network Components in Figures 4 and 5 respectively are combined in Figure 3 above as a general network model using the object oriented PRM architecture.

NETWORK ARCHITECTURE MODELING

There are many applications for Modeling & Simulation (M&S) of the cyberattack, including: acquisition analysis support, operational support, wargaming exercises and training. Acquisition analysis typically involves the use of simulation to model warfare for the purpose of understanding the capabilities, performance, and interactions of forces and systems in combat. The result sought by the analyst is a better understanding and a quantitative assessment of forces, systems, doctrine, and tactics in warfare conditions. Operational support involves using simulation to help assess possible outcomes of an Operational Plan (OPLAN), or Course of Action (COA), used in promoting creative thinking through the visualization of the battle space, allowing users to assess a range of likely plans, tactics, and outcomes. As a result of this assessment, users can evaluate the strengths and weaknesses of candidate OPLANs and COAs from various perspectives. The objective is to provide a continuum of operational support that includes plan development, analysis, assessment, and operational action/response. Within DoD, wargaming and training involves interfacing with Command and Control (C2) systems to inject simulated entities into the exercise or experiment and to make these visible to participants viewing the exercise via operational C2 systems. In this way, simulation tools may be used to augment live exercise participants and systems with simulated ones. In general, simulations are used to gather insight to a potential situation or product by modeling the interaction between it and its relevant environment boundary or stimulation conditions. Through simulation, an understanding is gained regarding the behavior of the system modeled under these selected conditions. Based on the results various strategies for the operation or a system's operational limits are refined. Our collaborative team of subject matter experts created this multi-hypothesis Bayesian cyber modeling solution aimed to model cyberattacks at the network level. As such, various network node types are represented in the constructive simulation environment.

A frequent observation regarding the simulation of cyber events is the misconception that their results are fully deterministic (i.e., an intrusion is either successful or its not). Quite the contrary, the outcome of many attacks can be random and therefore probabilistic. For example, the current state of the system has a significant impact on both probability and effect. Consider the results of Denial-of-Service (DoS) attacks: no effect, various degrees of degraded operation, system crash, etc. – all of which are dependent on current network traffic conditions, hardware/resource utilization, application load, and other factors. System misconfigurations can occur randomly; for example, lockdowns are sometimes reversed inadvertently or even deliberately. Human behavior can considerably vary, for example:

- Susceptibility to social engineering,
- User adherence to acceptable use policies and procedures, and
- Insider threats, particularly malicious and erroneous actions of operators and administrators.

Even incident response and recovery can vary depending on countless factors, including the availability of network and system administrators and their respective skill levels, the time required to bring nodes and applications back up after a crash, and so forth. With so many events having indiscriminate outcomes, the need to construct statistical models to study the effects of cyber phenomena on systems becomes clear.

The cyberattack simulation models computer systems and network elements, including hosts, routers, firewalls, Intrusion Detection Systems (IDS's), switches, hubs, and links. With the team's collective experiences in network systems and cyber, attributes associated to a given node were designed to reflect network elements and potential vulnerabilities in real world. A network's host node characteristics can be represented explicitly in our simulation. Attributes like node type, connectivity, and defense capabilities are all important parameters in cyber effects modeling. Through execution of the simulation logic, end users of the tool can use this

simulation capability to answer "what-if?" questions for resiliency and risk assessments. To support cyber analysis, the simulation tool can be set up to run in a constructive manner. Since cyberattacks are never deterministic in the real world, Monte Carlo techniques could be exercised to model the probabilistic events in a cyber incident. This is useful in modeling different cyber actors and their attack behaviors.

Simple 2-Node BN Example

To demonstrate our approach, consider the pure BN shown in Figure 6 representing the threat agent and two assets (mail server and web server) in a computer network. It represents a model for calculating the likelihood of a Nation State threat actor's movement between the assets, as represented by the insert at the lower left in the figure. It is a merger of three BNs extracted from Figures 4 and 5 above.

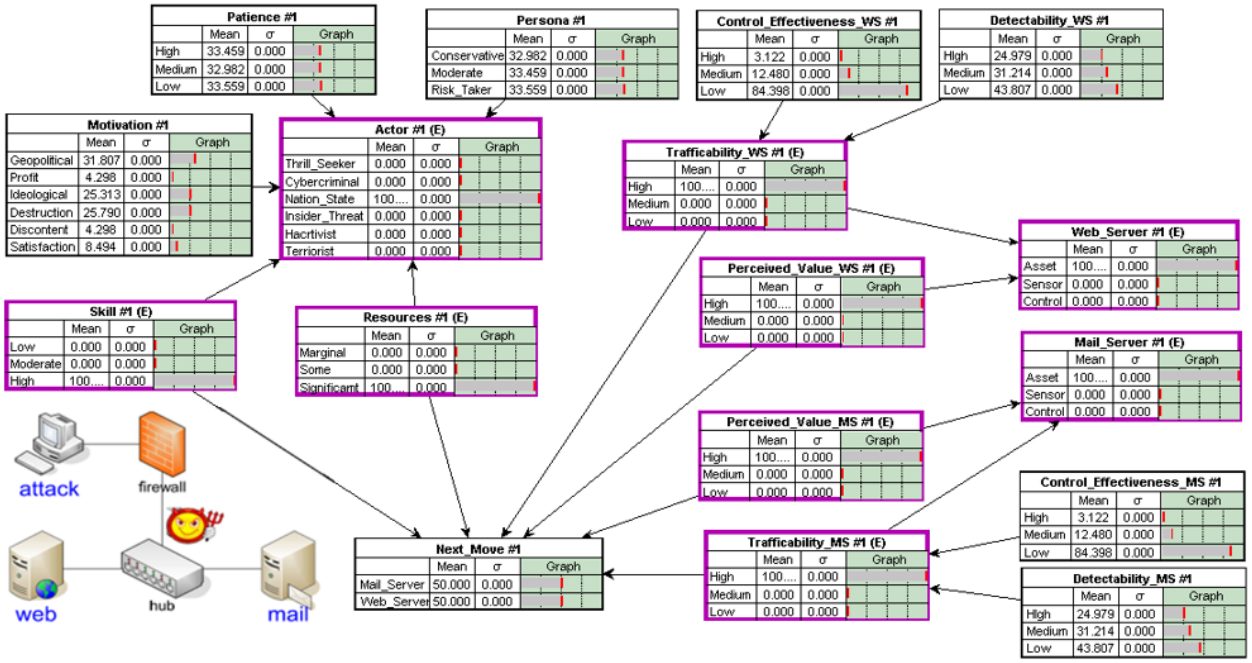

Figure 6:Phase I Demonstration Network Topology and Bayesian Network when everything is equal then 50:50 Likelihood of Moving to either Asset

This simplified representation is different from the decision network shown in Figure 3. Here only Bayesian inference is represented to determine the likelihood of a movement, without any decision or utility node logic. The compiled network shows the status and feedback when external evidence has been imposed on the nodes as indicated by purple borders. The feedback is the result of imposed Bayesian inference embodied in the "child" node Conditional Probability Tables (CPTs). This results in probability distributions calculated across the states in every node, as the simulation is played out.

The construction of the CPTs is perhaps the most critical step in constructing a BN. It is a very tedious task which is achieved by one of three approaches:

- Cause-and-effect observations obtained in real life or controlled experiments,
- Monte Carlo (MC) simulation results giving output from applied inputs in a physic-based computer model, and
- Subject Matter Expert (SME) opinion based on experiences, rules of thumb and/or "lessons-learned" data.

The CPTs used in Figure 6 were populated by our SME team during a series of workshops. The process of populating the CPT involves evaluating all the possible cause-and-effect influences from the independent parent variable nodes collectively have on the selected dependent child node holding the CPT.

Figure 7 illustrates the CPT population process for a simple two parent (Variable1 and Variable2) and one child (Variable3) with variance in both the evidence of the parent states and child CPT. The example CPT shown at the top of Figure 7 illustrates the basic property that the sum of each row associated with the parent nodes across the states of the child in the CPT must sum to 100% probability. Note that the size of the CPT (number of cells) is equal to the product of the number of variables times the product of the number of parent and child states, which in Figure 7 is 2*3*4 = 24 mean value and 24 variance cells (48 total).

Figure 7 is a very trivial BN and can easily be managed. The threat agent model in Figure 4, on the other hand is more complex and its CPT contains a total of 5,832 cells for the mean values and variance. Managing and populating such large CPTs quickly becomes unwieldy.

Learning algorithms for BNs center around recalibrating the CPTs as new data is obtained. In the cyber world such training data would be obtained from documented forensics of successful cyber breaches. Such training data would automatically recognize bias in the system or new data sets. Indeed, as discussed in the next section, our simulation results exposed some inadvertent bias that were introduced in the model CPTs, by our expert team.

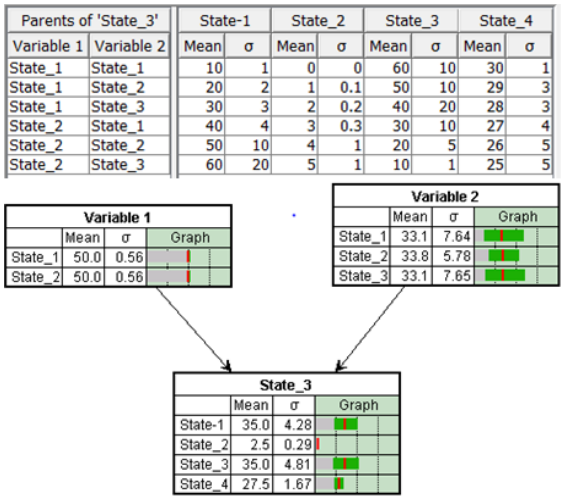

Figure 7: Example BN and CPT with Variance in Evidence and Inference

SIMULATION RESULTS

Figure 6 above shows the resulting 50:50 "Next Move" probabilities for the situation where the perceived value and trafficability to the Web Server (WS) and Mail Server (MS) are identical. The figure is for the Nation State threat actor, with the perceived value and trafficability set to "High", but the same result occurs regardless of which cyberthreat is selected so long as the perceived value and trafficability of the targets are identical regardless of the state settings[1]. When the perceived value and trafficability state values of the target assets are different, then the likelihood changes accordingly.

Table 2 gives the results for three additional simulations beyond the "All Equal" result in Figure 6, as shown by columns in the table:

1. The trivial result of 100% likelihood of moving to the WS when its value is high and trafficability is high (i.e., easy to "crack") and the MS value is low and difficult to navigate with a low trafficability.
2. A 95:5 likelihood between the MS:WS when the perceived value of both assets is high but the trafficability to the WS is low (i.e., difficult to penetrate) and the MS trafficability is high (easy to penetrate).
3. The case when there is hard evidence that both the Web Server and Mail Server are Assets, but the perceived value and trafficability of the Web Server are unknown and are determined by the Bayesian inference. The result is a 74.475% likelihood of movement to the Mail Server, because it's trafficability is high with a high perceived value. As expected, the threat takes the easier path.

Table 2:Example Results of Next Move Sensitivity Study

BN Nodes	State	All Equal	100% WS	95:5 MS	75:25 MS
		State Likelihoods			
Skill	Low	0	0	0	0
	Moderate	0	0	0	0
	High	100	100	100	100
Resources	Marginal	0	0	0	0
	Some	0	0	0	0
	Significant	100	100	100	100
WS Trafficability	High	100	100	0	51.145
	Medium	0	0	0	22.899
	Low	0	0	100	25.956
WS Perceived Value	High	100	100	100	63.39
	Medium	0	0	0	36.61
	Low	0	0	0	0
MS Trafficability	High	100	0	100	100
	Medium	0	0	0	0
	Low	0	100	0	0
MS Perceived Value	High	100	0	100	100
	Medium	0	0	0	0
	Low	0	100	0	0
Next Move	Mail Server	50	0	95	74.475
	Web Server	50	100	5	25.525

[1] *Note that all the simulations were performed using GCAS' SOU™ BN software. To simplify this simulation exercise, the variances associated with each node state were set to zero.*

CPT Bias

This research was conducted over a 4-month period, so short cuts were taken to achieve the "proof of concept" end goal. One place where additional work is merited is in the specification of the Bayesian Conditional Probability Tables (CPTs) which provide the inference from the parent independent variables to the child dependent variable. As discussed previously there are several approaches to populating a CPT with experimental data/observations being the preferred method. For our effort we relied on the Subject Matter Expert (SME) opinions of our team members who tediously reasoned thru the cause-and-effect logic. This is a reasonable approach as a first start, but it is subject to biases introduced by the SME intrapersonal and interpersonal uncertainties.

Such biases were discovered in the Cyberthreat Agent model (Figure 4) when the model was exercised using extreme situation cases in a sensitivity analysis study. For example, imposing the "hard" result for each Threat Actor type in Figure 4 above gives the probability distribution results shown in Figures 8 below. Comparing the distributions driving the likelihood of the various threat actors, only ***Motivation*** really determines what type of threat actor is present and the ***Patience*** and ***Persona*** variable nodes are non-entities, meaning they are basically uniformly distributed under all circumstances. ***Technical skill*** and ***Time-Resources*** are slightly more robust, but still pretty much inconsequential. For example, Figure 8b indicates that the Cyber-criminal is principally motivated by profit, with all the other variables uniformly distributed. The fact that ***Patience*** and ***Persona*** do not contribute is not surprising because the values in the CPT for these variables are identical in most cases. However, the values for ***Technical Skill*** and ***Time-Resources*** in Figure 4 appear to be washed out for some reason, which is surprising.

Thrill Seeker Motivation

	Mean	σ	Graph
Geopolitical	9.029	0.110	
Profit	21.510	0.234	
Ideological	5.678	0.069	
Destruction	13.039	0.157	
Discontent	3.995	0.050	
Statisfaction	46.749	0.316	

Cybercriminal Motivation

	Mean	σ	Graph
Geopolitical	6.407	0.084	
Profit	64.890	0.283	
Ideological	6.323	0.081	
Destruction	7.130	0.093	
Discontent	2.567	0.034	
Statisfaction	12.682	0.154	

Nation State Motivation

	Mean	σ	Graph
Geopolitical	32.715	0.292	
Profit	5.506	0.066	
Ideological	20.313	0.213	
Destruction	25.552	0.247	
Discontent	5.348	0.064	
Statisfaction	10.567	0.120	

Insider Threat Motivation

	Mean	σ	Graph
Geopolitical	9.694	0.114	
Profit	16.415	0.176	
Ideological	9.423	0.109	
Destruction	10.248	0.119	
Discontent	36.424	0.285	
Statisfaction	17.796	0.184	

Hactivist Motivation

	Mean	σ	Graph
Geopolitical	19.734	0.199	
Profit	6.084	0.070	
Ideological	25.596	0.249	
Destruction	11.493	0.139	
Discontent	19.061	0.193	
Statisfaction	18.033	0.184	

Terrorist Motivation

	Mean	σ	Graph
Geopolitical	19.519	0.205	
Profit	3.352	0.040	
Ideological	25.372	0.240	
Destruction	26.044	0.245	
Discontent	16.062	0.169	
Statisfaction	9.650	0.108	

Figure 8: Examples of Motivation Likelihood Distribution for Different Cyberthreats

Ideally, a calibrated database or resources could be used to fuel an improved CPT. Some agencies have psychologist that profile criminals and threats, including monitoring the dark web, and such participation in providing profiling details could be utilized in the follow-on effort. Indeed, our initial research does provide a list of key information that needs to be collected in the future to make the model more relevant. For example, it is unlikely that we will be able to quantify the number of resources each threat actor type has at its disposal during a cyberattack. However, published data does exists that rank the different Nation State actors relative to each other, (e.g., Russia, China, Iran, and North Korea), as to their technical skill and resources capability. Such intelligence information could fuel the model.

A similar sensitivity analysis was conducted on the Network Component Model (Figure 5). The CPT associated with the primary Network Node Type child appeared to produce generally expected results. The secondary Trafficability child of the Control Effectiveness and Detectability parents were less robust but nevertheless still acceptable. The main criticisms of the Network Component Model were its degree of completeness and interpretation. For example, there is a question of how the MITRE ATT&CK® framework (Brock, (2020), (MITRE, 2020) vulnerabilities can be tied into the Control Effectiveness variable.

RECOMMENDATIONS FOR FUTURE WORK

This paper is the first in a series of planned publications of follow-on work as our research evolves. In our future work we will use Multiple Hypothesis Tracking (MHT) developed for missile tracking/forecasting by Bayesian Inference

extended with Second Order Uncertainty (SOU) to account for non-linear effects and Interacting Multiple Model (IMM) filter to capture the uncertainty in threat's maneuvers. Bayesian SOU is used for both the threat maneuver hypothesis in MHT and in the determination of the transition probabilities between various sensors in the IDS. The multiple hypothesis process patterns for the threat vector are those documented in the MITRE ATT&CK framework.

MHM leverages utility/prospect/decision theory in a Bayesian context to hypothesize possible future movements as multiple threads. The result is a set of potential threat Course of Action (COA) maneuvers prioritized by the relative risk of the threat's decision options. To improve the precision of the hypothesis, SOU estimates will be used, as well as PRMs for creating an object-oriented architecture for representing the threat agents and various cyber network topologies.

In IMM/MHT processing, the association decisions are deferred, forming multiple (competing) hypotheses. These hypotheses get propagated forward in time, with the expectation that future measurements will resolve any ambiguity in the general hypothesis. Without proper management of the hypotheses there will be exponential growth in the number of hypotheses generated. Although there is a slight increase in the processing, the MHT has been shown to produce a more robust tracking solution than non-MHT solutions.

ACKNOWLEDGEMENTS

This material is based upon work supported by the United States Air Force AFRL/SBRK AFWERX under Contract No. FA8649-21-P-0059. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the United States Air Force AFRL/SBRK AFWERX.

REFERENCES

Blackman, S. (1986). Multiple Target Tracking with Radar Applications., Norwood, MA: Artech House,

Blackman, S. and Popoli, R. (1999). Design and Analysis of Modern Tracking Systems. Artech House,

Blackman, S. (2004). *Multiple Hypothesis Tracking for Multiple Target Tracking*, IEEE A&E Systems Magazine Vol. 19, No. 1 January

Borsotto, M and Savell, CT. (2006). *Decision Making under Probability Intervals*. GCAS Incorporated, US Navy Contract N00178-04-C-3117

Brock, C. (2020). *What is the MITRE ATT&CK Framework?*, Digital Guardian

Canadian Center for Cyber Security. (2021). "Cyber threat and cyber threat actors" https://cyber.gc.ca/en/guidance/cyber-threat-and-cyber-threat-actors

Crowdstrike (2021) https://www.crowdstrike.com/cybersecurity-101/indicators-of-compromise/

Crowdstrike (2022) https://www.crowdstrike.com/resources/reports/global-threat-report/

Crowdstrike (n.d.) https://www.crowdstrike.com/resources/crowdcasts/the-1-10-60-minute-challenge-a-framework-for-stopping-breaches-faster/

datatracker.ietf (2016). The Incident Object Description Exchange Format Version 2, https://tools.ietf.org/html/rfc7970

Dynamic Bayesian Network (n. d.). https://en.wikipedia.org/wiki/Dynamic_Bayesian_network

Getoor, L., Friedman, N., Koller, D., Pfeffer, A. and Taskar, B. (n. d.). "Probabilistic Relational Models", https://ai.stanford.edu/~koller/Papers/Getoor+al:SRL07.pdf

Howard, RA and Matheson, JE, (1984) *Influence diagrams, The Principles and Applications of Decision Analysis*, Strategic Decisions Group, Menlo Park, California, USA, Pages 719-762,

Jensen, VF. (2001). Bayesian Networks and Decision Graphs. Springer, New York, NY

Kalman Filter (n. d.). https://en.wikipedia.org/wiki/Kalman_filter

Kim C., Li, F., Ciptadi, A. and Rehg, JM. (2015)Kim C., Li, F., Ciptadi, A. and Rehg, JM. (2015). *Multiple Hypothesis Tracking Revisited*, IEEE International Conference on Computer Vision (ICCV)

Koller, D and Pfeffer, A. (1998). "Probabilistic Frame-Based Systems"; AAAI-98, pp 580-587,

Lee, RM, Assante, MJ and Conway, T, (2016) "Analysis of the Cyber Attack on the Ukrainian Power Grid – Defense Use Case", SANS E-ISAC https://ics.sans.org/media/E-ISAC_SANS_Ukraine_DUC_5.pdf(link is external)

Li, XR and Jilkov, VP. (2000). Survey of Maneuvering Target Tracking Part V: Multiple-Model Methods, Fall 2000 Lecture Notes: Israel Institute of Technology,

LMCO1 (n.d.) https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html

LMCO2 (n.d.) https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/Gaining_the_Advantage_Cyber_Kill_Chain.pdf

Markov Chain. (n. d.). https://en.wikipedia.org/wiki/Markov_chain

MITRE. (2020). *MITRE ATT&CK Framework* https://attack.mitre.org/matrices/enterprise/ (2020)

NIST, (2019) https://csrc.nist.gov/publications/detail/sp/800-160/vol-2/final

Pearl, J. (1997). Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference, Morgan Kaufmann, San Francisco, CA,

Pearl, J. (2000). Causality Models, Reasoning and Inference, Cambridge University Press, New York, NY,

Pfeffer, AJ. (2000), Probabilistic Reasoning for Complex Systems. Ph. D. Dissertation, Stanford University, CA,

RedCanary (2022) https://redcanary.com/threat-detection-report/techniques/credential-dumping/

REDLEGG. (2020). "7 Types of Cyber Threat Actors and Their Damage" https://www.redlegg.com/blog/cyber-threat-actor-types

Schwoegler, S, Blackman, S, Holsopple, J and Hirsch, MJ. (2011). *On the Application of Multiple Hypothesis Tracking to the Cyber Domain*, 14th International Conference on Information Fusion

Schwoegler, S, Blackman, S, Holsopple, J and Hirsch, MJ. (2011). *Multiple Hypothesis Tracking for the Cyber Domain*, SPIE Vol. 8137

Shimkin, N., (2009). Multi-Model State Estimation- Dealing with Model Uncertainty, https://www.webee.technion.ac.il/people/shimkin/Estimation09/ch9_multimodel.pdf

Stotz, A and Sudit, M. (2007). *INformation Fusion Engine for Real-time Decision-making (INFERD): A Perceptual System for Cyber Attack Tracking*, 10th International Conference on Information Fusion,

Streith, RL and Luginbuhl, TE. (1995). Probabilistic Multi-Hypothesis Tracking, NUWC-NPT TR 10, 428,

Woodson, SG and Savell, CT. (2018). *Decision Making under Uncertainty*, GCAS Incorporated, MDA Contract HQ0147-16-C-7805

Yadav, T. and Rao, A.M. (2015) *Technical Aspects of Cyber Kill Chain*, in J.H. Abawajy et al. (Eds): SSCC 2015, CCIS 536, pp. 438–452, DOI: 10.1007/978-3-319-22915-7 40, Springer International Publishing, Switzerland

Zhang, NL. (1998). "Probabilistic Inference in Influence Diagrams", Computational Intelligence, Vol. 14, No. 4, pp. 475-497