Big Data for Cyber-attack Management

Terry Janssen, PhD, SAIC
Chief Scientist & Cyber Strategist
Cyber Operations
Washington, D.C. USA
terry.l.janssen@saic.com

Nancy Grady, PhD, SAIC
Technical Fellow, Data Science
Emerging Technologies
Oak Ridge, TN USA
nancy.w.grady@saic.com

Abstract—Big Data has become big business. This paper explores a means of improving Cyber-security using Big Data technologies, ontology, and decision support for preventing or reducing losses from cyber-attacks. Because of the priority of this threat to national security it is necessary to attain results far superior to those found in modern-day security operations centers. Focus is on the potential application of ontology engineering to this end, and a cyber-attack management system (CAMS) is proposed. Because of the complexity of this problem domain and the requirement for quick, just-in-time action to prevent loss, we are investigating the application of formal methods with Big Data and ontology. The approach is overviewed and issues are discussed.

Keywords—big data; ontology; cyber-security; modeling, search; discovery; analytics; variety; metadata

I. INTRODUCTION

The last few years have seen tremendous increases in the amount of data being generated and used to provide capabilities never before possible. “Big Data” refers to the new engineering paradigm that scales data systems horizontally to use a collection of distributed resources, rather than only the earlier vertical scaling that brought faster processors and more data storage into a single monolithic data platform. Big Data technologies have the potential to revolutionize our capabilities to handle the large datasets generated in any cyber data analytics. The challenge, however, is not just in handling the large volumes and high data generation rate, but in leveraging all available data sources to provide better and faster analytics for attack detection and response. In this paper we will discuss Big Data analytics, Metadata and Semantics for data integration, and applications to cyber security and cyber data management.

II. BIG DATA

Big Data has several defining characteristics including volume, variety (of data types and domains-of-origin, and the data flow characteristics of velocity (rate) and variability (change in rate) in which the data is generated and collected;

Traditional data systems collect data and curate it into information stored in a data warehouse, with a schema tuned for the specific analytics for which the data warehouse was built. Velocity refers to a characteristic that has been previously referred to as streaming data. The log data from cell phones for example flows rapidly into systems, and alerting and analytics are done on the fly prior to the curation and routing of data or aggregated information into persistent storage. In a Big Data Architecture this implies the addition of application servers to handle the load. Variability refers to changes in the velocity of data flow, which for cost-effectiveness leads to the automated spawning of additional processors in cloud systems to handle the load as it increases, and release the resources as the load diminishes.

Volume is the dataset characteristic most identified with Big Data. The engineering revolution began due to the massive datasets from web and system logs. The implication has been the storage of the data in its raw format, onto distributed resources, with the curation and imposition of a schema only when the data is read. The variety characteristic often is used to refer to multiple formats of data, recognizing that the much larger amounts of unstructured text, image and video data have vital information to be harvested. This results in more sophisticated curation and pre-analytics to extract useful information, but not a change in architecture. Variety more broadly refers to the use of data from multiple domains. While volume and velocity are revolutionary in the information technology (IT) engineering, the variety characteristic drives a revolution for the organization in both the engineering and the mission by allowing previously impractical or impossible analytics. Techniques for handling variety will change our analytical capabilities.

A. Big Data Analytics

Much of the development of Big Data Engineering is a result of the need to analyze massive web log data. Massive Web logs were first filtered by page for aggregate page counts, to determine the popularity of
Big Data analytics then moved into what is termed “Social Network Analysis” (SNA) to analyze a link-node structure. Beginning with Google’s PageRank algorithm, a huge field of analytics has developed to understand the relationships between nodes represented by their link structure. This has applicability to cyber through determining the appropriateness of activity between servers. The complexity in cyber is that additional information on the “nodes” must be integrated in order to analyze for the appropriateness of any activity.

The last fifteen years have seen the extension of a number of analytics techniques to the horizontal big data scaling paradigm. The fundamental analogy to new analytics opportunities for the cyber community comes from the extension of Web log and SNA analysis to use the massive amounts of data to determine session patterns and the appropriateness of activity between resources. The challenge is that cyber must also deal with the attributes of any resources, adding in a variety of other contextual datasets into the analysis.

B. Variety

Traditional systems handled the variety of data through a laborious integration process to standardize terminology, normalize into relational tables, choose indexes, and store into a data warehouse that is tuned for the specific analytics that are needed. Naturally this is an inflexible process that does not easily accommodate new data sources, changes into underlying data feeds, or new analytical requirements.

For Web log analysis, this extension to customer session analytics only required the assignment of a customer or visitor ID to the session, allowing integration with a purchasing history. In the cyber analytics case, the integration point is not so simple. The integration of packet data, with server log data, with port-to-port connectivity data, with server type data, with network router settings, etc. provides a much-more complex use case, needing a more sophisticated way to integrate such a variety of data, some of which carries a number of additional attributes that are needed.

Recently variety datasets have been addressed through mashups that dynamically integrated a couple of datasets from multiple domains to provide new business capabilities. Early mashups demonstrated this value, for example in the integration of crime data with real estate listings; a valuable analysis that was not possible prior to the availability of open datasets. The limitation to such mashups is that they typically consist of the integration of a limited number of datasets, with the integration variables being manually selected. This type of manual integration is not sufficient for analytics across a variety of large volume datasets with complex inter-relationships.

Variety is the Big Data attribute that will enable more sophisticated cyber analytics. The requirement is for an automated mechanism to integrate multiple highly diverse datasets in an automated and scalable way. This is best achieved through a controlled metadata.

III. METADATA

The executive branch has been pushing an open data initiative to move the federal government into being a data steward. The goal in releasing the data is to better serve the public and promote economic growth through the reuse of this data. The difficulty in using this data arises from the lack of metadata descriptions of the datasets. Data reuse requires as much information as possible on the provenance of data; the full history of the methods used for collection and curation and analysis. Proper metadata increases the chances that datasets are re-purposed correctly - leading to analytical conclusions that are less likely to be flawed.
Two mechanisms are used for dataset integration in a relational model. In the relational model lookup tables are established to translate to a common vocabulary for views, and a one-to-one correspondence is used to create keys between tables. In a NoSQL environment, joins are not possible so table lookups and or keys cannot be used for data integration. The connection of data across datasets must reside in the query logic and must rely on information external to the datasets. This metadata logic must be used to select the relevant data for further analysis, implying the need for both standard representation as well as additional attributes to achieve the automated data retrieval.

A second approach is used to speed the data integration process for manual mashups of diverse datasets. Often XML wrappers are used to encapsulate the data elements, with the nomenclature for each dataset provided in the wrapper, based on user interpretation of the data elements. This approach allows rapid integration of data through the wrappers (as opposed to a lengthy data warehouse integration), but it is not an approach that can be automated integration, nor can it be used for large volume datasets that cannot be copied due to their volume. Even in a mashup, wrapper terms used in the metadata are themselves subject to interpretation, making reuse of data elements difficult.

Without metadata referenced to well-understood standard terminology applicable across domains, the diverse datasets cannot be integrated automatically. Furthermore the integrating elements must be applied outside the dataset storage, implying that the integration logic must reside in the metadata layer.

IV. SEMANTIC TECHNOLOGY

Semantic technologies are crucial for the future handling of big datasets across multiple domains. While we have methods for unique concept identification arising through the Semantic Web, these technologies have not made inroads into traditional data management systems. Traditionally the ETL process has been used to enforce standard terminology across datasets, with foreign keys to external tables for the related information. This is not a scalable solution, since the introduction of a new data source requires the careful construction of foreign keys to each other dataset in the database. This lack of extensibility to add in additional sources highlights the limitations of horizontal scalability in current approaches. In addition there are limitations on the continued expansion in large data warehouses, highlighting their inability to continue to scale vertically.

Semantic technologies have not yet made inroads into Big Data systems. Big datasets that consist of volume tend to be monolithic, having no attempt to integrate across datasets. The data is typically stored in its raw state (as generated), and in the initial big data engineering no joins were allowed. Given this, most Big Data Analytics approaches apply to single datasets.

For solutions addressing the integration of variety datasets, the ability to integrate the datasets with uniquely defining semantic technology is thus a fundamental requirement. Two overarching requirements need to be addressed to use ontology for the integration of big data: the construction of the ontology, and the use of the ontology to integrate big datasets.

Ontology scaling

The standard method for data access through an ontology is to ingest the data into an ontological database, where the data elements are encoded along with their extant relationships. This does not work in a Big Data scenario, since ontological databases do not have the horizontal scalability needed to handle data at high volume, velocity or diversity. Further exacerbating the problem is that some of the data needing to be integrated is not owned by the analytical organization and cannot be ingested but only accessed through query subsets.

Separate ontology for metadata

The implementation of an integrating ontology would consequently need to reside in the metadata for browsing and querying. While this metadata could be browsed manually, the real value comes if it can be actionable; such that selections over the metadata ontology would automatically construct queries to the big data repository. A number of ontologies relative to the cyber domain already exist, encompassing resources, attack event ontologies, etc. The key is to incorporate the appropriate elements and their relationships needed to describe the elements in the desired datasets. Our intent is not to recreate a cyber ontology from scratch, but to leverage those that exist to develop a first order ontology specific to the integration of the relevant cyber datasets. Focusing on first order logic will enable the ontology to be actionable to dynamic data integration.

In order to serve as the facilitator for data integration for automated integration, this first order ontology would need to contain elements such as:

- Data Element definitions
- Dataset location
- Data Producing Resource characteristics
- Resource connectivity

For analytics, additional mid-level ontologies would be needed to provide reasoning over the data, such as time and location. Domain specific ontology elements would include for example resource attributes by resource type, translations such as IP to location, and derived attack pattern components.

The key to the use of a semantic representation for the metadata is to separate the semantic metadata from the data storage. In order to leverage the scalability and speed of high-volume NoSQL solutions, the ontology will need to reside in its own scalable environment. Data exploration would thus require a
mechanism to browse the metadata within the ontology, with a seamless transfer mechanism to flow down into the data.

**Probabilistic Challenges**

One significant challenge in the use of ontology for automated data analytics across datasets resides in the need for probabilistic reasoning. Typically in ontology representations, triplets are considered “facts”, implying full confidence in the data elements being described. In the real world, such luxury is typically non-existent. Resources will continually be updated, and there will be a latency prior to the new configurations being updated in the ontology. Attack chains will have multiple possible paths with probabilistic representations of each link type. Activity counts must be evaluated with a statistical significance test to determine if an activity is truly of concern. Such counts will have variations relative to time-of-day and day-of-week. Using an ontology for such probabilistic analytics will require the ability to analyze activity under some uncertainty. Much work has been done on probabilistic ontology like MEBN which inserts Bayes theorem in ontology nodes [1].

V. APPLICATION TO CYBER-SECURITY

The goals of big data applied to cyber-security are generally to improve the effectiveness and timeliness in these four categories of activities:

1. Identity Management
2. Fraud Detection & Prevention
3. Governance, Risk & Compliance
4. Security Management

For the later, security management, one sub-goal is to attempt to achieve near real-time awareness of security inside and outside of a networked enterprise, and another is just-in-time (JIT) response to attempt to prevent loss from a split second attack. Perimeter security can be improved, as part of a defense-in-depth approach, by JIT blocking and filtering. Traditional “after-the-loss” scans and forensics are the status quo in the industry. In this research, we aim to prevent loss by JIT awareness and response. Actionable data can decay in seconds, and losses can occur in seconds, so it clear that loss prevention requires JIT, if not near real-time, response.

We believe that Big Data analytics combined with near real-time situation assessment and JIT response will provide an additional level of defense-in-depth. Traditional “firewalls” imply filtering of IP addresses and malware patterns that are updated manually as new information is collected. Big Data will provide more complete and more timely information and knowledge that makes it possible to correlate, analyze and use the data in near real-time to do loss prevention actions; automatically in some situations. These actions typically include:

- Incoming message block
- Outgoing message block
- SSL inspection Initiation (encrypted channel)
- Terminate connection
- Quarantine data or hardware unit
- Lock user’s account
- Notifications

Examples are: (1) near real-time update of a firewall with new IP addresses or URLs, for immediate update into the firewall for filter/block, and (2) near real-time (NRT) termination of a session (port cutoff) as soon as it is suspected that it is a connection established by a bad actor, or is infiltrated. Surgically precise NRT responses can maximize the prevention of loss of confidentiality and integrity, while minimizing loss of network availability to others that are not directly involved.

To illustrate how this would work in practice, consider these three scenarios.

First, logs collect data on all machines with open connections to locations outside of the enterprise, with a JIT analysis including the variety of additional context datasets. One connection is open to China, but the attributes of the particular user connecting imply they are not supposed to connect anywhere in Asia. Within split seconds the command is given by the system to cutoff the port and a security violation is logged for follow-up.

Second, consider a user who initiates an identical second software application, as determined by an identical hash. This is a specific type of pattern, wherever an identical hash is observed in multiple places it is an indication of a compromised account. In
the NRT response the connection is terminated/cutoff, and the data and memory images are quarantined and saved for forensics. Off-line cyber forensics is scheduled, the account's user-password is disabled, the user is notified to call for a new password, and the situation details are preserved for future reference; and for potential analysis and learning of bad actor behavior.

Finally, consider that analysis of the data results in a discovery that a server has ten times more .avi files than typical for a server of that type, which is indicative of a user setting-up and running an unauthorized file server. The user’s account is disabled and she is notified to call in, the files on the server are quarantined and preserved for forensics, and a forensic investigation is scheduled.

Some of these examples of potential NRT response, and some are JIT with man-in-the-loop; they are notional for conveying general concepts and they are not necessarily recommended for implementation.

For practical application now, we envision that the following questions can be answered with properly implemented Big Data technologies that span the variety of datasets:

- What data is available on malware X attacks globally?
- How many machines did an event land on?
- What ports were leveraged?
- What users effected?
- What machines were compromised?
- What was leaked?
- Sensitive info?
- Insider or Outsider?

More difficult questions for the future would be:

- What should I expect from this attacker within the next hour? Next week? Next month? (Based-on historical data on this attacker.)
- What unsafe actions are my users doing, rank ordered by risk significance?
- What suspicious activity occurred today?
- Tabulate statistics on Vulnerabilities versus Attacks, and visualize the results.
- Where is the greatest risk within the enterprise?

The latter “future set” of questions requires substantially more research and development in topics like machine learning and reasoning, and is well beyond the scope of the current paper. For example, can ontology as proposed here help us reason about risk based on the topology of devices and controls; theoretically this is deterministic and machines should be able to do better than man. Our intent is to model perimeter security of a large, enterprise network and collect real-time data, reason about risk in real-time based on the topology of devices and controls, and respond to threats in a just-in-time to attempt to prevent loss. Given the appropriate set of data and generation of a set of reasonable hypotheses, can we use Big Data to do evidence collection to support or refute those security risk and threat hypotheses, in-time to prevent loss?

**Approach Summary**

A good overview of cyber ontology engineering can be found in [4]. We begin with a canonical cyber-security model focused on malicious activity – the Diamond Model shown in Figure 1.

The four corners of the diamond, Victim, Infrastructure, Capability, and Actor (the one threatening the victim), account for all the major dimensions of a malicious cyber threat.

![Figure 1. Diamond model of cyber-attack showing bad actor and victim, the infrastructure of assets with a networked enterprise, and capabilities that bad actor uses to victimize.](image)

**Top Level Conceptual Model**
For our purposes here the above model is modified to include a cyberwar aspect. First, as with the previous model, a bad actor has capabilities to do malicious things to the network:

- steal information (loss of confidentiality),
- corrupt or destroy information (loss of integrity),
- disrupt or prevent a good actor from having access (loss of availability).

In addition is the good actor who counters the back actor. When the good actor ensues one of the three above types of loses we refer to her as “victim”. A good actor uses cyber defense capabilities to counter the bad actor, to protect her assets, as illustrated (Figure 2).

The seven core capabilities of Bad Actors are:

- Reconnaissance; Find opportunity using social media, spyware, penetration testing (pentest)
- “Weaponise”; Select/develop malware
- Deliver; Payload to destination
- Exploit - related to reconnaissance, e.g., keylogger to capture keystrokes and therefore passwords
- Install Malware; Compile/run-Time
- Command & Control; Call home/get orders
- Action; Transaction leading to the victim’s loss of confidentiality, integrity, availability

Twenty common controls have been identified in [5]. For the purposes of this research we have consolidated this to 9 common controls, plus JIT response as a key capability of good actors in responding in time to prevent loss rather than after-the-fact. These good actor capabilities are:

- Access control – specifying who, what, when and where of access
- Backup & restore - storage of system images periodically, for protection against loss
- Secure coding techniques - and collection of data from applications including exceptions and errors
- Configuration Management data
- Continuous monitoring events inside system awareness, and outside situation awareness (perimeter)
- Firewall & Deep Packet Inspection logging of all positive hits
- Inventory of all devices and software, in detail
- JIT response - actions to be performed in respect to current situation before a loss is incurred
- Risk Assessment - valuation of the assets and respective probabilities of loss given the controls
- Training - personnel data from questions-answers, training profiles with dates and assessments of training needs

In recent years the continuous monitoring control has taken on a special meaning for cyber compliance; here we broaden this to mean near real-time continuous monitoring to maintain comprehensive system awareness inside the enterprise and situation awareness outside of the perimeter (Internet). This entails:

- Discovery and inventory of software/hardware that can impact/effect cyber-security
- Monitoring all enterprise networking devices for configuration compliance with documented policy
- All devices with a known vulnerability
- All security related events from hardware and software

![Figure 2. Good-Bad Actor “cyberwar” Model](image-url)
• All threat vector data from inside and outside of the network, at all kill chain stages [2]
• All known malware patterns, signatures/ footprints
• All open source and human intelligence related to cyber-security (OSINT/HUMINT)
• Events from all logs related to cyber-security, including access control logs, intrusion detection and prevention logs, firewall logs, deep packet inspection logs, and workstation and server system logs.

The threat level, also known as the defense condition DEFCON level, is determined by cyber-threat situation assessment in relation to system awareness [7], and is primarily determined from cyber intelligence collection, of which there are two primary types:

• Intelligence, gathered by intelligence agencies by people in the field (HUMINT) and open source (OSINT) from sources like the web, and
• Network events/alerts collected by network sensors primarily located in the perimeter, also known as the demilitarized zone or DMZ [8], and inside the networked enterprise.

The Goal: Precision Response

As mentioned above, one of the most interesting aspects of ontology in cyber-security Big Data is its potential contribution to JIT response. Another is precision response – an absolute requirement in many situations to avoid loss without causing loss of availability to others. It is an open research question as to how precise, how real-time and well in general this idea would work. Based on the authors’ intimate knowledge of cyber-security in large networked enterprises, we hypothesize:

Just-in-time automated decision making from big data and automated response would be (a) more effective at preventing loss (CIA) than without it, and (b) the delays in availability would be relatively short, i.e., acceptable to the user population.

To this end a well-conceived and implemented cyber ontology could potentially use formal logic [12], or probabilistic formal logic [1,2], to infer situation awareness from an appropriate Big Data collection, and to infer just-in-time decisions for precision response. The goal is to minimize loss and maximize overall availability for all otherwise unaffected users. This of course would slow down messages for the effected users, because suspect emails, webpages and connections would be quarantined or delayed until the situation could be fully assessed by cyber-security specialists or forensic teams as appropriate; this is simply the cost of security. Streamlined workflows can minimize the time for quarantine to resolution, such as automatic routing of tasks like forensic investigation of potential damage from a suspected malware occurrence. Ontology with formal methods automation appears to be a viable solution for just-in-time decision, planning and response; the authors do not know of any viable alternative. Therefore we hypothesize here that a formal ontology approach is the future of cyber-security. Taking this a step further, we need to identify an architecture that would most likely provide the kind of real-time and JIT performance that is necessary to make this viable. Many just-in-time response requirements/situations are unique, but also, much commonality exists. For example, after intrusion there is a high probability that the attacker will try to set up a secure connection [SSL] to provide an encrypted channel for stealing data or other malicious activity; or attempt to get control at the kernel level to hijack the CPU; or attempt to open a command prompt to give commands to corrupt the system, like deleting hard disk or planting malware for later use.

In these situations it is important to know the signature of the attacker, her profile and previous
behavior patterns. We are learning that Big Data provides a wealth of previously unharnessed information. If this information exists relative to an attack, it can be used in predicting the next step in the attack. These are known as the *Kill Chain* steps, and they are described in sequence in detail in [3].

Figure 3. Potential application of formal ontology and big data to just-in-time (JIT) decision, planning and response.

Given an attack and prior information, the question then becomes one of 'how much time before the attacker will be successful' – how soon must something be done to prevent loss? This expected time lapse can be estimated in advance by considering what the attacker must accomplish at each step in the kill chain, and how much time they take based on historical data on attacks, e.g., the Bayesian probability of time to next step based on time between steps in previous attacks. (See [1,2] for an example of Bayesian probability.) Factors that affect lag-times include the status of the target device and data being targeted. If the attacker is targeting a known vulnerability such as a security hole in Windows 7 on the workstation where the attack is happening it will likely be seconds or less before potential loss, but if a patch has been applied to that system the night before, then the log file for that system, collected into the Big Data, is available to the ontology for potential inference that the risk is low and no response is warranted. If the patch has not been made the inference is 'high risk' and the system should be immediately quarantined until resolved. This of course might cause a significant inconvenience to the user, but that is a minor consideration compared to a high valued loss. It is obvious that this is complex and it will require adequate resources to implement. Whether it is worth the cost of implementation can be addressed by a typical risk assessment [10]. It is the belief of the authors that the potential costs of losses are orders of magnitude higher than cost of this implement. One estimate is that global cost of cybercrime is $1 trillion [13] and the amount of cybercrime is increasing – the time to address this problem is now. The cost of a system solution is a one-time expense compared to the ongoing losses without one.

To see this clearly one has only to compare this JIT Response approach to the types of processes done currently in most networked enterprises today: situation awareness data feeds are collected by a cyber-security operations center (SOC). There the data is processed manually and decisions are made off-line, often hours, days or months after loss. When a decision is finally made it might result in an IP Address being added to a firewall or other remedial action. These after-the-fact fixes are inadequate and cannot continue if losses are to be prevented. Can advanced technology like proposed here radically change our response capabilities? One thing for sure – we will not solve the problem if we don’t do this kind of hypothesis-and-test of ideas, and collect evidence to support or refute hypotheses like the ones suggested here.
Elicitation of Cybersec Knowledge

How hard, how expensive, how time consuming, would it be to collect this knowledge and implement it in an ontology with probabilistic reasoning and formal methods? Clearly the technology has advanced to the point of practical application. Our efforts have already made substantial progress in establishing the first principles necessary to build out the top level model shown in Figure 2. Hundreds of the core concepts have been identified.

As a first step in preparing to instantiate an ontology we have been mindful of what hundreds of organizations do in the current cyber-security management process in a global networked enterprise. Description of this workflow is beyond the scope of this paper; it suffices to say that system awareness currently resides in the minds of hundreds of professionals who track threats and malware, maintain the security devices like firewalls and the configurations and patches of thousands of network devices, monitor events and log files, create tickets when an anomaly is observed and perform remedial actions – termed Incident Response, Configuration Management, Vulnerability and Patch Management, Firewall, Intrusion Detection and Prevention, Deep Packet Inspection and Cyber Threat Assessment, Security Architecture and Design etc.

We propose to elicit all knowledge necessary for JIT assessment, decision, planning and response into this ontology. On first glance this may appear daunting, but based on the successes with ontology engineering in recent years, and the high stakes, we believe this not only practical, but necessary, to better understand how to solve this national priority problem.

Cyber-security management has the characteristics of a successful knowledge elicitation and ontology engineering endeavor. The information is in digital form, and cyber-security processes are repetitive - meaning that the same indications of an attack are well documented and observed in typical network operations routinely, and the remedial steps are documented and used routinely. This is not to say the cyber-security experts are not highly knowledgeable and skilled – just the opposite. This knowledge can be coded and reused in the parts the machine does best; man should continue to do the parts that she does better than machines. With this expectation we will meet of goal stated up-front of flipping the current situation to one where defense of the network is optimized and efficient, lowering cost of defense, and making it very hard and expensive for the attacker.

There are areas of cyber threat assessment within Big Data analytics that have not been fully explored. For example, Twitter has received much attention in recent months. What would we find in the tweets if we were looking for malware and cyber-attacks, and related that to all the knowledge we have to-date, including information on bad actors and their motives? Would link analysis provide better knowledge of cyber-criminal groups and their plans to launch cyber-attacks (where, when, how, who, what target, etc). As with all applications of twitter data there is much noise and many meaningless links, and this would certainly be the case here, but all we need are a few ‘golden needles from the haystack.’ This is only a tiny example of what Big Data analytics entails.

Within the networked enterprise, and beyond the perimeter in the Internet, every device is capable of creating events. Most large networks already capture the data from these events, usually from log files. For example, SPLUNK is a Big Data solution that was invented to capture network events, and is rapidly being adopted worldwide for this and other purposes. We currently use SPLUNK with a large population of diverse types of events, but for this approach to work properly we will need to capture even more events and data. To effectively counter a cyber-attack we need to collect available data related to all aspects of cyber-attack and defense (see Figure 5).

Essentially Big Data analytics, from a variety perspective, provides the technology that makes possible a collective memory of bad actor behavior and pro-active steps, successful and unsuccessful, in defending the confidentiality, integrity and availability of the networked enterprise and all of its assets. This data would include bad actor profiles, and all prior attempts that were foiled, or not, in the kill chain. It would also be a collective memory of forensic methods and results of intrusion and after-loss findings.

A successful demonstration of cyber-attack prediction and detection is of very little value without JIT response to prevent losses. A meaningful demonstration needs to include decision and planning to enable JIT response. In many cases the response requires a work flow that must be established from course-of-action planning to enact more than one step in sequence.

The remainder of this paper is focused on an approach to improve cyber situation awareness and response through Big Data integration with ontology, analytics, and machine reasoning for enabling just-in-time response, before loss is incurred from an attack. To this end: (1) a systems approach is developed, (2) an ontology is engineered for use in the system,
(3) a prototype demonstrates machine reasoning. However, the common logic used in the prototype would need to be transformed in a much more near real-time (NRT) solution, but this should be achievable using the results an approach similar to [18]. The important take away from this paper is that a system for JIT Cyber Attack Management is more of an engineering endeavor than research—it can and should be built now to manage the threat that is exponentially growing out-of-control [16]. More can be done to improve network design and architecture for marginal gain in cyber security, but the problem cannot be fully addressed without also having the near real-time awareness at the perimeter and JIT response as advocated here.

VI. CYBER-SECURITY MANAGEMENT SYSTEM

The system concept being described here is shown in Figure 4.

Figure 4. Four primary functions (left) to identify suspected and confirmed attacks, and respond; system does machine-best-at part and dashboard used for man-best-at part; DS (bottom) short for decision support.

This conceptual design consists of the integration of a diverse Big Data capability, including system awareness and situational awareness.

Figure 5. Wide variety of Big Data useful to Cyber-security Management in two categories: System Awareness (inside networked enterprise) and Situation Awareness (inside and outside of the network perimeter).

An architecture for Big Data and ontology used in hypothesizing cyber-attack, and used to correlate data and provide corroborative evidence for testing hypotheses is depicted in Figure 6.

Figure 6. Notional dashboard with visualizations related to a cyber-attack; shows just-in-time situation awareness with a variety of evidence from Big Data supporting a hypothesis of a widespread attack; all data values shown are fictitious and meaningless.
A process for cyber-attack management is illustrated in the flow diagram in Figure 7. Big Data analytics and data mining is the driver, operating 24/7 in a security operations center. As suspicious events are discovered hypotheses (H) are generated. Further evidence is collected from the Big Data and by semi-automated forensic investigation (strictly manual is too slow to prevent loss in many cases), until it is warranted to infer that an attack has been confirmed (evidence supporting attack > rebuttals), or until the evidence indicates it is not an attack (rebuttals > evidence supporting attack) and the evidence collection and hypothesis test process is terminated for that instance i (H_i). The security operations center's specialist is notified, as indicated by the icon (red circle with T inside). If H_i is accepted then the risk assessment begins for that H_i. Risk is the impact of the attack at that specific location in the network times the probability it will succeed (R_i = I_i * P_i). Several factors are important in risk assessment in this context, because the impact depends upon what assets are reachable from that point in the network where the attack is taking place and the status of the devices at that point in the network. For example, if a device is being targeted that has all software patches up-to-date the risk is much less than if they are not. In the latter case a known vulnerability exists on the targeted device that the attack can exploit and success. Also it depends upon the point in the kill chain: at the beginning of the kill chain during reconnaissance the risk is much less than it is the last step of setting up an encrypted tunnel (SSL) to pass data back home. There are many other factors too numerous to enumerate here.

If the risk is greater than a pre-established threshold, or a decision is made by the man-in-the-loop, then a course-of-action planning process is initiated that can result in multiple steps in a prescribed sequence, and including the alternative plans that suffice in countering the attack. The plan that is selected is one that is optimal in terms of a cost/benefit ratio. If no sufficient plan is generated or if no plan is below threshold on cost then the H_i is terminated, and the security operations center's specialist is notified. If a sufficient plan is generated then it is executed. Man is in-the-loop at all steps in the cyber-attack management process and she can intervene at any time.

VII. ONTOLOGY ENGINEERING

Development of a cyber ontology needs to be carefully focused and right-sized, because full cyber ontology development is a huge and largely unnecessary undertaking. It is only necessary to develop a cyber ontology to support our system objectives and requirements: Big Data integration for NRT awareness at the perimeter and JIT response. For this reason many aspects of cyber ontology can be omitted for our purposes here. Our ontology includes only requisite ingredients from the following:
- Agent Ontology, e.g., Person, Organization, Sensor
• Role Ontology, e.g., Data Consumer, Cyber Threat Analyst, Hacker, Malicious Actor, Asset Role
• Capability Ontology, e.g., Access Control Capability, Continuous Monitoring Capability
• Infrastructure (Artifact) Ontology, e.g. Communication Network, Computer Disk, Computer Memory
• Big Data Process Ontology, e.g., Data Curation, Mash Up Process, NRT Data Stream, Data Analytics Process, Network Intrusion Response Process
• Cyber Attack Event Ontology, e.g., Confirmed Phishing Incident, Hacking, Data Loss, Data Theft, and
• Information Content Entities, e.g., Malware Pattern, Access Control List, Cyber Incident Report, Forensics.

Our first step in ontology engineering was to examine the principles of cyber-security and attempt to identify the canonical form of this cyber-security problem, in context of cyber-attack management. Several hundred core concepts were identified and represented in set notation for ease of manipulation. All are related to the top level model illustrated in Figure 2; a small sample is shown in Figure 7. Of all these concepts, the only ones selected are those necessary/required for ontology engineering to achieve a proof-of-concept demo of the primary functionality of the above system.

The top level of our ontology is shown in Figure 9, and the three sublevels are shown in Figures 10, 11 and 12.

Figure 9. Upper level of the Cyber Ontology.

Figure 8. Examples of sub-concepts in set notation [7].

| System Awareness | { devices | events | performance } |
|---|---|---|
| Situation Awareness | { threat vectors | malware | bad actors | behavior patterns | anomalies } |
| Precision Response | { Blacklisting | Blocking/Filtering | Access Control | Intrusion Prevention | Port Cutoff | Address Translation | Account Disabling | Whitelisting exceptions | Data Loss Prevention | DEFCON Level Response } |
| Blocking/Filtering | { SPAM | malware | threat vectors | Blacklisted IP addresses & URLs } |

Figure 10. Lower level ontology on bottom left of Figure 9.

Figure 11. Lower level ontology on bottom middle of Figure 9.

Figure 12. Lower level ontology on bottom right of Figure 9.

Figure 13. Big Data is integrated on demand via instantiation with ontology to provide awareness of cyber attacks and security/defense posture.
The ontology is core enabler of the required functionality necessary for the system above. Although beyond the scope of this paper, our goal is a proof-of-concept prototype of the entire process, but only for a few appropriate types of attacks and respective plans as defined by a fairly rigorous test set. Big Data elements for proof-of-concept have been partially selected from the types of data illustrated in Figure 5.

Ontology engineering tools are being evaluated for “most suitable” for implementing this ontology for use in the system as described above. A trade study will need to be conducted for tools can be selected for implantation of a production system capable of meeting the above objectives in a large, global enterprise network. For the purpose of demonstrating the concept we have selected a tool from HIGHFLEELT.com that provides for first order logic within ontology; a tool that one of the authors has used successfully in the past. Some research questions are overviewed in the next section, and they need to be addressed before all of the factors are established, along with appropriate metrics, to conduct assessments for “best tools and techniques” for production grade deployment.

Due to page limit constraints it is impossible to discuss all aspects of the cyber ontology development here, but it is a few aspects need to be mentioned here. For example, there are many good resources for specifying and instantiations these ontologies to a level useful in CAMS; most notable are efforts by MITRE [5]:

- Malware Attribute Enumeration and Characterization (MAEC)
- Cyber Observable eXpression (CybOX), a structured language for cyber observables

And another is an effort begun by Mary Parmelee of MITRE: a semantic framework of loosely-coupled modular ontologies built upon the Security Content Automation Protocol (SCAP). SCAP is a suite of specifications that standardize the format and nomenclature by which security software products communicate software flaw and security configuration information:

- Open Vulnerability and Assessment Language (OVAL), a language for representing system configuration information, assessing machine state, and reporting assessment results.
- Common Platform Enumeration (CPE), a nomenclature and dictionary of hardware, operating systems, and applications.
- Common Configuration Enumeration (CCE), a nomenclature and dictionary of security software configurations.
- Common Vulnerabilities and Exposures (CVE), a nomenclature of security-related software flaws.

An important aspect of the above is interoperability across automated security systems based on the above OVAL, CPE, CCE and CVE standards, and this is key to the Cyber Attack Management System proposed here.

VIII. Near Real-time Awareness & Control

In-depth discussion of architecture and design at very high Internet backbone bandwidth is a topic beyond the scope of this paper. However, the above proposed system cannot be implemented within a large enterprise network interfacing the Internet, without hardware and software capable of functioning at near real-time at that speed (>10GB/sec). Many network devices, and security devices in particular, are capable of interacting CAMS at NRT. One network appliance (hardware and software) that can scale to this speed and volume of network traffic is the Cloudshield, the CS-2000, a multi-function solution appliance [15, 19].

Several features of the CS-2000 make it feasible CAMS to achieve the type of real-time awareness and control: (1) it is not pre-configured network capability set programmed into the system, making it a viable candidate for application of CAMS code, (2) it allows rule sets to be defined and processed in NRL that in conjunction with CAMS can analyze, make decisions, and take action on packet data received from the network. For example, it can be configured to execute inspection of any packet data, capture of portions or all packet data, modification of packet data, insertion...
of new packets, drop or discard of packets, and algorithm processing. These heuristics (rules) are coded by a programming language, called RAVE, that is designed to make the development of packet processing policies and applications easier. Decisions and actions can carried out upon network traffic including control or altering of traffic flow, stateful filtering of packets, and deep packet inspection.

Figure 15 shows the Deep Packet Processing Module (DPPM) which includes its own processors, memory, and physical interfaces; it implements a RAVE execution engine to carry out the logic defined by a loaded RAVE application. Each Up to two DPPM can be configured to independently execute different RAVE applications.

The physical computer blade that provides management access to the system is called the Application Server Module (ASM). The ASM is physically plugged into the same chassis enclosure as the DPPM blades. The ASM includes its own processor, memory, disk storage, and physical interfaces. There is a single instance of the ASM in the evaluated CS-2000 configuration used to manage DPPM blades within the same CS-2000 chassis:

- CAMS can send rule sets that permit or deny information flows, and modify user attribute values, over a TLS protected channel
- CAMS can issue commands via an application programming interface (API), referred to as GODYN that provides for dynamic data update/control of the CS-2000, or the JSON (JavaScript Object Notation) API can be used to manage the same information.

The above provides adequate capability to demonstrate a CAMS proof-of-concept prototype achieving JIT response to prevent loss in cyber attacks.

IX. RESEARCH QUESTIONS

Our achievements-to-date has been relatively easy given the maturity of the requisite technology; nothing has been outside of routine engineering. However, cyber security is a hard problem and it is doubtful that the approach taken here, or any other, will be a complete solution. Furthermore, the capabilities of CAMS will need to rapidly advance with the rapidly advancing sophistication of cyber attacks [16]. The research questions are reported separately in the following categories [20]:

- Big Data and Analytics
- Ontology and Probabilistic Reasoning
- Just-in-Time Decision Making
- Design and Architecture

X. FUTURE STEPS

We are planning further research and development, beginning with the Big Data Analytics necessary to more fully identify, understand and respond to cyber attacks. In parallel we would
like to develop a proof-of-concept prototype that will result in a detailed design of this cyber-attack management system complete with all the requisite parts of the ontology and Big Data integration. The key to the success of this prototype will be to focus on one narrow aspect of cyber-attack-defense; if one is successfully implemented and demonstrated it can be used to extrapolate the resources needed for development and implementation in a large production environment.

Our next steps are to

- Select a subset from the core concepts
- Select an ontology engineering tool
- Design an empirical assessment
- Develop a rigorous set of test cases
- Select the Big Data elements
- Implement a proof-of-concept prototype including ontology component
  - Develop ontology
  - Populate the prototype
  - Conduct the assessment
  - Report the results

For this work to be meaningful to an eventual production level deployment it needs to be conducted in close cooperation with a large enterprise network operation to (1) learn-from and utilize real attack-defend related Big Data and (2) glean the performance and functional requirements demanded by a high performance operating environment.

REFERENCES

SAIC is a premier technology integrator providing full life cycle services and solutions in the technical, engineering, intelligence, and enterprise information technology markets. SAIC is Redefining Ingenuity through its deep customer and domain knowledge to enable the delivery of systems engineering and integration offerings for large, complex projects. SAIC’s approximately 15,000 employees are driven by integrity and mission focus to serve customers in the U.S. federal government. Headquartered in McLean, Virginia, SAIC has annual revenues of approximately $4.3 billion. For more information, visit saic.com.