Network Security Data Analytics Architecture for Logged Events

Mark E. DeYoung
IT Security Lab
Virginia Tech

Randy Marchany
IT Security Office
Virginia Tech

Dr. Joseph Tront
Dept. of ECE
Virginia Tech

Abstract

Data-driven network security and information security efforts have decades long history. The deluge of logged events from network mid-points and end-points coupled with unprecedented temporal depth in data retention are driving an emerging market for automated cognitive security products. Historically, new technologies like this are delivered as non-contextualized black boxes. We frame network security data analytics within the context of intelligence activities and products and go on to propose network security data analytics as a framework to develop and evaluate cognitive security products that can satisfy operational needs. Finally, we discuss functional design requirements, limiting factors, and initial observations.

1. Introduction

Data-driven network security history stretches back to statistical anomaly detection in Dorthy Denning's 1986 Intrusion Detection Expert System and continues through the late 1990s with Peter Vaxon's work on Bro Network Security Monitor and Martin Roesch's SNORT. It is extended through the 2000s to today with a wide variety of commercial end-point and mid-point security tooling that includes signature based, threshold based, rules/heuristics and anomaly detection techniques.

Unfortunately, tunable parameters of techniques are often masked. Vendors pack unknown methods into black boxes labeled as intrusion detection systems (IDS) or intrusion prevention systems (IPS). Both produce alarms and events that are fed into security information and event management (SIEM) systems for filtering, correlation and eventually review by human analysts.

2. Problem Domain and Solution Domains

Contemporary network security introduces new challenges. Rapidly growing volume, velocity, and variety of data is frequently used to characterize ``Big Data''. A flood of events from network mid-points and end-points can rapidly overwhelm over-tasked security analyst. Inexpensive high capacity storage has enabled the collection, aggregation, and retention of event data with unprecedented temporal depth and volume[1] [2]. Data diversity has also accelerated, internationalization and localization efforts have vastly increased the variety in data encoding. System events are generated in diverse dialects of semi-standards like syslog[1] and several generations of Microsoft Windows event logs[3]. Network security practitioners must seek out new methods.

The problem domain of network security has historically drawn solutions from statistical models and computational algorithms. The explosive growth of big data presents an opportunity to apply automated cognitive techniques drawn from Artificial Intelligence related data mining, machine learning, and natural language processing. As shown in Figure 1 we seek to apply a broader solution domain that includes data science approaches with the requirement of an underlying information technology (IT) infrastructure.

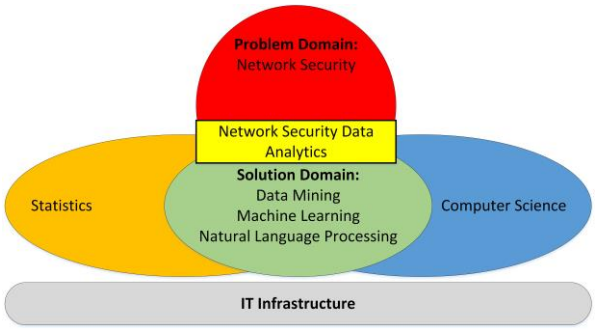

Figure 1 Network Security Data Analytics

Adoption of encryption for data-in-motion and data-at-rest is increasing[4]. Unfortunately, encryption reduces the analytic utility of traditional monitoring systems and forensic data collection. There is little point in collecting network packets or web traffic that will never be decrypted for analysis. An alternative is to collect and centrally aggregate observations of systems behavior from logged events. Event logs are essential to system health monitoring, forensics, and regulatory compliance[3]. Log collection, management and aggregation has gone from an

[1] e.g. RFC 3164 *The BSD Syslog Protocol* is obsoleted by RFC 5424 The Syslog Protocol. Implementations of both standards are in active use.

"untapped market" in 2006 to one with a "Top 47 Log Management Tools" in 2014[3], [5]. Logged events could become the foundational data for automated security analytics. The automation of cyber security and cyber defense by cognitive means is extremely appealing. Automated monitoring and threat analysis could speed responses to cyber threats beyond human reaction times. It gives tantalizing glimmers of agile and rapid cyber defenses. There is already a rapidly emerging market for commercial cognitive products. The cyber security market is predicted to grow from $77 billion in 2015 to $170 billion by 2020; with security analytics comprising $2.1 billion in 2015 and $7.1 billion in 2020[6]. The historical, and dangerously unstated, risk is that commercial cognitive security products will also be delivered as non-contextualized black boxes (e.g. vendor does not define throughput limitations, indicate detection accuracy, or characterize countermeasures for adversarial learning).

A "Holy Grail" is to automatically produce intelligence artifacts that enable people to make informed decisions about cyber security events. An unstated risk is blind adoption of sophisticated techniques that are not relevant to a contextualized operational environment. To this end we are designing a network security data analytics platform to support the development and evaluation of repeatable and reproducible network security methodology that produces interpretable results.

3. Intelligence Production Frameworks

Security requires an intelligence production methodology. Similar intelligence production frameworks are used to inform marketing, business, law enforcement, and military organizations. At a strategic level we can frame network security data analytics with the context of activity groups and product categories defined by existing intelligence production frameworks. Additionally, we can decompose network security data analytics into related activities of collection, processing, analysis, and dissemination. Another perspective is the successive refinement of observations from an operational environment into actionable intelligence products[7].

We combine the perspectives (product categories and activities) in Figure 2. Core operational requirements and considerations for the collection, processing, and analysis phases are shown in Figure 2. In this paper we will address the collection, processing and analysis phases. The blocks labeled "Planning and Direction" and "Dissemination and Integration" are more administrative in nature and are not explicitly included in this study.

Collection is the gathering of raw data from the operational environment. For the purposes of network security analytics the raw data under consideration is log events produced by mid-point and end-point systems. Examples of data sources include: system events logged into syslog or Windows event logs; alerts from IDS, IPS, and SIEM systems; network flow data; and compliance scan results. We will focus on logged events throughout this paper with the assumption that many raw data sources are trivially converted to time stamped sequences of semi-structured events in logs.

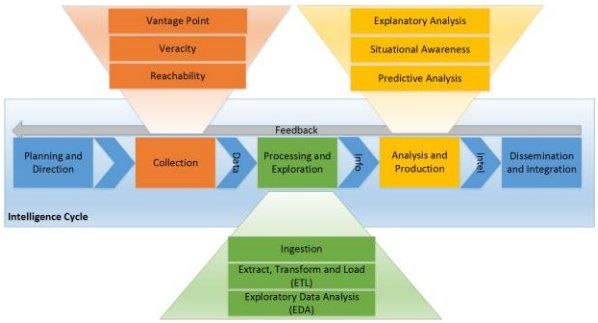

Figure 2 Intelligence Cycle – activities and product categories.

Processing and exploration is often about learning *of* events to develop an understanding of their syntactic structure, encoding, and feature characterization. Loading data in a normalized format for preliminary exploratory analysis is one of the first challenges in processing and exploration. Because log events and logs are produced in a wide variety of formats we need methods to Extract, Transform, and Load (ETL) data. Transformation and data normalization is essential to prepare data for analysis. There is a wide variety in data encoding, syntactical structure, and feature characterization even in a widely accepted formats like syslog[8]. Exploratory Data Analysis (EDA) can range from classical statistical methods to automated processing with data mining or machine learning techniques[9], [10]. EDA is "an analysis approach that focuses on identifying general patterns in the data, and identifying outliers and features of the data that might not have been anticipated"[11]. Analyst often overlap ETL and EDA tasks and frequently lump the task set together as "data munging". Some data munging tasks include: renaming variables, data type conversion, recoding data, merging data sets, inputting missing data, and handling erroneous values.

Analysis and production are more focused on learning *from* events. There are several relationships between multiple events that must be considered: spatial, temporal and semantic. Some temporal categorizations are:

- **Retrospective** explanatory analysis; "what happened?"

- **Concurrent** situational awareness; "what is happening?"
- **Prospective** predictive analysis; "what will happen?"

There are also spatial relationships and semantic relationships that should be considered. For example, in Figure 3 we show a device that is moving across the state of Virginia. At 8:05 a file was opened in Blacksburg, VA (retrospective) in the present at 9:05 the device is in Roanoke, VA and has just finished reading the file (concurrent); then in the near future, 9:10 the device closes the file while it is located in Lynchburg, VA (prospective).

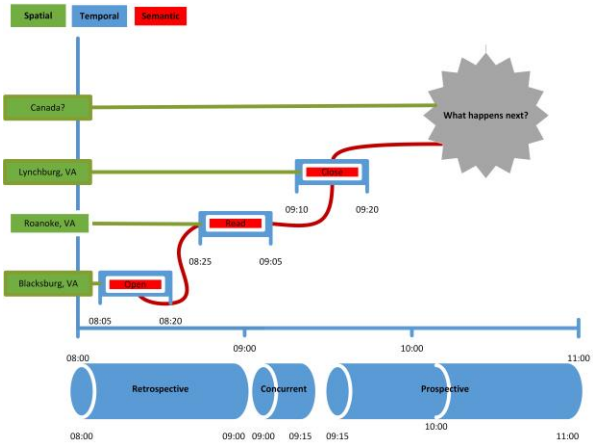

Figure 3 Inter-event relationships

Largely, we are attempting to detect and report on harmful anomalous conditions. Analysts conduct retrospective forensics, monitor dashboards for situational awareness, and in some cases attempt prospective analysis. Primary tasks for network security analysts are: detection of anomalous conditions; evaluating anomalies detected by IDS, IPS and reported through SIEM systems; and recommending remediation actions. This is problematic because of the asymmetry between probability of anomalies and consequence of the events with given false positive rates. People quickly loose trust in systems that produce overly frequent, non-informative alarms[12]. An SIEM with high false positive rates is likely to be ignored or disabled.

4. Functional Requirements

As a first step towards an analytical platform the IT Security Lab (ITSL) in conjunction with the IT Security Office (ITSO) has evaluated multiple design iterations and variations to determine functional requirements for a log aggregation and analysis system. We are collecting a subset of authentication and network events from Virginia Tech's operational network. The proof-of-concept system also performs ETL processing to support EDA activities. From these design experiments we propose the functional requirements shown in Figure 4.

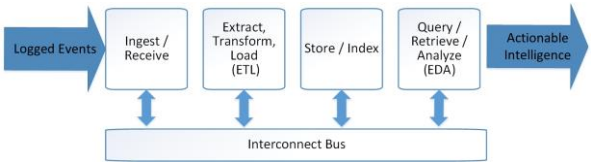

Figure 4 Functional Requirements

The system must ingest logged events at an acceptable rate and then transform and normalize data. Next the data must be stored and indexed to support performant query retrieval for EDA. Because we desire to compose the system from substitutable components we use a message bus to interconnect the sub-systems.

5. Analytic Considerations and Limits

A primary concern in network data security analytics is accurate detection and reporting of anomalous conditions that have a negative impact on systems. Collection involves many concerns including vantage point (i.e. sensor placement and positioning) and sampling frequency. Also, sensors should be designed to produce accurate and precise data. Additionally, controllability and reachability are classical control problems that are also relevant to collection efforts. And finally, veracity is a concern for the logged events produced by mid-point and end-point systems. Cyber systems exists in an adversarial environment where compromised systems and the event data they produce should not be fully trusted.

All of these concerns have the potential to introduce partial observability that impacts the statistical accuracy, fidelity, and precision of raw data. We can generalize the phenomenon of partial observability as the inability to measure state and maintain situational awareness of a sensor. Several causes of partial observability include:

- inability to monitor sensors due to data integrity
- data loss due to communications failures
- tampering, data has lost veracity
- sensors move out of view, loss of vantage point
- inability to interpret measurements (e.g. change in event log format due to software updates)

Carefully considered approaches to collections concerns can reduce uncertainty.

The impact of uncertainty and partial observability must be addressed in processing/exploration and analysis stages. There is no single standard for producing events or recording them in logs and realistically there never will be. Logs are produced for performance monitoring, error detection, and to

demonstrate regulatory compliance. Additionally, the syntax, data types, and even encoding of an event's data are dependent on interpretations and specific implementations of logging tools. Likewise, there is no standard to indicate the units of measure, frequency, velocity, range of motion, or semantics between logged events. This leads to two essential learning problems in learning from logs of event data:

- Learning *of* logged events: determine encoding, syntactical structure, and feature characterization.
- Learning *from* logged events: recognizing temporal, spatial, and semantic relationship patterns between events.

The first learning problem is typically approached as a data munging exercise when an analysts attempts to ingest a new data source. The analyst must iterate through ETL and EDA cycles to uncover undocumented syntactical features and obscured data types to ensure the data will be usable in future analysis. The second learning problem is not often addressed during processing and exploration but instead is handled during analysis and production.

When we have recorded historical observations, preferably with known ground-truth outcomes, it is possible to apply retrospective analysis of predictive factors within probabilistic bounds (due in part to partial observability). In effect, can we use our knowledge of the past to help predict the future?

Spatial, temporal, and semantic context should all be considered[13]. During analysis and production we characterize event relationships with rules for ordering events temporally, spatially, and semantically. Temporal ordering is impacted by the difficulty of maintaining synchronized time in a distributed system but simplified somewhat by the increasing monotonicity of time. Spatial considerations have become increasingly important due to the high degree of mobility provided by wireless communication systems. Spatial considerations are important for attribution efforts. Direct human interactions with computing systems are detectable because humans are physically present in a single location during a given time window. While we cannot infer intent, we can infer causation for the first immediate system reaction to human input. Semantic relationship between events are important to detecting sessions and processes.

6. Conclusion

This paper focused on an architectural approach network security data analytics. Many factors surround the development of an effective analytical platform such as the partial observability of data, model complexity, and computational combinatorics. All must be considered during the design, implementation, and analysis process. Further, the basic design should be extensible so that it can be used to analyze complex relationships with auto-correlative dependencies (i.e. spatial, and spatial-temporal). Lastly, a design that supports testing and validation is important to meet the goal of interpretability.

7. References

[1] A. A. Cardenas, P. K. Manadhata, and S. P. Rajan, "Big Data Analytics for Security," *IEEE Security & Privacy*, vol. 11, no. 6, pp. 74–76, 2013.

[2] S. Sagiroglu and D. Sinanc, "Big data: A review," in *Collaboration Technologies and Systems (CTS), 2013 International Conference on*, 2013, pp. 42–47.

[3] S. Northcutt, J. Shenk, and D. Shackleford, "The Log Management Industry: An Untapped Market," SANS, SANS Institute InfoSec Reading Room, Jun. 2006.

[4] Ponemon Inst., "2016 Global Encryption Trends Study," Ponemon Institute, Research Report, Feb. 2016.

[5] A. Lurie, "Top 47 Log Management Tools," *ProfitBricks Blog*, 19-May-2014. [Online]. Available: https://blog.profitbricks.com/top-47-log-management-tools/.

[6] D. Pereira, D. Schatsky, P. Sallomi, and R. (Bob) Dalton, "Cognitive technologies in the technology sector: From science fiction vision to real-world value," *Deloitte University Press*, 15-Dec-2015. .

[7] CJCS, "Joint Publication 2-0: Joint Intelligence." Chairman of the Joint Chiefs of Staff (CJCS), 22-Oct-2013.

[8] Assuria Ltd, "In Syslog we trust?," Assuria Limited, 5675d13, Mar. 2012.

[9] J. W. Tukey, *Exploratory Data Analysis*, 1 edition. Reading, Mass: Pearson, 1977.

[10] G. J. Myatt and W. P. Johnson, *Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining, 2nd Edition*, 2nd ed. John Wiley & Sons, 2014.

[11] O. US EPA, "What is EDA | CADDIS: Data Analysis | US EPA," 2009. [Online]. Available: https://www3.epa.gov/caddis/da_exploratory_0.html. [Accessed: 24-Apr-2016].

[12] T. Sanquist, T. Sheridan, J. Lee, and N. Cooke, "Human Factors Aspects of Anomaly Detection Systems," presented at the Committee on Human-System Intregation; National Research Council, 12-Feb-2009.

[13] M. A. Hayes and M. A. M. Capretz, "Contextual Anomaly Detection in Big Sensor Data," in *2014 IEEE International Congress on Big Data*, 2014, pp. 64–71.