A Network Malware Vaccination Solution

Randall Mora, *President/CEO, Avum, Inc.*

Abstract—This paper describes our research and development for (a) malware recognition and characterization, (b) malware capturing, and (c) processes to continually reduce the probability of a malware detection false positive (P_{FP}). We will demonstrate the building of an intelligent layer on top of metadata that can be used to serialize events for training a model on known malware behavior, as well as normal behavior. We intend to illustrate the use of actionable code for the tagging or encapsulation of endpoint events with logic that facilitates building models to be used in real-time or near real-time analysis of malware.

1 Introduction

THE goal is to build a software solution that uses attacks we know about to characterize them by the method that could be used to model and determine unknown attacks. The framework would enable a cooperative set of mitigations which could close the perimeter, so to speak, for nearly all attacks. The approach provides mitigation at a high level of abstraction to provide substantial layers of protection. The number of attack vectors that can be used is limited – there are only so many ways an assault can be launched on a device. Our goal will be to block any malicious activity using the same method, regardless of changes to its appearance or vector, preventing the malware from fulfilling its mission even if it already exists on a device. Each method will be defined at a sufficiently specific level, allowing a targeted mitigation approach for preventing the attack.

Our objective is to research the possibility of fusing collected endpoint data from log events, registry or process events, or other auditable events related to adversary behavior patterns. From this data, we will produce a comprehensive, unitary set of metadata that can be modeled to categorize normal and abnormal behavior within a network. The goal of this modeling is to target malicious activity and map it to intelligent metadata signatures.

2 Problem

Increasingly sophisticated malware attacks present an omnipresent threat that is becoming increasingly daunting. As we progress into the digital age, the drastically increasing number of attacks against Internet-connected systems since 1988 [1] suggests that malware intrusion has become an epidemic [2]. In 2010 alone, 286 million different types of malware accounted for more than 3 billion total attacks [3]. This statistic, although shocking, is an apparent truth that proposes a dilemma that needs to continue to be researched and addressed.

> Being believers in Moore's law, we understand that technology moves swiftly, with malware intrusion techniques moving just as swiftly. This truth drives our cyber security development to remain current by adjusting to a rapidly changing security environment.

• *R. Mora is Avum's Founder, President, and CEO, he plays an integral role in the company's research and development while providing strategic and management direction as the company's top officer . E-mail: randall@avum.com*

Manuscript created September 6, 2018.

There are a number of agencies dedicated to solving this problem, and many detection and prevention techniques have been developed and implemented. Current malware detection approaches fall into two main categories: anomaly-based detection and signature-based detection. The former relies on the knowledge of what constitutes normal behavior to be able to recognize malicious behavior, and the latter relies on known malicious code and behavior "signatures" to characterize malicious activity [2]. Even with these developments in malware detection technology, highly motivated malware developers consistently fuel the creation of new and innovative ways of bypassing detectors to infect home users and target commercial entities, and government agencies alike [4][5][6]. Contemporary malware makes use of a wide range of techniques such as packing, code obfuscation, polymorphism, and metamorphism to evade detection [7]. The consistent threat of dynamic intrusion realizes ongoing research in the development of new prevention techniques.

3 Objective

We will focus much of our research on the development of an intelligent, complete data abstraction to characterize the metadata of system events. Our design (Our Solution, Section 4) outlines how a potentially unlimited number of independent agents, collaborating through a defined workflow, could facilitate probabilistic model and learning algorithm development to derive and extract meaning and relevance from malware metadata. Our approach is to gather log events, registry or process events, or other auditable events related to adversary behavior patterns that would indicate an attack and produce a set of comprehensive metadata to characterize normal vs. abnormal behavior. A system of Intelligent Agents (IAs) would then wrap this metadata with an intelligent layer of actionable code, creating an encapsulated object. This intelligent layer allows for additional IAs to message the object in order to fuse relevant data into metadata that can be serialized and modeled to define normal versus abnormal behavior within a network. The goal is to map malicious activity to collectable metadata.

We will research and report on the capability of developing comprehensive, intelligent metadata that can characterize normal and abnormal events that can be modeled on a system. We will leverage the Avum's IA Framework that provides an advanced task queuing structure for IAs to effectively work collaboratively to encapsulate, sequence and fuse malware metadata (Our Solution, Section 4). The intelligent metadata we intend to research would consist of a layer of actionable code encapsulating previously found system event metadata prior

to queuing. The actionable code present in these encapsulated objects will enable events to be messaged by intelligent agents. These messages will facilitate and drive the construction of models. Model development and execution will identify patterns, trends, anomalies, and other relevant data that will be used to discover and characterize malicious behavior, while analyzing and computing the likelihood of a false positive (P_{FP}).

4 Solution

Our approach is to use actionable code to help accumulate metadata in a way that can be used to model behavior across multiple domains of information (mostly Windows log events, but also, if necessary, registry or process events, or other auditable events). Our solution is to build an intelligent layer on top of the malware metadata (encapsulate the metadata with serializable code that can respond dynamically to requests about the endpoint or the details of the actual data in the Windows log events). This encapsulation will enable the modeling of malicious and non-malicious behavior within the domain in which we operate.

The intent of our research is, as proposed in the challenge RFP, to develop a "technology capable of discerning behavioral characteristics of a large corpus of malware in a relatively short period of time, and then [analyze] these behavioral characteristics to identify temporal indicator-producing patterns, leading to the detection of malicious activity on endpoints in real time." To achieve this goal, we will leverage technology we have already built, our IA Framework. We will also continue research in other areas to ensure a complete sphere of research.

The goal of our research is to develop an abstract representation of all event types within a network. With this abstraction we will then identify the unique footprints of malware events. We envision a series of IAs that are grabbing already collected endpoint metadata and wrapping it in an additional, intelligent, actionable layer of code. Our goal is to use this encapsulating layer of actionable code for messaging of events, in order to create strings of data that can be used to train the models that are used in real-time and near real-time.

Log events will be the target source for our malware endpoint data. Log events include registry or process events, and other auditable events related to adversary behavior patterns. To collect the log events' endpoint data, we will use tools already built into the existing operating systems. However, if after further analysis, any additional tools are required, they will be reviewed/cleared prior to deployment.

By following the recommendations set forth in the National Security Agency/Central Security Service's Information Assurance Directorate's paper titled: "Spotting the Adversary with Windows Event Log Monitoring," [8] the events we intend to collect/focus on as indicators of malicious behavior are the following:

- Application whitelisting events
- Application crashes
- System and services failures
- Windows update errors
- Windows firewall status
- Clearing of event logs
- Software and service installation
- Account usage
- Kernel driver signing
- Group policy errors
- Windows defender activities
- Mobile device activities
- External media detection
- Printing services
- Pass the hash detection
- Remote desktop logon detection

As this will be an iterative process, additional indicators will be included in the study as required and deemed necessary. We will complete the work to outline the metadata that needs to be collected (i.e., Windows event logs and other auditable events) then build our data sets that will be used as input to the tier of IAs that encapsulate and queue relevant events for sequencing and fusion. Our goal is to categorize disparate metadata such that it can be modeled and used to flag abnormal events.

The queuing system/machine learning approach we envision will work well within a framework similar to our IA Framework (see Figure 1). As encapsulated metadata is queued by a series of IAs, additional IAs are assigned to processing this information within the queue. Different IAs are responsible for processing different parts of the building task chain: fusing data, reordering the task queue, creating new tasks, removing and adding metadata units, serializing fused data, adding/pulling it to/from the datastore, and pulling data to train a model that can be used to categorize normal behavior and, in turn, determine malicious behavior. All of these tasks will be completed with the aid of our intelligent layer surrounding the metadata. A diagram of how we envision this system is shown in Figure 1 on the following page.

This system will also employ a machine-learning approach, first being deployed on a test system without malware and designed to learn "normal" behavior on the network. This normal behavior is defined as events that take place throughout any given day and do not present any malicious threats. After normal behavior is modeled, malware will then be introduced to the test system. Analysis will occur in real-time– as information is processed into metadata by our team of IAs. Events that are out of the ordinary will be flagged. Our system will work as a vaccination does [9]: once normal behavior is learned and a piece of malicious software is introduced, the anomalous activity's associated metadata signatures are flagged. As we introduce more and more malware-related log events into our environment, we will learn malware-associated system events and hence be able to build a tolerance to them. This will aid in the development of the prevention of zero-day attacks.

Our test system, closely monitored, will be private and secure to protect others from the threat of malicious activity. Understanding that malware may cease to perform its malicious intent on an enclosed network, we will research additional methods to extract malware information. This will also potentially gather previously concatenated data sets. Through this process, system events that are associated with malware intrusion will be categorized and studied.

There are far-reaching implications of the completion of our desired research. When an abnormal event occurs, based on the eventual probabilistic models of our system activity, a probability score could be provided to the event. We have been researching probabilistic methods that would serve as the architectural basis for our model construction. One of which, an application of Bayes' theorem, is a method we have previously researched with regard to our intelligent agent technology. Bayes' theorem allows for the combining of our prior data and the observed data to allow us to create a conditional distribution based on the current set of observed data.

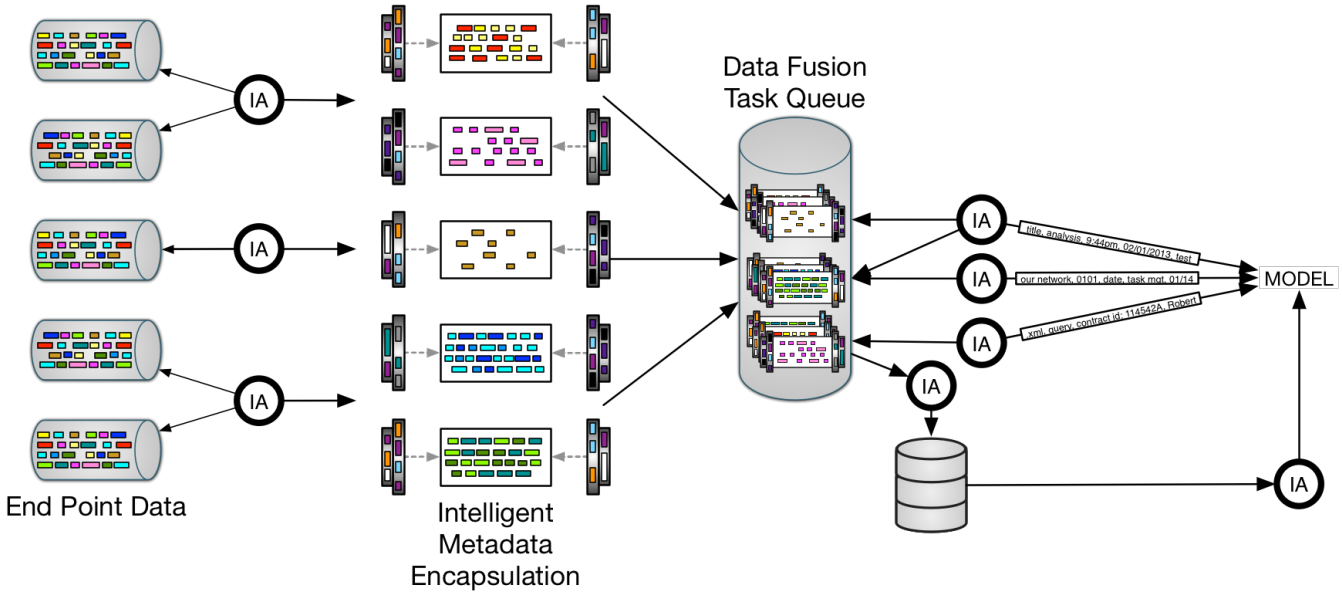

Fig. 1: Network Malware Vaccination Solution

Simply stated, Bayes' theorem provides a means of updating existing knowledge with new information, a key component of our intelligent agent driven model construction. We begin with a prior belief, π, and after learning information from data y, we change or update our belief about θ and obtain $p(\theta \mid y)$ (the probability of θ given y). These are the essential elements of the Bayesian approach to data analysis, which are present in the formula for Bayes' theorem below [10].

$$p(\theta \mid y) = \frac{p(\theta, y)}{p(y)} = \frac{p(y \mid \theta)\pi(\theta)}{p(y)} = \frac{p(y \mid \theta)\pi(\theta)}{\int p(y \mid \theta)\pi(\theta)d(\theta)}$$

By utilizing probabilistic methods such as Bayes' theorem (as well as others), we can answer the question: "What is the likelihood this intrusion alert is a false positive?" We will perfect our false positive evaluation algorithms/techniques by thoroughly testing against both benignware and malware alike. As we gather more information from our own system and from the research of others, our intelligent agents are iteratively updated to more effectively fuse data across domains. This facilitates the construction of more successful models, and increases the probability of detecting false positives [11]. Future research will also involve providing response options for detected malicious behavior.

5 Case Study

The following outlines our process to conduct the research required to achieve the desired goals.

Discovery and Data Collection of Malicious and Non-Malicious Metadata

We will research, present and finalize the initial endpoint metadata that will be used for the study. This can and will change as the study progresses. The process of characterizing the malicious and non-malicious behavior will change this data set. It is the goal of this task to create a comprehensive starting point. We have been investigating sources from which we will obtain the malware samples to facilitate our research, including Agency-provided sources (1999 MIT/LL DARPA Intrusion Detection Evaluation Data Set, PREDICT Data Sets, and HSIP Gold), as well as publicly available samples and data sets (data.gov, aws.amazon.com/publicdata sets/, offensivecomputing.net, malwareblacklist.com, malshare.com, kernelmode.info, etc.). We will pull data and samples from these sources. Through gathering endpoint data samples of malicious and non-malicious behavior, we will uncover known malicious and non-malicious behavior indicators that will facilitate the design of the actionable code that will encapsulate the event. We will then work to arrive at a definition of an endpoint with actionable intelligence attached. We will want to iterate through as many ideas as possible to build the best definition. As the study progresses, this representation of an endpoint event is expected to change. Our goal is to be able to message the event so that it can create strings of data that can be used to train models that will be used in real-time or near real-time.

System Analysis and Detailed Design of Infrastructure for Processing Endpoint Metadata

The intention is to use actionable code for training and building models of malicious and non-malicious behavior. We will define the logic that will encapsulate the endpoint events, define the logic that will exist in an IA for the fusion of relevant events across endpoints, and perform a complete functional specification of actionable code that will be written to support characterizing the metadata.

Build the Actionable Codebase That Will Be Used to Encapsulate the Events and Queue and Consolidate Actionable Endpoint Metadata

At this point, we will build the actionable code that will encapsulate the endpoint metadata, the methods that will serialize these endpoint objects, and the methods that will return strings for the training and execution of our models. We will

also define and create IAs to bring this actionable intelligence together for modeling. We will create IAs for reading and queuing endpoint data, encapsulating the endpoint metadata, the fusion of encapsulated endpoint objects, and for building training sets for the models. The time allotted for building additional IAs needed will be reviewed after discovery and analysis.

Model Discovery and Development

Our goal is to start simple and iteratively change the model as we continually add complexity from more endpoint data and additional malicious activities. After revisiting discoveries, we will first perform a final determination of the modeling tools that can be used for this aspect of the project. We will then proceed to modeling malicious and non-malicious activity. Our plan is to build a non-malicious network and an encapsulated network with released malware in it, then monitor and build training sets with our IAs. This will be an iterative process of adding malware, monitoring, and then wiping and recreating the environment for testing and training the network model.

Iterate Data Gathering and Model Training and Execution While Analyzing the Results

We will perform analysis on the results of our characterization and test and revisit methods for increasing the probability of determining malicious behavior and reducing the probability of false positives. We will make any necessary changes to our study then iteratively analyze and retest.

Proof of Concept – Execution

We will next prepare, socialize, and document our proof of concept test case and presentation of the work completed to-date, review demonstrations/deliverables and revisit key aspects of the study, and discuss successful and unsuccessful areas of the design.

6 Conclusion

Our solution plays to the Cyberspace threat by applying our strengths – by drawing on and adapting our depth of experience in developing intelligent agents. The agents will operate within and across domains, will mine disparate data, will compress out irrelevant data, and will fuse remaining data to create hybrid intelligence used to detect, predict, and prevent attacks in Cyberspace.

Acknowledgments

Mr. Mora would like to acknowledge the Avum team, for always being there to provide support, advice, and assistance; with special acknowledgment to David Michalczuk, for not only building Avum's secure networks but for his never-ending care in the diagrams that represent what we do.

References

[1] CERT/CC, Carnegie Mellon University. *http://www.cert.org/stats/*, Last updated February 12, 2009.
[2] Idika, Nwokedi and Mathur, Aditya P. *A Survey of Malware Detection Techniques.* Department of Computer Science, Purdue University, West Lafayette, IN. February 2, 2007.
[3] Reavis, Jim. *The Ongoing Malware Threat: How Malware Infects Websites and Harms Businesses – What you can do to stop it.* Symantec. 2012.
[4] Prince, Brian. *Defense Department Confirms Critical Cyber-attack.* Eweek. August 8, 2010. *http://www.eweek.com/c/a/Security/Defense-Department-Confirms-Critical-Cyber-Attack-551206/*
[5] Sherstobitoff, Ryan. *Analyzing Project Blitzkrieg, a Credible Threat.* McAfee Labs, 2013. *http://www.mcafee.com/us/resources/white-papers/wp-analyzing-project-blitzkrieg.pdf*
[6] Aldridge, Jim. *Remediating Targeted-threat Intrusions.* Mandiant Company. *https://dl.mandiant.com/EE/library/BH2012_Aldridge_RemediationPaper.pdf*
[7] Ma, Weiqin, et al. *Shadow attacks: Automatically Evading System-Call-Behavior Based Malware Detection.* Journal in Computer Virology 8.1-2 (2012): 1-13.
[8] National Security Agency/Central Security Service. *Spotting the Adversary with Windows Event Log Monitoring.* Revision 2. *http://www.nsa.gov/ia/_files/app/Spotting_the_Adversary_with_Windows_Event_Log_Monitoring.pdf*, December 16, 2013.
[9] National Center for Immunization and Respiratory Diseases. *How Vaccines Prevent Disease. http://www.cdc.gov/vaccines/vac-gen/howvpd.htm, Last updated April 25, 2012.*
[10] Mora, Randall P. and Hill, Jerry L. *A Service-Based Approach for Intelligent Agent Frameworks.* Presented at the International Telemetering Conference (2011) 1-8. Submitted June 2011, Published June 2011.
[11] Christodorescu, Mihai, et al. *Semantics-Aware Malware Detection.* Security and Privacy, 2005 IEEE Symposium on. IEEE, 2005.

Randall Mora Mr Mora is Avum's Founder, President, and CEO. In Avum, he has built, and continues to build, a professional services and computer sciences research and development firm that specializes in large-scale systems and software development. With over 35 years of experience in systems architecture, development, and deployment, Mr. Mora is currently focusing on cognitive computing approaches and enterprise architectures to solve complex mission-critical solutions for achieving interoperability between cross-domain disparate systems. Mr. Mora is currently applying this technology across multiple contracts, including major government and commercial business applications..