AIVRTEK Anomaly Behavior Analysis (ABA) Methodology

www.avirtek.com

Problem Statement

The current cybersecurity techniques are mainly labor intensive (e.g., patch update), signature based, and not flexible enough to handle the current cyberspace complexity, dynamism, and epidemic-style propagation of attacks. Furthermore, the organization boundaries are gradually disappearing so that the idea of creating a defendable perimeter becomes useless and, on top of that, the cyber-attackers that we need to protect against can be insiders who are trusted and have full access to computing system resources and services. Consequently, it is not feasible anymore to rely on human-manual-intensive management tasks that have failed to promptly secure our cyber systems and services. It is critically important that we develop an autonomous decision making system that can adapt, react, and learn from real-time computer systems.

Limitations of Current Cyber Security Solutions

Current cybersecurity technology and tools have failed for many reasons, some of which are:

- They are mainly signature based solutions that cannot detect new and novel cyber-attacks
- They use many isolated and heterogeneous tools for monitoring performance, fault, and security that make it extremely difficult for human to comprehend and manage in a timely manner
- They are typically developed as threat response (defensive) technologies that inherently cause operational issues as they 'respond and repair' attack damages.
- They are manually intensive activities that make them too slow to respond and act in a timely manner against malicious threats (e.g., according to a recent survey of 50 benchmark companies, the average detection of a cyber-attack is 18 days and more than 200 days for detecting an insider threat if successful).

Anomaly Behavior Analysis (ABA) Methodology

We have developed an Anomaly Behavior Analysis (ABA) methodology to recognize any non-self behavior in computers, protocols, users, or applications. The ABA methodology has been successfully applied to analyze the behavior of different network protocols (IP, TCP, UDP), wireless networks, HTTP, DNS, and email protocols [1, 3, 4, 5, 6]. In this task, we will use this methodology to analyze the operations of the cyber DNA data structures. To explain our ABA-based Self-Recogniation methodology, we will describe two examples: Anomaly detection in the Domain Name System (DNS) protocol (application layer protocol) and detection of malicious components embedded in data objects (e.g., files, HTML, XML, Image, etc.).

Example 1: Anomaly Detection in DNS Protocol

The main function of the DNS protocol is to provide its users with a name service that translates human friendly names to network level IP addresses. In the 1980s, performance of the DNS protocol was very important issue. Hence, security issues were completely ignored and consequently the DNS protocol suffers from severe security flaws.

The two most important features of the DNS protocol that are exploited by cyberattacks are its hierarchical architecture and caching mechanism. Cache poisoning attacks exploit the vulnerability in caching recently discovered name resolutions [11]. Since many Internet services rely on DNS protocol, a wide range of cyberattacks, such as phishing fraud or identity theft, use the DNS cache poisoning approach. Furthermore, DNS servers can be shutdown by Denial of Service (DoS) and Distributed DoS (DDoS) attacks. DNS protocol can also be used by cyber attackers to participate in DDoS attacks against the other Internet services, similar to the ***smurf*** attack [14], and is referred to as DNS amplification attack [15].

In general, two techniques have been investigated to address the DNS security problems: Preventive approach and Intrusion Detection approach. The preventive approach is based on adding security mechanisms to the protocol in order to prevent or harden from the attack. A good example of this approach is the DNSSEC [16] in which the DNS protocol is redesigned from scratch with security considerations. The second approach is based on Intrusion Detection Systems [21][22][23] which monitor the DNS traffic to detect cyberattacks against the DNS protocol.

AVIRTEK's ABA approach for the DNS protocol is based on continuously monitoring and analyzing the temporal behavior of the DNS protocol to detect any anomalous behavior that might be triggered by DNS attacks. The DNS protocol according to RFC1034 [9] and RFC 1035 [10] is a Query-Response protocol. It means that a DNS server will listen to its dedicated port (standard UDP port 53) and for each Query it receives, it sends back an appropriate Response.

In anomaly behavior analysis, it is necessary to characterize the normal DNS traffic in order to detect abnormal behaviors that exploit the existing vulnerabilities in the DNS protocol. Figure 6 shows an exemplary logical state diagram that characterize the normal

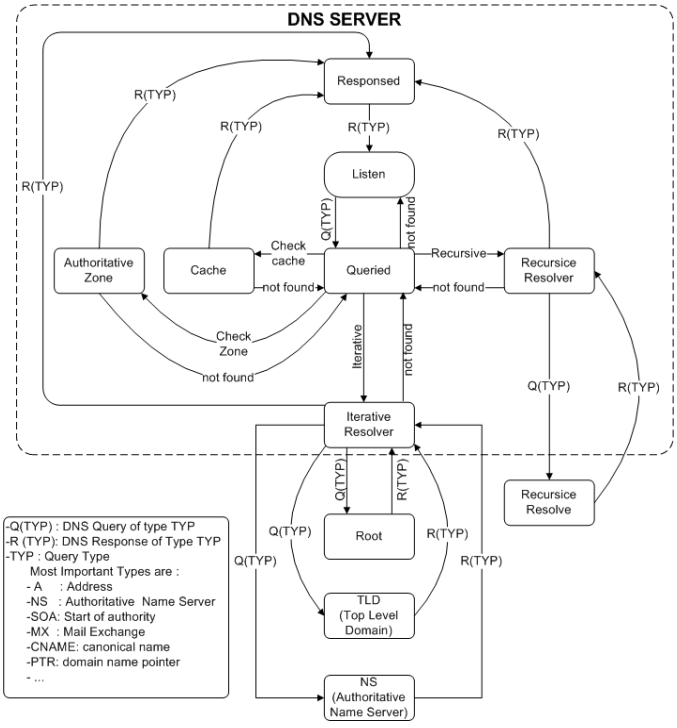

Figure 1. DNS protocol state diagram.

operations of the DNS protocol. The transition conditions of the protocol are different types of queries and responses (Q(TYP) and R(TYP)), where TYP indicates the type of query or response. Our anomaly behavior of the DNS protocol is based on the assumption that the cause of anomaly can be localized to shorter subsequences within the actual sequence [14]. This method uses a sliding window of fixed size ***n*** to extract ***n***-length subsequences, known as ***n-grams***, from the actual sequence. Since most of the DNS attacks generate anomalous protocol transitions, the use of ***n-gram*** behavior analysis can then be used to detect accurately the DNS attacks.

We use machine learning techniques to learn the normal transitions when the DNS protocol is operating normally. In the training mode, we generate normal DNS traffic as well as abnormal traffic. During this phase, the ***n-grams*** of the DNS traffic are generated and the statistical properties of each pattern are stored in the database. If during the testing mode, the statistical properties of the observed ***n-gram (anomaly score)*** are significantly different from those stored for normal ***n-gram*** DNS traffic, it will be considered abnormal.

We evaluated the ABA for the DNS protocol with different types of DNS attacks (e.g., birthday attack and Dan Kaminsky cache poisoning, DNS Amplification, and DNS Tunneling) that were detected accurately with no false alarms. Figures 2 and 3 indicate the distribution of the anomaly score (*a-score*) for normal and DNS attack flows. The ***X*** axis shows the anomaly score and the ***Y*** axis represents the percentage of flows which match that score.

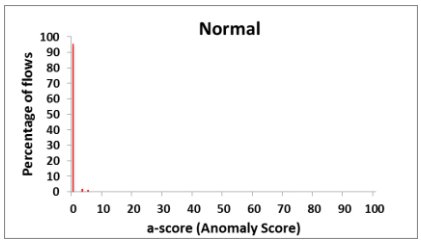

Figure 2: Anomaly Score Distribution for Normal Flows

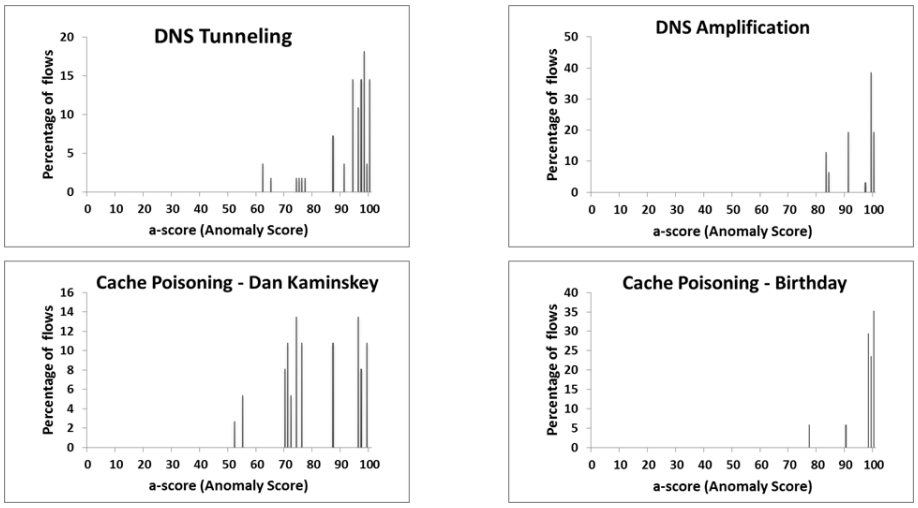

Figure 3: The anomaly score distribution for different type of attack traffics

It is clear from these diagrams that the normal and abnormal traffic are properly separated and consequently can detect DNS attacks with almost no false alarms With an award from the US Air Force, AVIRTEK applied ACS methodology to detect the existence of malicious components embedded within any data objects or files.

Anomaly Detection in Data Objects and Files

Our approach focused on performing two types of anomaly analysis: Anomaly Structural Analysis (ASA) and Anomaly Dynamic Analysis (ADS) as shown in Figure 4. In the ASA approach, we analyze the structure of large number of data files of one type (e.g. HTML, XML, RTF, etc.) and use advanced machine learning algorithms to characterize the normal structure of the meta data and body components in that file. In the ADS approach, the dynamic analyzer renders the file being analyzed in a virtual sandbox environment that protects the host computer from infection. Most often malware doesn't become active until either a length of time has expired, a hyperlink or other key has been clicked, or some other event has triggered it. The dynamic analyzer generates data structures that can be used to analyze the dynamic behavior and how it interacts with operating system services, file systems, networks and system supported run-time libraries. This dynamic analysis will enable us to detect the behavior of a hidden malicious component once becomes active. The overhead of running the ADS is high and it will be used after the ASA analysis is performed to make sure we can detect any hidden malicious components that cannot be detected by the structural analysis.

•

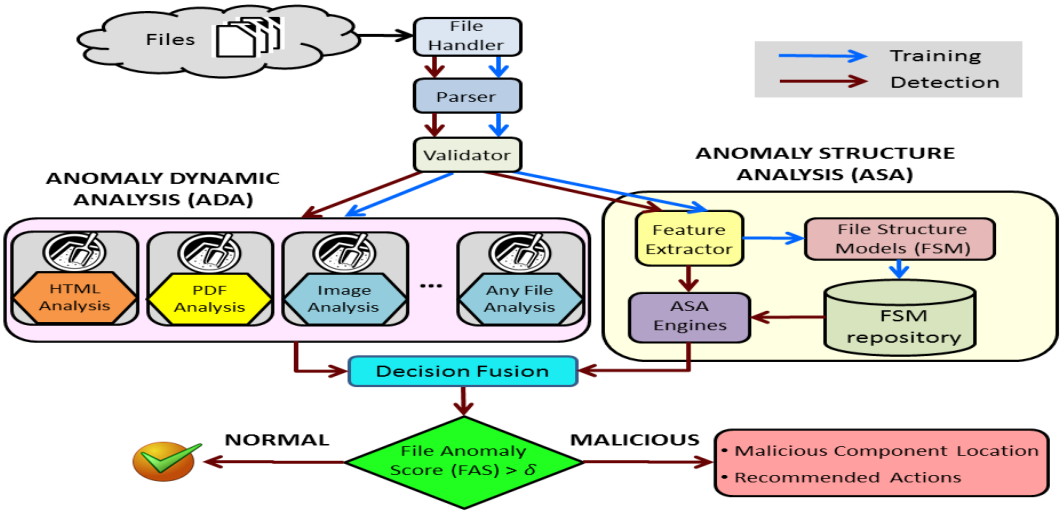

Figure 4: Anomaly analysis approach of data objects and files

Early in the project, a web crawler tool was developed to collect the required data file types from different web sites. The vision is that AMAP will continuously download files from the internet, and if they were classified as normal, features would be extracted and added to the training database.

For example, Figure 5 shows the process we follow when we download files from the Internet and the parameters that can be used to characterize normal structures. The downloaded files first pass through a Decompressor which will separate packed files into their individual major elements (html, java script, shell script, image, table, etc.). During this process we will be generating statistical metrics that will be used to perform the two types of anomaly analysis (ASA and ADA) that will be used to determine whether a file is malicious or normal. Figure 3 shows the metrics that we used to characterize the normal statistical properties of HTML key words, headers, HTML tags, JavaScript, and other components as will be explained when we discuss our approach to develop HTML Behavior Analysis Unit (BAU). In addition, the dynamic behavior of the file will be analyzed using an isolated virtual machine environment. If a threat is detected at any of the scanners, the trustworthiness of the file will be updated.

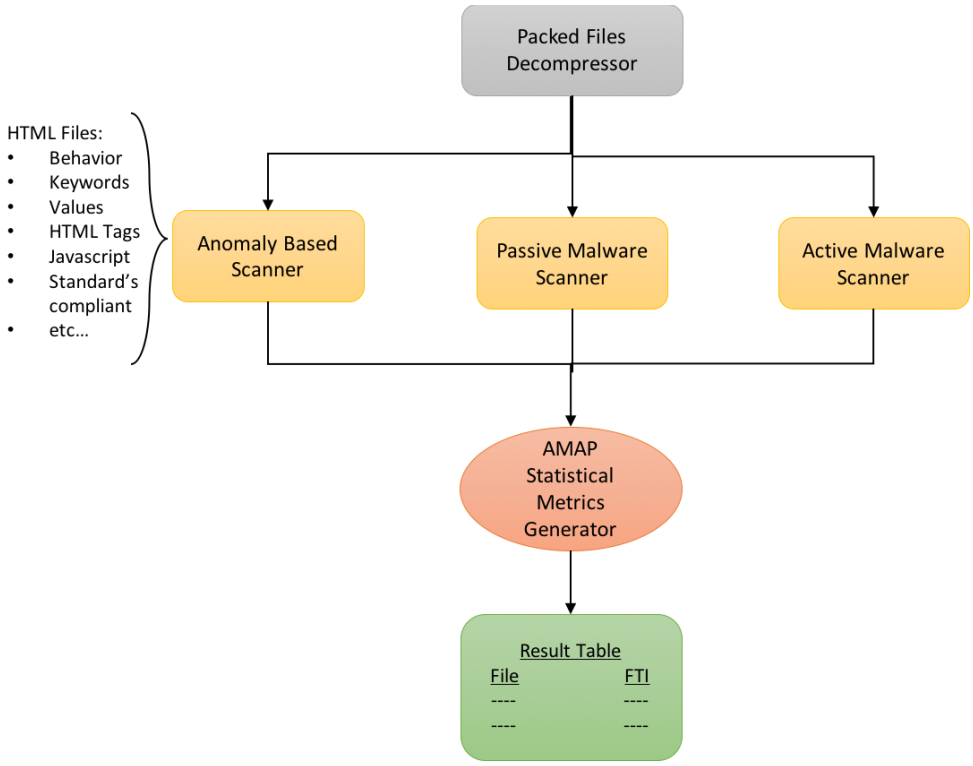

Figure 5. ABA approach to analyze downloaded files.

AVIRTEK has developed a Data Analytics Engine (DAE) shown in Figure 6 to perform the required ABA on any file or data objects. The first stage in the data analytics engine is to select the features to be used for analysis that will use information theoretic based approach to select the most relevant features that can support the required anomaly analysis. The next stage to use to the collected data files (normal and malicious) to identify the machine learning algorithms that can meet the required performance requirements in terms of detection accuracy, and low number of false alarms that will be validated in the second stage.

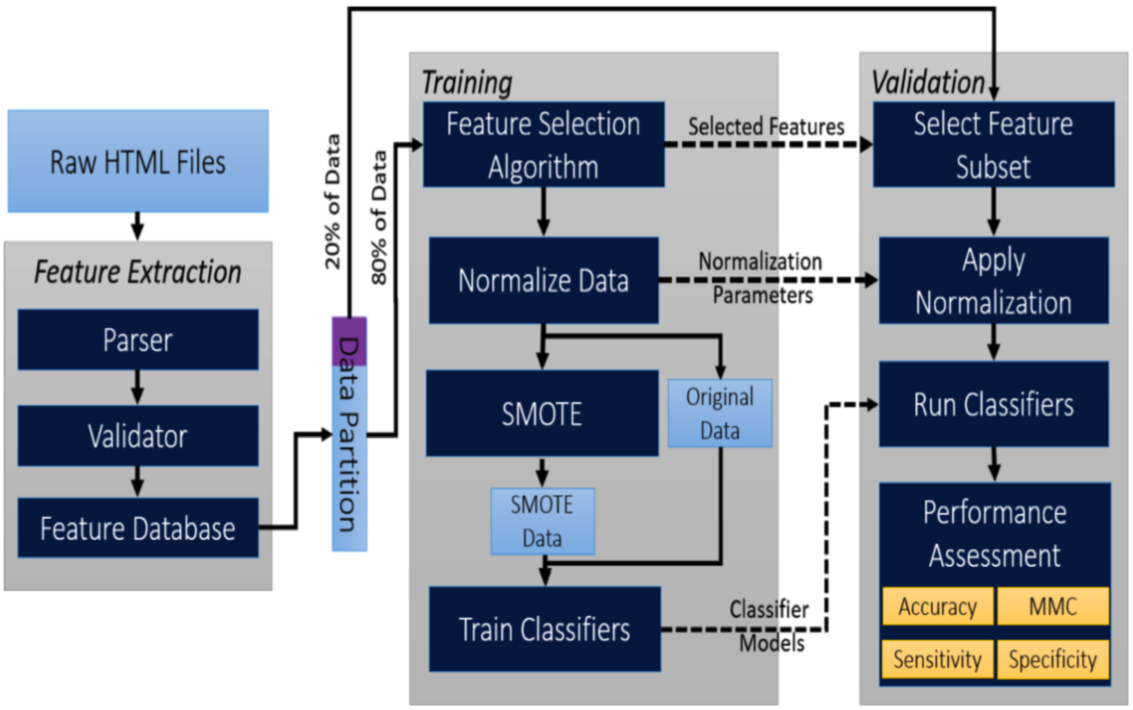

Figure 6: Data Analytics Engine

The ABA methodology for detecting malicious components in files and data objects was developed with funding of more than $1 Million dollars SBIR Phase I and Phase II award from the Air Force Research Labs. The DAE shown in Figure 6 is used to to develop the ML-based models. During the training phase, we used a dataset of 10,762 entries. A preliminary evaluation of the ASA approach on an unseen dataset of 6,945 entries were used to test the performance of the obtained classification tree. The results of the evaluation are shown in Table 1.

Table 1: Anomaly structural analysis results

Correctly Classified Instances	6946	99.299%
Incorrectly Classified Instances	49	0.7005%
Mean absolute error	0.008	
Root mean squared error	0.072	
Relative absolute error	48.329%	
Root relative squared error	75.254%	
Total Number of Instances	6995	

-

4.3 Autonomic Management

The researchers at AVIRTEK and the NSF Center for Cloud and Autonomic Computing (CAC) at the University of Arizona have developed and successfully implemented a general autonomic computing environment that has been the basis for AVIRTEK ACS autonomic management capabilities. AVIRTEK is currently commercializing this architecture in its ACS-based cybersecurity products. By adopting the Autonomic architecture shown in Figure 12 we implement Autonomic Management using two software modules: the Observer and the Controller modules. The Observer module monitors and analyzes the current state of the managed cyber resources or services. The Controller module is delegated to manage their operations and enforce the operational policies. In fact, the Observer and Controller pair provides a unified management interface to support self-management services by continuously monitoring and analyzing the current managed resource conditions in order to dynamically select the appropriate response to correct or remove anomalous conditions once they are detected and/or predicted. The Observer monitors and gathers data about the logical and physical resources and analyzes them to build the knowledge required by the Controller in order to carried the most effective responses to cyberattacks.

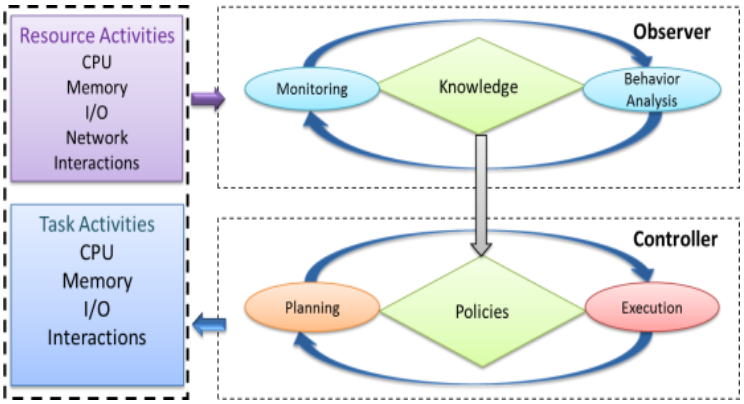

Figure 12: Autonomic management architecture. Architecture

4.3 Self-Protection Agents (SPAs)

The self-protection agent is a software module that runs on any managed resource (computers, servers, etc.) to take the recommended actions once the self-recognition agents detect any "non-self" behavior in the monitored computers, applications or users. To reduce the false alarms in "non-self" user behavior detection, we adopt a Challenge-Response approach by requiring each user to provide the correct answers to a prior selected set of questions as shown in Figure 13.

If the SRA detects any "non-self" user behavior at runtime, it will ask the user to answer one or more random questions from the list as shown in Figure 14. If the user does not answer the question correctly during a specified period of time, an alert is generated to the SPA to take the appropriate responses (lock user account, lock the machine, shutdown computer, etc.). If the user answers the questions correctly, the adaptive learning module will be triggered to adopt the self-recognition user model.

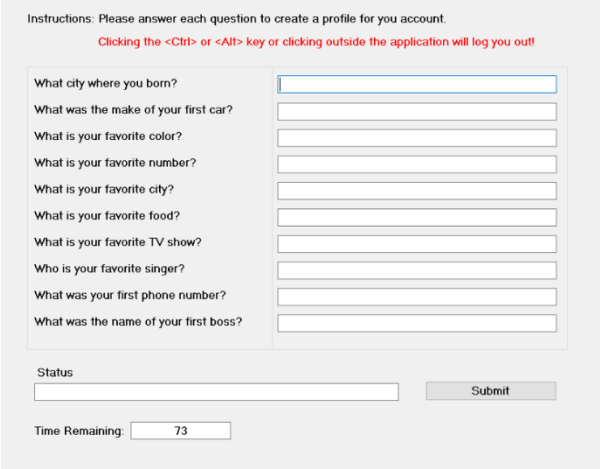

Figure 13: Questionnaire

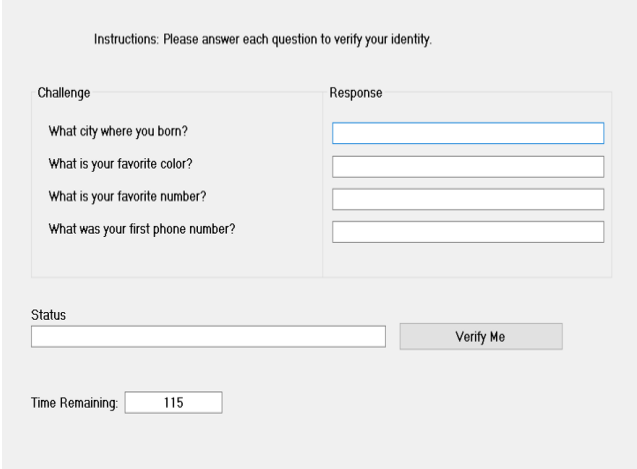

Figure 14: SRA Random Challenge

About AVIRTEK, Inc.

AVIRTEK is a cybersecurity companies developing autonomic cybersecurity products and services, located in Tucson, Arizona, and founded in 2006. AVIRTEK, is a startup company from the National Science Foundation (NSF) Center for Cloud and Autonomic Computing at The University of Arizona (nsfcac.arizona.edu), a center funded by the Industry/University Cooperative Research Center of the National Science Foundation, is pioneering innovative autonomic management solutions that are based on strategies used by biological systems to deal with complexity, heterogeneity and uncertainty. AVIRTEK's technologies enable us to deploy cyber infrastructures that can self- configure, self-heal, self-protect, and self-optimize their resources and services. AVIRTEK research and development capabilities were developed with more than $ 8 Million dolars of funding from US Air

Force, Army Research Laboratory and Navy. To learn more about AVIRTEK products and services, go to http://www.avirtek.com