Systems for Detecting Advanced Persistent Threats

a Development Roadmap using Intelligent Data Analysis

Johannes de Vries, Hans Hoogstraaten

Fox-IT
Delft, The Netherlands
{jdevries,}@fox-it.com

Jan van den Berg, Semir Daskapan
Engineering Systems & Services Department
Delft University of Technology
Delft, The Netherlands
{j.vandenberg, s.daskapan}@tudelft.nl

***Abstract*—Cyber-attacks against companies and governments are increasing in complexity, persistence and numbers. Common intrusion detection methods lack the ability to detect such - what are commonly termed - advanced persistent threats. A new approach is needed that takes the stepwise characteristics of this type of threats into account and links analysis methods to attack features. This paper takes up this challenge. First, an analysis framework is proposed to relate complex attack attributes to detection and business aspects. Second, the framework is used to define a development roadmap for designing advanced intrusion detection systems; such systems can analyze network traffic and client data at multiple network locations using both signature and anomaly detection methods derived from the intelligent data analysis field. Third, a test case is provided showing the potential power of the proposed development roadmap.**

***Keywords*—intrusion detection; advanced persistent threats; development roadmap; intelligent data analysis; cyber security**

I. Introduction

At present the cost of cybercrime, criminal activities on cyber infrastructures, is considered to somewhere between 100 billion to 1 trillion US dollars annually worldwide [1]. Cybercrime is attractive to criminals because they run a low risk at being caught and prosecuted for their crimes. The result is that a complete industry has evolved aimed at committing cybercrimes. Governments on the other hand have also found that cyberspace can be used to spy on other states and can be an arena for warfare [1].

Virus scanners, firewalls and intrusion detection systems where created with the purpose to reduce the economic damages from cybercrimes. Cyber criminals and spies in turn created more advanced means to breach the security measures. Some of those attacks are called Advanced Persistent Threats (APTs) [2]. An APT is a form of multistep attack that is executed with more stealth and is targeted specifically to achieve a specific goal, most often espionage. APTs use different steps, just as normal multistep attacks, in order to reach their goal. However, APTs are different in the sense that they are more often based on so-called 'zero-day exploits' (not publically known security flaws in software) and advanced means of attack like social engineering [2]. APTs are currently the largest threat to companies and governments [3].

This paper proposes a new way of analyzing multistep attacks like APTs by linking attack characteristics to detection methods as used in network or host intrusion detection systems (N/HIDSs). The intelligent data analysis algorithms in these methods are one of the key issues in detecting various activities related to APTs. The proposed framework considers aspects of attack methods, detection methods and impact on business.

The remainder of this paper is organized as follows. Section 2 gives some background on multistep attacks, and on current applications of intelligent data analysis methods for intrusion detection. Section 3 introduces the new framework for analysis of APTs. Section 4 presents a roadmap to the development of systems that detect APTs with the use of intelligent data analysis methods. An application test case is provided in section 5, after which section 6 concludes the paper.

II. Attacks and Data analysis Methods

A. Advanced Persistent Threats

More sophisticated cyber attacks consisting of multiple steps are being researched since the beginning of this century. The approaches to multistep attacks assume that most steps of a multistep attack are detected [5] [6] [7]. The emergence of a new breed of multistep attacks, often called Advanced Persistent Threats, can be considered to be a new form of this type of attacks [2]. They differ from the multistep attacks in the sense that they are executed with more stealth by skilled attackers who are very persistent in achieving their goal. The heavy use of zero-day exploits, which are exploits unknown to software vendors and security companies, makes detection more difficult. Social engineering and targeted emails to direct users to websites to install malware are also common traits of APTs. APTs are generally considered to have reconnaissance steps, steps to gain a foothold in a network or host, steps to look for resources and, finally, steps for proprietary data extraction [2] [3] [8]. A well-known example of an APT is named Operation Aurora. This attack was aimed at several high value companies and used multiple zero-day exploits, social engineering and encryption for obfuscation, making it very hard to detect such attacks [2] [3]. Defending an organization against APTs requires keeping software and

978-0-7695-5014-5/12 $26.00 © 2012 IEEE
DOI 10.1109/CyberSecurity.2012.14

defenses up-to-date. But this is not enough considering the use of unknown exploits. An improved approach to intrusion detection is required to detect APTs. However an appropriate attack analysis framework and design roadmap for (N/H)IDS is lacking.

B. Intelligent Data Analysis for Intrusion Detection

When designing (N/H)IDS generally spoken there are three approaches to intrusion detection [9]. The first approach uses signature detection. A signature detection system compares a data sample to the signatures in the system. When a signature matches, a warning is issued. Such systems are reliable and have a relatively low false positive rate. The main problem is that such systems are not capable of detecting unknown characteristics of attacks [9]. But unknown attacks can be detected if some characteristic of a known attack is used. For example if a directory listing is made on a command prompt by some malware, it can be detected. Or a NOP sled indicating a buffer overflow attack can also be detected independently of the used malware.

Anomaly detection is the second approach. Anomaly detection methods learn what is considered to be normal behavior in a network or computer system, and report anomalies as alerts. Two different groups of methods are used in learning what normal behavior is. The first are called supervised learning methods. These methods use labeled datasets to understand what is normal and what, possibly, is an attack. These methods are relatively successful without having too many false classifications. The second group of methods concerns unsupervised learning algorithms. These methods use unlabeled data to find anomalies but usually generate a lot of false positives [9].

The third approach combines signature and anomaly detection: signature detection is used to ensure detection of known attacks, and anomaly detection is used to create a means to detect attacks unknown to signature detection [9].

III. ATTACK ANALYSIS FRAMEWORK

Section two mentioned the lack of an appropriate attack analysis framework and a design roadmap for (N/H)IDS. A new framework for analyzing attacks is therefore proposed here. The step-wise structure of attack methods used in APTs are used to define the framework.

A. Analysis Framework

The new framework (see Fig. 1) contains seven columns. The first three columns describe attack related aspects, in this way providing a detailed description of the attack. This description provides features for detection, which are the input for the detection related columns more at the right. The first column contains the different steps of an APT attack. The number of steps determines the number of rows in the framework. The second column contains low-level attack methods that are used for each attack step. E.g., in the gaining access phase, a password guessing attack might be tried. The third column contains features of the attack methods in the second column. In case of a password guessing attack, such features are the number and frequency of login attempts. They can be used for detection. For unknown methods like zero-day exploits, these features might not be known exactly. In such cases the goal of the attack step and the goal of the attack method can be used to specify indicators or possible expected changes in the behavior of the systems in use.

The content of the columns in the framework should be ordered so that information in the different columns can be combined like attack features to attack methods, locations to attack features, etc. The relationships and influences between the different columns should become clear when one reads the framework from one side to the other. Drawing a linked tree like structure in the rows could be used to make the relations between the columns more structured and easier to understand [19].

The fourth column contains possible detection locations of features. These might be, for example, in a demilitarized zone (DMZ) of the network, in a server log, or on several workstations. The location determines the possibilities for detection and for the data that become available for analysis methods to be used for the detection of attacks. Some attacks might have multiple detectable features giving a choice for detection methods or detection locations.

The fifth and sixth columns describe additional detection

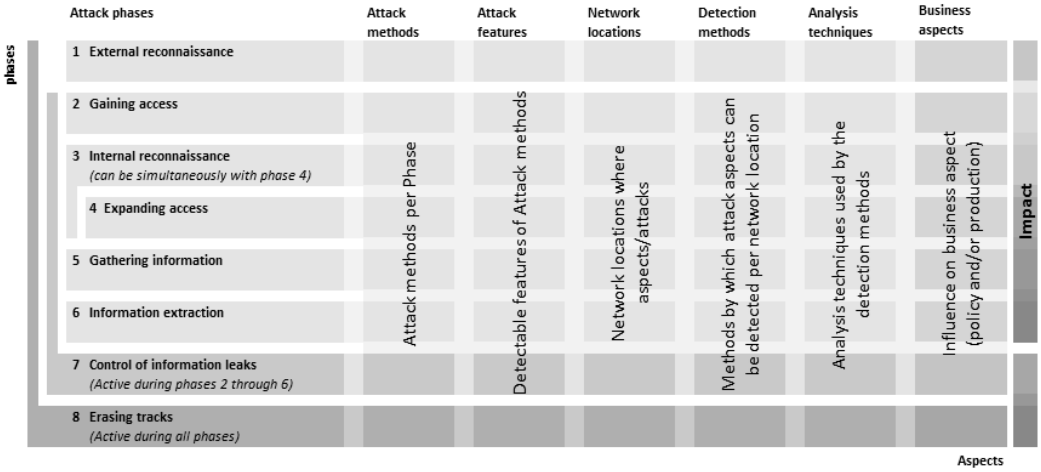

Fig. 1. An analysis framework to relate attack characteristics to detection location and methods.

issues. The fifth column contains possible detection methods. Methods include approaches for, for example, network intrusion detection, host intrusion detection, or log analysis. The sixth column should be filled with analysis methods used. This is the place where intelligent data analysis algorithms are placed in the model. The kind of input data determined by the contents of the first five columns determines what input becomes available for analysis: the sixth column indicates the methods to be used for detection of the attack features at the proposed locations in the previous columns.

The last column in the framework contains business aspects related to attacks and detection methods. The impact scale on the uttermost right hand side of the column indicates that the possible business impact of an attack increases with the progress of an attack through its steps. Detection at later stages reduces the time available for counter actions and increases the chances of occurring information breaches. Therefore, attacks should be detected as early as possible. Business aspects are also posing limits on the design of a detection system. Privacy concerns for example might come up with certain detection methods. Or the cost of a system might become too high with certain design choices, making the system unattractive.

B. Application of framework

Applying the framework to APT attacks starts with choosing the number of attack steps. Here, eight steps have been chosen. These eight steps describe distinct activities when looking at the goal of the steps. The steps are similar to the seven steps determined by others [7], [2], [8]. The first step is external reconnaissance. The second step is gaining access to the network. The third step is internal reconnaissance. The fourth step is expanding access by obtaining administrator rights for example. This step could be performed simultaneously with the third step. The next step is gathering of information on a single location in the network and preparation for extraction: the actual sending of the gathered information to a location outside the network is a separate step because it has a distinctly different goal and has a high impact. The seventh and eighth steps are control activities during the execution of an attack and are carried out from the moment an attack has been identified.

The content of the columns four to eight is composed based on the results of the attack analysis in the first three columns. For example: Emails with a link to a website that contains malware can be used to gain access to a network. Emails can be actively scanned to see if they contain links. Emails can be scanned at different locations in the network; on workstations, mail proxies or as network traffic. These choices of location provide the possible choices for detection methods. These methods in turn can use different analysis methods. The result is that the framework provides options for the design of an APT detection system. For an example with more information on the filling-in of the framework we refer to [19], that contains other and much more information than could be provided in this paper.

IV. A ROADMAP FOR DESIGNING APT DETECTION SYSTEMS

This section proposes a roadmap for designing intrusion detection systems that use intelligent data analysis to detect sophisticated APTs (SAPTs).

A. The Framework Used as Development Roadmap for Design

The analysis framework presented in section III gives insight into *what* needs to be detected, *where* it can be detected, *how* it can be detected, and *why* it needs to be detected. The concrete insights obtained influence the design of an APT detection system. The attack related columns of the framework answer *what* needs to be detected; the steps of an APT attack, the methods that can be used, and the attack features that can be detected. The detection location column of the framework contains the information *where* the attack related features can be detected. Combinations of attack features and detection locations limit the choices of detection methods and analysis methods. The question of how to detect is therefore affected by the answers to the 'what and where' questions. The detection and analysis methods columns contain the possible answers to the question of *how* to detect attacks. *Why* attacks need to be detected is answered by considering the business aspects. The motivation for detection also provides limitations to choices on analysis and detection methods.

The above-given considerations resulted into the insight that the proposed analysis framework can also be used to define a roadmap for designing sophisticated intrusion detection systems. We have visualized this idea in Fig. 2 and will elaborate it in the following sub-sections.

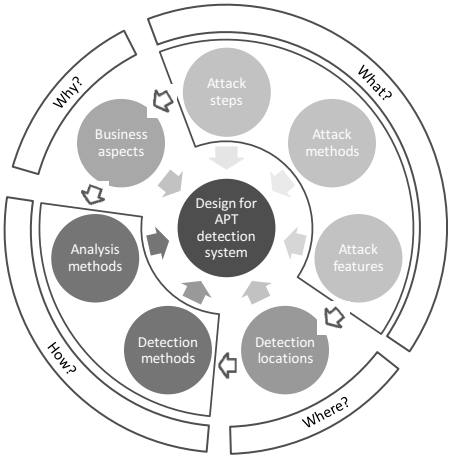

Fig. 2. The analysis framework columns as input for ID system design.

B. Roadmap-based system design, general aspects

The proposed roadmap consists of four questions that will be explained below.

1) What must be detected?

This specifies the type of attack. As said, APTs are multistep attacks in which each step has a different purpose and uses different attack methods. Distinguishing these steps provides an overview of the progress of an attack. Combining events also offers a means to identify an APT amongst more

common attacks. As was also already mentioned above, APTs can use known attack methods, but they often (also) use zero-day exploits. This prevents detection by common defenses. However, changes in behavior of successfully attacked clients or servers could be detected. For example, different behavior can consist of an unusual change in access frequency to data sources or in the number of connections to the Internet.

2) Where can APTs be detected?

Research on distributed systems for detection of complex attack scenarios, like APTs, shows that multiple analysis methods and correlation on their results is the most successful approach for detection [16]. Therefore, for capturing network data multiple probes should be deployed capturing traffic in different physical network segments. Probes are the system elements that gather data. A probe can be a physical device to capture packets on a network, but also be software that looks at program behavior in memory.

Warnings and data from the different analysis elements needs to be combined and analyzed to detect APT attack sequences. Already processed data gathered centrally minimizes the amount of network traffic, but does introduce a single point of failure. Redundancy (duplication) of a central analysis system can reduce the risk of failure but increases the costs of the system. Using the local analysis elements to look for APTs requires sharing of all warnings between all local elements. This increases the amount of network traffic dramatically and might not be possible on workstations due to performance issues. An alternative is to let local analysis elements look for parts of attack sequences visible within their own data. The result is that sequence analysis is also partly distributed across the network reducing the impact of a failure of the central analysis element.

3) Why should APTs be detected?

The economic damages due to a successful cyber attack can be very high. The expected financial impact of attacks is the main influence on investments in security measures [17]. Research by Iheagwara et al. shows that the return on investments in intrusion detection depends on the system's ability to reduce the occurrence and the impact of an attack. This ability depends on the system design and the choice of analysis methods [15]. Their research shows that a system with multiple sensors covering all physical network segments gives the best detection result. Others have shown that the application of multiple data analysis algorithms further improves the detection rate in distributed systems [16]. The effectiveness of the system, which is its ability to detect attacks, needs to be high. This should be combined with a high accuracy resulting into a low number of false warnings since, otherwise, people working in Security Operating Centers (SOCs) tend to pay less attention to alarms that go off. A distributed design with multiple algorithms is supported when taking an impact reduction point of view. The costs of such a system, on the other hand, might become too high. The maximum accepted cost of a system can be calculated by a cost/benefit calculation [15]. Theoretically, the expected financial impact of potentially occurring attacks is the maximum justifiable amount to be invested if these attacks can be prevented by the system. The result is that the prevention of high-impact attacks like APTs warrant higher investments.

4) How should APTs be detected?

Detection of unknown attack methods that are popular in APTs do require anomaly detection. Section II showed that anomaly detection for intrusion detection is still suffering from a large number of false positives, especially when unsupervised learning algorithms, algorithm which are trained without human intervention, are used. Unsupervised learning methods eliminate the need for training dataset creation and can add to the detection by signatures. An advantage of unsupervised methods is that they adapt their view on what is normal with changes in network use. This also brings a risk: An attacker can train the algorithm by slowly starting the attack letting the algorithm get used to the attack related traffic patterns [9]. Anomaly detection by supervised learning algorithms tends to perform better, but requires attack free or labeled datasets for training. Creating such a dataset for each installation and for each local analysis element is time consuming manual activity. Signature detection has proven to be reliable and capable of detecting attacks based on general signatures [9]. Using human made signatures as a baseline method ensures a reliable system without high installation costs. Most signatures can be reused in different installations spreading the costs of signature creation over multiple systems.

a) Anomaly detection data analysis

Anomaly detection methods can use data that describes behavior for unsupervised learning methods. By applying clustering algorithms, this can be done through comparing behavior of network clients. This approach often creates false positives if the input data from the probes contain clients with different patterns of normal behavior. For example, a client that behaves differently might belong to a different department. Knowledge of the network and careful choice in placements of probes can prevent such problems.

Possible clustering algorithms that have shown good results are *k*-means clustering and self-organizing maps. To prevent false classifications, semi-supervised methods can be used. Semi-supervised methods use a limited number of labeled events instead of completely labeled training sets. The labeled events should identify the different clusters and create a start for clustering algorithms [9].

Detection of anomalies in a central analysis element is more difficult. The warnings created by matching signatures and changes in behavior have to be combined to identify possible APTs. The large number of possible sequences of low-level attack methods in multistep attacks like APTs makes it hard to identify sequences of events that belong to the same attack [16]. The consequence of the large number of possible sequences is that it is harder to define normal behavior. Unsupervised learning by clustering algorithms can still be used to identify sequences of anomalous behavior but they will generate a high number of false classifications. The false classification rate can be improved by combining the results from different clustering algorithms like shared nearest

neighbor and *k*-means. Event sequences classified as anomalous by both algorithms have a higher chance of being a true positive than those that are classified as anomalous by only one algorithm. Such a voting approach is called boosting [9]. More complex approaches [7] use knowledge about lower level attacks to correlate events to create attack scenarios.

b) Other applications of intelligent data analysis

Intelligent data analysis can also be used to improve the performance of signature detection and to automate the creation of signatures. Examples are the creation of decision trees for rule application to reduce the analysis time when there are a lot of rules in the system [18]. Another option is to implement rule-learning approaches. An example is fuzzy rule-based anomaly detection [9]. This approach uses labeled datasets to create rules that define the clusters of normal and anomalous behavior. The labeled dataset can be derived from data clustered by a clustering algorithm The accuracy of this dataset can be increased by using decisions on reported alerts to manually label the data or clusters. This approach could improve the accuracy of the local analysis elements of a system.

C. Roadmap-based system design, concrete aspects

To give the above-given considerations concrete form, we here introduce a set of four basic elements a sophisticated intrusion detection system can be built from. The four elements are (*i*) a probing element for gathering data, (*ii*) a low level analysis element for analyzing data locally, (*iii*) a high level analysis element to globally analyze data (i.e., data collected from various locations of the computer network at stake that are analyzed centrally to draw conclusions at global system level), and (*iv*) a reporting element to inform SOC workers in appropriate ways on what is going on by, for example, using a set of dashboards. A basic architecture of an ID system capable to detect ATPs is given in Fig. 3:

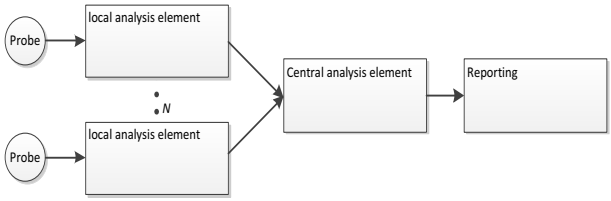

Fig. 3. A basic architecture of an ID system capable to detect ATPs.

Local analysis elements might be rather simple or very sophisticated depending on the precise functionality required (see previous sub-section). A basic architecture for such an element is shown in Fig. 4:

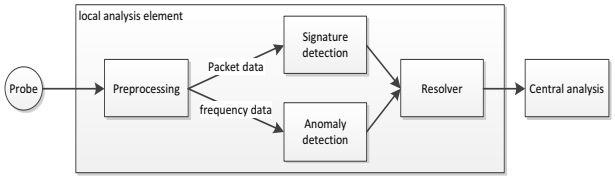

Fig. 4. A basic architecture of a local analysis element.

A central analysis element is by definition rather sophisticated but further depends on the precise functionality required (see also the previous sub-section). A basic architecture for such an element is shown in Fig. 5:

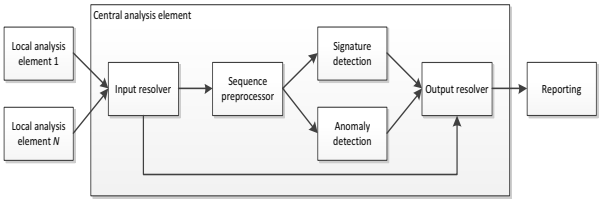

Fig. 5. A basic architecture of a central analysis element.

It should be clear that in complex network environments the four basic elements introduced should be combined in smart ways to create an effective, robust and business-related ID system for detecting APTs. This is not further elaborated here but instead, a test case is introduced below showing (*i*) an example APT attack for a basic computer network, and (*ii*) a description of how this attack can be detected using ideas from the design roadmap described above.

V. A TEST CASE

We start by sketching an example APT attack as might occur in a given basic computer network containing only three network segments. The structure of this example APT follows the structure of eight steps discussed before. It employs similar approaches as used in the Operation Aurora attack. The first segment, in the basic network, is the public network or the Internet. The second segment contains public services like web servers. These can be reached from the public network and internal network, but do not have access to other internal network segments. (This segment is often called a DMZ.)

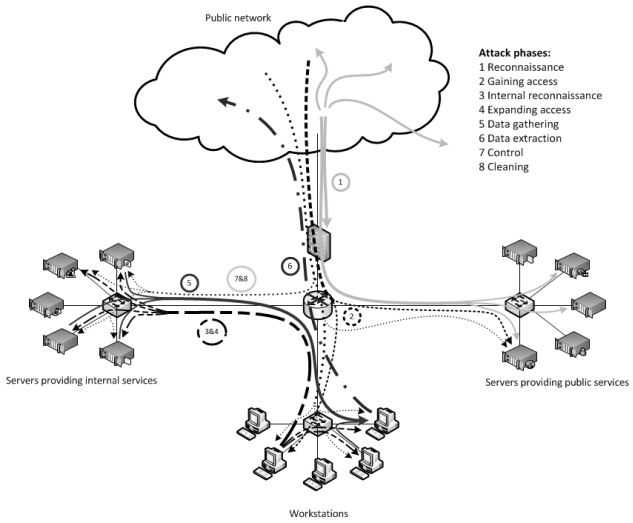

Fig. 6. An example APT attack as might occur in the network shown.

The workstation segment represents segments in networks containing workstations and other network clients that do not provide services. While the attack is taking place, in each step of the eight mentioned steps, deeper parts of the network are reached. The arrows in Fig. 6 show these advances through the network. The related traffic streams are mixed with the normal traffic streams making them possibly hard to detect. However, the attack related changes in traffic content, volume and destinations can be used for detection purposes.

Now we will describe in detail - by using the roadmap design issues introduced in section IV – how the APT can be detected and be reported upon.

Step 1. Reconnaissance: The first step of all attacks is reconnaissance of the target organization. An attack starts by browsing corporate websites for names and mail addresses, checking DNS registrations to find public accessible services and checking the open web for social media profiles of people claiming to work at the target company. The main goal is to find handles for social engineering approaches and version information on servers and website content management systems, to find exploitable vulnerabilities.

Activities in this step are performed in open source on the Internet and are therefore outside the monitored network of the target company. Browsing of the corporate website can be logged, but this does not give any information on attacks since this is considered to be normal behavior. Providing information about the company is what the corporate website is for. Other scan methods of the web site or other services can be detected because they deviate from normal visits. However probing public available services is done in such a frequency a specific attacker is very hard to distinguish in this step. Possibly a correlation can be made with later found evidence. This will however not detect the attack.

Step 2. Gaining access: After the first step the attackers proceed to use the profile information of employees to construct phishing emails, which look legitimate. These emails contain a link to an infected website which uses a zero-day exploit to install a malware component on the victim's workstation.

This step is the first one that generates specific detectable events. The signature detection element in the first probe detects the presence of the hyperlink in the email. The mail is therefore flagged and relevant information, like the link address, addressee and source is recorded. The system starts looking for traffic to the address in the link. This is also recorded with relevant information like the address of the internal workstation from which a connection to the linked website is initiated and a timestamp. The malware installed from the website is not detected by the signature detection block in the local traffic analysis element because of the lack of a signature. The recorded information from both the mail and the website visit is reported to the central analysis system as separate events. This system logs them as a possible attack, but does not report it since the probability of this series of events being normal is too high. The execution of the malware binary on the workstation can be detected as suspicious because the binary was not seen before. Automated binary analysis could further determine the use of suspicious function calls or obfuscated parts. These are also reported as separated events.

Steps 3&4. Internal reconnaissance and expanding access: Once the attackers have gained a foothold in the network through the malware, they will try to expand their access to other parts of the network. The malware starts to monitor connections to servers in the network, gather information about installed programs and network users to identify server addresses, network structure and possibilities for expanding access. Un-patched programs or operating systems create possibilities for further expansion of the attackers' access to network clients and servers. The attackers also perform active reconnaissance on the network themselves by connections through the malware clients.

These steps generate traffic over the internal network (like connections to internal services) and traffic to the Internet (like command and control traffic to the malware). The signature element in the probes can detect port scans characteristics and other known reconnaissance techniques. Such behavior might be found if the attacker has access to a workstation of an employee that does not use the financial applications while the attacker looks for a means to connect to these applications.

The anomaly detection block using traffic data frequencies identifies the unusual traffic between clients and servers raising another possible suspicious event. Individually, this event might not be suspicious. It could be legitimate traffic which just doesn't occur often. But reported to the central analysis element, it is linked to the previous events resulting into a sequence of events which could be classified as an attack. A low level warning is raised indicating that an attack might be ongoing and manual analysis is desired.

Expansion of user privileges on an infected workstation or infections of clients in the same segment as the infected machine goes unnoticed by the probe since the first does not create suspicious traffic (assuming that the command and control traffic to and from the internet is obfuscated). The latter activity generates traffic which does not pass the probe and can therefore not be detected.

Steps 5&6. Gathering and extracting information: After a while the attackers are successful and have found the wanted technical documentation and have access to the financial systems of the target. They slowly gather all the information on one of the clients they control and prepare the information for extraction. Finally they extract the information to a legitimate file storage application on the Internet to make the extraction look as normal as possible. They also continue snooping around for other data they can sell and extract this as well.

The anomaly detection element will observe a change in traffic volume in step 5. This is classified as suspicious and reported. In the central analysis system this increases the warning level. This does require that the increase of volume is on the system which was initially flagged by the email. If this is not the case, the system will still report the anomaly but it

won't be linked to the already present warning in the central system.

A change in traffic volume to the Internet from a workstation is classified as an attack and is given a high warning level in the central system. This might be legitimate traffic, but in such a case it is preferable to have a false positive then a false negative. The chance at a false positive is lower when there is a direct link to other steps or in combination with other aspects for example, if the workstation has no other activity that day indicating the employee has a day off. Weaker links, like individual steps, reported with different workstations involved but in the same network segment also reduce the chance of a false positive. Such relations can be identified by the signature element in the central analysis system. If the uploading to the Internet is done from internal services it can be immediately flagged as suspicious by both signature and anomaly detection since this can be considered to be uncommon behavior for almost all internal services.

Steps 7&8. Control and erasing tracks: The attackers have continuously monitored progress through direct access through a backdoor created by the malware and by updates from the malware to servers on the Internet. After extraction of the last of the wanted information, the attackers hide their tracks by uninstalling the malware.

The removal of the malware will be unnoticed by the network probes since it occurs on the workstations. The command and control traffic of the malware clients and the traffic generated by the activities of the attacker through the malware could be detected by the network probes. Signature detection will only work if such traffic contains known suspicious elements. Statistical profiles have a better chance at detecting such traffic as unusual especially when the communication is encrypted. It is probably hard to distinguish such a change and this would therefore only justify a warning when at least steps two or three are detected.

VI. Validation

The test case provided in the previous section can be considered as a first step in validating the framework: it shows the potential power of the proposed roadmap for the development of systems capable to detecting SATPs in the future.

In addition to that, some experts in the field have been consulted to give their opinion about the proposed approach. Conversations with those experts during the research project confirmed that the chosen design approach is a possible way to go [19]. E.g., probing at strategic locations in networks to capture traffic characteristics between different segments is considered a good approach needed to get an overview on the situation. In practice however, the number of probes to be installed in a network is not only determined by technical means but also by customer wishes and financial constraints.

The application of IDA methods for detection of attacks is a point to which some experts remain sceptical. On the other hand, application of such methods in the central analysis element of the system is considered to have high potential. The overall consensus is that IDA methods may give added value, especially in statistical profiling. According to the opinion of most experts, the classical (and most commonly used) signature approach remains to be the core technology in detection. We would like to observe here that this experts' opinion might be somewhat biased because the interviewed experts have learned to work with and depend mostly on signature based systems. Their experience with other methods, especially those based on IDA, appeared to be very limited, which contributes to their current sceptism.

True validation can only be achieved by implementing systems based on the proposed development roadmap. This is what is currently undertaken in the company where this research started.

VII. Conclusions

APTs are a new, more sophisticated, version of known multistep attack scenarios. These APTs form a problem for current detection methods because these methods depend on known signatures of attacks and APTs make heavy use of unknown security holes for attacks. The approach presented in this paper uses a framework for analysis of attacks which links low level attack methods to detection methods and intelligent data analysis methods. The framework is used as a roadmap towards a system design capable of detecting APTs.

Applying the framework in the described way results in a design in which detection methods are being selected based on appropriate analysis of occurring APT behavior (changes). Taking business aspects as well as attack related aspects into account, a layered system design and the use of multiple analysis algorithms turns out to be the most natural choice. Signature detection is used to provide a more accurate detection of known attacks. Anomaly detection is necessary to detect unknown attack methods, which remain undetected by signature detection. An important problem with anomaly detection is that is has a relatively high false positive rate. Methods to increase the accuracy, like boosting, can be used as one way to reduce the number of false positives. Next to the local low level analysis elements, redundant global high level analysis elements are needed as well in order to link together suspicious behavior found in various attack steps. It is assumed that this will further reduce false positive rates. Despite of this, human analysis of warnings is assumed to remain necessary.

Three recommendations for future research can be made. First, the features used for analysis are determining whether an attack can be detected based on anomaly detection algorithms. Preprocessing of data is therefore perhaps the most important step in detection. Research into good features for detection is needed to improve anomaly detection. Second, the design approach in this paper still required analysis of alerts by experts. Creating a better user environment requires more research into the activities of these experts. Questions like: "What kind of information do they require" and "When do they require this information?" should be answered to create an adaptive user interface. Finally, a new reference dataset for research in intrusion detection is needed to get more relevant

information on the success rate of algorithms. Data from real incidents would be very useful to create such a dataset. Attacks are constantly changing, especially APTs, making a representative dataset hard to create.

REFERENCES

[1] N. Kshetri, The global cybercrime industry: economic, institutional and strategic perspectives, Springer, 2010.

[2] C. Tankard, "Persistent threats and how to monitor and deter them," Network security, vol. 2011, no. 8, pp. 16-19, 2011.

[3] Symantec, "Symantec Internet Security Threat Report," Symantec, 2011.

[4] V. Igure and R. Williams, "Taxonomies of Attacks and Vulnerabilities in Computer Systems," IEEE Communications Surveys & Tutorials, vol. 10, no. 1, pp. 6-19, 2008.

[5] P. Ning, Y. Cui and D. Reeves, "Constructing attack scenarios through correlation of intrusion alerts," in Proceedings of the 9th ACM conference on Computer and communications security, New York, 2002.

[6] S. Cheung, U. Lindqvist and M. Fong, "Modeling Multistep Cyber Attacks for Scenario Recognition," in Proceedings of the DARPA Information Survivability Conference and Exposition, Washington, 2003.

[7] S. Yang, A. Stotz, J. Holsopple, M. Sudit and M. Kuhl, "High level information fusion for tracking and projection of multistage cyber attacks," Information Fusion, vol. 10, pp. 107-121, 2009.

[8] GOVCERT.NL, "National Cyber crime and Digital Safety Trend Report", GOVCERT.NL, The Hague, 2011.

[9] S. Dua and X. Du, Data mining and machine learning in cybersecurity, Taylor & Francis Group, 2011.

[10] M. Tavallaee, N. Stakhanova and A. A. Ghorbani, "Toward credible evaluation of anomaly-based intrusion-detection methods," IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, vol. 40, pp. 516-524, 2010.

[11] S. Mukkamala and A. Sung., "A Comparative Study of Techniques for Intrusion Detection," in 15th IEEE International Conference on Tools with Artificial Intelligence, Sacramento, 2003.

[12] S. Wu and W. Banzhaf, "The use of computational intelligence in intrusion detection systems: A review," Applied Soft Computing, vol. 10, pp. 1-35, 2010.

[13] P. Garcia-Teodoro, J. Diaz-Verdejo, G. Macia-Fernandez and E. Vazquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," Computers & Security, vol. 28, pp. 18-28, 2009.

[14] J. Davis and A. Clarck, "Data preprocessing for anomaly based network intrusion detection: A review," Computers & Security, vol. 30, pp. 353-375, 2011.

[15] C. Iheagwara, A. Blyth, T. Kevin and D. Kinn, "Cost effective management frameworks: the impact of IDS deployment technique on threat mitigation," Information and Software Technology, vol. 46, pp. 651-664, 2004.

[16] C. Zhou, C. Leckie and S. Karunasekera, "A survey of coordinated attacks an collaborative intrusion detection," Computers & Security, vol. 29, pp. 124-140, 2010.

[17] T. Rakes, J. Deane and L. Rees, "IT security planning under uncertainty for high-impact events," Omega, vol. 40, pp. 79-88, 2012.

[18] C. Kruegel and T. Toth, "Using Decision Trees to Improve Signature-Based Intrusion Detection," in RAID 2003, Pittsburgh, 2003.

[19] J. de Vries, "Towards a roadmap for development of intelligent data analysis based cyber attack detection", Master's Thesis, Delft University of Technology, July 2012.