An analysis framework to aid in designing advanced persistent threat detection systems

J.A. de Vries[a], J. van den Berg[a], M.E. Warnier[a], H. Hoogstraaten[b]

July 5, 2012

ABSTRACT

Cyber-attacks against companies and governments are increasing in complexity, persistence and numbers. Attackers take more time and effort to remain undetected than previously known multistep attacks. Common intrusion detection methods lack in their ability to detect such complex attacks. A new approach to detection is therefore needed which takes the stepwise characteristics of these new threats into account and which links analysis methods to attack features. An analysis framework is proposed to relate attack aspects, like attack steps and features to detection and business aspects. The framework can be used as a roadmap towards a detection system design. Using the framework as a roadmap results in a system design which analyses network traffic and client data on multiple locations in a network. These analyses are performed with signature and anomaly detection methods.

1. INTRODUCTION

Computers have become part of our everyday lives and the internet is connecting users and companies to each other on a worldwide scale. Malicious activities on cyber infrastructures date back to the eighties and resulted in defenses against viruses and unauthorized access. 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].

The emergence of viruses, worms and other malicious activities on the internet and its precursors resulted in the creation of defenses. 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. This rat race is still continuing today. Attackers are targeting specifically and try to remain undetected while they look for proprietary information. These attacks are often called Advanced Persistent Threats (APTs). An APT is a new form of multistep attack which is executed with more stealth and is targeted specifically to achieve a specific goal, most often espionage [2]. Just as normal multistep attacks are APTs also executed in different steps to obtain their goal, but APTs are different because attackers make more use of zero day exploits, which are unknown security flaws in software, and other advanced means like social engineering [2]. APTs are currently the largest threat to companies and governments because detection of APTs is often failing in current defenses [3].

This paper proposes a new way of analyzing multistep attacks like APTs with the aim of linking attack characteristics to detection methods like network intrusion detection systems (NIDS) or host intrusion detection systems (HIDS). The intelligent data analysis algorithms in these methods form the key in detecting activity related to cyber attacks. The proposed framework considers aspects of attack methods, detection methods and impact on business. The lessons learned from the framework have been applied to design a system to detect APTs.

This paper is organized as follows. Section 2 gives some background on multistep attacks, especially APTs, and on the current applications of intelligent data analysis methods in intrusion detection. Section 3 introduces the framework proposed for analysis of APTs. Section 4 presents an approach to detection of APTs with the use of intelligent data analysis methods. A reflection on the design approach is given in section 5, after which section 6 concludes this paper.

2. ATTACKS AND DATA ANALYSIS METHODS

2.1 Cyber attacks

Computer networks contain a lot of information. Much of this information is protected to ensure confidentiality, integrity and availability. Intentional breaches of security are called cyber attacks. Cyber attacks exist in many different forms. They range from simple denial of service attacks to complex cyber espionage attacks. Analysis of attacks is a continuous process to keep up with the ingenuity of attackers. Attack taxonomies can help in categorizing attacks based on their characteristics. These characteristics of attacks and attack families can be used to design signatures for detection [4]. Taxanomies can also be used as checklist in the development of intrusion detection system that look for these signatures by ensuring coverage of all known attacks. Attack taxonomies do not necessarily focus on single low level attack methods like virusses or vulnerability exploits. Taxonomies can also contain high level attacks which are sequences of low level attack methods [4]. Defensive systems like virus scanners and firewalls are used against the low level attacks. More elaborate cyber attacks consisting out of multiple steps are being researched since the beginning of the century. The steps in these multistep attacks containt different attack methods to achieve a step specific goal. Reconnaissence steps, for

[a]Faculty Technology, Policy and Management, Delft University of Technology, Netherlands

[b]Fox IT, Netherlands

example, can use portscanning as attack method. In scientific literature these attacks are commonly called multistep attacks or attack scenarios. Ning et al. for example tried to correlate low level attacks to reduce the number of warnings given by intrusion detection systems [5]. Chueng et al extended on this idea by using information from different systems aimed at low level attacks to detect multistep attack scenarios [6]. Another detection approach to multistep attacks is by Yang et al. who used data fusion on alerts from multiple intrusion detection systems to identify multistep attacks. They also presented an attack guidance template with seven attack stages. The first stage contains recon attacks from an external network, their final stage is reaching an attack goal on the internal network [7].

2.2 Advanced Persistent Threats

The approaches to multistep attacks mentioned above all assume that most, or even all, steps of an multistep attack are detected. The emergence of a new breed of multistep attacks, often called Advanced Persistent Threats, can be considered to be a new form of multistep attacks [2]. These attacks differ from the attacks analyzed in the previous paragraph 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. APT are generally considered to have a reconnaissance step, steps to gain a foothold in a network, steps to look for resources and finally 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 for common defenses to detect the attack [2] [3].

Defending an organization against APTs requires in the first place to keep software and 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.

2.3 Intelligent data analysis in intrusion detection

In general there are three different approaches to intrusion detection [9]. The first approach is signature detection. A signature detection system compares a data sample to the signatures in the system and when a signature matches a warning is issued. Such systems are reliable and have a low false positive rate. (A false positive is a classification error stating that an attack occurred while it was not the case. A false negative is the opposite; no warning of an attack is given while an attack is happening.) The problem is that such systems are not really capable of detecting unknown attacks [9].

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 attacks. 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 possibly what attacks are. These methods are considered to be relatively successful without too many false classifications. The second group of methods are unsupervised learning algorithms. These methods use unlabeled data to find anomalies. These methods 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].

2.3.1 Single method approaches to anomaly detection in literature

A study on anomaly detection methods in intrusion detection by Tavallaee et al. shows that classification methods are the most commonly used methods for anomaly detection. The most commonly used classification algorithms used are Neural networks, Hidden Markov Models, Support Vector Machines and Bayesian networks. Other method categories identified are statistics based methods, clustering methods and a group miscellaneous [10]. The success rates of the application of anomaly detection methods in research are often above 95% accuracy [11] [12]. This success is primarily because classification methods with supervised learning are used on the well known DARPA 99 dataset which was created in 1999. This dataset provides a labeled dataset with attacks and one without attacks for training. The result is that more accurate supervised learning algorithms can be used for anomaly detection without creating a labeled dataset which is expensive. The results of methods tested against the DARPA dataset are disputed because the DARPA dataset is considered to be outdated and is as such less reliable as benchmark [13]. Other methods based on statistics like frequency time series data or clustering methods are less popular but they can be used for unsupervised learning. Popular clustering methods in literature are Shared Nearest Neighbor, k-Means and Self-Organizing Maps.

The biggest challenge in applying machine learning algorithms to anomaly detection problems is the choice of data and data features for analysis. The choice of data type, for example IP packet data against stream data, determines if attacks can be detected. This is even more important in the choice of data features, for example addresses, protocols, duration, etc. for traffic data. A high number of data features slows down analysis while too few can make attacks undetectable [14].

3. ATTACK ANALYSIS FRAMEWORK

Section two introduced taxonomies as a means to create structured lists of attacks which can be used in the development of intrusion detection systems. The zero-day exploits and other advanced means used in APTs can be placed in these taxonomies as members of certain families but this generally does not provide enough information for the signature based detection systems supported by taxonomies. A new framework for analysis is therefore proposed to give better insight into the structure of APTs and the detection of APTs. The structure and attack methods used in APTs are used to determine the structure of the framework. The number of steps and the attack methods used provide detectable features and possible detection locations of the features.

Possible detection and analysis methods are placed in the framework in relation to the attack aspects and the possible detection locations. Finally business aspects of the attack aspects and detection methods were added to the framework to capture the influence of business aspects on the design of APT detection.

3.1 Analysis framework

The new framework proposed in this paper contains seven columns. (Figure 1) The framework tries to give a means of analysis of attacks linked to detection. The first three columns contain attack related aspects. They provide a detailed description of the attack. This description provides features for detection which are input for the detection related columns. The first column contains the different steps of an attack. The number of steps in this column determines the number of rows in the framework. The second column contains low-level attack methods used per attack step. The third column contains features of the attack methods in the second column. These features 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 changes in behavior of systems which might be expected. The content of the columns in the framework should be ordered so that information in the different columns can be related to each other: Attack features to attack methods, locations to attack features, etc. Drawing a tree like structure in the rows could be used to make the relations between the columns more visible.

The fourth column contains possible detection locations of features. These might be, for example, in a DMZ, in a server log or on workstations. The location determines the possibilities for detection methods and the input data for analysis methods used for detection of attacks. Some attacks might have multiple detectable features giving a choice for detection locations and/or detection locations. This can be useful knowledge when designing a detection system.

The fifth and sixth columns contain detection aspects. The fifth column contains the detection methods. Methods are for example network intrusion detection, host intrusion detection or log analysis. The sixth column should be filled with analysis methods used in the detection methods. 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 the input data for the analysis algorithms. The sixth column indicates the methods which can 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 right hand side of the column indicates that the possible impact increases with the progress of an attack through its steps. An attack should be detected in as early a state as possible, but detecting a later step reduces the time available for counter actions and increases the chances of information extraction. The impact can be seen as an incentive for detection. Business aspects are also posing limits on the design of a detection system. Privacy concerns might for example come up with certain detection methods. Or the cost of a system might become too high with design choices making the system unattractive.

3.2 Application to APTs

Applying the framework to APT attacks starts with choosing the number of attack steps. For this paper eight steps were 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 yang et al. [7] and the steps described by Tankard [2] and GOVCERT [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 a higher impact. The seventh and eight steps are activities concerned with controlling and executing the attack and measures to prevent detection of attack methods used. The last two steps are active during the entire attack.

The content of the other columns is build on the attack analysis in the first three columns. For example: Emails with a link to a website which 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 choice of location provide a choice of detection methods.

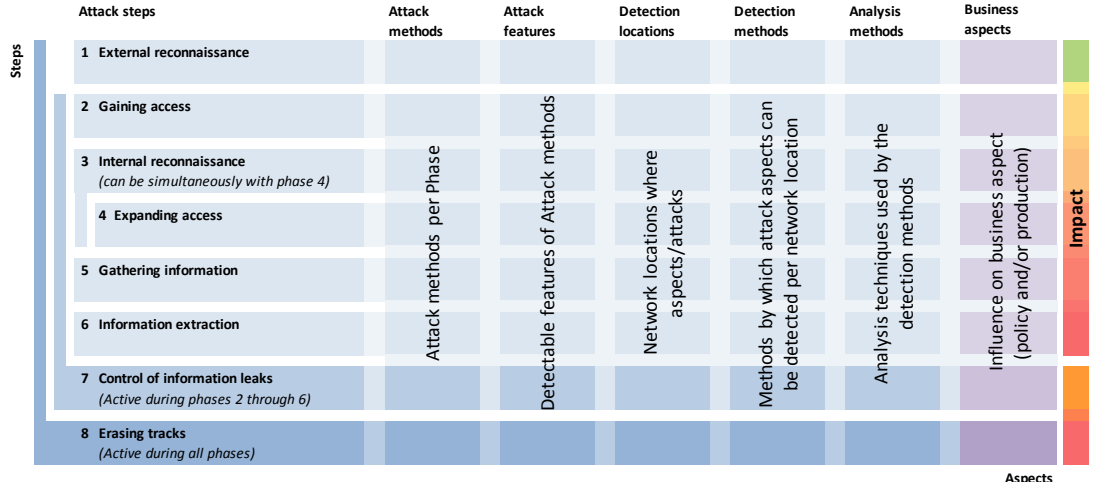

Figure 1; overview of the framework

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.

4. APT DETECTION DESIGN

The previous sections showed that multistep attacks like APTs can be analyzed per step and for each step different attack methods are possible. APTs are different from multistep in the sense that they often use unknown exploits and they approach their targets carefully and selectively. Attackers involved in APTs try very hard to remain undetected. The result is that common detection methods run the risk of missing an APT. This section proposes an intrusion detection system using intelligent data analysis to detect APTs.

4.1 The framework as roadmap for design

The framework presented in the section 3 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 what, where, why and how questions and their impact on each influences 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 influenced by the answers to the what and the 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 the business aspects. The reasons for detection given by the business aspects also act as limitations to choices on analysis and detection methods.

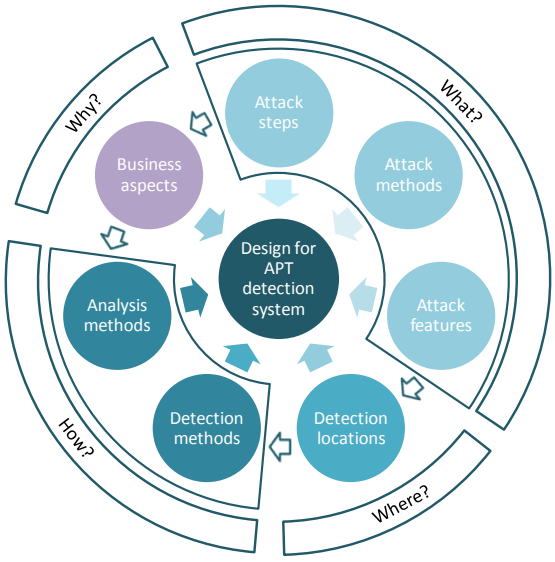

Figure 2; framework columns and their design input.

Using the framework to analyze attacks and possibilities for detection creates a roadmap for the design of an APT detection system. The answers to the what, where, why and how questions are input for a system design. This approach to a system design uses the attack analysis as the driver for design choices and helps to support design choices while ensuring that the business requirements are met.

4.2 System design using the roadmap

The following section uses the framework as a roadmap to design a system to detect APTs. The four questions from Figure 2 are used as design steps.

4.2.1 What must be detected?

Section 2.2 presented APTs as a new and advanced threat against which current defenses are not adequate. The design in this section should be able to detect APTs. 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 provides a means to identify an APT amongst more common attacks. APTs can use known attack methods but they often use zero day exploits, which are unknown, to gain access. This prevents detection by common defenses, but a change in behavior of successfully attacked clients and servers could be detected. Different behavior can mean a change in access frequency to data sources or connections to the internet which are unusual. Differences in behavior can be in network usage but also in software on workstations and servers. Not all attack methods used by APTs generate network traffic. Privilege escalation attacks on workstations for example do not necessarily generate network traffic.

To find APTs one should not only look at known attack methods but also a behavior differences. Not only network traffic, but also network clients should be monitored for attacks.

4.2.2 Where can it be detected?

Research on cost effective network intrusion detection systems shows that multiple sensors on a network gives the best change at detecting an attack [15]. 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]. APT attacks are advancing deeper into a network with each step. Gathering data at different locations increases the changes of detecting different steps of an APT. Detection locations can be somewhere in the network or on workstations and servers. For capturing network data multiple probes should be deployed capturing traffic in different physical network segments. Probes are the system elements which gather data. A probe can be a physical device to capture packets on a network, but a probe can also be software which looks at the behavior of programs in computer memory.

Performing behavior analysis and signature detection locally creates distributed system without a single point of failure. It also eliminates the need for a high performance system capable of handling all data gathered by all probes. The local analysis elements produce warnings about detected signatures and suspicious changes in behavior. These warnings need to be presented to security analysts. But they also need to be analyzed for attack sequences indicating ongoing APTs. Warnings and data from the different analysis elements needs to be combined and analyzed to detect APT attack sequences. Gathering the data centrally minimizes the

amount of network traffic, but does introduce a single point of failure. Redundancy of the central analysis element can reduce the risk of failure but increases the costs of the system. Using the local analysis elements to look for APTs requires the 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 which are 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.

4.2.3 Why?

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]. The return on investments in intrusion detection depends on the system's ability to reduce the impact of an attack. This ability is depends on the system design and the choice of analysis methods according to Iheagwara et al. [15]. Their research shows that a system with multiple sensors covering all physical network segments gives the best detection result. Zhou et al. show that using 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 as high as possible. This should preferably be combined with a high accuracy, which means that the systems gives a low number of false warnings.

A distributed design with multiple algorithms is supported from a reduction of impact point of view. The costs of such 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 this means that the expected financial impact is the maximum amount to be invested if attacks can be prevented by the system. The result is that attacks like APTs, which have a high impact, warrant higher costs. But this is limited on the one hand by the accuracy of the system and the impact of attacks itself.

Capturing and creating behavior patterns is considered to be invasive to the privacy of network users when it shows personal information or behavior of individual users. A system design needs to deal with these privacy issues if such a detail level is necessary for detection of APT.

4.2.4 How?

The previous steps showed that a system which can detect APTs needs to look at known and unknown attack methods. The low level attack methods are executed at different locations in the network. Captured traffic can be used to find many attack methods, but there are also attack steps in APTs which do not necessarily produces any network traffic. The system design should also look for traces of APTs at workstations and servers. A distributed system gives the most effective and accurate results. This means that different data types need to be analyzed for which different analysis methods are needed. The efficiency and accuracy of the analysis methods need to be high enough to warrant investments in the system.

Section 2.3 showed that anomaly detection for intrusion detection is still suffering from a large number of false classifications, especially when unsupervised learning algorithms are used. Anomaly detection by supervised learning algorithms perform better but they require attack free or labeled datasets for training before they can detect anomalies. Creating such a dataset for each installation and for each local analysis element is hard to accomplish. 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 more reliable system without high installation costs. Most signatures can be reused in different installations spreading the costs of signature creation over multiple systems.

Detection of unknown attack methods which are popular in APTs do require anomaly detection. 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 [9].

4.2.4.1 Anomaly detection

Known attacks can be detected by signature detection. Changes in behavior can also be detected by describing normal behavior in a signature, but this requires many specific signatures which makes this approach unattractive. Anomaly detection methods can use data that describes behavior for unsupervised learning methods. This can be done for example by comparing behavior of network clients by means of clustering algorithms. This approach can create false classifications if the input data from the probes contains clients with different normal behavior. For example: A clients which 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].

Anomaly detection 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 which 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 which are only classified as anomalous by one. Such an approach is called boosting [9].

More complex approaches like the one proposed by Yang et al. [7] use knowledge about lower level attacks to correlate events to create attack scenarios. Yang et al. try to match alert sequences to known attack sequences and tries to match the results to information exposure sequences. The information exposure sequences are seven stages ranging from external reconnaissance to achieving an internal network goal. These stages are very similar to the eight steps from the framework defined in chapter two. Yang et al. state that alert correlation methods are still in the infancy state and that a lot of research is still required.
The approach of using knowledge about the structure of APTs by labeling the events from the local analysis according to the steps they belong to can help to create better event sequences for anomaly detection.

4.2.4.2 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 which define the clusters of normal and anomalous behavior. The labeled dataset can be derived from the clustered data from the anomaly detection block described above. The accuracy of this dataset can be increased by using decisions on reported alerts to manually label the data. This approach could improve the accuracy of the local analysis elements of the system.

5. REFLECTION ON THE FRAMEWORK AS A ROADMAP

Using the framework as a roadmap for a system design must result in a system that is able to detect APTs. This section reflects on the system designs ability to do so.

Creating a distributed system creates the possibility to analyze data from different data sources. Using both signature and anomaly detection in the analysis elements gives more possibilities for detection of attack features. The anomaly detection is needed to detect unknown attack methods. However, general signatures for zero-day exploits might be preferable because such rules show more clearly why an anomaly is reported. This would also make the system more reliable because the chosen unsupervised learning algorithms still have a relatively high number of false classifications.

Sequence analysis is crucial for the detection of APTs in the central analysis element. Research on the other hand shows that sequence analysis for multistep attacks is far from accurate at the moment [5] [7]. The system design is able to find individual steps of an APT link these to the correct steps. But there is no input from the framework on how to link low level attacks to high level attacks. Linking steps is relatively easy when different steps are detected within a small timeframe on a single location. The system can do this on the local analysis level. But on this is much harder on the central analysis level. Smart filtering of low level attacks to reduce the dataset helps to improve the results.

All warnings, from the local analysis elements and the central analysis element, are reported to experts for analysis. These experts can take the appropriate action. The decisions of these experts on the reported alerts can be used as input in the system to improve the accuracy of the system. This approach can be used to implement semi-supervised learning.
The dependence on experts also calls for an efficient user interface design. A more intelligent approach to presentation of alerts and data in the network improves the effectiveness of the system.

6. CONCLUSIONS

APTs are a new, more persistent and target, version of known multistep attack scenarios. These APT 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. Using the framework in such a manner results in a design which uses a selection of analysis methods based on an analysis of APTs. The result is that business aspects as well as attack related aspects point towards a distributed system design and the use of multiple analysis algorithms. 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. The problem with anomaly detection is that is has a relatively high false positive rate. The expected detection error is even higher for anomaly detection for high level attacks.

Anomaly detection remains necessary even with the high false positive rates. Methods to increase the accuracy, like boosting, can be used to reduce the number of false positives. But human analysis of warnings remains necessary.

The proposed framework helps to analyze attacks and to determine which analysis methods are needed for detection.

6.1 Research recommendations

The features used for analysis are determining if an attack can be detected by anomaly detection algorithms. Preprocessing of the data is therefore perhaps the most important step in detection. Research into good features for detection can therefore help to improve anomaly detection.

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. Attacks are constantly changing, especially APTs, making a representative dataset hard to create. The DARPA dataset on the other hand is more than ten years old and cannot be considered representative for today's attacks.

Bibliography

[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, "Nationaal Trendrapport Cybercrime en Digitale Veiligheid 2010," 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.