CHALMERS

Setting up and Fine Tuning a Security Operations Centre

Master's Thesis in Secure and Dependable Computer Systems

VASILEIOS FRILIGKOS

Chalmers University of Technology
Department of Computer Science and Engineering
Göteborg, Sweden, 2013

Setting up and Fine Tuning a Security Operations Centre

Challenges and best practices in deploying a fully operational Security Operations Centre

CHALMERS

Examiner: Tomas Olovsson

Chalmers University of Technology
University of Gothenburg
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden Telephone + 46 (0)31-772 1000

Abstract

Given the need for every company to be cost-effective, it comes as no wonder that Management does not wish to allocate many resources to information security while on the same time demands a perfect and exhaustive coverage of its infrastructure and applications. This paper will deal with possible ways to maximize the efficiency of a Security Operations Center (SOC), a specialized team responsible to centralize and manage the totality of security operations regarding an IT infrastructure, in order to protect proactively and respond, in real time, to security events.

Using a fully operational network of a company as the base for the experiments, multiple real life attack scenarios were reproduced in order to study the results, adapt the defensive mechanisms using the acquired feedback and present the gained experience. These results can be used as an exhaustive guideline for anyone interested in setting up efficiently a Security Operations Center. Through the best practices proposed by the paper, a security analyst will be able to adapt and fine tune a SOC to the specific context of the organization in question, while making sure that no critical elements are overseen or forgotten. Moreover, this paper will give answers about how to provide to the upper-management layers of an organization a service that will minimize security risks and mitigate security events while being cost-effective and efficient. Detailed descriptions of the necessary tools for the centralization, monitoring and resolution of security events as well as how should they be configured and fine tuned are also included.

Even though the description of all the procedures is exhaustive, through the conducted experiments it was made clear that in order to have an efficient SOC, which translates to a constant, realistic and reacting monitoring and protection of an IT infrastructure, a continuous and systematic procedure needs to be implemented in order to update, adapt and fine tune the techniques employed by the SOC, depending on the evolution of the demands, the needs, and all kind of changes associated to the organization in question. Therefore, there are no plug and play solutions that can be deployed and then forgotten, no matter the cost. Human expertise is always required and plays a crucial role to the whole procedure of protecting the IT infrastructure.

Table of Contents

1. Introduction

1.1 Background

In today's corporate environment, information technology (IT) security is becoming an ever more complex matter. Corporate infrastructure includes many heterogeneous assets and the main disadvantage defensive security experts have against attackers is that one unique point of exploitation is usually enough to generate negative results for the organization. Trying to implement the proper defensive mechanisms in order to mitigate these risks is a constant challenge for security experts. Moreover, the mere detection of an attack, a malware infection and propagation or any other type of malicious activity is a difficult task due to the constantly growing and evolving complexity of hardware and software. In combination with the rapid advance in the sophistication level of attacks and malware code, a need for a holistic and more importantly realistic view of the whole IT infrastructure has emerged. Attacks usually involve more than one component and will probably affect multiple assets. This is why multiple security solutions have emerged so as to provide a constant monitoring system that will detect any suspicious behavior and try to mitigate any attempted exploit. Intrusion detection/Intrusion prevention systems (IDS/IPS), host based security solutions as well as Internet protection technologies have appeared in order to cover the need for continuous and multi-layered security awareness.

1.2 Definition of a SOC

Inevitably, the problem of being over flooded with security data is becoming a real challenge to deal with. While heterogeneity of security solutions is required as explained above, it also means that warnings, alerts, actions or even just plain logs include a wide range of formats and standards. Moreover, almost every piece of equipment is providing logging functionalities of different verbose levels raising the volume of data to be treated to unmanageable levels. Only to make things worse, keeping your IT infrastructure secure involves reaction delay. One cannot simply take too much time to collect, detect, interpret and prioritize threats and warnings because by the time this procedure will be done, the results may already be disastrous.

In order to deal with these problems of security information, organizations cover their needs with Security Operation Centers (SOC) either as an integrated part of the organization itself or as a service provided by an external collaborator/ company. The SOC's role is mainly to provide situational awareness by continuous monitoring of the IT infrastructure and real-time alerting of security related incidents. It constitutes actually of an operational supervision system dedicated to protect the IT infrastructure and react in real-time in case of an intrusion or an attack.

1.3 Thesis outline

1.3.1 Context

The company in which I conducted my thesis has a long history of security activity which was mainly focused around penetration testing. This means that companies that needed to verify the security level of their infrastructure, network or applications gave permission to IT security experts to try to discover and exploit vulnerabilities. There was though a rising need and demand for a service of constant vulnerability assessment and monitoring in order for the companies to be sure that their infrastructure is always protected, compliant with multiple regulations, secure

against malware and attackers and safe for its users. In this context, the company decided to mount an Information Security Operation Centre (I-SOC) which would provide all these services for the company itself but also for other organizations as a Managed Security Service Provider (MSSP).

1.3.2 Goals of this thesis

My role entailed researching and documenting the best practises as well as testing, benchmarking and finally deploying the proper tools in order to implement the methodologies required for such a task. In this paper I present all the requirements, the challenges met and finally the discoveries and results around the setting up of an Information Security Operation Centre. First, there is a listing of the requirements and the issues that have to be dealt with in order to achieve operational functionality for a SOC. In the second part, there is a presentation and explanation of the methodology and architecture that was used as a model in order to define, implement and test the different solutions and methodologies. And finally, there is a section where all the results and conclusions will be presented, regarding of course the initial requirements.

2. Optimizing a SOC

2.1 Problematic

Since the early years of the appearance of Intrusion Detection and Intrusion Prevention Systems (IDS/IPS), it became clear that logging suspicious activity is a prerequisite in order to have a realistic view of the perimeter in question and can help identify and even prevent malicious activities. This is why such systems became popular and are nowadays a must-have regarding security for organizations. However, due to the high volume of data there is a risk of being drowned in logs and loose the essential information. Nowadays, this issue has only become more critical as every piece of equipment (firewalls, routers, switches, web servers etc.) is able to log any suspicious or potentially dangerous activity.

Therefore, the challenge to identify suspicious behaviour in modern infrastructures leans more towards the proper selection and filtering of logs and log sources than the actual information of the logs. This means that despite the fact that the actual logged data contains some critical information, the task to sort it out of a high volume of other logs without generating false alarms constitutes the main problem of security monitoring.

2.2 Scope of the thesis

The goal of this project is mainly to define the best practises on how to consolidate, centralize, aggregate and correlate security events across the whole range of an IT infrastructure in order to provide a security awareness as well as risk prevention and incident responding to security related events. All these, while maximizing efficiency and minimizing response time and costs. For this reason, multiple cases and scenarios are taken into consideration in order to reflect as well as possible realistic needs and situations. However, proper configuration of security components as proxy servers, IDS/IPS, antivirus server-client solutions etc does not enter the scope of this thesis. There is no discussion about how to properly set up an IT infrastructure from a security point of view, neither on how to select and adapt security technologies regarding specific needs as all these subjects are already thoroughly analyzed.

2.3 Definitions

As log source we define any system or program that generates log data regarding some activity. Besides programs and systems that provide logging capabilities on top of their normal functionality, ex. routers that log incoming/outgoing packets, there are also dedicated programs which are used exclusively as monitoring and logging components of a system, ex. IDS/IPS.

Not all logging activity constitutes security related activity. More often than not, logging systems are used in order to identify functionality problems, like loss of service. These log data should be distinguished from security related logs because it only adds up to the already high volume of data that has to be treated. However, we should be careful not to exclude information that may have potential security value.

In order to better define and focus the questions to be answered, a clear definition of the services required by a SOC must be provided. A security operations centre needs to provide real time, global, monitoring of the whole infrastructure of a given organization. All types of equipment and devices must be supported and all security incidents must be detected and identified in almost

real time. Human overhead has to be minimized by augmenting effectiveness of log treatment and definition of workflows related to incident response. Finally, historical data have to be treated properly so as to allow and facilitate forensic procedure in case of post mortem incident analysis.

2.4 Issues that need to be addressed

As a security analyst who has to monitor efficiently the IT infrastructure, I summarised the required tasks to the following questions:

1. ***What equipment must log data?***

   Do we need to receive logs out of every piece of our infrastructure or only out of main pieces? Do we need to log intermediary or end devices/systems? Or both?

   The main issue is that usually information of a particular activity is logged multiple times by different equipment. For example, an IP packet passing through a firewall, a router and a switch will be logged three times. And if there are more than one sub networks logging will be multiplied accordingly. Therefore, there is a need to decide if the complete logging procedure offers some extra information or if it is just duplicated and useless data. In the case where logging an event only once suffices, it also important to define which point is the optimal one to perform the logging.

2. ***Which level of verbosity should we apply to each log source?***

   Is every piece of information useful or should we filter the events that are logged in order to keep the essential information?

   Every piece of equipment that provides logging facilities has different levels of verbosity and information included in the log. Activity is classified regarding the function performed and regarding the severity of the action. Furthermore, logs can include all the activity or just the trace or just a count.

3. ***Is there a need for a dedicated component that will log activity on our system?***

   Do we need an IDS or IPS and if yes, is there a need for a host based or network based monitor component? Or both?

   IDS/IPS offers a much better overview and understanding of the ongoing activities of a system. Still, not all organizations are in possession of one. Moreover, there are network based IDS/IPS's that monitor the networks activity and there are host based components monitoring activity on one specific host. Deciding which system is actually necessary according to the security requirements and standards is important in order to avoid useless and encumbering data.

4. ***Where should we store the log data?***

   Is there a need to centralize such data? Should the logs be kept on a local aggregator or a remote one?

   This issue regards mainly the MSSP point of view. In the case where one external provider is responsible for the security log management, it has to be decided where the logs will be stored. This has to do with the level of sensitive information contained in the data and the policy/willingness of the client to share it with its security service provider. Obviously,

managing remotely log data adds some operational overhead as well as some restraints to the procedure. This is why the client has to calculate carefully the advantages and disadvantages of the chosen solution.

5. *Which protocols and procedures should be used in order to establish proper log transmission between the sources and the storage?*

Since the service will be provided for clients whose infrastructure is remotely located, there is an apparent need to define security and functionality requirements of data transmission. Even in the case of an integrated SOC inside an organization, there are many sub networks that might communicate over a wide area network (WAN) and possibly over Internet. Security levels have to be ensured for such communication due to the sensitive nature or data transmitted. Furthermore, data loss has to be prevented or at least detected because absence of logs can disrupt the monitoring effectiveness and affect gravely the required awareness.

6. *Define log monitoring retention and security requirements/policies.*

In short, for how long should we store the log data and under which format, encrypted or not, as well as who should have access?

In order for a SOC to be able to provide forensic analysis of events, log information has to be stored for a long term period under a suitable format that will allow easy and quick data retrieval while minimizing their size and complexity of storage. On the same time, sensitive data have usually to be encrypted as well as tamperproof, depending on the potential compliance requirements required by authorities.

Answering clearly to these questions will allow us to proceed to the next step in the chain of security information management and treatment by having a proper understanding of the lower level of log information. Once we have these issues addressed and well defined, the security analyst has to decide what policies and rules must applied in order to categorize and treat properly large amounts of log data. In other words, it is made possible to move on to a higher level of abstraction.

Specifying the roles and methodologies used by a SOC, makes it possible to better understand the procedures that need to be followed. As mentioned before, the SOC has to be able to cover a wide range of devices and software components. Actually, it needs to be able to integrate all different kinds of log data, no matter the diversity and complexity of protocols used. Moreover, all these versatile formats need to be treated by the same system in order to provide correlated events. Let us not forget that situational awareness across the whole organization has the benefit of being able to correlate multiple, unrelated at first glance events so as to detect more complex patterns of attacks or propagation techniques of malware. Which leads to the next question.

7. *What kind of procedure must be used then in order to be able to accept all vendor-related data log?*

Furthermore, what protocols of normalization have to be used in order to provide a common format for all types of log?

Managing to receive properly all the logs in a central system under a unique format of representation constitutes the first level of log treatment. Now, as a second level of treatment, logs must be aggregated depending on their content so as to produce aggregated events that contain all the necessary information without the size and complexity of individual log records.

8. ***The rules for such aggregations as well as the place where this procedure will take place have also to be well defined.***

The advantages of aggregated events over individual ones are that size is reduced, useless information is discarded and the processing power required for the rest, more complex procedures is cut down. Aggregated events are also able to provide a first level of alerting, depending on simple rules based usually on the size of unique logs over a period of time.

Already, by using the aggregation procedure correctly, we are in place to have a better view and reaction on incidents. However, attacking methods are becoming more complex and obfuscated while on the same time, many logging components produce false positive alerts that can frustrate and render a monitoring unreliable and ineffective. For such cases, a higher level of event correlation and structural analysis is required in order to mitigate these issues. After all, SOC as an operational supervision system must be able to provide threat control and prevention as its final service.

9. ***Events have to be correlated using specific rules that will create context against which future events will be checked and evaluated.***

On the same time, structural analysis will be used in order to check predefined patterns that may lead to unwanted results. At this point, it is important for the SOC to have a complete view of all the systems affected in order to define the criticity of a potential attack and to be able to provide risk management.

Security policies and regulations have to be taken into consideration in order for the SOC to be able to generate alerts based not only on the importance of individual systems but also on requirements set by the organization.

Finally, when correlation and structural analysis are through and well defined and our monitoring system is fine tuned in order to detect attack patterns and to avoid alerting on legitimate activities, we have to define the manner and the actual procedure of alert generation and alert transmission.

10. ***We have to define who must be notified as well as the actual content of the alert.***

Furthermore, since reducing human overhead is one of the main advantages of a SOC, clearly stated and effective workflows have to be defined and attached to generated alerts, in order for the appropriate personnel to be notified and act accordingly.

As a first step to better understand and define the needs and best practises of setting up a security operating system, the before mentioned questions will help as a guide. In the following chapter, the methodology which was used in order to encounter in real life situations all these challenges will be presented in details. There are already recommendations for some of the questions mentioned but in order to be able to test and

decide upon realistic data, a proof of concept scenario was performed based on a topology as closer to reality as possible.

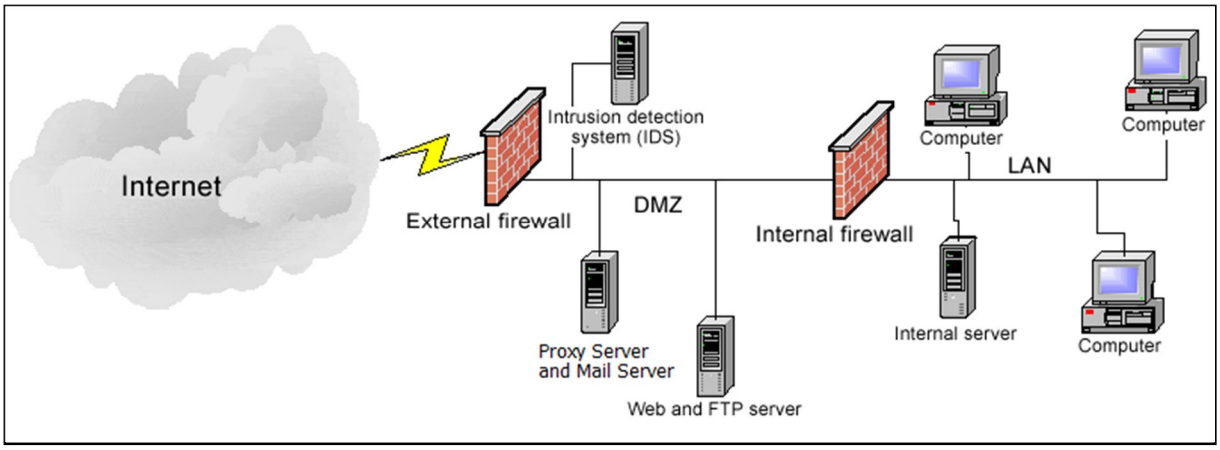
3. Proof of Concept

3.1 Topology

As a case study, a typical IT infrastructure was replicated representing a typical client network with respect to the basic notions of security. This translates to an IT infrastructure where topological, there is a border firewall facing the public interface (Internet) which we will refer to as the Border Firewall. On the internal perimeter there is a Demilitarized Zone (DMZ) where the public servers that need to be accessed from the Internet are located, and another firewall (Internal Firewall) which protects the internal, private network of the infrastructure. The Border Firewall is a stateless firewall with a "default allow" policy concerning traffic. The Border Firewall's rules act as a hoarse filter which serves to mitigate a broad spectrum of attacks and exploits by denying packets which are obviously illegal (inbound traffic from the public interface with private source addresses, with broadcast destination addresses, protocols destined to hosts that do not support them, fragmented packets etc). In addition, in order to have a high functionality and accessibility for the DMZ zone, we need to be lenient on the Border Firewall rules.

Inside the DMZ there is an IDS/IPS in order to identify intruders or suspicious behaviour in general and there is also a Proxy Server which acts as a secure forefront for internal users accessing the Internet. We can also find some other generic-purpose servers (Web, Mail, DNS etc.). The Proxy Server is also an essential component because it allows seamless and transparent communication for internal users while on the same time it protects them against malicious sites and malware by providing an indirect, filtered communication.

On the border of the internal network we find the Internal Firewall which is a Statefull Firewall equipped with well defined rules that provide a fine granularity on what is allowed and what is denied ("default deny" policy). As the large amount of traffic is already dealt with at the Border Firewall, we can apply a stricter policy of rules and we can dedicate more required resources for the necessary functions of a statefull firewall. Inside the internal network there are workstations with configured antivirus software and devices for general purpose use, like printers etc.



IT Topology

The infrastructure is divided into multiple sub networks accordingly and there are also intermediary network devices (routers and switches) at different joint points of the infrastructure. Furthermore, each sub network is mutually isolated by each other using virtual local area networks (vlan) allowing inter-communication only through properly configured inter-vlan routing devices, e.g. routers.

In order to have reliable and realistic results for the proof of concept, the case study used was actually based on an existing infrastructure of a real, functional network. I was allowed to configure arbitrarily any piece of equipment, as seen fit to the needs of the scenario, as long as my changes did not affect the overall functionality of the IT system. This constraint only contributed to a more realistic approach to the problems at hand because performance for an IT infrastructure is still the primary concern and should not be hindered as a result of the procedures of the SOC. The main modifications involved security logging details, rules and transmission.

The topology used constitutes a simplified, typical network that nevertheless includes all the necessary and important equipment found in a fully functional IT system. The same case study was used in parallel in two different Proof of Concept scenarios. In the first scenario -scenario A-, the SOC was an integrated part of the organization itself, meaning that the centralization of all the equipment can take place within the network. However, in the second scenario -scenario B- I tried to emulate the MSSP mode of a SOC where the collection and centralization of the log data has to happen through external links to an external company providing the SOC services.

3.2 Terminology

From the point of view of defined modules for the needs of the SOC, we distinguish 5 categories:

1. ***Log sources:***

   This module includes all devices and software components capable of providing activity data according to specified rules and filters.

2. ***Log collectors:***

   Either as an agent configured on a log source either as a stand-alone module, a log collector is responsible for the collection of log data out of each source and for the proper transmission to a central system of aggregation.

3. ***Event analyzers:***

   This module takes as input log data transmitted from all over the organization and provides the intelligence and analysis required in order to categorize and prioritize accordingly the various incidents.

4. ***Storage equipment:***

   A dedicated module reserved for the appropriate long term storage of the event's data. Historical data will be stored there in order to be easily recovered in case of forensic analysis or simply for statistical use.

5. ***Overview/Control equipment:***

   An overview console dedicated for the security analyst that will provide a centralized control module in order to allow easy configuration and management for the operational monitoring system. It will also provide a visualization interface of the monitoring and the incidents generated.

4. Approaches

4.1 Scenario A

4.1.1 Description

The two scenarios differ with respect to the placement of the SOC regarding the organisation which will be monitored. As a first scenario, we will use a SOC that constitutes an internal component of the company and it is therefore located, logically and physically, inside the organisation itself.

Advantages of such a solution are that the SOC can go in greater depths in order to better comprehend the needs of the IT system as well as the criticality of each separate component. Furthermore, by being a part of the company itself the SOC can easily get familiarized with any potential specific procedures, policies or infrastructure particularities.

Thus, it can adapt its monitoring and alerting to better serve its purposes. However, an inside-company SOC solution will lack the expertise of a clearly security focused provider as well as the wider area of coverage that a MSSP has in order to cover its various clients. Finally, for small to mid-sized companies, it may be a cost ineffective solution to implement a SOC with all the required material and personnel needed.

4.1.2 Components

So, regarding the first scenario, the components acting as log sources taken into account are:

- Border Firewall
- Internal Firewall
- Proxy Server
- DNS Server
- Mail Server
- Active Directory Server
- IPS
- Web Server
- Routers of the internal network
- Switches of the internal network
- Workstations (Windows PCs including antivirus software)

The emplacement of the SOC inside the company itself permits an easier manipulation of data flows as well as a larger use of bandwidth and thus verbosity and volume of logs.

4.1.3 Logging requirements

In order to better understand the level of verbosity and the type of logs collected, we need to define the different categories of logs used in order to distinguish between them and classify them. As there are multiple, vendor specific classifications and methodologies used, we will present a generic one that can cover and serve all the potential log sources.

Two distinct categories exist defining verbosity level, content and priority, and each log is assigned one value for each category.

Content includes as subcategories:

- System: Includes all normal activity associated with the functionality of the log source in question.
- Error: Includes all abnormal activity which may be a result of unexpected behaviour of the log source.
- Security: Includes all security related incidents, detected as such by the log source itself. This feature is supported by almost every log source but not entirely.

Concerning the priority of each log, we distinguish between:

- Debug: Includes debug level data, useful only to configure or correct a component.
- Information: Includes informational logs related to expected to be seen behaviour.
- Warning: Includes activity that involves low level severity results that need to be reported in order to avoid potential loss of service or damage.
- Urgent: Includes high level severity incidents that need an urgent attention in order to prevent loss of service or potential damage that may also affect the system involved.
- Critical: Includes high level, critical incidents that most certainly will have severe, negative results on the functionality of the component and/or the system involved.

We proceed now to a more precise presentation of the verbosity level of the logs collected by each log source. We will note the minimum priority requirement for a log source to start logging activity, implying that all the other higher levels are automatically included.

- Border Firewall: System Urgent; Error-Urgent; Security-Warning;
- Internal Firewall: System Urgent; Error-Warning; Security-Information;
- Proxy Server: Error-Urgent; Security-Information;
- DNS Server: Error-Warning; Security-Information;
- Mail Server: Error-Urgent; Security-Information;
- Active Directory Server: Error-Warning; Security-Information;
- IPS: System Urgent; Error-Warning; Security-Information;
- Web Server: Error-Warning; Security-Information;
- Routers of the internal network: Error-Critical; Security-Urgent;
- Switches of the internal network: Error-Critical; Security-Urgent;
- Workstations (Windows PCs including antivirus software): Error-Critical; Security-Urgent;

The idea behind the desired level of logs collected from each log source is that for critical security related equipment we need the most data we can have, taken into account the volume of data generated. For example, there too are many switches being used to collect all the activity data they can provide, which will eventually over flood the monitoring system with useless information, impairing the discovery of other potential threats. That is why we only take into account critical errors and urgent security events.

Regarding the log collectors, we find one in the DMZ which is responsible to collect data from the:

- Border Firewall

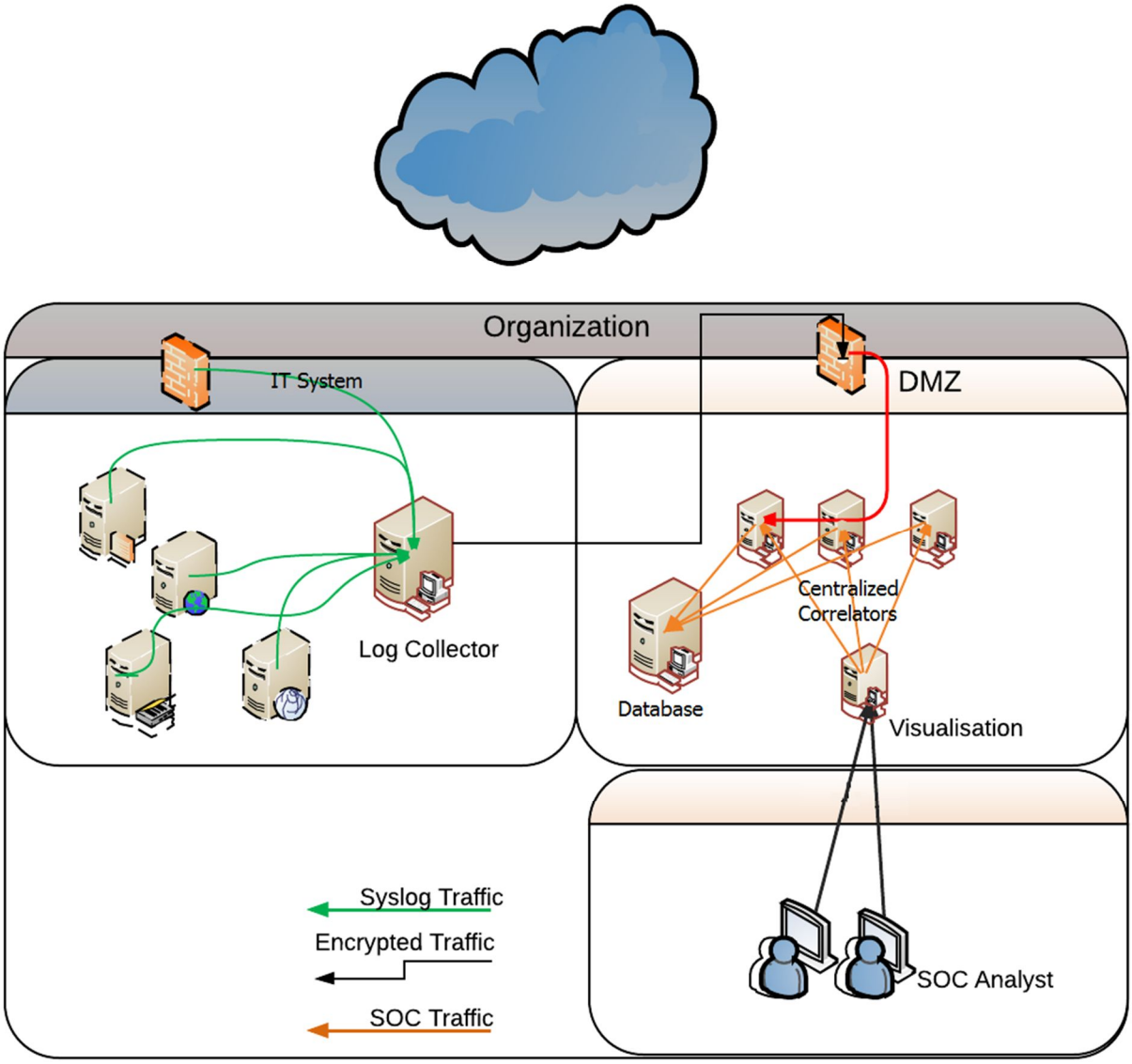
- Proxy Server
- DNS Server
- Mail Server
- Active Directory Server
- Web Server

And one in the internal network collecting data from the:

- Internal Firewall
- IPS
- Routers
- Switches
- Workstations (Windows PCs including antivirus software)

In order to facilitate the scenario, we assume that all internal components are on the same subnet and thus one collector suffices to manage all log sources. Otherwise, there would be a need for a local collector on the same subnet of each component that would forward the logs gathered to another central collector.

All log collectors forward the data gathered to a central unit found in the internal network, under an exclusive subnet, which includes the events analyzer. The data after being treated and possibly enriched by the event analyzer are sent to the storage component, which is also found in the same subnet. As we are covering only the needs of the company itself, the overview component can be also found in the internal network.



Internal SOC

4.2 Scenario B

4.2.1 Description

In the second scenario, the SOC is acting as an MSSP and the communication channels need to be externalized. Moreover, the available bandwidth is reduced in order not to impact the functionality of the client's IT system.

4.2.2 Components

The components acting as log sources taken into account are:

- Border Firewall
- Internal Firewall
- Proxy Server
- DNS Server
- Mail Server
- Active Directory Server
- IPS
- Web Server
- Workstations (Windows PCs including antivirus software)

In this scenario, the SOC is acting as a remote service provider and the logs have to be transferred over a WAN to a physically distant location. A local collector is responsible for the collection of all the necessary data inside the client company. Then, this collector is communicating through a secure tunnel with the event analyzer. This means that the logical schema of the network remains the same, in spite of the fact that log collector and event analyzer are in different physical connections.

4.2.3 Logging requirements

As before, we will note the minimum priority requirement for a log source to start logging activity, implying that all the other higher levels are automatically included.

- Border Firewall: System Critical; Error-Urgent; Security-Warning;
- Internal Firewall: System Urgent; Error-Urgent; Security-Information;
- Proxy Server: Error-Urgent; Security-Warning;
- DNS Server: Error-Urgent; Security-Warning;
- Mail Server: Error-Urgent; Security-Warning;
- Active Directory Server: Error-Urgent Security-Warning;
- IPS: System Urgent; Error-Urgent Security-Warning;
- Web Server: Error-Urgent Security-Warning;
- Workstations (Windows PCs including antivirus software): Error-Critical; Security-Urgent;

As noticed, the verbosity level of the collected logs has dropped in order to reduce the necessary bandwidth and cost of the data transfer.

Regarding the log collectors, we follow the same principles and topology as in scenario A, with the only difference that the event analyzer now resides in another location.

5. Technologies employed - SIEM

5.1 Context

In order for the SOC to satisfy all the requirements, various tools and solutions have emerged. The main component that encompasses all the functionalities of a SOC is called Security Information and Event Management (SIEM) and it constitutes a technology which arrived in the beginning of 2000. Originally, there were two distinct technologies, Security Information Management and Security Event Management (SIM/SEM) which were later combined into one solution.

SIM provides collection of logs over a wide area of sources, proper storage of the collected data over a defined period, ease of access to this data (reliable and fast research by indexing and enriching the stored data) and ultimately, an interpretation of the data transferred through the logs.

SEM solutions provide the advanced correlation (chronological or content-driven) of multiple, distinct logs in order to produce a meaningful overview of complex attack patterns. This means that it is responsible to analyze various incidents, which may be related or not, and bind them into a security event raising an alert in case of malicious action, all that in near real-time.

SIEM solutions gained important momentum by combining the functionalities of these two existing technologies in one seamless system that covers all the security monitoring and alerting needs providing a consolidated overview of the actual security state.

5.2 Functionalities

A SIEM can provide the following:

- Attack detection (simple or complex) by constant monitoring of the IT system of an organization.
- Conformity control based on regulatory restraints and security policy by detecting anomalies.
- Incident response and forensics capabilities.
- Long term archives.
- Reporting functionalities (dashboards, executive reports, trend reports etc.)

SIEM technologies became the new trend in the world of corporate IT security and this is the reason why many major security companies acquired or seek to acquire SIEM vendors as a means to integrate SIEM solutions into their security suites [1].

Of course, the SIEM is a security tool and with its many positive sides, it should not be forgotten that, as every tool, it is as good as its user allows it to be. A team, properly skilled and familiarized with the tool has to be assigned in order to adapt and fine-tune the SIEM to the specific target IT system. Moreover, SIEM technologies are susceptible to generate a large amount of false positives. Even after an initial phase of learning where the team running the tool will eliminate a good percentage of these false positives, there will still be some left that will require human intervention. Finally, even if it is evident, it should be mentioned that a SIEM does

not in any way stop or take any action to stop an ongoing attack. It is there just to detect the complex patterns of an attack and generate an alert in as close to real time as possible.

5.3 Tool selection

In order to find the proper SIEM solution to be used for the needs of the SOC, a research was conducted on different vendor technologies. The main points that were taken into consideration for the decision, in an order of priority were:

1. Support for MSSP mode of function, as it is the main reason for the company I did my thesis for, to deploy a SOC. This translates to a requirement for a distributed solution where different client-organizations send their security data to a centralized system, found in the service provider company, which will provide the analysis. It also means that the SIEM solution must support scalability and deployment flexibility regarding the integration of new assets.
2. Functional effectiveness over all the aspects and procedures of the SIEM mentioned above in combination with ease of deployment and installation.
3. Possibility to run all pieces of the SIEM solution on virtual machines. This gives a big advantage as it facilitates resource allocation in case of a greater demand due to evolution of the existing IT system or due to a new system that needs to be integrated.
4. Technical support in the country where the company resides. This was an important requirement because response time on technical errors or misconfigurations is considered a priority as to ensure of the constant real-time monitoring of the target organisation.
5. Reputation of the SIEM solution and engaged clients or company partners involved. Already tried solutions which have positive results and good reputation constitute a reassuring factor for potential client-organisations.
6. Cost-effective solution in regard to the pricing policy adopted by the company for its MSSP model.

Regarding the above mentioned criteria, two solutions were chosen for the realization of the PoC which were also benchmarked.

The technical details for both solutions are presented as a complement to the scenarios described in section [4. Approaches]

5.4 Technical definitions

1. Data collection:

   Log sources can be divided into two categories:

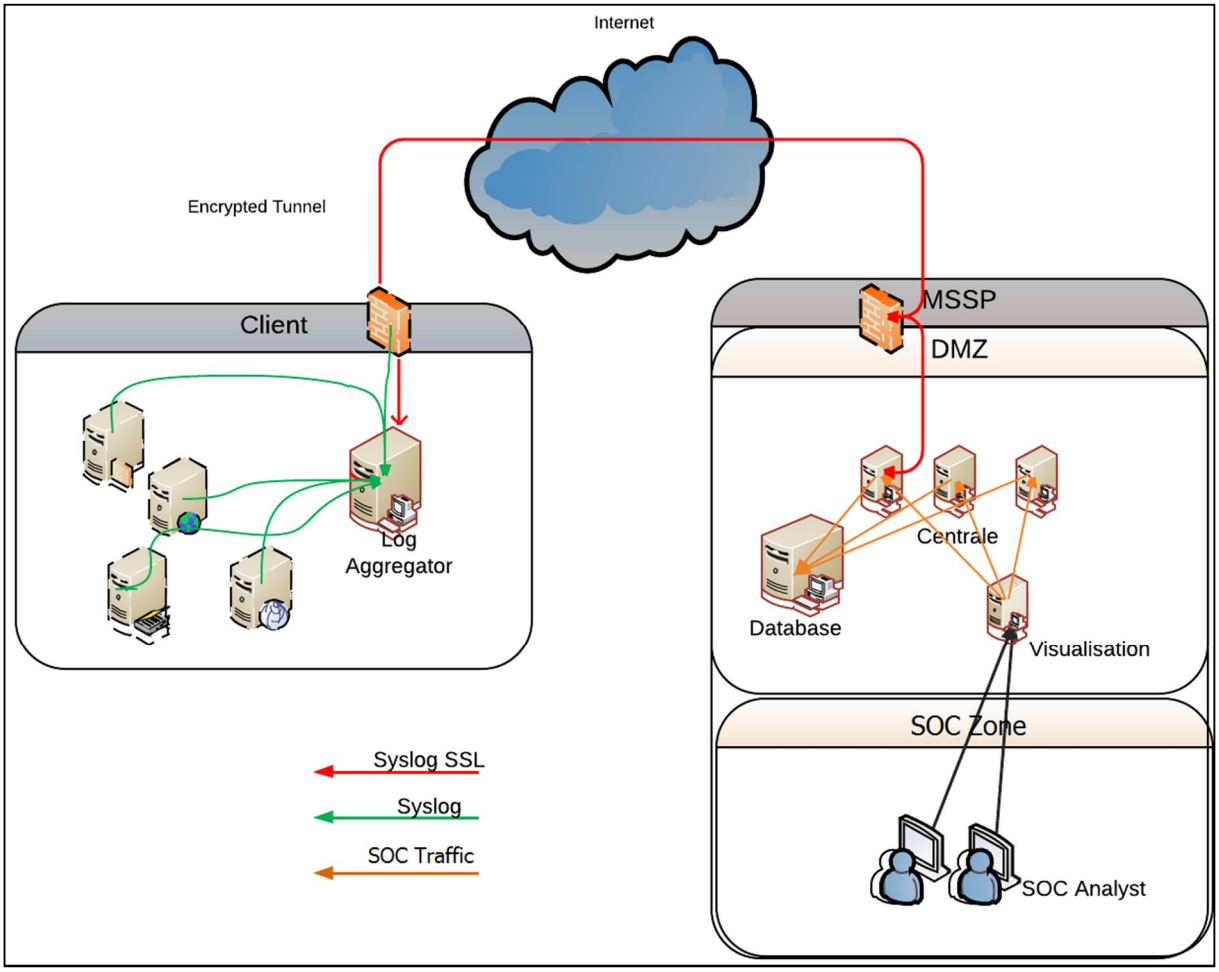
   I. Log sources with integrated functionality to transmit log data.
   II. Log sources generating log data without the innate ability to transmit it.

   For the second category, a special agent has to be deployed as to collect the logs locally and then transmit it towards the appropriate log collector.

   Also, for some components belonging to the first category, the supported data transmission may not satisfy the requirements for safe and reliable data transmission. For example, many networking devices support UDP log transmission that does not constitute a reliable nor secure protocol. Such components also require an agent to be deployed.

A totally clientless approach is more attractive because extra components may interfere with the original functionality and may demand resources that can impact the effectiveness of the system in question. Furthermore, having to install an agent and ensure that it is up and running on a large perimeter may prove to be quite a challenge.
However, it is something that cannot be avoided since there is no regularization in respect to log transmission over different vendors. The agents are low-resource, small pieces of software that can be installed on different Operating Systems.

2. Data collectors:
   Depending on the architecture of the target IT system, communication channels between different zones may be restricted. For these cases, a log collector that is responsible for the consolidation of all the log sources is placed on the same sub-area. This log collector communicates then with either another log collector, giving a distributed multi-layer architecture, either directly with the analyzer responsible for the specific IT system, giving a flat architecture. The log collectors should support data enrichment so as to be able to forward log data on behalf of the respected log source in such a way that the receiving end of the data can distinguish between different components despite the fact that the data is actually coming out of one and only device.
   Data collectors are also responsible for filtering and aggregating logs. It is very important to apply such a distributed model of filtering and aggregation in order to avoid potential bottlenecks, both in regard to bandwidth as well as in regard to processing power.
3. Data analyzers:
   The data analyzers are found in the internal scope of the company-provider and they provide all the intelligence required to correlate between disperse events. For this reason, they demand a lot of resources. They can run on virtual machines and they can be deployed in such a manner that allows new analyzers to be added in parallel, in case of extra power needed.
4. Overview components:
   SIEM also provides an overview component presenting real time activity and alerts. It is also able to generate dashboards and reports as well as present workflows in case of specific incidents. Good visibility of the monitored perimeter is also affected by good presentation and proper analysis of the data collected.



MSSP SOC mode of function

6. Results

In order to evaluate the results and the effectiveness of the configurations, multiple common attack scenarios were conducted. The efficiency of the I-SOC is measured depending on the detection or not of the associated attack, taking into consideration the complexity of the attack as well as the time required for a successful detection and identification of the threat.

6.1 Attack scenarios

6.1.1 Brute force attack to an administrative interface (ssh, application interface)

This scenario is straight forward. Multiple connection attempts either to an exposed ssh server or to an exposed web administration page. The attack scenario includes two variants, one which results to a successful connection after some requests and another one without any success.

6.1.2 Vulnerability exploitation of critical server

a. Authentication bypass of an SQL server
   Attempt to bypass the authentication procedure of an SQL server by exploiting known vulnerabilities.

b. Abuse of misconfigured DNS server (open relay)
   Exploitation of a DNS server which permits openly (from any host, not restricted to its own network) recursive queries in order to conduct a reflected attack.

c. Abuse of misconfigured SMTP server
   Exploitation of an SMTP server which permits to everyone (from any host, not restricted to its own network) to send emails to any destination in order to contribute to spam campaign.

d. Application level attack to web server (SQL injection)
   Exploitation of an application level vulnerability on a web page allowing the attacker to dump the database.

6.1.3 Virus/Trojan infection in the internal network

Malware infection through email attachment, spreading itself throughout the internal network.

6.1.4 Covert channels of communication

This kind of attack involves the use of a side channel in order to circumvent monitoring therefore bypass control. As an example, an attack including a DNS server abuse from an internal user in order to establish a DNS tunnel and circumvent data traffic policies regarding data exfiltration was used.

6.1.5 Detection of Advanced Persistent Threat

Advanced persistent threat (APT) includes a variety of attacks and techniques sharing a common factor, the advanced level of sophistication and complexity of the attack in order to avoid detection and treatment. As an example of such attack we used the infection and compromise of the critical infrastructure (Web Server) without identification of the entry point/procedure (in our case, phising e-mail leading to trapped web page distributing malware).

6.2 Attack results

6.2.1 Brute force attack to an administrative interface

The attack was conducted first on an exposed ssh server that was accessible from the Internet.
The tool that was used - hydra [2]- is simply launching multiple connections to the server using different credentials every time until it succeeds to identify a valid account with its password. The same principle applies for the web administration interface.

The detection of the attack was successful and occurred within 5 minutes after the launch of the automated tool. The rule of detection consists of a check of the number of connection attempts from a specific IP address over a period of 3 minutes. If the defined threshold is reached, a medium level event is generated and a second rule gets into effect looking for successful connections –valid credentials- coming from that same IP address for the next 5 minutes. The two rules leave no uncovered time window as by the time the second rule will expire (after 5 minutes of no successful connection) if the attack persists, a second medium level event will be generated by the first rule. In the case where the second rule is triggered, a high level alert is issued and immediate action must be taken. The I-SOC is already alerted with the generated "brute force attempt" event and by the time the potential successful connection occurs the response team will be ready to take action, following the predefined action plan. If the attack never succeeds but is persistent nonetheless, proper action must be taken in order to blacklist the source IP address, or even better, deny by default all connections initiated by IP addresses which are not found in a predefined whitelist (administrators etc.).

One variant of the attack includes the use of TOR [3] as a proxy for the brute force attack, resulting in a wide range of source addresses rendering the before mentioned detection rules practically useless. For this case, another rule is used which is based on behavioural analysis of the traffic. Using this method, during a learning period multiple metrics of the network usage are being monitored and registered. One of which is, global connection attempts to an administration interface over a defined time period. If this recorded value is exceeded two times over, then a medium level event of "potential brute force attempt" is generated. In order to be able to detect a successful attack, the SIEM needs a list of allowed IP address ranges that are entitled to access the associated administration interface. Successful connection from another IP address generates a medium level event "Unknown address connected to interface". Correlation of the two before mentioned events over a period of 30 minutes issues a high level alert of "Potential successful brute force attempt". In order to avoid false positive alerts it is necessary to go through an adaptation period where the exact values of thresholds will be defined and the whole procedure will be fine tuned so as to ignore legitimate traffic.

6.2.2 Vulnerability exploitation of critical server

- ***Authentication bypass on SQL server***

  Normally, the detection of such an exploit is detected by the IDS/IPS that monitors the traffic and the payload that is being transferred. One use of the SIEM that can contribute to this procedure is the crosscheck of the exploit by correlating the IDS/IPS alert with a successful connection by an unauthorized source, given a predefined list of legitimate users. In such a way, the false positives will be dropped and the level of trust to the IDS/IPS can be enhanced.

- ***Abuse of misconfigured DNS server (open relay)***

  DNS servers are bound to serve requests originating from the network they are supposed to serve and besides that, they are supposed to respond openly to requests regarding their own domain name(s). However, wrongly configured DNS servers will reply to anyone on the Internet asking information about other domain names. Abusing this configuration, attackers can use the DNS server to launch reflected DoS attacks [4] spoofing the source address of the request.

  Detection of this type of attack was successful using three rules. The first rule is responsible for identifying an overcharge of outgoing DNS packets from the host included in the DNS servers list, comparing the amount of actual traffic with the "normal" traffic, defined during a learning period. The second rule is almost identical but it concerns internal traffic to the DNS server. The correlation rule used is triggered when the first event is generated without the presence of the second event, for the same DNS server. In other words, if there is an unusually high amount of traffic coming out of the DNS server without an augmentation of internal traffic to the DNS server, a medium level event "Augmented external DNS action" is generated. Further investigation of the event is necessary in order to verify the validity of the generated event.

- ***Abuse of misconfigured SMTP server***

  The scenario as well as the detection of this attack is equivalent to the misconfigured DNS case study. The same rules and principles are used and the detection was equally successful.

- ***Application level attack to web server (SQL injection)***

  The attack scenario involves identification of an SQL-injectable parameter in a web application hosted on the associated server and following that, a successful database-dump exploiting the injection point.

  The first part of the attack is treated by the IDS/IPS which is responsible to identify potentially dangerous requests that include SQL commands. However, as this kind of application-vulnerability scanning is rather common, the generated event remains a low level event. A second rule that checks SQL server activity and traffic is responsible to generate a low level event if the activity is found augmented over a time period. Correlation of these two events in a time span of 30 minutes issues a high level alert that

needs to be taken care of in order to mitigate the attack. Detection of such an attack was successful but it necessitated a high amount of traffic originating from the SQL server, i.e. a database dump or a full listing of tables. Further fine tuning of the procedure involves a defined time plan where high volume traffic generated by legitimate activities on the SQL server are scheduled, in order to avoid false positives generated by maintenance or otherwise normal use of the SQL server.

6.2.3 Virus/Trojan infection in the internal network

This scenario involves the initial infection of a weak point of the internal network, in our case a professional laptop without an anti-virus software that visited a malicious site where it got infected, and afterwards the propagation of the malware in the internal network. The malware is part of a botnet that compromises its host in order to use its resources according to the commands it receives from a command sender. This functionality necessitates a channel of communication towards the command centre and possibly communication towards other locations so as to download more pieces of malware and/or update itself. It also implies a network activity which will be related to the activity of the botnet, for example participation in a DDoS attack [5].

The detection patterns of such a scenario can greatly vary due to the many variations of communication and infection techniques employed by modern malware. In our case study, the malware performs a network scan in order to detect other hosts on the network and identify possibly vulnerable or suitable targets. It also performs DNS requests in order to obtain the IP address of its command centre and then, it receives its commands at random intervals through a web service listening on a specific TCP port.

The first level of detection is taking place when the malware probes the network in order to identify potential targets. Normally, internal activity is considered legitimate; nevertheless some scanning patterns can be identified by the IDS/IPS as “Potential internal scanning” which will create a low level event. A script that is responsible to obtain and in real time update a list of IP addresses that are blacklisted by antivirus companies and the security community as related to malicious activity is used in order to alert whenever there is network activity involved with these targets. However, the IP address of the command centre as well as the malware repository were not included in this list during the infection and propagation of the malware. If this was the case, a high level alert would be issued and the activity would be investigated by the I-SOC. Still, the SIEM was able to identify the infection using a behaviour analysis rule based on antivirus reports, collected by the workstations of the internal network. There are many antivirus solutions that provide agent-central server architecture, which is able to centralize malware activity across the entire IT infrastructure. Antivirus agents are deployed on every workstation and a central server is responsible for the security policy applied. In case the security policy is compromised, the central server sends an event including all the necessary details to the SIEM, which in its turn evaluates the information based on specified thresholds and generates an alert accordingly. The I-SOC can then conduct a forensics analysis on the event, using indexed data generated across the whole IT infrastructure in order to identify the initial infection point. For example, malware analysis can reveal destination address, source TCP/UDP ports or other indicative patterns used by the malware which can be used in order to identify the actions taken by the malware. In that

way, the I-SOC can determine which parts of the network have been compromised and take proper actions to disinfect them.

6.2.4 Covert channels of communication

There are many ways to circumvent security policies regarding sensible data exfiltration or denied data communication, for example http traffic to sites outside the interest of the organisation. One common scenario involves the abuse of DNS requests in order to tunnel traffic inside DNS traffic. By requesting a domain/subdomain name which is controlled by the circumventor, the legitimate internal DNS will launch a request that will reach eventually the target DNS. It is possible to inject http or other traffic inside the domain name request which will be interpreted by the target DNS accordingly, i.e. stripping down the DNS layer and redirecting the underlying protocol properly. This procedure is called DNS tunnelling and can be detected easily by monitoring the volume of DNS traffic. By definition, the originating traffic will come from inside the network, which permits to put in place a rule detecting spikes in internal DNS activity from particular hosts. Further correlation with the actual domain/subdomain name requested can generated a "Possible DNS Tunnelling" alert associated with the source address. Servers or hosts that demand high volumes of DNS traffic for their legitimate functions have to be whitelisted in order to avoid false positives.

6.2.5 Detection of Advanced Persistent Threat

Advanced persistent threats (APT) constitute a major subject of discussion and research [6][7][8].If we would like to dissect the term and explain the etymology of its parts, APT contains three distinct components:

1. Advanced stands for the skills and expertise of the attacker regarding any aspect related to IT security, from technical skills to social engineering techniques. This implies that the attacker may have personalized tools or malware, avoiding traditional detection modules.
2. Persistent means that the attacker has an actual goal for which he is willing to devote time and energy to see it accomplished. There is a carefully designed plan with specific steps and procedures that need to be followed. Even in case where his plans are thwarted, he will change course and keep on going until he reaches its goal. He is probably funded and rests highly motivated and focused.
3. Threat actually says that this whole action is intended to cause harm in one way or another in order to achieve the personal goals.

What APT really means is that a group of persons are specifically dedicated towards a plan of action against a target and they are willing to spend their efforts to see it accomplished. The piece of malware involved will probably involve zero-days [9] and great effort will be dedicated to make their keep their actions stealthy. The victims that will be targeted are not anonymous users around the world but hosts of specific interest because of their role or their access rights.

As an example of an APT attack, behaviour of different well known and analyzed APT campaigns have been used. Summarizing a "default" tactic of an APT, taking always into consideration the great diversity in techniques and methodology applied, we could say that an APT attack initially uses spear target phising attacks as an entry point and then spreads through the internal network. A channel of communication is necessary in order for the malware to receive

commands and usually advanced techniques are employed, like for example fast-flux (a technique where multiple IP addresses are associated to a single DNS record, using a high frequency in order to bypass ACLs)[10] [11]. Often, a variant of remote access trojan (RAT) [12][13] is then used, assuming control of the infected machines and finally, an exfiltration covert channel is used in order to transfer the sensible data outside the target organisation.

Detection of such complicated attack patterns can be really difficult. Still, by analysing and breaking down the sub components we can use correlation rules to detect behavioural patterns that fit the general APT description. *"As APT depends on remote access and control, the network activity associated with remote control can be identified, contained and disrupted through the analysis of outbound network traffic"* [14]

Starting with the initial attack vector, assuming that a phising email campaign or a water-hole attack [15] is employed by the attacker, malicious traffic or well known Windows registry modifications over a large part of the target network, can be indicative of a targeted attack against the organisation. Surely, zero-days or altered variants of known attacks are not detected by signatures but since the payloads delivered are usually common or slightly modified, we can focus on payload detection of known RATs or other malware.

Continuing with the necessary communication channels or the exploit and assuming that a fast-flux communication is used by the attacker, we can use a feature of many SIEM solutions that provides scheduled searches using some scripts. For example, using a script that checks DNS traffic of specified hosts we can identify an excessive amount of different IP replies for a specific domain and alert about this domain. In that way, we can identify an attack despite the fact that multiple IP addresses are used. Furthermore, abnormal amounts of traffic compared to a baseline can also help the detection procedure.

In the case of APTs, a strong and well defined organisation security policy can facilitate the work of the I-SOC. For example, if the security policy dictates that all http(s) traffic must pass through a proxy server, then detection of such traffic by the firewall originating by a host which does not belong to the approved PROXY-Servers group will provide another indicator of suspicious activity. Another example, if the security policy states that no .rar files are allowed as outbound traffic then a host that is sending such files will raise a warning. A last example of security policy can be that outbound traffic, for example, towards Chinese IP addresses should be beforehand communicated to the I-SOC. In that way, IP geolocalization performed by the SIEM will reveal forbidden traffic for a host.

Unfortunately, all these detection techniques are difficult to be correlated to unique alerts with all the necessary details and often a manual investigation is required in order to reveal or reject an actual attack. This however, is not a limitation of the SIEM or the correlation techniques employed because it is really important to understand that all these tools and methods actually provide a situational awareness over the monitored IT infrastructure. Human intervention and intelligence will always be required and especially in the case of APTs, where the attacker will make its best to avoid traditional detection modules.

In the case of the scenario used, the detection of such an attack campaign was successful after some forensics conducted by the I-SOC. However, it is important to underline the fact that the all

the techniques employed by the APT like attack were known and expected; which is not actually the usual scenario of a successful APT campaign. Still, the results were encouraging taking into consideration that no direct detection was used but instead behavioural and side channel detection rules were responsible for the detection.

6.3 Gained experience from experiments

During the PoC, it became clear that network devices such as routers and switches generate a high amount of logging activity that was not particularly used in the detection and correlation rules. As there is always a need of available resources so as to deal with peaks of traffic, it was decided to keep only a handful of main routers, responsible for the greater part of traffic distribution. Moreover, for these pieces of equipment only events related to overly excessive amount of traffic in comparison to a baseline was transmitted to the SIEM.

Also, regarding personal workstations, there is no need to transmit directly the antivirus module's alerts as there are specific servers responsible to centralise this information. We can then use their logging capabilities, reducing the overhead to the SIEM by providing multi layered architecture of logging. It is important to understand that overlapping pieces of information that do not add any value to the awareness of the infrastructure must be avoided.

Despite the initial configuration where a high level of verbosity was used including original logs in the transmitted events, it soon was evident that this information was not used. Incidents that required further investigation could not solely investigated using the data in the SIEM, necessitating to access different modules. The SIEM provided though the alerting and the guidance as to where to look to further analyze the event.

The existence of an IDS/IPS was essential to the majority of the scenarios. Network IDS/IPS is necessary and every organisation should have in its disposal. Host based IDS/IPS were also helpful but only in a multi-layer architecture, using an intermediary server that will centralise all the workstations' data and will then emit events to the SIEM.

During the different scenarios, under the MSSP mode of function for the SOC both storage architectures yielded the wanted results and therefore the question of data storage should be solely answered by the needs and the security policy of the organisation-client.

This problem was easily mitigated using tunnelling. A secure tunnel (in our case an SSH tunnel) was used in order to transmit the necessary data from a local collector deployed inside the IT infrastructure, responsible for the collection and centralisation of the log sources, towards the correlator which was deployed in the network of the MSSP. Using this technique, we can ignore the underlying protocol of transmission and ensure that all the security requirements are met by properly configuring the tunnel. Of course, precaution should be taken so as to automatically restore the tunnel in case it goes down or in the event of a malfunction, to raise a high level alert to the SOC in order to manually restore the communication.

Depending on the desired compliance and its regulations, data retention should be adapted in such a way as to also provide a useful platform for post-event analysis and forensics. For example, PCI compliance requires that logs are protected and moreover, that log trails are equally protected in order to ensure that no data alteration has happened. Some regulations

provide different levels of security requirements to be applied on raw events and different ones on alerts. A carefully designed access list should also be defined as to ensure that only legitimate users have access to the stored data. Finally, user's activity should be monitored and stored under the same requirements, in case of a trail's audit.

It is really important to have a SIEM tool that can integrate all types of log formats, no matter the vendor or the technology used. For this reason, regular expression languages (regex) [16] are commonly used. Regex has the potential to interpret and integrate all kinds of input, provided that the expression is correctly written. A fact that it is not trivial due to the high complexity of regular expressions as the diversity of integrated logs grows. For that reason, it is more effective to use multiple regular expressions instead of one complex expression that encompasses all the log data formats. In any case, every regex put into place should be thoroughly tested and simulated against a sufficient number of associated logs in order to avoid mistakes that will lead to unmatched expressions and therefore, to logging data loss. This procedure should take place even in cases where a SIEM vendor already has integrated regular expressions for specific security components as almost all security modules allow personalisation of transmitted logs.

Another issue arises from the vendor related diversity of logs. The source-format can vary but the end event-format should be persistent and compliant to our needs. For that reason, different standards exist and the one used for the needs of the PoC Intrusion Detection Message Exchange Format (IDMEF) [17] was used. IDMEF provides a common protocol (XML formatted) for events' normalisation, according to which all security related events can fit into. There are many others alternatives out of which many vendor proprietary standards. It is important to find one that provides the desired functionality as well as the required flexibility to integrate all our logs.

The duty of log aggregation is undertaken either by the log collectors either by the log correlator. Since log collectors are meant to be only relays for log flows and probably, given a typical IT infrastructure, there will be a need for a large number of them, it would be better to move the aggregation on the correlator. However, in large environments, as the one used for the PoC, the correlator is easily saturated in terms of CPU and memory when both aggregation and correlation is conducted on the same module. Of course, we could just augment the CPU and memory allocation of the correlator but without satisfying results, as in cases of surcharge the correlator was very fast saturated when both aggregation and correlation demands spiked. For this reason, the aggregation rules were distributed amongst the log collectors, dividing the charge and providing a better anticipation of high traffic peaks. As with log integration, log correlation rules have to be tested extensively in order to define the overhead they impose on the log collectors and make sure that no saturation of resources will happen. The proposed decentralised architecture of aggregation offers another benefit in the case where resources' allocation was not correctly conducted, resulting in an overcharged aggregator which is rendered useless by high volumes of traffic (CPU/Memory saturation). In such a scenario, only a segment of the monitored infrastructure is affected (due to the segmented architecture) and the SOC has still visibility on the rest of the network. In that way, the SOC can still operate normally whereas in the case of a centralised aggregator-correlator, the SOC looses all visibility over the infrastructure and therefore, has to conduct its work using other tools or methods.

Event correlation is the final step in the collecting procedure that allows detecting and identifying complicate attack patterns and rejecting false positives. A seen in the simple example of brute forcing an administration interface, we can use predefined rules that will keep track of different events from all over the infrastructure in order to obtain the desired results. However, this procedure is demanding a lot of system resources especially for rules that need to correlate multiple events and over a long time span. For that reason, a classification according to the criticity of the associated equipment has to be conducted. Priority should then be given to critical components over less critical units. Criticity can be measured using the relative impact of a potential attack and the lack of its detection. This procedure demands a good knowledge of the target infrastructure and has to be manually conducted using a common method for events taxonomy. It constitutes one of the main disadvantages of a MSSP mode of SOC over an in-house SOC that inherently will have a better understanding of the needs and architecture of the IT infrastructure. However, the MSSP SOC can compensate by providing its IT security expertise and experience.

If everything is properly configured, the SOC can now receive alerts and events based on a whole infrastructure span. Each generated event should also have a proper, documented response procedure. Ticketing procedures with defined workflows were used for the needs of the PoC; for every alert, there is an associated ticket that will be transmitted using a ticketing framework, regulating the nature of the event, the actions to be taken, the maximum allowed delay as well as the responsible group to undertake the task. The actions to be taken are also documented apart and should be regularly reviewed in order to encompass new equipment or mitigation techniques. For all the actions, the SOC has the supervising role and often is responsible to initiate the relevant procedure.

6.4 Summarized results

Out of our PoC results we can summarize the gained experience to the following:

1. First of all, define the needs that drive the creation/deployment of a SOC.
    a. Proactive IT Security
    b. Organisation-wide, real time, consistent visibility
    c. Policy compliance-regulations required for business model
    d. Threat Intelligence
    e. Incident management
2. Decide whether an internally developed SOC would best fit the needs of the organisation or an external, MSSP-mode SOC is preferred
    a. Staffing challenge of appropriate personnel
    b. Coverage of service (24/7, workdays..)
    c. Urgency to deploy solution
    d. Sensibility of organisation's information, infrastructure schemas that will be shared with an external partner
    e. Infrastructure requirements – Cost effective solution
3. Once the above mentioned are clear, define the scope of the monitored perimeter
4. Define a clear, global security policy, prioritizing needs and requirements
5. Deploy the proper frameworks and tools that will offer the possibility to the SOC to implement all the required actions

6. Fine tune attack detection patterns based on particular needs and requirements. Take into consideration improvements in order to discard false positives
7. Implement a continuous procedure to review, evaluate and adapt policies, procedures and detection mechanisms based on feedback in order to ameliorate the situational awareness provided by the SOC, reduce risks and reaction time, ease administration requirements and maximize effectiveness by
8. Implement a continuous process of threat intelligence in order to stay ahead of emerging threats
9. Implement a continuous process of vulnerability management
10. Follow and adapt to any changes of established procedures due to infrastructure modifications
11. Define workflow procedures in order to provide reliable and near real-time incident handling
12. Once everything is in place, perform tests in order to evaluate solutions and procedures in place. Adapt and fine tune based on results.

7. Conclusion

A common problem encountered in modern IT organisations is that the existence of too many security modules is drowning the security teams with data. Antivirus solutions, global and personal firewalls, normal and reverse proxies, IDS/IPS, network modules and many other components provide an exquisite amount of security information that needs to be monitored and acted upon 24/7. In order to provide a solution for this problem a security team with a specific role was conceived, equipped with the proper tools that will make it possible to deliver a constant, consistent and close to real-time monitoring and reaction. This team was named SOC which stands for Security Operations Centre. The purpose of this team is to provide a situational awareness across the whole IT infrastructure by collecting, normalizing and correlating heterogeneous events in order to generate alerts and dismiss false positive events.

Purpose of this paper is to design, implement and then adapt depending on the results, all the necessary procedures required for the setting up of a SOC as well as for its fine tuning in order to maximize its efficiency. It is also important to document and analyze properly the results in order to facilitate and serve as a guideline for the potential implementation of a SOC by a security analyst. For these purposes, a fully operational network of a company was used, in order to conduct the appropriate experiments regarding the best practises and challenges for setting up a SOC.

The main tool used for the whole procedure undertaken by the security team is a relatively new generation of software, including all the required functionalities, that goes by the name SIEM (Security Information and Event Management). SIEM vendors claim that their SIEM solution can manage and accomplish all the required tasks, offering all different kinds of functionalities. Through the experiments however, it is was made clear that it constitutes no perfect solution and that in any case, proper configuration and constant fine tuning is required to be performed by the members of the SOC team. Sure, the SIEM can be a great framework that will aid the work of a security analyst by providing a common interface to generate alerts and conduct digital forensics over a large perimeter but without the intelligence and intuition of well trained and experienced IT security personnel, it does not inherently have the capabilities to go through the task by itself. A fact that makes MSSP (Managed Security Service Provider) mode of the SOC an attractive solution to companies or organisations that often do not have or cannot afford the level of IT security expertise, required for these operations.

The security modules used as well as all the applied intelligence is described in details as to have a transparent and clear image of the functionalities of the SOC. The distinct functionalities of each separate security tools of the multi-layered defensive tactics are not described in depth as they do not constitute the main focus of this paper. The results of the conducted PoC (Proof of Concept), which included a wide range of IT attacks and exploits (from simple brute force attacks up to Advanced Persistent Threats) demonstrated the great value of a fine tuned SOC equipped with all the proper tools. All the attack scenarios were successfully detected even in the case of advanced threats and complex patterns of attack, using data enrichment and correlation of multiple layers across the infrastructure. An important remark that needs to be underlined is that it is extremely difficult for a malware or an attacker to bypass simultaneously all different security solutions, without raising an alert. And that is exactly the main advantage of having an overall situational awareness across the whole infrastructure.

However, one challenge that had to be met is the treatment of the many false positives. An equally important fact because the existence of too many false positives can actually "hide" a real alert or even worse, make the security team indifferent to alerts. That is why in many cases multiple, redundant levels of data correlation where used as to confirm a valid alert and discard the “noise”.

Finally, it should be noted once more that no level of software complexity can provide the added value of a security expert. Moreover, the procedure of proactively protecting and mitigating threats is a constant one because attacks as well as infrastructure evolve rapidly and constantly, requiring adaptation to the new situation in order to have a consistent overview of the IT security across the organization.

8. References

[1] Gartner 2012-SIEM, 2012

[2] Hydra, 2013

[3] TOR - https://www.torproject.org/, 2013

[4] David Dittrich,Jelena Mirkovic,Peter Reiher,Sven Dietrich, "Internet Denial of Service: Attack and Defense Mechanisms", 2005

[5] Evan Cooke, Farnam Jahanian, Danny McPherson, "The Zombie Roundup: Understanding, Detecting, and Disrupting Botnets", 2005.

[6] SANS Technology Institute, "Assessing Outbound Traffic to Uncover Advanced Persistent Threat", 2011

[7] Colin Tankard, "Advanced Persistent threats and how to monitor and deter them", 2011.

[8] John W. Moore, "From Phishing To Advanced Persistent Threats: The Application Of Cybercrime Risk To The Enterprise Risk Management Model", 2010.

[9] SANS Institute InfoSec Reading Room, "Responding to Zero Day Threats"

[10] T. Holz, C. Gorecki, K. Rieck, and F. C. Freiling. "Measuring and Detecting Fast-Flux Service Networks. In Proceedings of the 15th Annual Network & Distributed System Security Symposium (NDSS)", 2008.

[11] ICANN Security and Stability Advisory Committee (SSAC). SAC 025: SSAC Advisory on Fast Flux Hosting and DNS, 2008.

[12] Colin Tankard, "Advanced Persistent threats and how to monitor and deter them", 2011

[13] Poison Ivy – Remote Access Toolkit

[14] Dambala, Inc., 2010

[15] Cliff Joslyn, Sutanay Choudhury, David Haglin, Bill Howe, Bill Nickless, Bryan Olsen, "Massive Scale Cyber Traffic Analysis: A Driver for Graph Database Research"

[16] SD Shanklin, TE Bernhard, GS Lathem , "Intrusion detection signature analysis using regular expressions and logical operators", 1999

[17] Intrusion Detection Message Exchange Format, 2007