# University of Piraeus
## Department of Digital Systems

Master Thesis

# A Methodology for Building a Log Management Infrastructure

Vasileios Anastopoulos

February 2014

# Advisory Committee

Sokratis Katsikas, Professor
University of Piraeus

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Thesis Committee

Sokratis Katsikas, Professor
University of Piraeus

Konstantinos Lamprinoudakis, Associate Professor
University of Piraeus

Christos Ksenakis, Assistant Professor
University of Piraeus

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

TABLE OF CONTENTS

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**LIST OF TABLES**

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## LIST OF FIGURES

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Abstract

The collection of log data is a challenging operation for organizations that wish to monitor their infrastructure for security or other reasons, while compliance to standards is often mandated by the type of its activities. In this thesis the problem of performing real-time security monitoring on a large scale infrastructure is approached through the proposal of a methodology for the implementation of a log management infrastructure. Already available and related work is used to compose parts of the proposed methodology, avoiding to “reinvent the wheel” where possible, while the discipline of social network analysis is employed to make and justify decisions that where formerly made either intuitively or based on experience and best practices. The methodology concludes at the creation of the repository of the necessary data, while their actual exploitation from their owner is not addressed. Security issues are an essential part of the methodology and embedded were necessary.

The proposed methodology addresses all the critical aspects of a log management infrastructure starting from the documentation of the log requirements and the details of the infrastructure that will be monitored. It continues with the analysis of log generation issues, which devices will be used and what log data need to be generated, how these data will be collected and managed in storage. The additional and critical issues of time synchronization, data preprocessing and infrastructure scalability are also analyzed, concluding with the proposal of a performance measurements process to measure the efficiency of the log management infrastructure and adjust it where it is necessary. The security issues are also examined as separate steps of the methodology.

The result and contribution of this master thesis is an innovative methodology that can be used as step-by-step guide for the implementation of a log management infrastructure in an organization.

This work can be expanded with the addition of log analysis and visualization tasks, as well as managerial issues such as the definition of Standard Operating Procedures (SOP) and the assignment of roles to the personnel. The opportunities offered by the cloud and virtualization technologies are also included as future work.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Περίληψη

Η διαδικασία της συλλογής δεδομένων καταγραφής αποτελεί πρόκληση για τους οργανισμούς που επιθυμούν να παρακολουθούν την υποδομή τους για λόγους ασφάλειας ή για διαφορετικούς , ενώ, η συμμόρφωση σε πρότυπα συχνά υπαγορεύεται από το είδος των δραστηριοτήτων του. Σε αυτή τη διπλωματική εργασία το πρόβλημα της παρακολούθησης σε πραγματικό χρόνο μιας υποδομής μεγάλης κλίμακας, προσεγγίζεται μέσω της υλοποίησης μιας υποδομής διαχείρισης αρχείων καταγραφής. Ήδη διαθέσιμη και σχετική εργασία χρησιμοποιείται για τη σύνθεση της προτεινόμενης μεθοδολογίας, αποφεύγοντας την "επανεφεύρεση του τροχού" όπου αυτό είναι δυνατόν, ενώ, χρησιμοποιείται η ανάλυση κοινωνικών δικτύων για τη λήψη τεκμηριωμένων αποφάσεων οι οποίες μέχρι τώρα λαμβάνονταν διαισθητικά ή βάση εμπειρίας και βάση βέλτιστων πρακτικών. Η μεθοδολογία ολοκληρώνεται με τη δημιουργία αποθετηρίων με τα απαραίτητα δεδομένα, ενώ, δεν πραγματεύεται την εκμετάλλευσή τους στην πράξη από τον κάτοχό τους. Τα ζητήματα ασφαλείας αποτελούν ουσιαστικό κομμάτι της μεθοδολογίας και είναι ενσωματωμένα σε αυτή όπου απαιτείται.

Η προτεινόμενη μεθοδολογία πραγματεύεται όλες τις κρίσιμες όψεις μιας υποδομής διαχείρισης αρχείων καταγραφής, ξεκινώντας από την καταγραφή των απαιτήσεων και των λεπτομερειών που αφορούν την υπό παρακολούθηση υποδομή. Συνεχίζει με την ανάλυση θεμάτων που αφορούν στην δημιουργία των αρχείων καταγραφής, ποιες συσκευές θα χρησιμοποιηθούν και τι δεδομένα πρέπει να παραχθούν, πώς θα γίνει η συλλογή τους και η διαχείρισή τους όσο είναι αποθηκευμένα. Επιπλέον αναλύονται, τα κρίσιμα ζητήματα του συγχρονισμού, της προ-επεξεργασίας και της επεκτασιμότητας της υποδομής, καταλήγοντας με την πρόταση μιας διαδικασίας μέτρησης της απόδοσης, ώστε να μετρηθεί η απόδοσή της και να προσαρμοστεί όπου είναι αναγκαίο. Επίσης, τα ζητήματα ασφαλείας εξετάζονται ως ξεχωριστά βήματα της μεθοδολογίας.

Το αποτέλεσμα και συνεισφορά αυτής της μεταπτυχιακής διπλωματικής εργασίας είναι μία καινοτόμος μεθοδολογία που μπορεί να χρησιμοποιηθεί ως οδηγός βήμα-προς-βήμα για την υλοποίηση μιας υποδομής διαχείρισης αρχείων καταγραφής για έναν οργανισμό.

Η υπόψη εργασία μπορεί να επεκταθεί με την προσθήκη ενεργειών ανάλυσης και οπτικοποίησης των αρχείων καταγραφής, καθώς και θεμάτων διαχείρισης όπως τη δημιουργία τυποποιημένων διαδικασιών λειτουργίας και την ανάθεση ρόλων στο προσωπικό, ενώ, ως μελλοντικό έργο συμπεριλαμβάνονται οι τεχνολογίες "νέφους" και εικονικών μηχανών, λόγω των δυνατοτήτων που παρέχουν.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Chapter 1

# Introduction

The collection of log data is a necessary operation for every organization. Reading the log files is like reading what your systems have to tell you. A security or other type of incident cannot or is difficult to identify and track without the necessary data, extended both in time and detail. Organizations collect data for various reasons among which the most critical ones are related to internal and external security issues. They also use them to detect and track suspicious behavior, as well as to support forensic investigations when needed. The state of the art in security attacks, i.e. the Advanced Persistent Threats (APT), are most of the times detected combining and correlating the log files from various sources [1]. Their importance is obvious by the various standards and laws that require their collection and storage. Depending on the type of company or organization different standards, and consequently requirements, are mandated. Despite the logs usefulness and value for their owner, their management is not achievable at no cost. The dispersion of the log sources, the variety of logging formats and the volume of the generated data make the log collection task a challenging one. Data preprocessing is necessary to bring the data to the adequate format for their actual exploitation, for the tasks of analysis, correlation and visualization. It is not surprising that data preprocessing, correlation and analysis are among the top challenging tasks organizations have to perform [1].

The problem that is dealt with in this master thesis is the implementation of a log management infrastructure in a Wide Area Network (WAN). The underlying problem is the need to perform real-time security monitoring of a WAN consisting of geographically dispersed and heterogeneous devices.

In literature there are available methodologies and best practices for the implementation of a log management infrastructure. Most of them examine and analyze part of the problem, others provide an abstract approach to the problem, while others are too technical and tailored to a vendor specific product. In addition, many of the above make decisions and provide solutions based on experience, without properly justifying them. This work is motivated by the need for a methodology that covers the whole process of implementing a log management infrastructure, addressing both high level and low level issues, accompanied by the adequate documentation.

The proposed methodology is summarized in the following block diagram, Fig.1. It starts with the capture of the log requirements and the collection of data about the WAN that needs to be monitored. The requirements may be dictated from a standard or the security policy of the organization. The available assets are recorded, as well as the network topology and bandwidth of the network links. The fulfillment of the above tasks is left abstract, no specific product or method is proposed. Continuing, a method is proposed to derive what needs to be

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

logged for the purposes of the log management infrastructure and different implementation architectures are examined. The infrastructure is divided into two tiers, the first is the log generation and the second is the log collection and storage tier. At the first tier, issues concerning the log sources are addressed. The second tier is separated into the log collection and the log storage sub-tiers. For the former, the placement of the equipment is decided using the social network analysis discipline and a method for estimating the sizing of the system is presented. For the latter, a life-cycle data management process is defined for the management of the stored log data and the storage needs are estimated. The methodology continues with the critical issue of time synchronization of the participating equipment. The preprocessing of the log data is discussed and the scalability of the log management infrastructure is evaluated. The scalability and preprocessing steps are placed at the end of the methodology, though they could be performed before the architecture of the log management infrastructure is decided. The specific security considerations that arise at each step are part of the proposed methodology, forcing their inclusion at the various decisions that are made. The methodology concludes with a performance measurement program, to measure different aspects of performance and apply corrective actions.

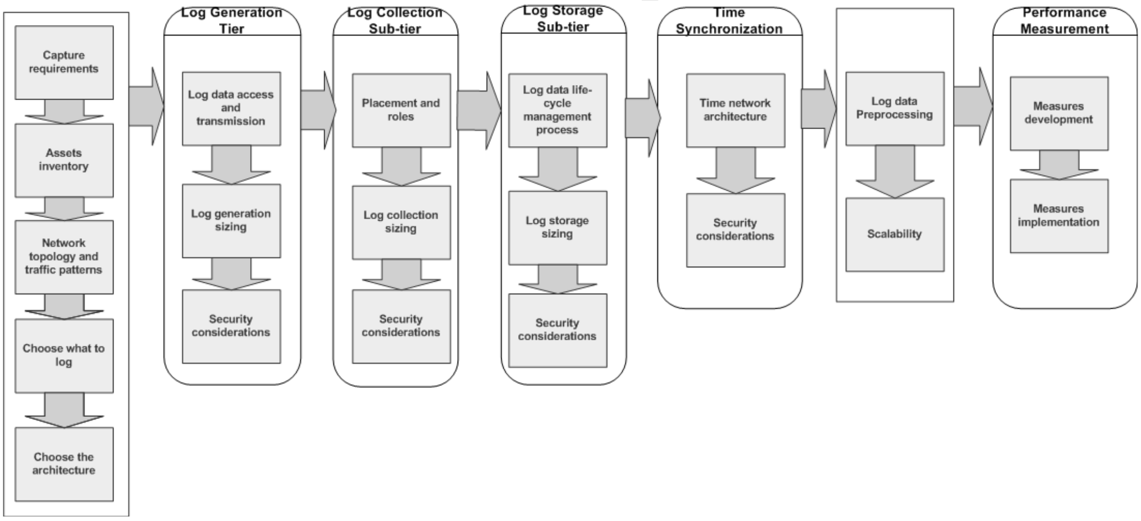


**Figure 1. Methodology Block Diagram**

The contribution of this master thesis is a methodology that integrates available methodologies, both high-level and low-level ones, with industry best practices and introduces the discipline of social network analysis to properly justify and document decisions, that would otherwise be made either intuitively or based on experience. The result of this work is a flexible methodology that can be used as a step-by-step guide for the implementation of a log management infrastructure or a specific part of it.

In Section 2 related work is discussed and in Section 3 the proposed methodology is presented. Following, in Section 4, it is applied as a case study on a real organization. The thesis concludes with Section 5, discussing the conclusions and proposing future work.

# Chapter 2

# Related Work

The log management technologies are presented from a high-level viewpoint in [2]. With this publication the United States of America (USA) National Institute of Standards and Technology (NIST) aims to assist organizations in understanding the necessity of log management. It can be used as a framework for the development, the implementation and the maintenance of a log management infrastructure, addressing most issues from a high level. Log management needs and challenges are presented along with the components, the architectures and the functions of a log management infrastructure. The definition of roles and responsibilities as well as the creation of feasible logging policies are also discussed. Though the authors discuss most aspects of the topic, the various issues and technologies are presented from a high level and though valuable guidance is provided, it cannot be used as a step-by-step guide. In [3] and [4] the authors address the capture of log requirements, the generation, the storage and the analysis of the log data, as well as relevant security issues. They present advantages and disadvantages of existing solutions and conclude into the proposal of best practices. It is a high-level approach that provides useful guidelines and discusses most key issues. A more low-level work is available in [5]. It presents a process for the determination of the organizational requirements, the creation of a repository of devices and the estimation of the volume of log data that is expected to be managed, the system sizing. It aims to facilitate an organization seeking to acquire a commercial product,  by effectively identifying its log management needs and providing criteria for the selection of the adequate solution. This work covers specifics parts of a log management infrastructure and cannot guide the whole process. In [6] the author follows a use cases modeling approach for the implementation of a Security Information and Event Management (SIEM) solution. The proposed method covers the definition of the log requirements and elaborates in the determination of what needs to be logged by an organization to meet its needs. An effective process is presented, which is adopted in the proposed framework, after being adjusted. In [7], the collection, analysis and visualization of the log data is performed in the cloud. It is a vendor specific solution tailored to a specific product. Though it cannot be used a guideline, it is an interesting document considering the advance and the expansion of the cloud technologies. The implementation of a private cloud infrastructure is feasible for large organizations, as well as the use of virtualization technologies in various components of a log management infrastructure, triggering new approaches to the log management process.

In social network analysis, the identification of the key nodes is a common task that is achieved using the measurements of centrality. In [8] various measurements are defined along with their possible interpretations and meaning, depending on the context. The degree centrality is used to identify the nodes that actively participate in the social network, the closeness

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

centrality to highlight the nodes that are close to other nodes, thus can easily and quickly interact with them, and the betweenness centrality to identify the nodes that hold a critical position and can consequently affect the social network. In [9] various methods of analyzing social networks are presented. One of them is the separation of the social network into a core and a periphery part based on the centrality of the nodes. The core of the network is formed from the well connected and well positioned nodes, marking them as important. The position and the connections of these nodes allow enable their quick access to the information circulating the network, in addition to the ease of interaction with other nodes. Removing them would affect the cohesion and stability of the social network. The opposite applies for the nodes of the periphery they are less important and their removal would slightly affect the network. More complex methods are proposed in [10] and [11], where the author addresses the inefficiency of the centrality measures in identifying important and key nodes, and divides the problem into two sub-problems. The first one is to find the set of nodes that if removed would maximally affect the communication among the remaining nodes, and the second one is to find the nodes that are maximally connected to all other nodes.

Concerning the measurement of the performance of the log management infrastructure, a process is presented in [12]. The authors provide a seven-steps process to establishing security metrics. Their aim is to provide an overview of the state of security metrics and to make proposals for the development of a metrics program. The USA NIST, with its [13] publication, provides detailed guidelines for the development and implementation of a performance measurement program. The topic is covered in detail providing guidelines for the identification of the adequacy of security controls, procedures and policies through the use of measures.

The proposed methodology benefits from the related work, integrating and adjusting processes and solutions already present. The high-level guidelines and methods are combined with the low level ones, with proposed best practices and vendor specific solutions. Social network analysis methods are used to identify the important components of the monitored infrastructure. The methodology proposed in this thesis differentiates from the related work in that it is a complete methodology, addressing both high level and low level issues, and innovates introducing the use of social network analysis to make and justify decisions, that would otherwise be made intuitively or based on experience.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Chapter 3

# Proposed Methodology

## 3.1 Capturing Requirements

The methodology starts with the recording of the log generation and collection requirements, i.e. what should be logged, from which devices and for which period of time. Though these depend on many factors and of course from the aims and the type of the organization the following are applicable in most cases.

**Applicable law and standards**: The applicable law and standards depend on the country and the type of the organization. The Sarbanes-Oxley (SOX), Health Insurance Portability and Accountability Act (HIPAA), Gramm-Leach-Bliley Act (GLBA), Federal Information Security Management Act (FISMA) and Payment Card Industry (PCI) Data Security Standard (DSS) are encountered often in the United States [2]-[4]. An organization may have to be compliant to more than one of the above depending on its activities and services it provides.

**Legal counsel:** In some cases the logging activities may result in collecting and storing personal or sensitive data, such as user account credentials, users browsing behavior, etc. In this case legal advice is required to ensure that no privacy or security issues will arise [3].

**Incident validation:** An incident, security breach, policy violation, etc, may not be detected for a long period of time. As the log records are the only evidence of its existence, the log management infrastructure has to able to validate a security incident both on host and network level and track it through out its path. As a consequence, the selected log generators should be able to provide log records of the necessary detail for the performance of such analysis or investigation [14].

**Reports:** Reports is a useful means of reviewing large amounts of log files, gaining understanding of what is "normal" and detecting the "abnormal". The specific reporting needs of the organization will indicate what has to be logged and in what detail. In [15] six categories of critical log information reports, that should be reviewed on a regular bases, are listed as follows:

- **Authentication and authorization**: Successful and failed authentication attempts, as well as the execution of privileged activities.
- **Change**: Changes in configuration of information systems and network devices.
- **Network activity**: Suspicious or dangerous network activity.
- **Resource access**: Access patterns across the organization.
- **Malware activity**: Events that are probably relevant to malicious software.
- **Critical errors and failures**: Indications of systems errors and failures.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**Security policy:** The security policy of the organization sets the barriers of acceptable personnel behavior and resources usage. The log files generated have to provide the necessary information to monitor users' compliance and hold them accountable in case of violation.

**Use cases:** The security events that are anticipated can be used for the logging requirements generation. Fraud, identity theft, data leakage and other security incidents can be used as use cases. These will result in the information, i.e. log files, that are necessary to detect, mitigate security incidents and to initiate the appropriate legal actions. Security incidents that have affected other organizations are a valuable resource for the creation of such use cases.

**Risk assessment:** It is the process of identification, estimation and prioritization of information security risk [16]. Estimating the risk of each asset is valuable in order to determine the systems criticality and derive, in consequence, the logging requirements.

Depending on the organization it may not be necessary to perform all of the above tasks. Only the applicable ones can be selected, enriched or otherwise adjusted to meet the specific needs of the organization. The output of this step is a list of requirements that the log management infrastructure should satisfy. For example, an organization that manages card holder information should be PCI compliant, resulting in the following requirements (sample from [3]):

- Logging of individual access to protected information.
- Logging administrative actions and use of privileged accounts.
- Daily review of logs and 24/7 monitoring for unauthorized access.
- Security and protection for logged data.
- At least one year of logs retention.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## 3.2 Assets Inventory

The full spectrum of the devices used from the organization is recorded [5] in this step of the methodology. This includes network devices (routers, switches, wireless access points), security devices (firewalls, Virtual Private Network(VPN) servers, Intrusion Detection Systems(IDS)/Intrusion Prevention Systems(IPS)), servers, desktop computers, mobile devices (tablets, laptops, smartphones), as well as network enabled printers and scanners. Its crucial to include standalone devices, e.g. a desktop computer that though not connected to the network, files may be transferred using removable storage media. An inventory of the available resources is compiled containing the type of each device, the operating systems and its versions, the services and applications running (Dynamic Host Configuration Protocol (DHCP), Active Directory, anti-virus software, etc). Each of these devices, services or applications, is a potential log generator that has to be recorded and documented.

The output of this step is a detailed list of the different types of devices, their multitude and the services or software that is installed and running.

## 3.3 Network Topology and Traffic Patterns

The position of each device in the network and the network bandwidth of each connection is recorded to compile the network topology diagram. The traffic patterns exhibited are necessary and included in the topology. These patterns can be derived using historical data from a performance monitoring package such as Cacti, Nagios, etc. In order to identify the traffic patterns, the available data has to be divided to different time periods such as working days, non-working days, peak working hours, weekends, etc according to the organization's activities. Mining for patterns can be performed using statistics analysis or data mining, though further elaboration is out of scope of this work. In case no historical data is available, at least the average and maximum observed bandwidth needs to be estimated.

The output of this step is the network topology diagram, enriched with the nominal bandwidth of each network link, the average and maximum bandwidth that are observed in its every day operation.

## 3.4 Choose What to Log

Choosing which devices should be logged and in what detail is not a trivial task. The multitude of the log generators and the amount of log files they generate can lead to huge log files difficult to analyze and demanding in computational and storage resources. To further complicate this task, log generators can be scattered to various places, the machine time they use may not be accurate or synchronized and in addition, no standard log record format is followed by vendors and developers. In [3] it is proposed that everything should be logged as the needs of a future forensic or other type of investigation, can not be safely predicted and the absence of the necessary data may impede them or make them inefficient. On the contrary, in [2] and [17], logging everything is considered a non realistic approach as it largely increases the storage, processing and network requirements. In order to determine what needs to be logged, the proposed methodology adopts and adjusts the process of selecting a SIEM system described in [6], abbreviated as Top-Down Bottom-UP Middle-Out (TDBUMO). The output of the three preceding sections is the input of this process.

**Top down:** The log management infrastructure, considered as product independent, is represented as the root node of a tree structure. The system types are grouped in the next level of the tree and are further detailed at the following tree level, specifying the versions of these systems. At the last level, each leave is a type of the logs that the parent node (system) can generate. This provides an understanding of the different types of log sources and the data flow inside the log management infrastructure. The output of this step is a tree structure, Fig.2, where the leaves contain all the specific types of logs that can be generated from these devices.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

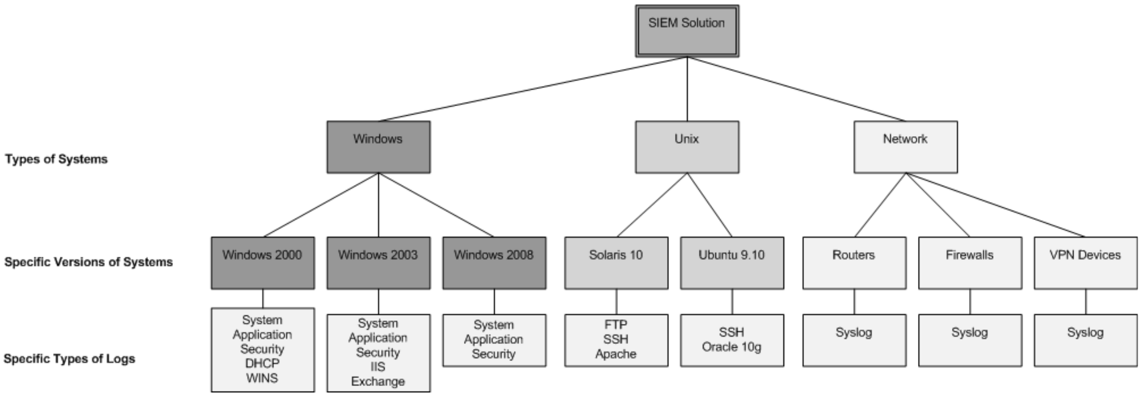


**Figure 2. High-level Log Files Tree**

**Bottom Up:** Starting from the leaves of the tree the logging capabilities of each device are documented in detail. The logging levels, the format of the records and the ways of accessing or retrieving the log files (agent, agent-less) are recorded. The output of this step is a detailed document of the specifics of each log file. Table 1 lists some fields that can be used for this task.

**Table 1. Log File Details Example**

| | |
|---|---|
| **Log file name** | error.log |
| **Log file description** | Error conditions |
| **Log file location** | /var/log/apache2 |
| **Log level** | Emergency, Alert, Critical, Error, Warning, Notice, Informational, Debug |
| **Log format** | Common Log Format (CLF) |
| **Record fields** | Remote host, Remote login name, Remote user, Time the request was received, First line of request, status of the *original* request, Size of response in bytes |
| **Application name** | Apache2 |
| **Application version** | Apache/2.2.22 (Ubuntu) |
| **Storage** | Raw files |
| **Rotation frequency** | Daily |
| **Access/retrieval** | Agent |

**Middle Out:** The specific types of logs that were identified and detailed in the previous steps, are mapped to the requirements that were gathered in the corresponding section. Either compliance needs, risk analysis or use case driven, the necessary fields are identified and selected for logging, along with the necessary verbosity level, to meet the needs of the organization. The record format of each log file is crucial, as the contained fields indicate whether the

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

necessary information is available or not. These specific record fields will also serve as the common element for the correlation of the log files, regardless of whether they were generated from the same device or not.

A matrix composed of the requirements and the specific types of logs is created. For each requirement, the log files that are necessary are marked on the matrix. At each marked cell at least the log level and the log format are recorded. Table 2 depicts a proposed matrix for this task, though the fields are indicative and more can be added based on the needs of the analyst.

**Table 2. Requirements Mapping Matrix Example**

| **Specific system** | **Specific type of log** | **Requirement** | | | |
|---|---|---|---|---|---|
| | | Requirement 1 | Requirement 2 | Requirement 3 | Requirement 4 |
| System 1 | Log file 1 | | Log level/format | | Log level/format |
| System 2 | Log file 2 | Log level/ format | Log level/format | Log level/format | Log level/format |
| System 3 | Log file 3 | | | | Log level/format |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Having completed the three steps of the TDBUMO process all the potentially useful log sources have been grouped and the details of their logging capabilities have been recorded. Thus, the proposed methodology has concluded so far to which sources will be included in the log management infrastructure and which of their log files contain the necessary data to meet the defined requirements. Defining the logging level provides the required verbosity and avoids the generation of large amounts of unnecessary and out of scope log records, that could increase the computational demands and harshen their analysis.

## 3.5 Choose the Infrastructure Architecture

In this step of the proposed methodology a high level decision on the log management infrastructure architecture is made based on the findings of the previous sections, the budget constraints, the goals of the organization in combination with its future plans and needs (scalability, integration with a SIEM system, organizational changes, etc).

A log management infrastructure is composed of hardware, software, networks and media that generate, transmit, secure, store, analyze and dispose log data [2],[4]. It consists of the following basic tiers:

- **Log generation**: This tier includes the devices that generate data, the log generators. These data are made available to the log servers of the following tier, running an application (agent), a service (e.g. syslog) or allowing servers to remotely access the log files (agent-less).
- **Log collection and storage**: This tier includes the log servers that accept the log data or

copies of them, sent from the previous tier. The collecting servers are called collectors or aggregators. The transmission of the data can be performed either in real-time or in batch mode. The data can be stored on the collectors them selves, on separate database systems, on Network-Attached Storage (NAS), on Storage Area Network (SAN) or other storage solution.

- **Log monitoring**: This last tier is composed of the user interface that is used to review the log files, to perform analysis and generate reports, as well as to administer the log generators and the collectors. The proposed methodology does not discuss this layer, but it is included in the design process to facilitate the integration of the resulting log management infrastructure with a SIEM or log analysis product.

Decisions have to be taken concerning the architecture of the log management infrastructure. An infrastructure can be centrally-managed, distributed or fragmented [4] and the transfer of the log data can be performed over the normal network of the organization, over a physically or logically separated logging network [2]. For the first decision a choice can be made among the following basic types or combinations of them:

- **Centrally-managed:** Having a single log management infrastructure is accompanied by the advantage of being able to review all the available log data. It is considered to be the only realistic approach in order to apply best practices [4], achieve accuracy in their collection and security in their storage.
- **Distributed**: For a large organization the centrally-managed approach is not considered feasible, due to the volume of data that has to be collected, stored and analyzed [2]. Using separate infrastructures is proposed, having each one handling a certain scope of the monitored infrastructure. The scopes can be defined based on the types of systems, the types of logs, the physical location of the equipment, etc. These separate infrastructures may operate independently or inter-operate through a central management point. Data logs are collected and stored in each separate infrastructure, enabling their fast retrieval and aggregation. This architecture is flexible and easy to adopt to organizational changes [4].
- **Fragmented**: In this type of infrastructure each department, network, system or even log generator, implements its own log management solution and only the corresponding administrators can have access to the log data. Though guidance may be provided from the organization, the application of policies and processes is not feasible. Log aggregation or correlation cannot be performed  and there is no efficient way for the organization to compile an overall picture of its network [4].

The second decision that has to be taken is to choose whether a separate and dedicated network will be used for the transmission of the log data or the normal one.

- **Physically or logically dedicated network**: Using a separate network poses performance and security advantages. For example, in the event of a malware infection the normal network may become unstable or overloaded, rendering the log data delivery unreliable. Handling a security incident is more efficient over a fully functional log network, than over a performance degraded one. Due to the difficulty and the cost of im-

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

plementing a separate network, a dedicated one can be selected for the connection of the critical devices (log servers, IDSs, etc), while the non-critical ones can operate over the normal network of the organization [2].

- **Over the normal network**: This approach does not have any benefit apart from the lower cost in terms of investing in hardware or software, as well as simpler configuration and ease of maintenance.

The above approaches are the basic ones and an organization may adopt a combination of them according to its specific requirements, physical dispersion and of course its available budget. The output of this step of the proposed methodology is a decision on the architecture of the log management infrastructure and the type of network to be used. The assets of the infrastructure are mapped and logically divided to the tiers defined by the selected architecture.

## 3.6 Log Generation Tier

The log generation tier includes the logging devices, log generators, that were selected during the TDBUMU process. For these devices the way of accessing their log files has to be defined as well as whether their local storage capabilities will be used or not. Security issues affecting the confidentiality, integrity and availability of the log generators are also addressed, but the most important is to estimate the volume of data that will be generated and transmitted.

**Log data access and transmission:** An agent can be installed on the log generator. This agent can access and parse the log files and its specifics depend on the software product. The use of regular expressions for the parsing of the log records is common practice. The transmission to the collector, i.e. a sensor of the product, is handled without intervention of the hosting system and it is most time transparent to the end user. A disadvantage of this approach is that changes in the log files format have to be reflected to the agent modifying the corresponding regular expressions and administrative overhead is added, as they need to be installed, configured and maintained. On the other hand it facilitates the log collection process, since the access, transmission and normalization of the log data are automatically performed, usually with a small footprint on the system resources. The transmission of the data is commonly performed over the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP), either in real-time or in batch mode. Depending on the product additional features as encryption or integrity checking may be available. Installing an agent is a good solution for servers and workstations [18].

An agent-less approach has the advantage of not requiring the installation of any additional software, thus having small impact on the systems and the administrative personnel. The server usually authenticates to each host and the log data are then pulled. A disadvantage of this approach is that no log filtering or normalization is performed on the individual host, increasing the amount of data that is transferred and processed on the receiving server. This can prove to be system intensive and slow across WAN connections. In addition, a credentials man-

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

agement method has to be employed, since the receiving server has to authenticate it self to each of log sources [2],[5].

A popular means of accessing and transmitting log data is the syslog protocol [19]. It is pre-installed in most Linux distributions and supported by network devices, such as routers and switches. Some popular implementations of the protocol are syslog, rsyslog, ng-syslog and Kiwi server. Rsyslog is currently the default implementation in Debian and RedHat based Linux distributions. It supports the transmission of messages over UDP, TCP, Reliable Event Logging Protocol (RELP) and Transport Layer Security (TLS), as well as storage of the log data in relational databases. Additional advantages are its filtering, relaying and caching features. The UDP protocol does not provide reliable delivery and it is not recommended by any means. The TCP protocol provides reliable delivery but with some caveats. In [20], the author addresses the "unreliability" of TCP. On a log generator, when TCP receives and buffers the messages it reports success to the client. Thus, in case of a system crash the syslog messages stored in the buffer are lost, while the client considers them sent. The RELP protocol is designed for reliable log delivery [21], though not mature and widely tested yet. Logs generated from routers and switches can also be accessed via SNMP traps, but the usage of syslog is encouraged due to its verbosity and its ability to identify more exceptions and degradation warnings in a network [22] compared to SNMP.

In literature [3],[4] it is recommended that log data should be transmitted without performing any filtering or preprocessing on the log generator, as the exclusion of some data may negatively affect future audits or forensic investigations. The configuration of the generators has to be tested to ensure that the required fields are indeed generated and transmitted to the collectors.

**Log generation sizing:** At this step the size of the log files that will sent from the generators is estimated. The number of devices and the log files that will be used are available from the output of the previous sections. A common and consistent metric used for this task is the Events Per Second (EPS), which is defined as the number of events a device can generate or receive in a second [18]. The EPS is estimated for each category of log generators, as they were grouped at the TDBUMU Top-Down part of the process. Depending on the time or other type of constraints, the log files can be collected from all the participating devices or from a single device, for each category and generalize the estimation for the whole category. The average and the maximum EPS are calculated. Depending on the amount of the available historical data, they can be mined for patterns, e.g. working hours versus non-working hours, working days versus weekends. For large data sets the application of data mining algorithms and statistical analysis provides valuable results. In [5] a data set of at least one week is recommended for the log generation sizing estimation.

Only the necessary log data, as derived from the TDBUMO process, are included in the estimation. The resulting data set is this way reduced to the events that are of interest and intended to be managed by the log management infrastructure. Specific time periods are selected and the following formula is calculated twice, one for the average and one for the maximum number of events [23]:

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

*EPS = number of events / time period in seconds*

Depending on the transmission method, e.g. syslog protocol, multiplying the EPS with the maximum message length (2048 octets for syslog [19]), the average and maximum bandwidth is estimated using the following formula:

*Bandwidth= EPS x (size of event)*

The event size can also be estimated from the log files already present on the log source. The maximum record length found in the log files and the average one, can provide a good estimation of what is usually generated from the specific log source. This process can only result into an estimation as log records vary in length and the log generation rate usually oscillates. In addition, the underlying protocols (Layer 4,3,2) add overhead to the transmission and additional protocol specific functions, such as IP fragmentation, add overhead and increase the amount of data that is finally transmitted over the network and form the value of the used network bandwidth.

**Security considerations:** Physical access has to be restricted. Privileges of system users must not allow the deletion or the modification of the log entries, limiting them to the addition of records. In case log files are stored locally on the log source, their integrity and availability must be protected. Operating systems and network devices provide security mechanisms to restrict access, while the use of encryption and hash functions can assure the confidentiality and integrity of the logged data. The log files should be regularly backed up to ensure their availability and recovery.

The output of this step is a decision on the method that will be used to access and transmit the log data, an estimation of the EPS that will be sent and the bandwidth that will be used, as well as the means of protecting the confidentiality, integrity and availability of the log generators.

## 3.7 Log Collection and Storage Tier

### 3.7.1 Log Collection Sub-Tier

In this tier two approaches can be followed. The first is the *syslog* based and the second is the vendor specific. The vendor specif approach depends on the product and its architecture. They are usually composed of agents installed on the log sources sending log data to a central collection point, after they have been preprocessed. The central collection point is divided in an analysis engine, a storage mechanism, the agents manager and the user interface. A hybrid approach is also feasible since most vendors support the *syslog* protocol in their products.

Following the *syslog* based approach, the second tier is composed of the log servers that receive the log data, either the original ones or their copies. The log servers can be as-

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

signed one or more of the following roles [2],[19]:

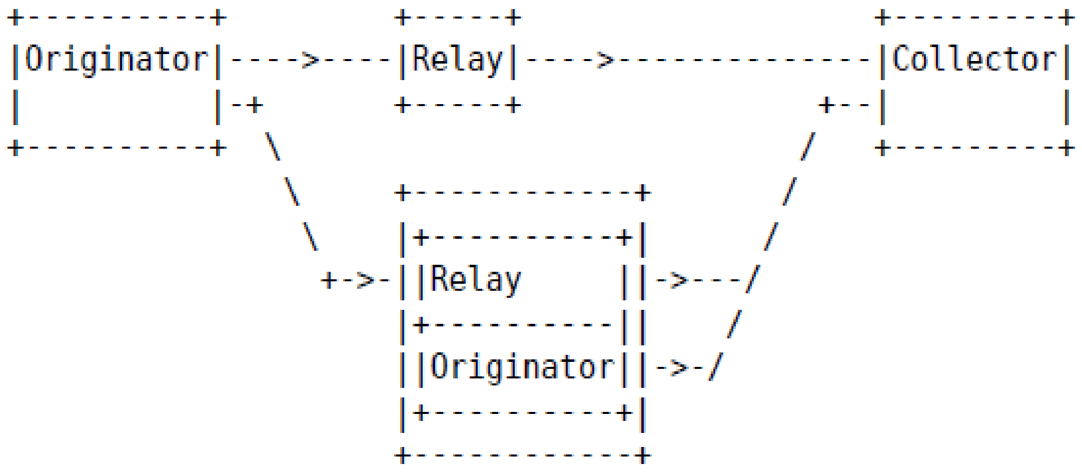
- **Originator**: The log server generates its own log files concerning its function. The log data can be stored locally or be transmitted to one or more destinations.
- **Cache/Relay**: The log server collects data from other log sources and simply forwards them to other log servers. They can be used to protect the collectors from peaks in traffic or be placed at the network or the collection bottle-necks.
- **Collector/Aggregator:** These servers receive and store log data either on themselves or on separate storage media. Each one usually servers a group of originators and/or cache servers.

It is a flexible tier and various combinations of the above server roles are possible, adding more levels as needed [2],[19],[22],[24]:

- Multiple log servers each performing a specific task. One server may be used for log analysis, an other server for live/production data [3] and one for archived data.
- Multiple log servers performing analysis and/or storage for a category or specific log generators. One server could analyze and/or store router log files, while another could perform the same tasks for server systems. Adding such levels benefits the infrastructure in terms of redundancy, as a log generator can switch to a back up server in case of system or communication failure.
- Two or more levels of distributed log collectors that preprocess or simply forward the log data to a next tier of more centralized collectors. A tier of caching servers can be included to protect the infrastructure in case of a traffic peak or an attack, as well as alleviate the problem of low network performance. This approach adds flexibility scalability and redundancy to the log management infrastructure.

A distributed hierarchy of log servers is proposed in [22], where the collectors are placed close to their generators in a hub-and-spoke fashion. Preprocessing of messages (filtering, separation per log source or application) prior to being forwarded is also advised to remove unnecessary data from the network. An example deployment combining *syslog* servers in different roles is depicted in Fig.3.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ



```
+----------+        +-----+           +---------+
|Originator|---->----|Relay|---->------------|Collector|
|          |-+      +-----+        +--|         |
+----------+  \                   /   +---------+
               \   +-----------+  /
                \  |+---------+| /
               +->-||Relay    ||->---/
                   |+---------||  /
                   ||Originator||->-/
                   |+---------+|
                   +-----------+
```

**Figure 3. Example syslog Deployment From [19]**

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕ

### 3.7.1.1 Placement and Roles

In this section the placement of the various components of the log management infrastructure is decided. As the components of each commercial product depend on the vendor, the proposed methodology elaborates in *syslog* implementations. Nonetheless, an agent can be considered as an originator and the component that collects the events can be considered as a collector, thus fit the architecture of the specific product to the proposed methodology.

The architecture of the log management infrastructure has already been decided in a previous section. This imposes restrictions and can also guide the placement of the servers. For example, if a distributed architecture has been chosen, this means that the originators will be divided into groups, each of which will send its log data to a specific collector. Choosing the central point of the infrastructure is usually dictated by the location of the organization's headquarters, the location of the Network Operations Centers (NOC) or other location of interest. The placement criteria that are considered by the methodology are the following [22]:

- **Geographic location**: For a WAN that extends across multiple locations the placement of a collection point to each region is proposed.
- **Collectors close to their originators**: The collectors have to be placed close to their originators. This is desired as a network problem or the spread of a malware infection could disrupt the log collection process.
- **Hub-and-spoke architecture**: Many originators forward the log data to a collector and many collectors forward them to a central point, or in the case of multiple layers, to a central collection point at the following layer.
- **Hierarchical:** The placement of the components has to follow a hierarchical fashion, starting from the originators and ending to the central collection point, where all the log data are collected and processed by the organization.

To address the above criteria Social Network Analysis (SNA) is applied to analyze the structure of the network and the characteristics of its components, as well as to detect and interpret patterns of social ties among actors [8],[9].

A *node* (or *actor*) is a social entity. It can be a discrete social unit (an individual), or a collective social unit (a group of people, a corporate department, etc). Though termed actors it is not implied that they have the ability to act. *Links* (or *social ties*) connect actors establishing a tie between a pair of actors. A *relation* is the collection of a specific kind of ties, formed among the actors of a specific set of actors. Social networks are composed of nodes and links. These nodes relate with other nodes through their links. When the links have a direction the network is *directed* and the link from node A to B is different from node B to A. When the link direction is not specified, the network is undirected and the link from A to B is not different from B to A. A node can have one or more attributes and a link can be binary of valued. Using graph theory notation $G=(V,E)$ is a social network $G$ with $|V|$ nodes and $|E|$ links among them. It is represented by a $|V| \times |V|$ adjacency matrix, were the existence of a link between node $v_i \in V$ and node $v_j \in V$, is indicated by a value in the $e_{ij} \in E$ cell.

The social network is constructed based on the network topology diagram and the network traffic. These data are already available from the output of the previous sections. The log sources are the actors/nodes and the network connections among them are the social ties/links of the social network. The network bandwidth (nominal, maximum and average) is used as the value of the links. Four relations are constructed resulting in four adjacency matrices. One for the nominal bandwidth, one for the maximum and one for the average. The fourth one uses binary values denoting the presence or the absence of a link.

The analysis of the social network aims to identify the important nodes of the network based on the placement criteria previously defined and the SNA theory. Though SNA can analyze many aspects of a network in this step of the methodology only the identification of the important nodes is required. A node is identified as important when it is close to other nodes (geographic location and closeness to the originators) and when it connects to many other nodes (hub-and-spoke fashion). The aim is to place the collectors to specific locations where the will be close to and easy to connect to as many originators as possible. For the purposes of the analysis, the total degree, the closeness, the eigenvector and the betweenness centrality [25] are selected among the available centrality measures.

- **Total degree centrality:** The degree centrality is the number of links a node has. It is distinguished into in and out degree, when the links are directed to or from the node, respectively. The Total Degree Centrality of a node is its normalized in plus out degree. Let $G = (V,E)$ be the graph representation of a square network and a node $v$. The Total Degree Centrality of node $v = deg / 2 * (|V| - 1)$, where $deg = card\ \{u \in V | (v,u) \in E\ V\ (u,v) \in E\}$ ([8] as cited in [25]). A node with high degree centrality is a well connected node and can potentially directly influence many other nodes [10].
- **Closeness centrality**: It is the average geodesic distance of a node from all other nodes in the network. The geodesic distance is the length of the shortest path between two

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

nodes. Let $G = (V,E)$ be the graph representation of a square network, then the closeness centrality of a node $v \in V$ is $v = (|V| - 1) / dist$, if every node is reachable from $v$ and $v = |V|$ if some node is not, where $dist = \Sigma d_G(v, i), i \in V$ ([26], as cited in [25]). The closest a node is to others, the fastest its access to information and greater its influence to other nodes [27].

- **Eigenvector centrality**: It is a measure of the node's connections with other highly connected nodes. It calculates the eigenvector of the largest positive eigenvalue of the adjacency matrix representation of the square network. To compute the eigenvalues and vectors a Jacobi method is used ([28], as cited in [25]).
- **Betweenness centrality**: It is defined, for a node $v$, as the percentage of the shortest paths, between node pairs, that pass through $v$. Let $G = (V, E)$ be the graph representation of a symmetric network. Let $n = |V|$ and a node $v \in V$. For $(u, w) \in V \times V$, let $n_G (u, w)$ be the number of geodesics in $G$ from $u$ to $w$. If $(u, w) \in E$, then set $n_G (u, w) = 1$. Now, let $S = (u, w) \in V \times V \, | d_G (u, w) = d_G(u, v) + d_G(v, w)$ and let $between = \sum(n_G (u, v) \times n_G (v, w))/n_G (u, w), (u, w \in S)$, then the betweenness centrality of node $v = between/((n - 1)(n - 2)/2)$ ([26], as cited in [25]). A node with high betweenness is important because it connects many nodes and a possible removal would affect the network.

It should be noticed that adding or modifying the placement criteria could result to additional measurements and analysis, though the above measurements are an efficient measure of the nodes' importance [10].

The binary relation is used to perform the measurements. After their calculation the nodes are sorted based on their total degree centrality in descending order. When a node is highly ranked, it means that it has many connections with other nodes with distance one, meaning one hop away. Placing a collector at this node's location is adequate, as it is close and directly connected to many log sources. A cache server could also intermediate at this node between the log sources and the collector. The fact that many log sources are adjacent to this node indicates that high traffic and high EPS rate is expected. The analyst then identifies were the total degree centrality decreases suddenly. This sudden drop can be used to map the nodes into layers. The low ranked nodes could form a layer that would forward its log data to a higher ranked node, in a hub-and-spoke mode. The same process is repeated for the closeness centrality. When a node in highly ranked, it means that it can be reached from other nodes with few hopes, i.e. with the intermediation of few devices. A highly ranked node location is adequate for a log collector since log sources are close to it and a device failure, that would make some network paths unavailable, is less likely to affect it. On the contrary, placing a log collector to a node location distant to the log sources would increase the risk of failing to deliver the log data. The administrative overhead is also lessen, avoiding the modification of firewalls or other devices that may otherwise need to be reconfigured to allow the log data to flow through the network. The process is repeated once more for the eigenvector centrality. A highly ranked node is a node that has a lot of links with nodes that are well connected too. A node with low eigenvector centrality is connected with nodes that have few connections. The location of a node with high eigenvector centrality is a suitable location for the placement of a

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

collector or for the central collection point.

At this point the locations where the collectors, or the cache servers, could be placed have been identified. The analysis continues with the identification of the nodes that if removed would increase the number of the social network components (maximal connected sub-networks), the "*boundary spanners*" [25]. These nodes, also referred as *gatekeepers*, hold a critical position in a social network as removing such a node results into sub-networks that do not link to each other. The value of each node is calculated as the ratio of the betweenness centrality to the total degree centrality of the node. Nodes with high betweenness centrality and low total degree centrality are identified as boundary spanners. If a router is a boundary spanner and for some reason fails to route the traffic, the result will be the partitioning of the log sources to sub-networks unable to communicate outside their subnet. As a result, a collector placed to the same location would fail to communicate with its originators. Placing the equipment of the log management infrastructure on such nodes, should be avoided. The SNA measurements do not mandate the placement of the components of the log management infrastructure. They provide a means of identifying the important nodes and the respective locations in the WAN. The measurements can be used independently or combined to form new metrics.

The methodology continues with the assignment of originators to collectors or cache servers. The originators have to be divided into groups, were each group will forward its log data to the same collector or cache server (more that one destination can be assigned). To achieve this the Newman algorithm is employed [29]. It is an agglomerative hierarchical clustering algorithm for detecting community structure in large networks. At the starting state of the algorithm each node is the only member of a community. At each step, the communities are repeatedly joined into pairs choosing the join that results in the greatest increase in modularity. *Modularity (Q)* is a network property proposing a specific division of that network into communities. If the division is good there will exist many links among the community nodes and only a few links between the communities. In [29] the modularity is defined as

$$Q = \frac{1}{2m}\sum_{vw}\left[A_{vw} - \frac{k_v k_w}{2m}\right]\delta(c_v, c_w).$$

where $k_v k_w/2m$ is the probability of an edge existing between nodes $v$ and $w$ if the links are formed at random, but respecting the nodes degrees. $A_{vw}$ is the adjacency matrix of the network and $A_{vw}=1$ if nodes $v$ and $w$ are linked and $A_{vw}=0$ otherwise. The δ-function $\delta(i,j)$ is 1 if $i=j$ and 0 otherwise and $c_v$ denotes that node $v$ belongs to community $c_v$. If the fraction of the links inside the community is the same as for the randomized network, then $Q=0$. High values in modularity indicate good division of the network into communities. The progress of the clustering can be represented as a tree that shows the ordering of the joins.

The grouping that outputs the Newman algorithm assists the analyst to assign the originators to collectors or cache servers, as well as to validate the placement, checking whether a collector has been placed close to each group. On the other hand, having placed many collectors into a small group may indicate an error in the placement.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

#### 3.7.1.2 Log Collection Sizing

The output of the previous step is used for the log collection sizing. The amount of log data that each originator transmits is already available as the output of the log generation tier, thus the estimated EPS that each collector will receive and retransmit, in the case of a cache server, is calculated. For example, a collector that receives log data from five log originators, will receive the sum of their corresponding EPS. The bandwidth calculation has been addressed into previous section and the use of volatile memory calculated as follows:

*Volatile memory= EPS x event size*

where *event size* is the size of each event that is transmitted or received by the collector. In the case of a *syslog* implementation this can oscillate from the minimum to the maximum *syslog* packet size. Depending on the log management solution and the configuration options, more parameters can be added to the formula. In *rsyslog*, for example, queues can be configured and allocate volatile memory to them. Based on this estimation the necessary hardware can be defined.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

#### 3.7.1.3 Security Considerations

When the *syslog* protocol was designed security was not a concern of high priority. The transmission over UDP has the disadvantage of unreliable delivery, there is no confirmation for the reception of the message. The *syslog* messages are transmitted in clear text over the network and neither the sender's nor the receiver's identity is authenticated. An eavesdropper can potentially capture the traffic and read the content of messages, as well as affect their integrity modifying their content. As a result a security incident can be hidden from the security personnel modifying the transmitted log data. A replay attack is also feasible, as the attacker can capture the *syslog* messages, alter their timestamps and send them once again to their destination. A masquerade attack can be effective as the attacker can impersonate another originator. There is no way to distinct the original from the replayed messages or distinguish a legitimate from an illegitimate originator. A Denial of Service (DoS) attack is also possible, when the attacker sends more data to the collector than it can handle, resulting in loss of the necessary log data due to the exhaustion of the computational and/or storage resources of the collector.

The unreliable delivery can be alleviated transmitting over TCP instead of over UDP. The confidentially, integrity and authenticity of the messages can be protected employing the TLS protocol. Each sender and receiver has to install its own certificate to verify its identity and perform the required encryption functions. Even this way the traffic remains exposed to traffic analysis and adds the administrative overhead of key management and devices' configuration and maintenance. The implementation of VPNs is a feasible solution, while some implementations provide the feature of sending the *syslog* messages through Secure Shell (SSH) tunnels. The use of TLS or SSH increases the bandwidth consumption and the time needed for the transfer of the data [4],[24]. The availability is also protected using TLS or SSH, since an illegitimate user can not send data to a receiver. In addition, configuring the receiver to have a rate limit or protecting it with a firewall, can pose restrictions to the receiving traffic and protect against some DoS attacks.

The output of the log collection sub-tier section is the placement of the log servers, the role each one will be assigned, the assignment of originators to collectors and the log collection sizing i.e. the EPS each server is estimated to receive, the required bandwidth and the volatile memory that will be consumed. The necessary security measures for the protection of the log data are also included in the output.

### 3.7.2 Log storage Sub-tier

#### 3.7.2.1 Log Data Life-cycle Management Process

Log storage is a critical component of the log management infrastructure. For an organization willing to effectively manage the volume of log data, a data life-cycle management process is required. In order to design such a process the data stages, the storage mechanisms, the amount of log data and the functions that will be performed need to be considered. The log data go through the following stages [4]:

- **Production/live data:** It is the data that are used for real-time analysis and on-going review.
- **Back up data**: These are a copy of the production data, intended to be used in case the first ones become unavailable.
- **Archive data**: It is data that are kept in long-term storage for regulatory or forensic reasons or for the benefit of the organization, such as the performance of data mining or statistical analysis.
- **Disposed data**: These data are no longer necessary for the organization and they are disposed from the infrastructure. Depending on the sensitivity of data and the security policy, they can be simply deleted or securely removed to avoid their recovery and a possible data leakage.

Depending on the data access requirements various storage mechanisms can be employed [3]:

- **On-line storage**: The data are stored in high-performance systems, such as NAS or SAN, where the access time is a few milliseconds. The data are available to a large number of users.
- **Near-line storage**: In these storage systems the access time is measured in seconds. The data are available for infrequent use to a small number of users.
- **Off-line storage**: The data are stored on external media. They cannot be accessed unless mounted to the system.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

The above storage mechanisms can be implemented using databases or raw files:

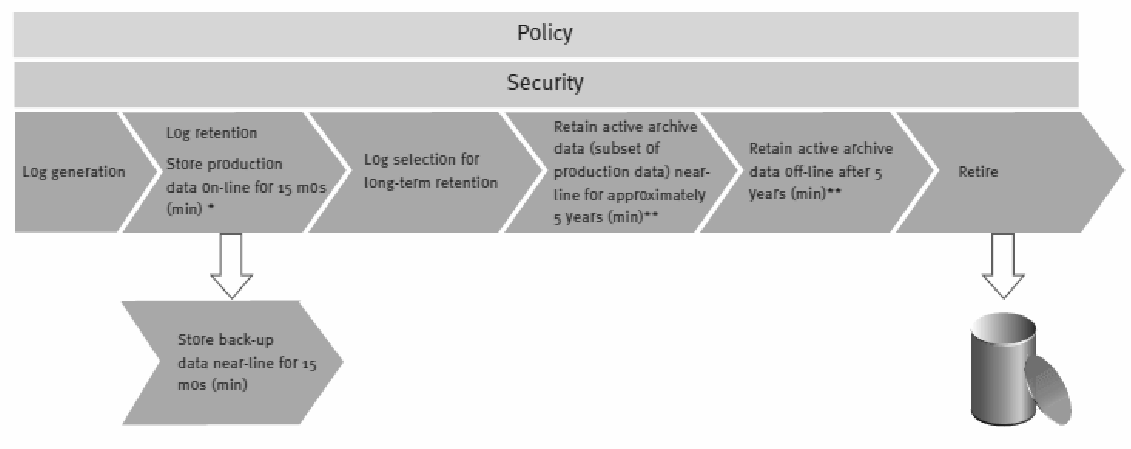
- **Raw files:** They have the advantage of speed both in write and read operations. In addition, they can store the data in their original format. This is important to maintain the data in forensically sound condition. On the other hand, its more difficult to perform the processing tasks and an application usually needs a parsing mechanism to access and retrieve them.
- **Databases**: They pose the advantage of facilitating the processing of the data. Each log record is divided into separate fields, each of which is stored into the respective database table columns. They are usually the bottle-neck of the log management infrastructure, as the database cannot insert messages at the same speed that the log server can collect and process them. Concerning the *syslog* approach, some of the its implementations include features that enable the configuration of message queues on the *syslog* server, were the log data are temporarily stored, until the database becomes available again [24],[30]. In addition, most databases offer different storage engines some of which are suited more for writing than for reading transactions (e.g. the MySQL MyISAM storage engine [22]). Configuring clusters of databases is also a solution to alleviate high rates of transactions as well as scalability problems.

Functions that are performed during the log data life-cycle are [2]:

- **Log rotation**: It is the function of closing a log file and opening a new one, based on its size or time parameters. This keeps the log files in manageable size and enables the narrowing of the log analysis to specific files, as they are already separated based on the rotation criteria (one log file per day, for example).
- **Log retention**: Logs are archived on a regular bases, usually as part of the standard procedures.
- **Log preservation**: Logs are archived due to special interest, like forensic or incidence handling reasons.
- **Log compression** The log data are compressed to save storage space. Care must be taken to use lossless algorithms.
- **Log encryption**: The data are encrypted to protect their confidentiality. It is a recommended function when they are stored in an external device or transferred by any means. Proprietary encryption algorithms should be avoided in favor of publicly known and tested ones (e.g. AES).
- **Log reduction**: The log records that are of no interest are removed to reduce the occupied storage space.
- **Log conversion:** The format of the log records is modified, e.g. converting relational databases data to XML files.
- **Log normalization**: The representation of the fields of the log records is altered, to facilitate analysis and reporting.
- **Log file integrity checking:** The message digests of the log files are calculated to ensure the apprehension of an integrity violation.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Based on the organization's requirements (compliance, security policy, etc) and intended use of the log data (data mining, statistical analysis, etc), a decision is taken on the stages of data that will be used and the functions that will be performed, resulting in the documentation of the data life-cycle management process. An example process is show in Fig.4 [4].



Policy
Security
Log generation
Log retention
Store production data on-line for 15 mos (min) *
Log selection for long-term retention
Retain active archive data (subset of production data) near-line for approximately 5 years (min)**
Retain active archive data off-line after 5 years (min)**
Retire
Store back-up data near-line for 15 mos (min)

**Figure 4. Example Log Data Life-cycle Management Process [4]**

#### 3.7.2.2 Log Storage Sizing

Having defined the life-cycle management process, the log storage sizing is estimated where it is required. As the placement of the log collectors and the expected EPS at each one is already known from the previous section, the anticipated volume of data is now estimated. For example, for the second phase of the process of Fig.4, the log retention requires the production data to be stored for a 15-months period. Assuming a *syslog* collector is expected to receive an average rate of 1,000 EPS of estimated event size 1,024 bytes, this equals to 1,000 *1,024= 1,024,000 bytes per second. For the 15-months period the required storage is: 1,024,000bytes * 38,880,000 sec = 36.21 Terabytes.

The storage mechanism is also deducted by the life-cycle management process. The volume of the data and its intended use and access patterns, are the key factors for this decision. If the data are to be analyzed or otherwise processed from the organization, then the deployment of a database solution is preferred. If only a few analysts are going to access the database, then a near-line storage solution would be adequate.

#### 3.7.2.3 Security Considerations

The storage sub-tier needs to be adequately protected to assure the confidentiality, integrity and availability of the log data. An attacker may gain physical access to the log storage and affect the data or remotely exploit a vulnerability of the storage system. Modification or destruction of the log data is feasible if access is achieved with specific user privileges. To mitigate this risk, the physical protection of the storage system has to be addressed and the oper-

ΠΑΝΕΠΙΣΤΗΜΙΟ

ating systems and/or database servers have to be properly configured for access control. Encryption mechanisms such as hash functions, symmetric ciphers and digital signatures, are effective in protecting the confidentiality and the integrity of the data. Their availability can be protected through a back up process and hardware redundancy. For example, the log data can be written to a selected hard disk drive were even the system administrator can not access, allowing access only to the security personnel. A hash function algorithm could be applied to each backed up file to calculate its message digest and a symmetric encryption algorithm could be applied in continuance, to ensure its confidentiality. The systems security is not further addressed in the proposed methodology as it should be part of the organization's overall security policy and mechanisms.

The output of the log storage sub-tier section is the log data life-cycle management process, the log storage sizing, i.e. the volume of data that will be stored in each collector, the storage mechanism and the security measures that are necessary for the protection of the data at this sub-tier.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## 3.8 Time Synchronization

A critical issue in the implementation of a log management infrastructure is the synchronization of the logging equipment. Due to geographic dispersion multiple timezones may be used or some devices might use their internal clock for the timing functions. Log correlation requires accurate and uniform timing, to combine the log data from the various sources and identify the events of interest. An approach to the normalization of log data, that are generated using different time sources, is to add or subtract the difference between the timezones or the difference in the devices configuration. Obviously this is not a realistic approach as it requires a list of each device's time configuration and of course the overhead of performing the normalization through a manual or an automated process.

The recommended solution is the employment of time synchronization technologies such as the Network Time Protocol (NTP) [31] and the Precision Time Protocol (PTP) [32]. The PTP protocol is used for the precise synchronization of clocks in measurement and control systems that communicate using packet networks and are implemented using technologies like network communication, local computing and distributed objects. It supports accuracy in the range of sub-microsecond and requires minimal network and local clock computing resources. The devices are organized in a master-member hierarchy where all the members are synchronized with the master clock.

Both protocols are suitable for a log management infrastructure, though the NTP is preferred in the proposed methodology due to its wide and long usage on the Internet and the familiarity of the administrative personnel with it. The NTP is a widely used protocol to synchronize computer clocks among distributed time servers and clients. The current version is version 4 (backwards compatible with version 3), that achieves potential accuracy to the tens of microseconds. It uses the UDP port 123. An NTP implementation can operate in three modes, as a primary server, as a secondary server and as a client. An NTP network usually gets the time from an authoritative time source such as an atomic clock. The NTP server with the attached authoritative time source forms the stratum 1. It can be a public server or a private one and distributes the time to the stratum below. The concept of stratum defines how many hops away is a device from the authoritative time source. In WANs it usually achieves synchronization at 10 milliseconds and 1 millisecond at Local Area Networks (LAN). It avoids synchronization with possibly inaccurate machines, by not synchronizing to a machine that is not synchronized it self and by comparing the time reported from more than one machines [33].

### 3.8.1 Time Network Architecture

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

In [31] three modes of operation are defined for the NTP protocol, which are further elaborated in [33] as follows:

- **Client/server Mode:** A client sends a request to usually more that one servers and expects the answers, which are processed after received. When synchronization is provided to a large number of clients, the servers are usually organized in groups of three or more servers, operating in symmetric mode to achieve redundancy. Each of these servers is a client for thee or more servers of the lower stratum.
- **Symmetric Mode**: This mode of operation is indicated when NTP servers provide mutual back up. When a peer fails to communicate with the server of the lower stratum, time data is received through its own stratum and group peers. In this mode the server both obtains and supplies time. This mode is recommended for redundant time servers connected through diverse network paths.
- **Broadcast and/or Multicast Mode:** An NTP server can broadcast time in a subnet. This broadcast is restricted in the specific subnet as it is not routed from the routers. Accuracy and reliability are degraded, but eases the administration when a large number of clients is synchronized to a few time servers.

The above modes of operations can be combined to form NTP architectures as follows [33]:

- **Flat peer architecture:** All the NTP servers peer each other while some of them, geographically dispersed, synchronize to external systems.
- **Hierarchical structure**: The network routing hierarchy is copied and used for the NTP hierarchy. The top nodes (core servers) of the hierarchy, synchronize to external systems and each layer has a client/server relationship with its above layer (lower stratum). It is the recommended architecture as it provides scalability, stability and consistency. Fig. 5 depicts an NTP hierarchy and the concept of stratum. The one-way arrow indicates a client/server mode and the two-way arrow indicates a peer mode operation.
- **Star structure:** All the devices have a client/server relationship with a few time servers in the core.

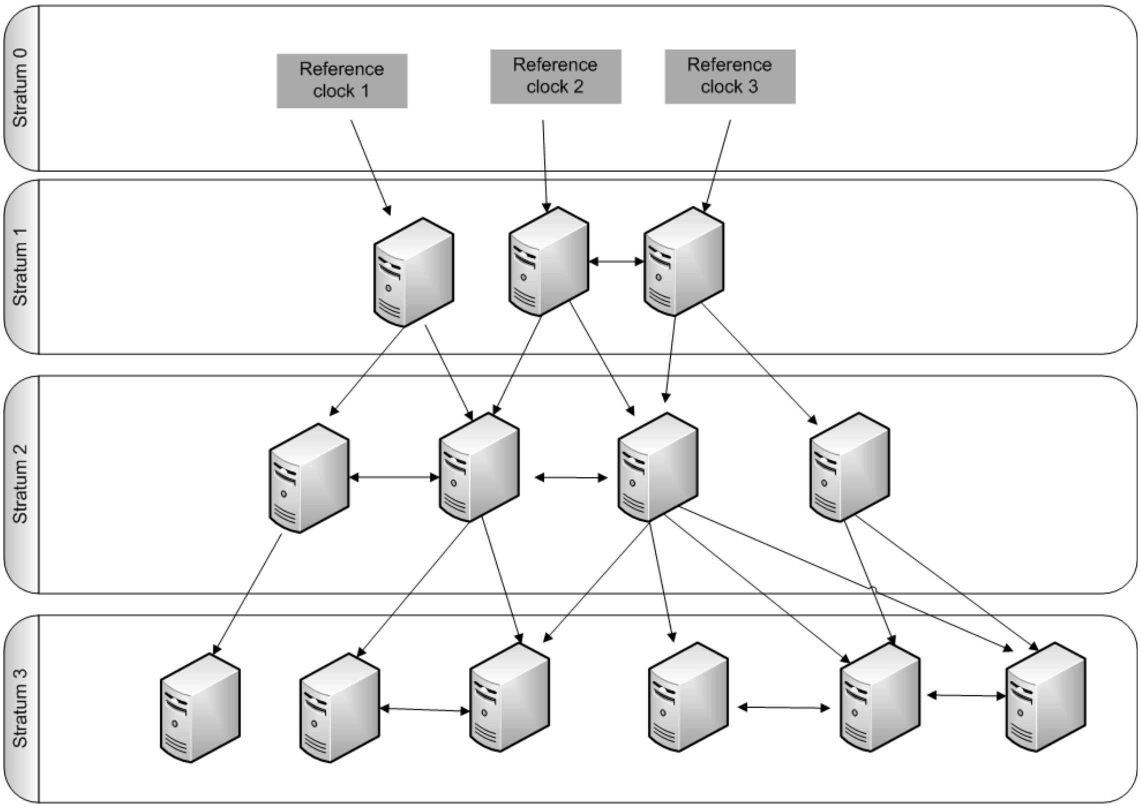


**Figure 5. NTP Hierarchy**

The proposed methodology uses again the SNA measurements already performed to derive the NTP servers locations. The closeness and the betweenness centrality are used in combination, to derive the NTP strata. In the case of a flat peer architecture the highest ranked nodes are connected to each other and some of them access a reference clock, while in the star architecture the highest ranked node, or a few highly ranked, will be accessed for time data from all the rest ones. If a hierarchical architecture is chosen, then the nodes need to be divided into groups to form the corresponding strata.

The nodes are sorted in descending order based on their closeness and betweenness centrality. The nodes that are highly ranked in both measurements will form stratum 1; selecting at least three nodes for this stratum is advised. Then the measurements are observed for sudden decreases in their values. The nodes above and below that point are mapped to two different groups. The first group forms stratum 2 and so on. It should be noticed that this process is not deterministic, though it is a means of identifying the adequate nodes. The actual selection is left to the analyst that may consider more factors.

For the NTP network design, an NTP server is recommended to access at least three lower stratum time servers. This is necessary for the NTP protocol to apply insanity check algorithms and assure the time accuracy in case of a server malfunction. In addition, operating in symmetric mode with peer servers provides the benefits of achieving back up and redundancy.

### 3.8.2 Security Considerations

Time synchronization is critical in security and especially in forensic investigations. Log data from various sources are collected and correlated and the sequence of events is reconstructed based on their time information. Time critical applications also need accurate time data in order to properly function. The NTP security model treats the time values as public; the aim is not to hide the data but to authenticate the sources. Security in time servers involves the client and server authentication and the integrity of the time values. Possible attacks on a time synchronization network are man-in-the-middle attacks as well as attacks on the availability. An attacker could transmit false time values, disrupt the protocol or exhaust the resources, to deny the service to legitimate applications. The packets could be modified and replayed by an attacker that has access on the network. An analysis of possible attacks, both active and passive, is provided in [34].

NTP supports the implementation of general purpose Access Control Lists (ACL). Thus, the server can be protected from unwanted clients. Another defense is the use of symmetric keys (starting from NTP version 3) to calculate message digests. When a packet is transmitted a Message Authentication Code (MAC) is concatenated to the NTP header and fields. When it is received the MAC is validated and the packet is accepted; otherwise it is ignored and an alarm is raised. From NTP version 4, X.509 certificates are supported with the use of the Autokey public key algorithm. The public and private keys of both the client and the server are used for the exchange of a secret key that is then used to calculate message digests [31],[34]. Employing symmetric or asymmetric encryption adds protection to the messages' authenticity; the confidentiality and the availability of the time information remain vulnerable. Enabling these features adds administrative and computational overhead when a large population of clients is served.

The output of this section is the NTP network architecture, the locations and the devices where the NTP servers will be installed, their separation into strata and the modes of operation of each device, as well as the servers they will reference in each mode of operation and the security settings.

## 3.9 Log Data Preprocessing

Log data preprocessing is a necessary and challenging part of the proposed methodology. The log data are generated from various types of devices and applications and still no common format is followed by vendors and developers. They often log the information they need for debugging and trouble shooting in a proprietary format (eXtensible Mark-up Language (XML),Comma Separated Value (CSV),etc). Some log files are not even meant to be human readable (binary files), but to be accessible using specific applications or to be used by specific applications. The time source that each originator uses can lead to inconsistently timestamped log records, thus impede the log correlation process. Time synchronization is addressed in the previous section of the methodology.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Preprocessing tasks of log data originating from different sources, include data transformation, data filtering based on the facility or the priority, data aggregation of frequently appearing records, as well as data reduction when not all of the available data are necessary for the analysis tasks. The log data can be processed on the log source, on transit or while in storage. In the *syslog* approach preprocessing tasks usually involve the modification of the time field and the aggregation of records. Instead of repeating the same event in a log file, it is recorded once along with the number of its occurrences. The log data can also be filtered and separated based on the facility, the priority or the type of the device. When a vendor specific approach is followed and an agent is installed on the logging devices, the log records are usually normalized prior to their transmission, thus performing the necessary preprocessing. An obvious drawback of this approach is the need for the agents to support all the necessary parsers, as well as the need to update those parsers every time the log originator alters something on its logging format.

The log files that will be managed by the log management infrastructure and their specific formats have already been recorded in a previous section, thus the fields that need to be modified can be easily identified and preprocessed. Depending on the log generation, collection and storage sizing and the estimated overhead, the components of the infrastructure that will perform the preprocessing tasks can be selected. This can be performed on the log generators, on the log collectors or in the storage mechanism.

The output of this step is the preprocessing tasks to be performed and the corresponding devices.

## 3.10 Scalability

Scalability of the log management infrastructure is an important characteristic and needs to be assessed. The needs of an organization may change rapidly due to compliance with a standard or due to an expansion of its activities. A new security policy or a security incident may drive the need for more accurate and voluminous log data collection. Scalability is considered in five dimensions [35] as follows:

- **Event volume**: What is evaluated is the ability of the infrastructure to handle a large increase in the number of events. This can be a long term change, due an expansion or change in the requirements or a temporary peak of the traffic, after a security or other unexpected event. The former could require the acquisition of additional equipment and the expansion of the log management infrastructure, while the later could be encountered with the reconfiguration of the present equipment.
- **Assets**: At this dimension the mechanism for compiling and updating the assets inventory is evaluated. The assets' modeling and the available attributes per asset type are considered, such as known vulnerabilities and patch history. These specific attributes, for example, would be used in the event of a malware infection, to include or exclude specific assets to the potential targets.
- **Locations**: Supporting logging in new geographic locations is a demanding task. The

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

log collection, transfer and storage are obvious implications, as well as the administration and maintenance of these remote devices. Agent installation, agent-less access, configuration and software maintenance issues are considered.

- **Capacity**: A demanding aspect of the infrastructure is the storage capacity. The storage mechanism is evaluated in terms of ease of expansion both geographically and in capacity. This is desired to be achievable under low economic cost and low administrative overhead.  Other functions such as compression or normalization may also have a significant impact on the computational resources, while solutions such as relational databases require the presence of a Database Administrator (DBA). The entire life-cycle management process is included in the evaluation.
- **Analysis:** The data volume and the correlation rules that can be processed in real-time are evaluated. An increase in the volume of log data or the correlation rules that need to me examined, may result in inefficient analysis, thus the infrastructure misses its targets. The ability to adequately scale the analysis tasks is evaluated.

## 3.11 Performance Measurement

The proposed methodology concludes with the development of performance measures to monitor certain activities and apply corrective actions if needed. The use of measures benefits the decision making and facilitates the achievement of the defined goals and objectives. The type of measures depends on the maturity of the program; a long running program is accompanied by refined processed and procedures, by documentation and historical data, as well as measurements/metrics collection mechanisms. The methods for collecting the measurement data should not be intrusive and due to their sensitivity, they should be properly managed. At this section the performance of the infrastructure is monitored after its implementation has been completed. The process of the implementation is proposed to be managed with the help of a project management methodology such as Project Management Institute (PMI) or Projects in Controlled Environments version 2 (PRINCE2).

In [13] the measures are categorized into implementation, effectiveness/efficiency and impact measures.

- **Implementation measures**: They are used to monitor the progress of a security control, a policy, a procedure, etc and it is usually measured as a percentage.
- **Effectiveness/Efficiency**: They are used to measure if an action is implemented correctly and results in the desired outcome.
- **Impact measures**: They measure the impact to the organization by a security control, policy, etc. Using such measures the value of the program is highlighted.

In order to develop the measures that will be used to measure the performance of the log management infrastructure the following process is followed [12], [13]:

- **Define a goal and objectives**: The goal of the measurement program is stated and it is broken down into objectives that guide to the achievement of the goal. The people

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

that will review the measurements/metrics and will handle the decision making participate in the goal and objectives definition.

- **Measures/metrics development and selection**: Through the development process, the measures that apply to an individual action can be defined, as well as the ones that apply to the whole program. In the former case the measures should be mapped to the specific action, while in the latter they should be mapped to the goal or to the objectives. To develop the measures a top-down or a bottom-up approach can be used. At the top-down approach, for each program objective the measures/metrics are generated. On the other hand, at the bottom-up approach, the measurements/metrics that are already available or easy to generate from the monitored tasks, are examined to result in whether they are adequate for the program objectives. Those that are suitable are selected and those that are not are rejected.
- **Targets and benchmarking**: The success of an action or objective is indicated by the achievement of the related measurement/metric targets. One approach is to set the target for a metric/measurement according to a baseline, while an alternative one, is not to set the target until the measurement/metric is run at least once. Benchmarking can provide comparative and meaningful results, through the comparison of the organization's performance to best practices and other organizations' practices,.
- **Data source and collection**: The data that will be used for the measurements/metrics are selected. The sources of these data and the way of accessing them is defined.
- **Program Review/Refinement:** The metrics/measurement program is reviewed to determine the accuracy of the measurements/metrics, the effort they require and the value they add to the organization.

Sample fields for measurements/metrics are listed in Table 3 [13].

**Table 3. Measurements/metrics Template Sample Fields**

| | |
|---|---|
| **Measure ID** | 01 |
| **Description** | Reconfigure firewalls on the servers |
| **Goal** | Update security policy |
| **Measure** | Percentage |
| **Type of measurement** | Implementation |
| **Formula** | # reconfigured / # installed |
| **Method of measurement** | Inspect the firewalls configuration files |
| **Target** | 100 % |
| **Frequency of measurement** | Weekly |
| **Responsible Parties** | IT administrators |
| **Data source** | Servers |
| **Reporting format** | Pie chart |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Having defined the measurements/metrics, an implementation process follows to ensure continuous use of the measurements/metrics and the improvement of the log management infrastructure performance [13]:

- **Data analysis:** Gap analysis is performed to compare the collected measurements/metrics with their targets, if defined. Causes of poor performance or areas to be improved are identified. Responsibilities to personnel can also be assigned.
- **Corrective actions**: If a gap is present the necessary corrective actions are identified. The range and the priority of these actions is determined, based on the risk mitigation goals.
- **Business cases development:** The results of the two preceding steps are used for the compilation of business cases to facilitate the allocation of resources and get support from the management.
- **Apply corrective actions**: The corrective actions are applied and iterative data are collected and analyzed to track the progress of the corrective actions, measure the improvement and identify were further improvement is needed.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Chapter 4

# Case Study

The methodology was applied on a real network, the GRNET network. As stated on its web site *"GRNET provides high-speed advanced networking services to the Hellenic academic, research and education community, serving all institutions, universities, research centers and schools through the Panhellenic School Network"* [36]. The necessary data for the case study are publicly available on the web site. Specifically, the network topology, the network devices and the bandwidth of the links is provided. Unfortunately these data are not enough for the full application of the proposed methodology, so wherever supplementary data are required they are assumed in order to demonstrate its application.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## 4.1 Capturing Requirements

The GRNET provides help-desk, host-master, networking, computational, VoIP and other services to academic institutions and research centers. It decided to implement a log management infrastructure to monitor and respond to security related events.

The GRNET network is managed by GRNET S.A., a limited company owned b the Hellenic State. In order to be compliant with the Greek law, enforce the use the security policy and to be able to respond to security incidents, the log management infrastructure has to:

- Log individual user access to protected information.
- Log the execution of privileged actions.
- Log the invalid access attempts.
- Log the security system events.
- Log suspicious network activity.
- Protect the confidentiality, integrity and availability of the logged data.
- Retain the log files for at least on year.
- Be scalable.
- Be low budget.

## 4.2 Assets Inventory

For the creation of the assets inventory the network topology and the data available on the web site were used. The network topology diagram, version of 2013-07-29, depicts the Layer 2 and Layer 3 network devices, Appendix I. It contains the routers and switches as well as the links connecting them. All the routers were included for the case study and from the switches only the ones directly connected to a router. In the cases where two devices were connected through multiple ports, only one of those connections was considered to simplify the data collection process, as it was performed manually. The result was a list of 17 routers and 51 switches. For the needs of the case study a server machine is assumed connected to each switch. At each server machine Debian 7 64-bit is installed, running a web and a file server. The web server is Apache2 and Samba is used for file sharing and authentication. Under this an assumption 51 servers are added to the inventory. Table 4 lists the assets of GRNET S.A.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

**Table 4. GRNET Assets Inventory**

| Asset | Category | Quantity | Software | Other |
|---|---|---|---|---|
| Cisco CAT3750 | Switch | 24 devices | Cisco IOS | 24 ports of 10/100/1000 Mbps |
| Extreme X450 | Switch | 17 devices | ExtremeXOS | 24 ports of 10/100/1000 Mbps |
| Juniper EX4200 | Switch | 10 devices | Junos | 24 ports of 10/100/1000 Mbps |
| Cisco GSR 12416 | Router | 15 devices | Cisco IOS | |
| Juniper MX960 | Router | 2 devices | Junos | |
| HP ProLiant ML110 G7 Server | Server | 51 devices | Debian 7.1 64bit<br>Samba<br>Apache2 | 1 Gbps NIC |

## 4.3 Network Topology and Traffic Patterns

The L2/3 network devices recorded in the assets inventory are connected through 75 connections. For each of these connections the nominal, the maximum and the average bandwidth recorder during the time period of one year (from 2012-09-21 to 2013-09-21), are used for the case study. The data were recorded from the GRNET network monitor tools available on the web site. The bandwidth data are provided for each Layer 2/3 device, for each port, for each Virtual Local Area Network (VLAN) and for user defined time periods. They had to be collected manually, since no reporting tools were available on the web site and this is the reason for which some switches and links, between the devices, were ignored. An assumption is made in this section for the missing data, it is assumed the in each LAN the server is connected to the switch with 100 Mpbs network links. The network connections between them and the bandwidth details (nominal, maximum and average bandwidth) are listed in Appendix II. The network topology diagram is available from the web site, Appendix I.

## 4.4 Choose What to Log

The TDBUMO process is followed to result into the log files that will be used to meet the log management infrastructure requirements.

**Top Down:** The devices of the inventory are grouped creating the following tree structure, Fig.6. The leaves of the tree list all the specific types of logs that can be generated from the log sources.

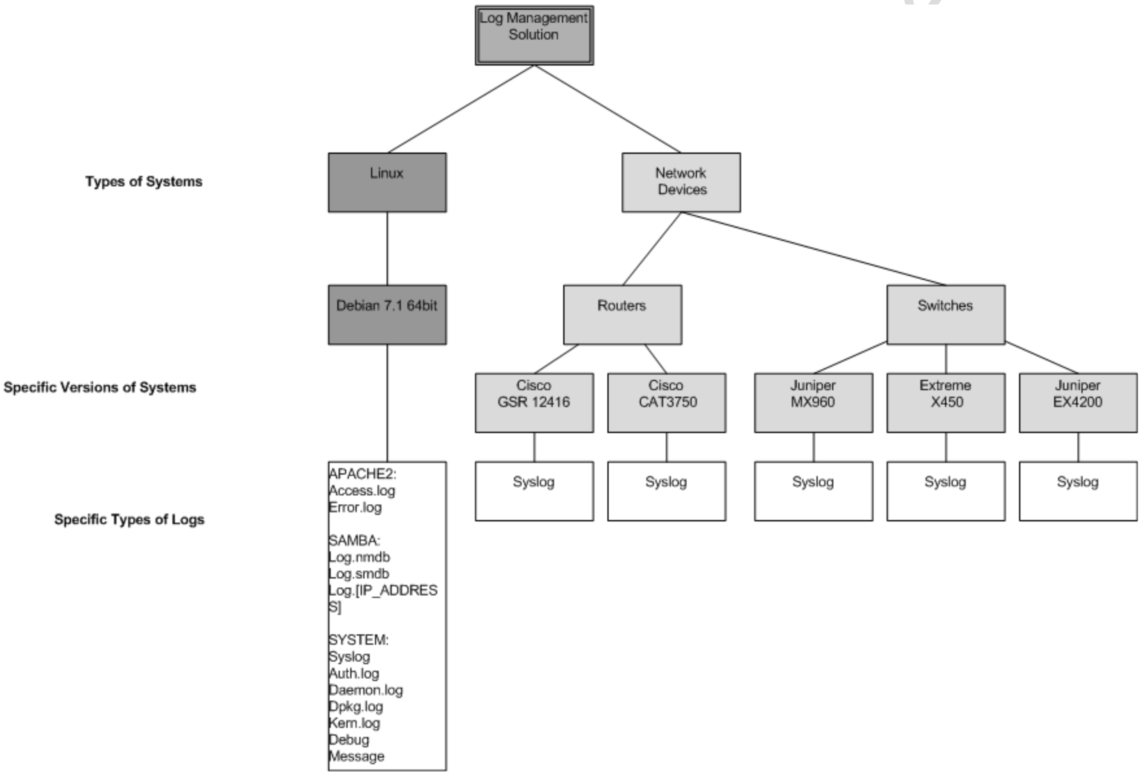


**Figure 6. TDBUMO Top Down Output**

**Bottom Up:** The specific log files identified in the previous step are further elaborated and their details are documented. Debian distributions use the *rsyslogd* daemon [37] for their logging operations, implementing the *syslog* protocol [19]. Routers and switches also support the use of the *syslog* protocol and are included in the analysis, [38],[39],[40]. The results of the analysis are listed in Appendix III.

**Middle-Out:** The output of the previous step is the list of the specific log files each system can generate and their corresponding details concerning the logging level, the log format, etc. It is used in combination with the output of the log requirements section to create a matrix, mapping the log files to each requirement, Appendix IV. This matrix of devices, log files, log levels and formats, is the output of the TDBUMU process and it is the answer to what needs to logged to meet the requirements of the log management infrastructure.

## 4.5 Choosing the Infrastructure Architecture

The network topology and traffic patterns section has already provided the network topology diagram of the GRNET S.A. with the corresponding bandwidths (nominal, maximum, average) of the traffic that flows through it. It is a WAN that extends throughout the country connecting academic and research institutions. Observing its topology, Appendix I, it is obvious that it is composed of smaller Metropolitan Area Networks (MAN).

In literature the centrally-managed infrastructure is considered to be the only realistic approach, to apply log management best practices and benefit from the analysis of the whole volume of log data. Due to the size of the network, the number of log sources and the anticipated log data volume, the distributed architecture poses the benefit of limiting each infrastructure to a specific scope, thus decreasing the demands in resources. A hybrid architecture, the combination of the centrally-managed and distributed architectures is chosen as the best solution for the GRNET.SA log management infrastructure.

Another issue that is examined in this step is whether the log management infrastructure should be implemented on a separate or on the normal network of the organization. The size of the network and the number of log sources leaves no choice but to use the normal network. Implementing a separate physical network would significantly increase the cost of the project and the time that is required for its deployment. Performance and security concerns endorse the implementation of a separate logging network, at least for the critical devices and parts of the network. This separation can be logical using relevant networking technologies, such as VLANs and VPNs. This solution increases the administrative overhead but alleviates the security and performance disadvantages of using the organization's normal network.

## 4.6 Log Generation Tier

The TDBUMU process resulted in the log devices and the specific log files that will be collected from the log management infrastructure, Appendix IV. All these devices (servers, routers and switches) support the *syslog* protocol, so it will be used for the access and transmission of their log data. The devices that have the ability to maintain local copies of the log files i.e. the Debian servers, will enable the feature of locally storage. A server, apart from participating to the log management infrastructure is still managed by a local systems administrator. He/she needs access to logs of sufficient detail to successfully perform maintenance and troubleshooting, thus the debug log level will be enabled and stored locally in raw files, while the Informational, Info and Level 1 for the system, Apache2 and Samba log files respectively, will be sent via the *syslog* protocol to the collectors. The routers and switches do not have the ability to log locally sufficient amounts of log data. The Warning level will be sent to the collectors using the *syslog* protocol.

Moving to the log generation sizing, no historical data are available from the GRNET.SA network, thus no patterns can be identified. Concerning the network devices a 10 EPS and 5 EPS rate is assumed for routers and switches, respectively and the event size is assumed to be

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

1,024 bytes. For the Debian servers 100 EPS are assumed, with 150 bytes of average event size. The log generation sizing is summarized in Table 5.

**Table 5. Log Generation Sizing**

| Device Type | Average EPS | Corresponding Bandwidth |
|---|---|---|
| Router | 10 | 0.0819 Mbps |
| Switch | 5 | 0.0410 Mbps |
| Server | 100 | 0.2100 Mbps |

Both the servers and the network devices have to be physically protected from unauthorized access. Details on the GRNET.SA infrastructure and security policy are not available but for the purposes of the case study it is assumed that they are properly protected and placed in computer rooms, where their health status is monitored by the local system administrators to assure their availability. In both the servers and the network devices configuration privileges are assigned to specific personnel, the users are allowed only to view the log files, not to delete or modify them. Regular back ups are performed and they are encrypted and digitally signed using asymmetric key encryption.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## 4.7 Log Collection and Storage Tier

### 4.7.1 Log Collection

#### 4.7.1.1 Placement and Roles

At this step the role and the location of each *rsyslog* server is deducted using SNA measurements. The social network is constructed using the already available network topology diagram of the GRNET.SA. Each device depicted in the topology is represented as a node in the social network and each connection among them is used to form a link. With these links four different relations are constructed, one with binary values (a link between two devices either exists or not), one with the nominal bandwidth as the link value, one with the average and one with the maximum. These data are available from the output of previous sections and its easy to be inserted in an SNA software tool. The analysis and the visualizations in this case study were performed using CASOS ORA version 2.3.6, a software tool developed by the Carnegie Mellon University [41].

The resulting social network is *G=(V,E)* with *|V|=68* and *|E|=75*. Thus the network is composed of 68 nodes that form 75 links between them. The nodes have 4 attributes i.e. structural properties. A sample set of nodes is listed in Table 6, the full data set of nodes is included as Appendix IX. The *Node ID* is the unique ID of each node used by the software, the *Node Title* is an attribute, a meaningful name for the nodes (the names that where used are the ones given by GRNET.SA in the network topology diagram). The *Device Model* is the model of the device represented by the node, the *Device Type* specifies the category and the *Vendor* indicates the vendor of the device. The number and the type of attributes depends on the specific needs of the analyst and are useful in the network visualizations.

**Table 6. Sample Node Set**

| Node ID | Node Title | Device Model | Device Type | Vendor |
|---|---|---|---|---|
| R-1 | XAN2 | Cisco GSR12416 | Router | Cisco |
| R-2 | THES2 | Cisco GSR12416 | Router | Cisco |
| R-3 | IOAN2 | Cisco GSR12416 | Router | Cisco |
| SW-1 | IONASW1 | Extreme X450 | Switch | Extreme |
| SW-2 | TEIEP@ioaSW | Extreme X450 | Switch | Extreme |
| SW-3 | UOISW | EX4200 | Switch | Juniper |

Each relation is represented by a *|V| x |V|* adjacency matrix. A sample of the relation that uses the average bandwidth as the links value is shown in Table 7. The matrices are populated using the data of Appendix II. The cells that contain a value indicate a link between these devices; the node with *Node Title* XAN2 is connected with THES2. Since the relation depicts the average bandwidth, the average bandwidth recorded between those to devices is 118.77 Mpbs. The cells with zero value indicate that no connections exists between these devices. If the value of the link was of no interest the presence of a link would be denoted with a Boolean (1 or 0) value.

**Table 7. Sample Relation Adjacency Matrix**

| | XAN2 | THES2 | IOAN2 | LAR2 | PATR2 | YPEPTH1 |
|---|---|---|---|---|---|---|
| **XAN2** | 0.0 | 118.77 | 0.0 | 0.0 | 0.0 | 0.0 |
| **THES2** | 0.0 | 0.0 | 50.52 | 328.58 | 0.0 | 0.0 |
| **IOAN2** | 0.0 | 0.0 | 0.0 | 0.0 | 305.16 | 0.0 |
| **LAR2** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **PATR2** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |
| **YPEPTH1** | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 |

After the matrix is complete ORA is used to visualize the network. Visualizations provide the analyst a birds-eye view of the network and the data, Fig.7. For each node the *Node Title* attribute is displayed and for each link the nominal bandwidth. The detail and the type of information that is included in the visualization depends on the software tool.  Adjusting this

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

visualization by simply using different shapes for the nodes based on the *Device Type* attribute, Fig.8, results in an informative visualization of the actual network. The routers are the circle shaped nodes and the switches are the square shaped nodes.

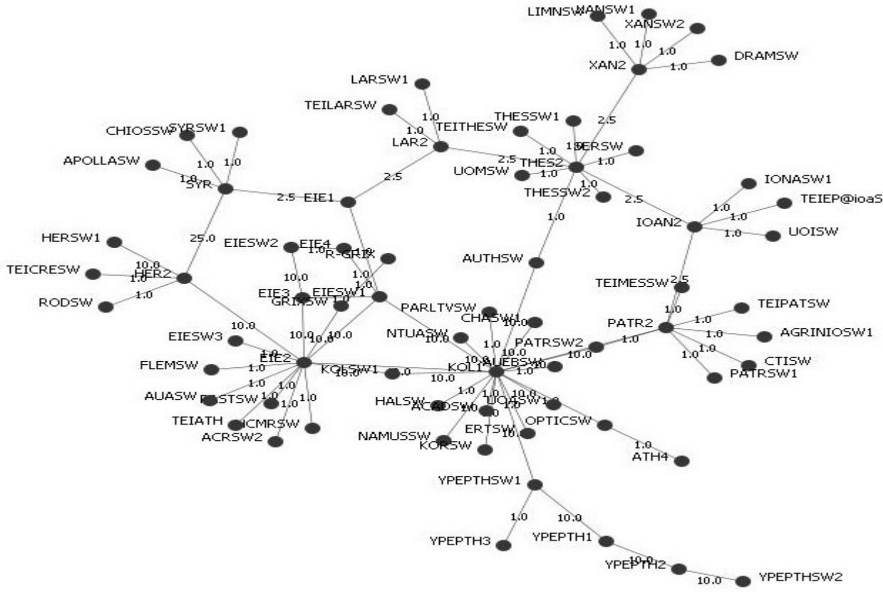


**Figure 7. GRNET.SA Social Network Visualization**

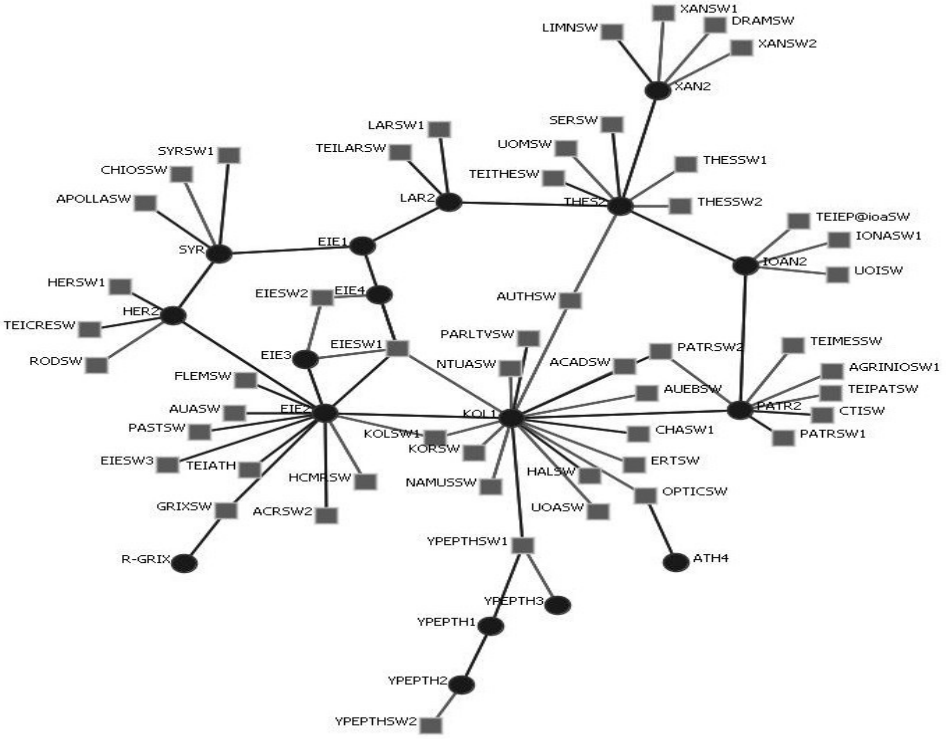


**Figure 8. GRNET.SA Social Network Visualization-Device Type**

In continuance, the total degree centrality of the nodes is calculated. The ten highest valued nodes are listed in Table 8 (full data set measurements in Appendix V) and visualized in Fig.9; the higher the value the biggest the size of the node in the visualization. These are the nodes that have the most direct links. This is interpreted in that placing a collector to the location of one of these nodes, the collector will be one hop away from many originators. KOL1 and EIE2 are the ones with the highest value, and then some clustering occurs, nodes 3 and 4, as well as nodes 5 to 9.

**Table 8. Ten Highest Ranked Nodes – Total Degree Centrality**

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.269 | 18.000 |
| 2 | EIE2 | 0.194 | 13.000 |
| 3 | THES2 | 0.134 | 9.000 |
| 4 | PATR2 | 0.119 | 8.000 |
| 5 | XAN2 | 0.075 | 5.000 |
| 6 | IOAN2 | 0.075 | 5.000 |
| 7 | SYR | 0.075 | 5.000 |
| 8 | HER2 | 0.075 | 5.000 |
| 9 | EIESW1 | 0.075 | 5.000 |
| 10 | LAR2 | 0.060 | 4.000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

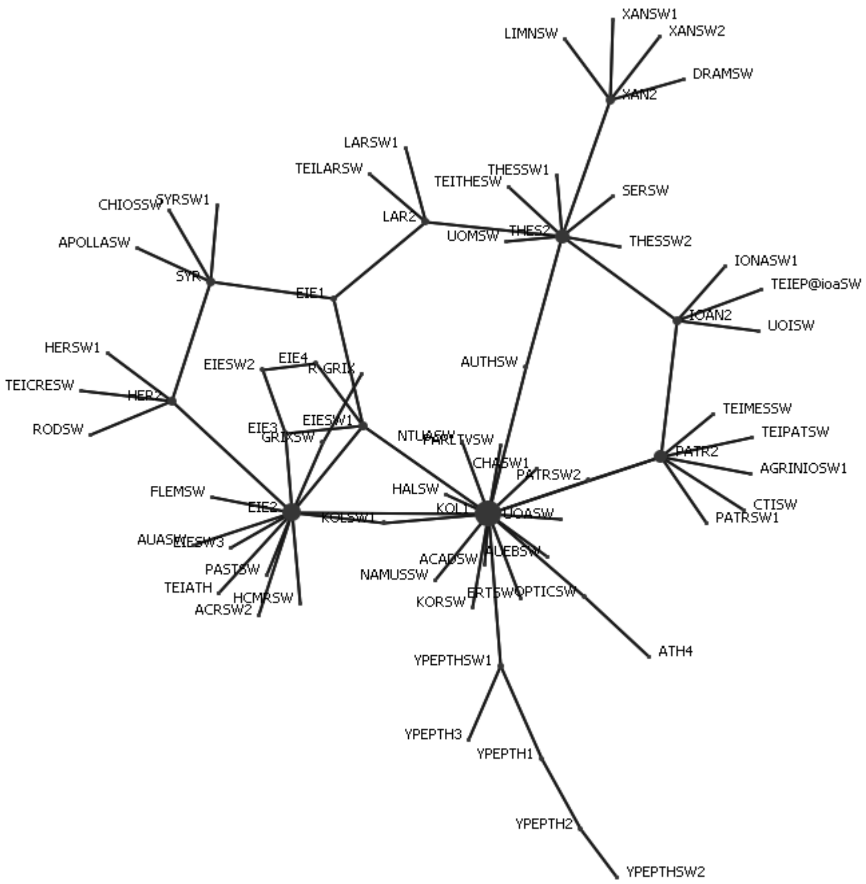


**Figure 9. GRNET.SA Social Network Visualization- Total Degree Centrality**

The closeness centrality is then calculated to identify nodes that are close to other nodes, measured in hops. The ten highest valued nodes are listed in Table 9 (full data set measurements in Appendix VI) and visualized in Fig.10 (again the size of the node is proportional to the value of the measurement). The nodes have almost the same value, that means that most of the nodes are easily reached from other nodes in the social network. Nodes KOL1 and EIE2 are the highest valued ones making them a good location for the placement of the collectors as they are close to all other log originators of the network.

**Table 9. Ten Highest Ranked Nodes – Closeness Centrality**

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.438 | 0.007 |
| 2 | EIE2 | 0.385 | 0.006 |
| 3 | EIESW1 | 0.366 | 0.005 |
| 4 | PATR2 | 0.360 | 0.005 |
| 5 | AUTHSW | 0.358 | 0.005 |

| | | | |
|---|---|---|---|
| 6 | KOLSW1 | 0.337 | 0.005 |
| 7 | THES2 | 0.330 | 0.005 |
| 8 | EIE1 | 0.327 | 0.005 |
| 9 | PATRSW2 | 0.321 | 0.005 |
| 10 | YPEPTHSW1 | 0.318 | 0.005 |

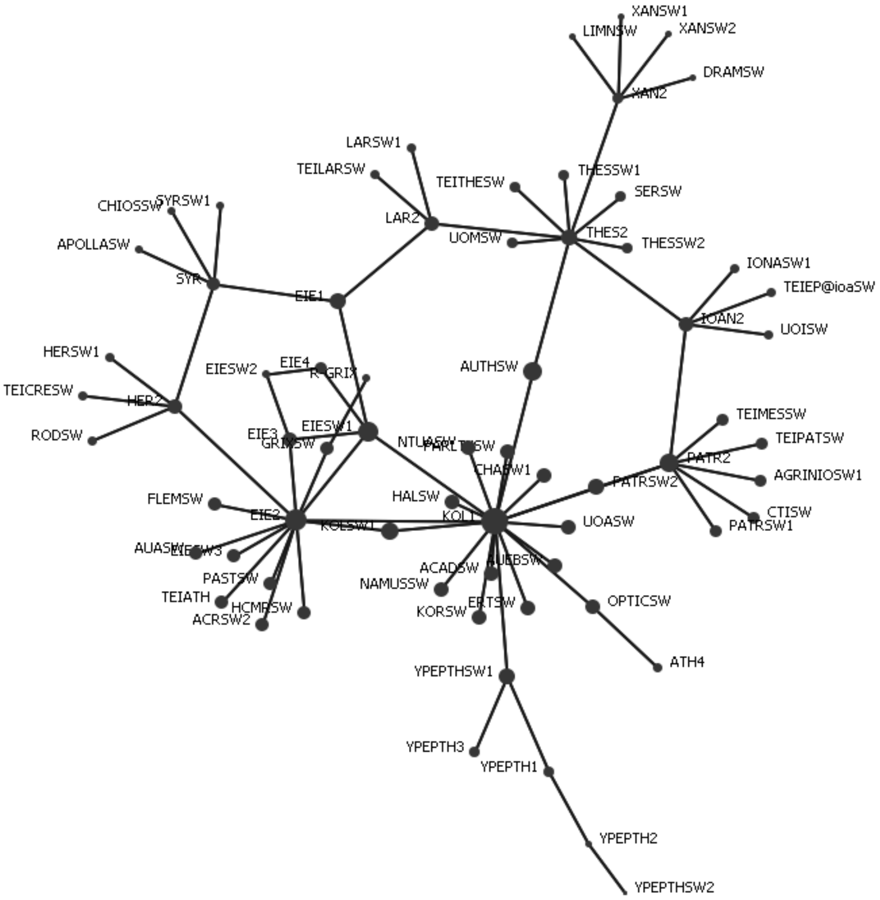


**Figure 10. GRNET.SA Social Network Visualization- Closeness Centrality**

The eigenvector centrality is measured then. The ten highest valued nodes are listed in Table 10 (full data set measurements in Appendix VII) and visualized in Fig.11 (the size of the node is proportional to the value of the measurement). A node is high valued when it is connected with also highly connected nodes. The top ranked is node KOL1, which indicates that it is location is a good point for the central collection of the logs. Based on this measurements, the KOL1 node location could be selected as the central point of the log management infrastructure. In Fig.11, it easily identifiable the importance of this node, as it is connected with nodes that are them selves connected to many other nodes.

**Table 10. Ten Highest Ranked Nodes – Eigenvector Centrality**

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.823 | 0.582 |
| 2 | EIE2 | 0.571 | 0.404 |
| 3 | EIESW1 | 0.367 | 0.260 |
| 4 | PATR2 | 0.299 | 0.211 |
| 5 | KOLSW1 | 0.287 | 0.203 |
| 6 | PATRSW2 | 0.231 | 0.163 |
| 7 | EIE3 | 0.206 | 0.145 |
| 8 | AUTHSW | 0.188 | 0.133 |
| 9 | YPEPTHSW1 | 0.185 | 0.131 |
| 10 | OPTICSW | 0.177 | 0.125 |

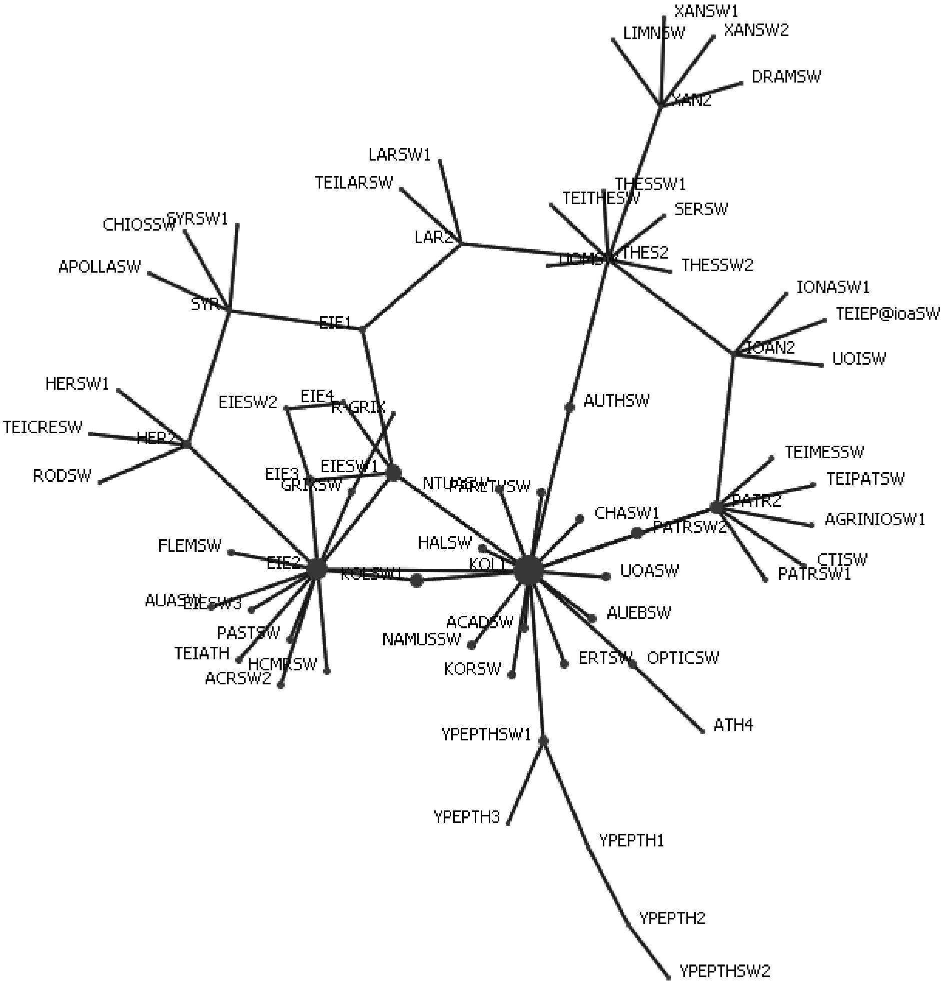


**Figure 11. GRNET.SA Social Network Visualization-Eigenvector Centrality**

The measurements continue with the betweenness centrality. The ten highest valued nodes are listed in Table 11 (full data set measurements in Appendix VIII) and visualized in Fig.12 (the size of the node is proportional to the value of the measurement). Nodes with high betweenness are important as many paths that connect pairs of nodes are pass through them, thus their removal i.e. a device malfunction or failure can disrupt the network communication. This measurement is used in continuance in combination with total degree centrality to identify the boundary spanners.

**Table 11. Ten Highest Ranked Nodes – Betweenness Centrality**

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.645 | 1427.100 |
| 2 | EIE2 | 0.368 | 813.033 |
| 3 | THES2 | 0.314 | 693.833 |
| 4 | PATR2 | 0.216 | 478.500 |
| 5 | AUTHSW | 0.181 | 400.433 |
| 6 | HER2 | 0.133 | 293.667 |
| 7 | IOAN2 | 0.129 | 285.500 |
| 8 | EIESW1 | 0.125 | 276.133 |
| 9 | EIE1 | 0.119 | 262.400 |
| 10 | XAN2 | 0.117 | 258.000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

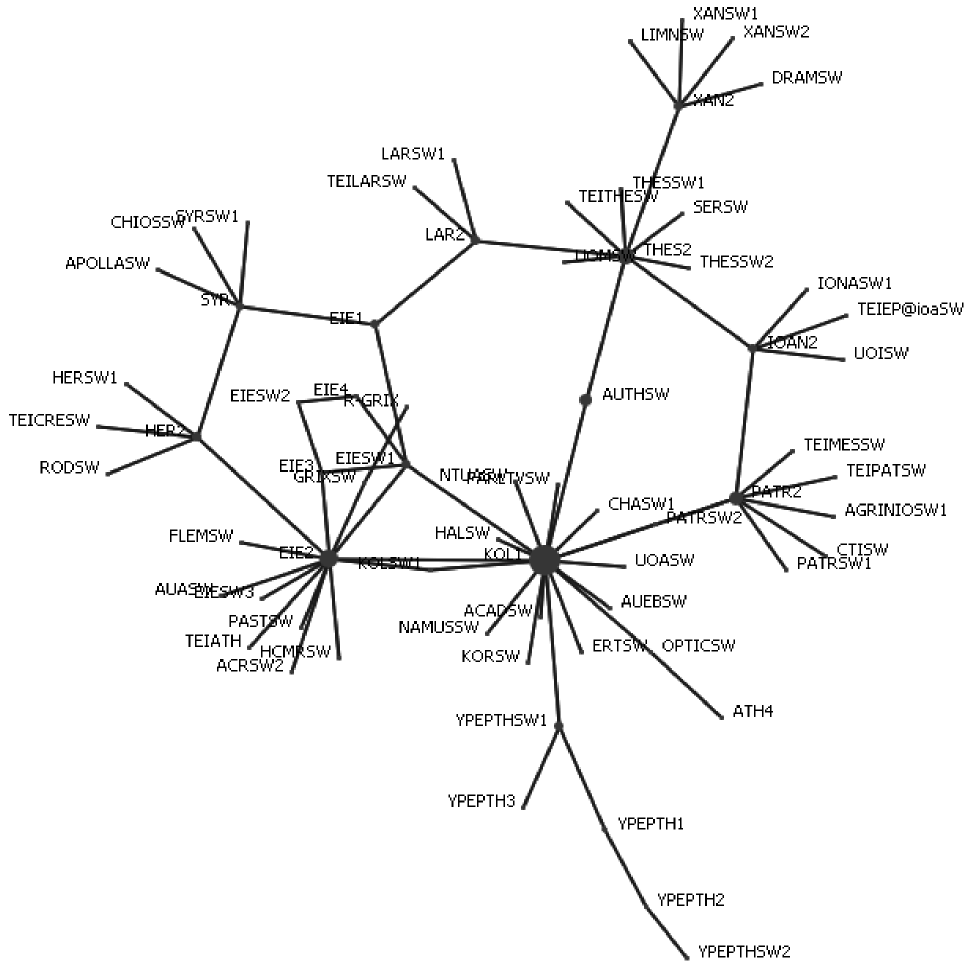


**Figure 12. GRNET.SA Social Network Visualization- Betweenness Centrality**

Having performed the above measurements the importance of each node, in terms of connections and distance, has been identified. It is now crucial to identify the nodes that have the structural property of being gatekeepers (or boundary spanners). A boundary spanner, is a node that if removed a new component is created from the social network [25]. The ten highest valued nodes are listed in Table 12 and visualized in Fig.13. The node that is most likely to disrupt the social network is node AUTHSW. This node has high betweenness centrality and low total degree centrality, this is interpreted in that many communication paths between pairs of nodes pass through this node, though it is connected with only few nodes. As shown in Fig.13, the removal of one of the boundary spanners will fragment the social network to more components. For a communications network this is translated into inability to communicate from the one subnet to the other. For the log management infrastructure this would result in loss of log data from a part of the monitored network. The placement of the collectors should consider such a scenario. If, for example, node THES2 got removed, the subnet containing XAN2, LIMNSW, XANSW1, XANSW2 and DRAMSW, would be separated from the rest of the network. If the disruption is temporary, the placement of a cache server or enabling the local storage of a *rsyslog* server placed at XAN2 would alleviate the problem and no log data would be lost, as they would be transmitted when the communication with the rest of the network would be re-established.

**Table 12. Boundary Spanners**

| Node Title | Boundary Spanner |
|---|---|
| EIE2 | 1.0000 |
| GRIXSW | 1.0000 |
| HER2 | 1.0000 |
| IOAN2 | 1.0000 |
| KOL1 | 1.0000 |
| LAR2 | 1.0000 |
| OPTICSW | 1.0000 |
| PATR2 | 1.0000 |
| SYR | 1.0000 |
| THES2 | 1.0000 |
| XAN2 | 1.0000 |
| YPEPTH1 | 1.0000 |
| YPEPTH2 | 1.0000 |
| YPEPTHSW1 | 1.0000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

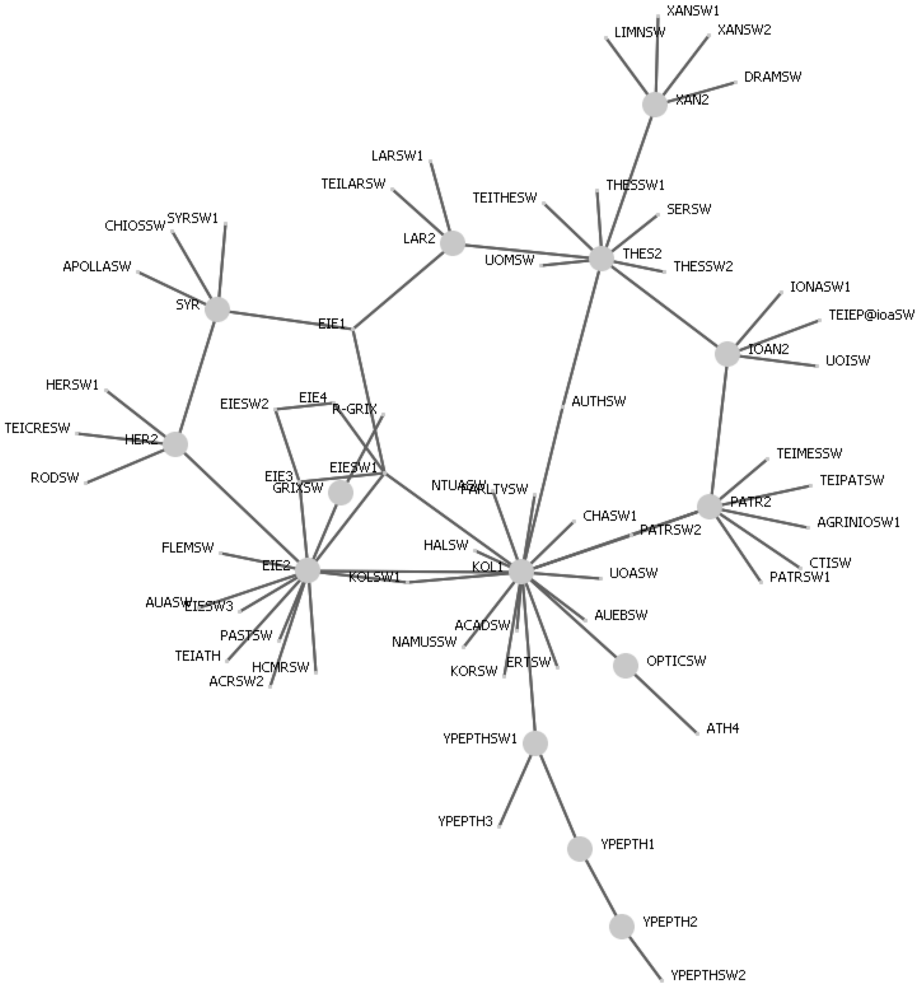


**Figure 13. GRNET.SA Social Network Visualization- Boundary Spanners**

The measurements performed are combined to deduct the role and the location of each *rsyslog* server. In Table 13, the ten top ranked nodes are compared based on their ranking, not the value of the measurements, across the measurements performed (full list of node in Appendix X). The results of this comparison is also visualized in Fig.14. This is indicative of the importance of specific nodes and as a consequence, it indicates the nodes that most satisfy the placement criteria. Nodes KOL1 and EIE2 are high valued in almost 90% of the measurements, EIESW1 in about 60% and THES2 at about 30%. These are the nodes that are closer to the log sources, directly connected to many other log sources and connected with devices that are highly connected too.

**Table 13. Recurring Top Ranked Nodes**

| Rank | Betweenness centrality | Closeness centrality | Eigenvector centrality | Total degree centrality |
|---|---|---|---|---|
| 1 | KOL1 | KOL1 | KOL1 | KOL1 |
| 2 | EIE2 | EIE2 | EIE2 | EIE2 |
| 3 | THES2 | EIESW1 | EIESW1 | THES2 |
| 4 | PATR2 | PATR2 | PATR2 | PATR2 |
| 5 | AUTHSW | AUTHSW | KOLSW1 | XAN2 |
| 6 | HER2 | KOLSW1 | PATRSW2 | IOAN2 |
| 7 | IOAN2 | THES2 | EIE3 | SYR |
| 8 | EIESW1 | EIE1 | AUTHSW | HER2 |
| 9 | EIE1 | PATRSW2 | YPEPTHSW1 | EIESW1 |
| 10 | XAN2 | YPEPTHSW1 | OPTICSW | LAR2 |

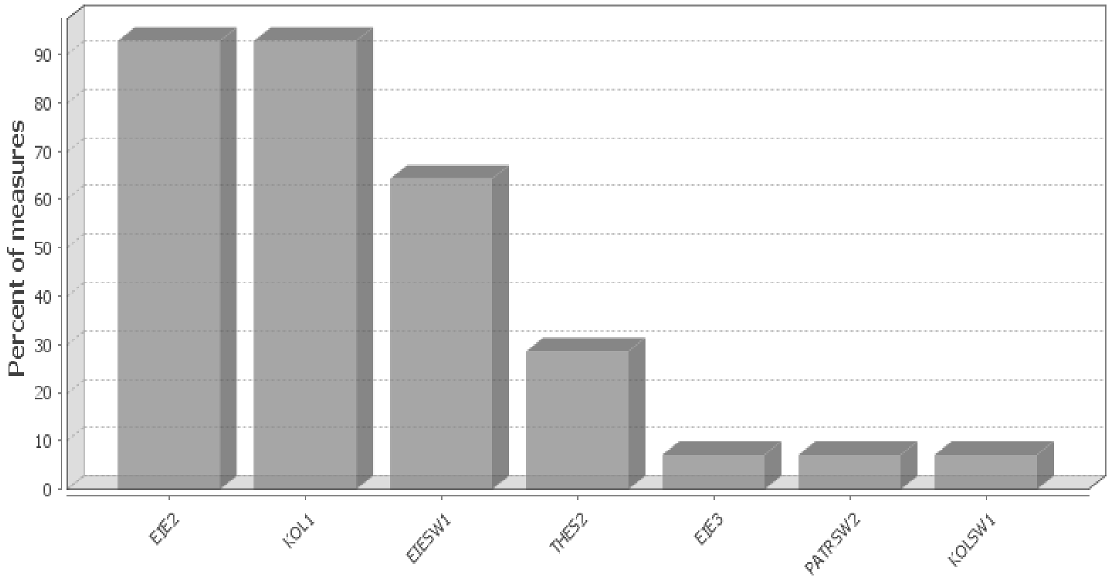


**Figure 14. Recurring Top Ranked Nodes**

Having identified the important nodes as well as the nodes with a special position (i.e. the boundary spanners), the methodology continues with the selection of the node locations where the *rsyslog* collectors will be placed, as well as with the definition of additional server roles such as cache servers. So far the social network analysis has reveled the structural properties of the network, the connection patterns and has identified the nodes that are important according to the placement criteria. The final decision on the placement of the servers and the roles assigned to them, is left for the analyst, as additional factors may need to be considered.

A distributed architecture has already been chosen for the log management infrastructure. This is interpreted in that the log originators will forward their log data to local *rsyslog* collectors, which will in turn forward them to fewer more central collectors, resulting in a few distributed ones. This follows a hierarchical structure that results to a final collection point, the root node. Due to the architecture the structure stops at the most central collectors i.e. to Layer 1 of the tree. The root node is not specified as the log data will be collected and managed at Layer 1, in a few collectors. This implementation is solution independent, as in the case of the implementation of a log management or SIEM product, the log data are already available for use. A SIEM, for example, does not need to correlate everything that is collected, it could filter the already collected data keeping only the ones required for its aims. If the central point of the infrastructure had to be transferred, then the log collection would not be interrupted and as soon as the new point would be ready only the analytical tool would have to be installed and connected to the storage points. Of course the distributed log data can always be forwarded to a single collection point, if desired.

Based on the measurements the nodes can be divided into layers as follows:

- **Layer 1**: KOL1, EIE2, THES2, PATR2, EIESW1
- **Layer 2:** HER2, SYR, LAR2, XAN2, IOAN2

This selection is also verified by the visualization of Fig.15. Though not displayed, the x-axis is the closeness centrality of the nodes increasing to the right, and the y-axis is the total degree centrality of the nodes, increasing upwards. The nodes with high total degree centrality are placed to the top of the figure and the ones with high closeness centrality are placed more to the right. The circle shaped nodes are the nodes that will implement the Layer 1 of the log management infrastructure, the square shaped ones the Layer 2 and the triangle shaped ones the Layer 3. The EIESW1 node is not top-ranked in total degree and closeness centrality though high valued, but it is high in betweenness and eigenvector centrality. From Fig.15, EIESW1 seams to fit better to Layer 2, but because it is connected to highly connected nodes (eigenvector centrality) and many nodes physically connect through it (betweenness centrality), it is included into Layer 1. The placement of the nodes on the social network is shown in Fig.16.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

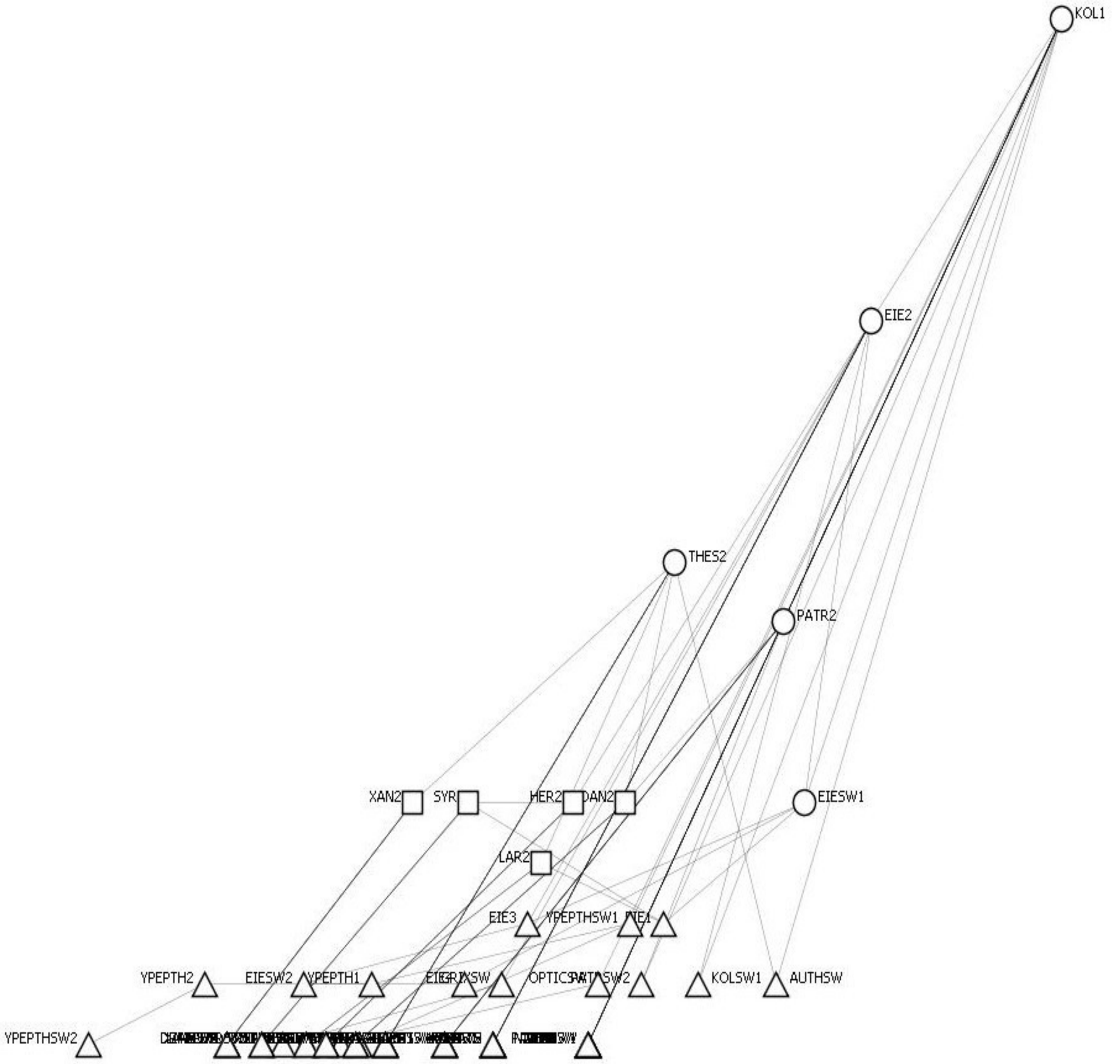


**Figure 15. Closeness to Total Degree Centrality Layout**

ΠΑΝΕ

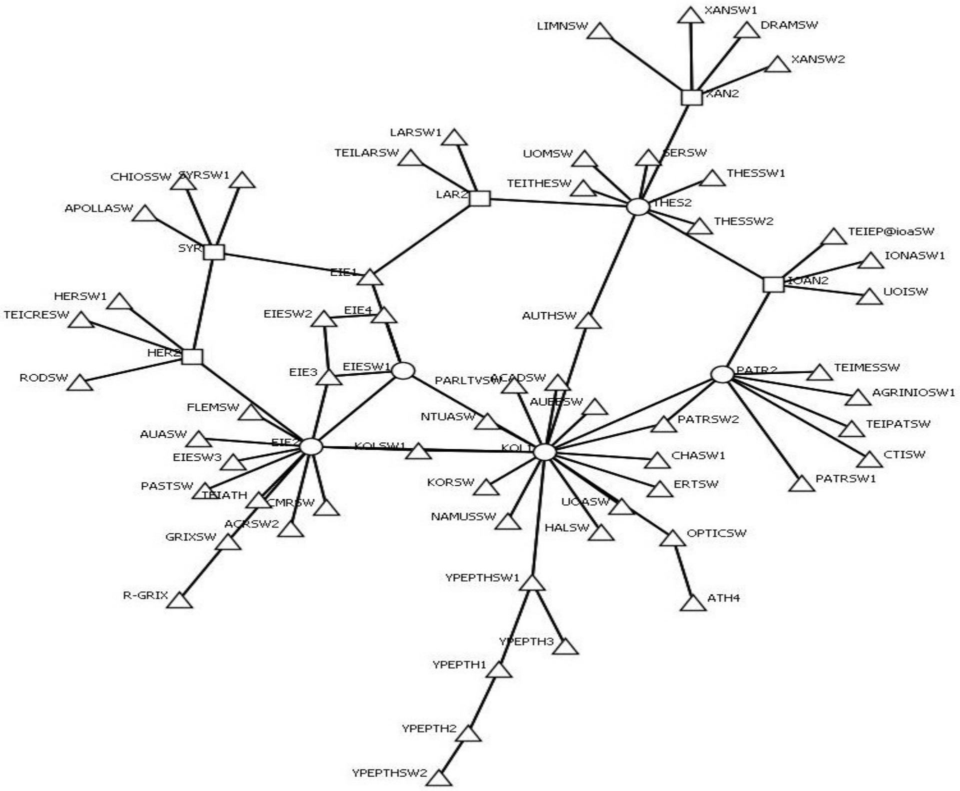


**Figure 16. Placement of the Layers of the Log Management Infrastructure**

So far the proposed methodology has resulted in the placement of the *rsyslog* servers assigned with the role of the collector. The Layer 2 collectors forward their log data to the Layer 1 collectors, from were they can be exploited for analysis, correlation or other tasks. The methodology continues with the assignment of nodes to groups i.e. log originators to log collectors. Though the assignment can be performed manually observing Fig.16, the Newman algorithm is used to facilitate and justify the process. Table 14 and Fig.17 depict the groups that the algorithm output for the GRNET.SA social network, while Fig.18 depicts these groups on the social network. The algorithm automatically divided the nodes to seven groups. Combining these groups with the separation of the nodes into layers, what is observable is that the selection of *rsyslog* server locations and roles based on the centrality measures, lead to the placement of at least one to each group. In groups such as Group-5, where two Layer 2 servers have been placed and Group-1, where two Layer 1 servers have been placed, the assignment of originators to collectors is performed considering the network topology, Fig.8. As a result, the collector placed in SYR will be assigned the SYRSW1, CHIOSSW, APOLLASW nodes, while the HER2 collector will be assigned the HERSW1, TEICRESW, RODSW nodes. These Layer 2 collectors will forward their log data to the Layer 1 collector in EIESW1. At Group-6, no server has been placed, thus a Layer 2 *rsyslog* collector will be placed in the location of YPEPTSW1.

**Table 14. Node Groups**

| Group | Size | Members |
|---|---|---|
| 1 | 14 | EIE2, EIE3, EIE4, R-GRIX, EIESW1, EIESW3, FLEMSW, HCMRSW, PASTSW, TEIATH, AUASW, ACRSW2, EIESW2, GRIXSW |
| 2 | 14 | KOL1, ATH4, OPTICSW, HALSW, NAMUSSW, AUEBSW, ACADSW, PARLTVSW, UOASW, NTUASW, ERTSW, KOLSW1, CHASW1, KORSW |
| 3 | 11 | THES2, LAR2, EIE1, THESSW1, SERSW, THESSW2, AUTHSW, TEITHESW, UOMSW, TEILARSW, LARSW1 |
| 4 | 11 | IOAN2, PATR2, IONASW1, TEIEP@ioaSW, UOISW, TEIMESSW, TEIPATSW, CTISW, AGRINIOSW1, PATRSW1, PATRSW2 |
| 5 | 8 | SYR, HER2, RODSW, TEICRESW, HERSW1, APOLLASW, CHIOSSW, SYRSW1 |
| 6 | 5 | YPEPTH1, YPEPTH2, YPEPTH3, YPEPTHSW1, YPEPTHSW2 |
| 7 | 5 | XAN2, DRAMSW, XANSW1, XANSW2, LIMNSW |

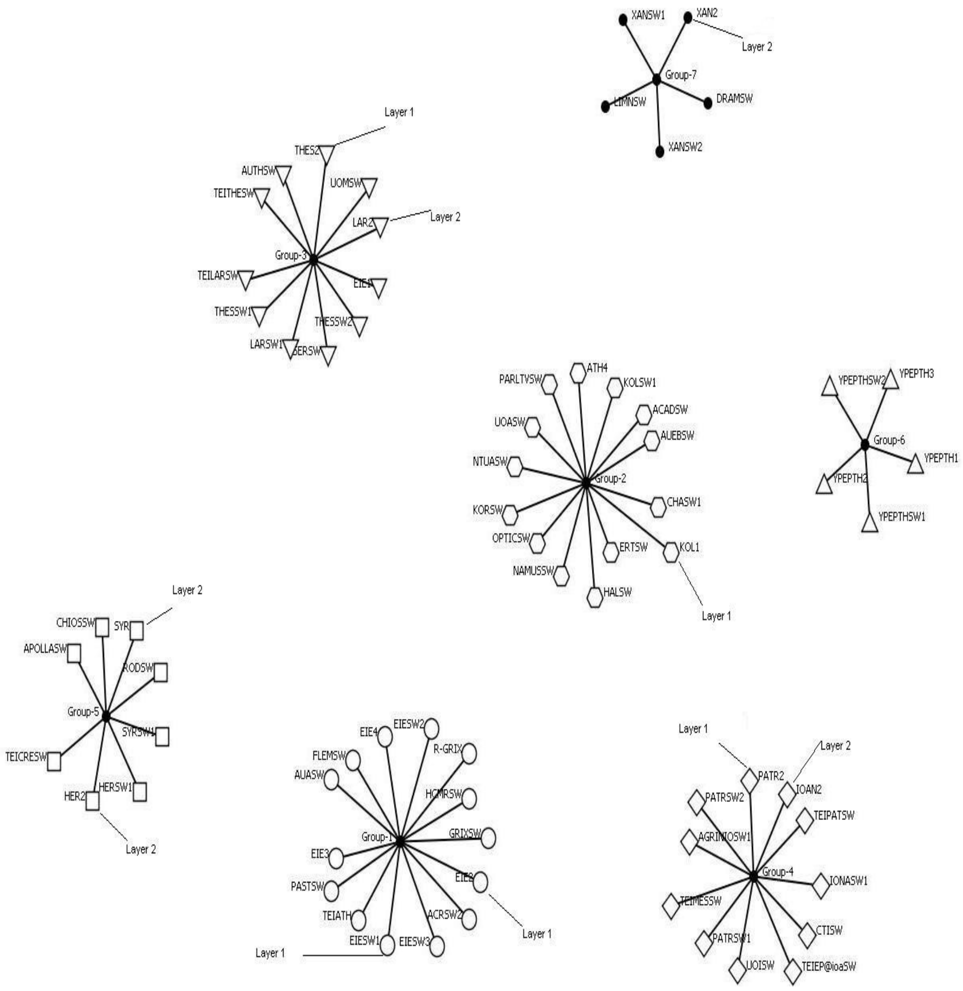


**Figure 17. Node Groups**

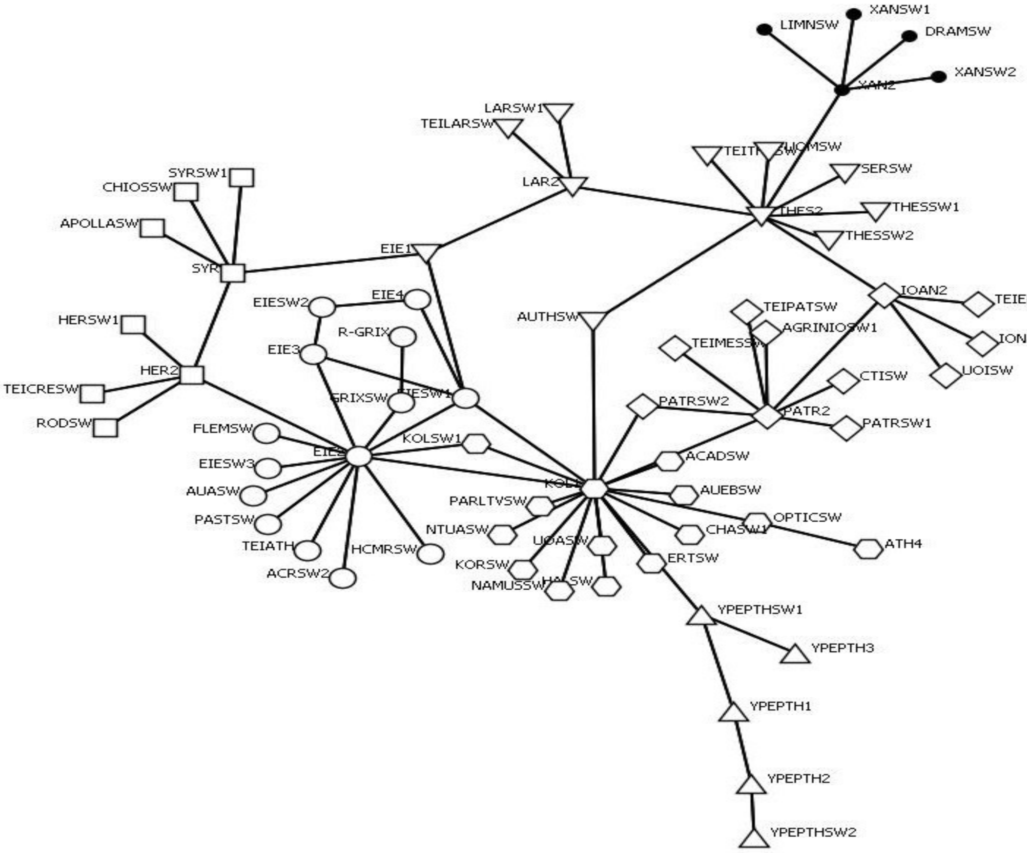


**Figure 18. Node Groups on the Social Network**

Starting from the originators and moving to the tree root node the originators are assigned as listed in Table 15 and visualized in Fig.19, where the rectangles are originators and the ellipses are collectors. The root is left abstract as it can be a log analysis or a SIEM product independent of the underline log management infrastructure.

**Table 15. Assignment of Originators to Collectors**

| Layer | Collector Location | Generators /Collectors |
|---|---|---|
| Layer 2 | HER2 | HER2, HERSW1, TEICRESW, RODSW |
| | SYR | SYR, APOLLASW, CHIOSSW, SYRSW1 |
| | LAR2 | LAR2, TEILARSW, LARSW1, EIE1 |
| | XAN2 | XAN2, LIMNSW, XANSW1, DRAMSW, XANSW2 |
| | IOAN2 | IOAN2, TEIEP@ioaSW, IONASW1, UOISW |
| | YPEPTHSW1 | YPEPTHSW1,YPEPTH1,YPEPTH2,YPEPTH3,YPEPTHSW2 |
| Layer 1 | KOL1 | KOL1, ATH4, OPTICSW, HALSW, NAMUSSW, AUEBSW, ACADSW, PARLTVSW, UOASW, NTUASW, ERTSW, KOLSW1, CHASW1, KORSW |

ΠΑΝΕΠΙΣΤΗ

| | | |
|---|---|---|
| | | Collectors from Layer 2: YPEPTHSW1 |
| | EIE2 | EIE2, R-GRIX, EIESW3, FLEMSW, HCMRSW, PASTSW, TEIATH, AUASW, ACRSW2, GRIXSW |
| | THES2 | THES2, TEITHESW, UOMSW, THESSW2, SERSW, THESSW1,AUTHSW<br>Collectors from Layer 2: LAR2, XAN2 |
| | PATR2 | PATR2, TEIMESSW, AGRINIOSW1, TEIPATSW, CTISW, PATRSW1, PATRSW2<br>Collector from Layer 2: IOAN2 |
| | EIESW1 | EIESW1,EIE3,EIE4,EIESW2<br>Collectors from Layer 2: HER2, SYR |

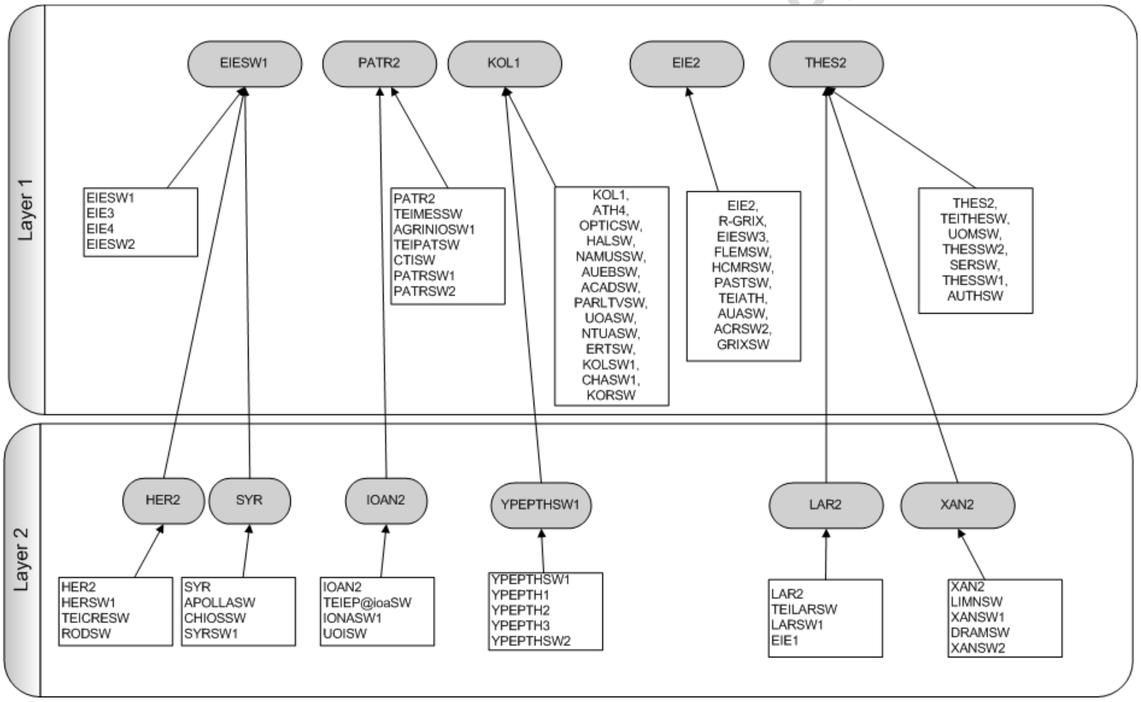


**Figure 19. Assignment of Originators to Collectors**

### 4.7.1.2 Log Collection Sizing

Having defined the locations and the roles of the *rsyslog* servers the system sizing (the volume of log data expected to process each server, the consumed network bandwidth and the required Random Access Memory (RAM)) is estimated. The EPS generated and the bandwidth required for each device has already been estimated in the log generation tier. The following table, Table 16, lists the devices that are assigned to each collector and the corresponding EPS, the bandwidth and memory consumption. The details of the estimation are listed in Appendix XI. The *rsyslog* implementation of the *syslog* protocol provides the features of filtering the messages and performing certain actions for each filter. For each of these actions a queue can be configured that will store messages on the volatile memory, the hard disk drive or both. This features have not been included in the estimation and are usually configured to handle peak traffic on the servers [24].

Table 16. Log Collection (rsyslog) Sizing

| Layer | Collector | Server Requirements | | |
|---|---|---|---|---|
| | | Total EPS | Total Bandwidth (Mbps) | Memory-RAM (bytes) |
| Layer 2 | HER2 | 125 | 0.4149 | 30,360 |
| | SYR | 125 | 0.4149 | 30,360 |
| | LAR2 | 130 | 0.4558 | 30,360 |
| | XAN2 | 130 | 0.4559 | 30,360 |
| | IOAN2 | 125 | 0.4149 | 30,360 |
| | YPEPTHSW1 | 140 | 0.5377 | 30,360 |
| Layer 1 | KOL1 | 320 | 1.4035 | 60,720 |
| | EIE2 | 160 | 0.7018 | 30,360 |
| | THES2 | 400 | 1.4496 | 91,080 |
| | PATR2 | 265 | 0.9528 | 60,720 |
| | EIESW1 | 380 | 1.2856 | 91,080 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Concerning the security of the sub-tier, the originators will transmit their logs over TCP to achieve reliable delivery to the collectors. The Layer 2 *rsyslog* servers will transmit their log data to the Layer 1 servers over TLS to ensure message confidentiality and integrity. The use of TLS could be expanded to all the supporting devices, if the implementation of a key management scheme is feasible and the administrative overhead acceptable. All *rsyslog* servers will be protected by a firewall to ensure acceptance of messages from a certain set of IP addresses with rate limiting, to protect against Denial of Service (DoS) attacks.

### 4.7.2 Log Storage

The log data that are collected by the *rsyslog* servers will go through four stages: live data, back up data, archive data and disposed data. The functions that will be performed are: log rotation, log retention, log compression and log integrity checking.

**Log data volume:** The location and the anticipated volume of log data for each collector is already estimated in the previous section. From the log requirements section it is defined that the log files should be retained for a period of one-year, prior to be disposed. Table 17 lists the collectors and the storage needed for each stage (the detailed estimation is included as Appendix XII). It lists the sizing of the storage for each collector and for the three data stages, though not all of the stages are applicable to all the collectors.

**Table 17. Log Storage Sizing**

| Layer | Collector | Storage Stages (Gbytes) | | |
|---|---|---|---|---|
| | | **Live data (5 days)** | **Back up data (5 days)** | **Archived data (1 year)** |
| Layer 2 | HER2 | 16.3346529007 | 16.3346529007 | 1192.4296617508 * |
| | SYR | 16.3346529007 | 16.3346529007 | 1192.4296617508 * |
| | LAR2 | 18.3945894241 | 18.3945894241 | 1342.8050279617 * |
| | XAN2 | 18.3945894241 | 18.3945894241 | 1342.8050279617 * |
| | IOAN2 | 16.3346529007 | 16.3346529007 | 1192.4296617508 * |
| | YPEPTHSW1 | 22.514462471 | 22.514462471 | 1643.5557603836 * |
| Layer 1 | KOL1 | 61.5084171295 | 61.5084171295 | 4490.1144504547 |
| | EIE2 | 30.7542085648 | 30.7542085648 | 2245.0572252274 |
| | THES2 | 59.3036413193 | 59.3036413193 | 4329.1658163071 |
| | PATR2 | 38.8491153717 | 38.8491153717 | 2835.9854221344 |
| | EIESW1 | 51.0638952255 | 51.0638952255 | 3727.6643514633 |

All the data stages have been estimated for the whole set of collectors, though not all of them will be implemented. For the Layer 2 collectors only the live and back up data stages are required, since their log data are forwarded to Layer 1 for archival. This way the data exist in both places, (in YPEPTHSW1 and KOL1 for example), thus in the event of a loss of communication between these two nodes the missing data can be retransmitted as soon as the connection becomes available. The stages that will not be implemented in each collector are marked with a trailing asterisk. The final destination of the log data is Layer 1, where all three stages are applicable.

**Storage mechanism:** The live data will be used for real-time security monitoring, so online storage systems such as SAN will be used to achieve high performance for the security personnel. The back up data will be used in case the live data become unavailable for any reason. A low cost solution is to use external storage that will be populated with the new live data periodically as part of the standard procedures. The archived data will also be kept in external media, as they are kept for analytical reasons such as statistical analysis, data mining, etc.

**Functions:** The log live data will be rotated every day, to keep the size of the tables manageable and facilitate searching through them. The backed up and archived data will be compressed to reduce their size and save storage space. Both functions can be included to the systems configuration and the normal execution will be checked as part of the standard opera-

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

tions. When they reach the state of disposal, after one year of retention, they will be disposed using a secure delete application (srm, wipe, etc)

**Storage engine:** The live data will be stored in a MySQL database servers using the MyISAM engine. This uses a table-locking mechanism that is optimal for simultaneous reads and writes. The back up and archived data, will be stored again in MySQL tables, using the ARCHIVE engine [22].

The following data flow diagram, Fig.20, depicts the log data life-cycle management process for the GRNET.SA log management infrastructure. Each cycle represents a data process or function and the two parallel lines represent storage, while in Fig.21 the infrastructure deployment is shown.

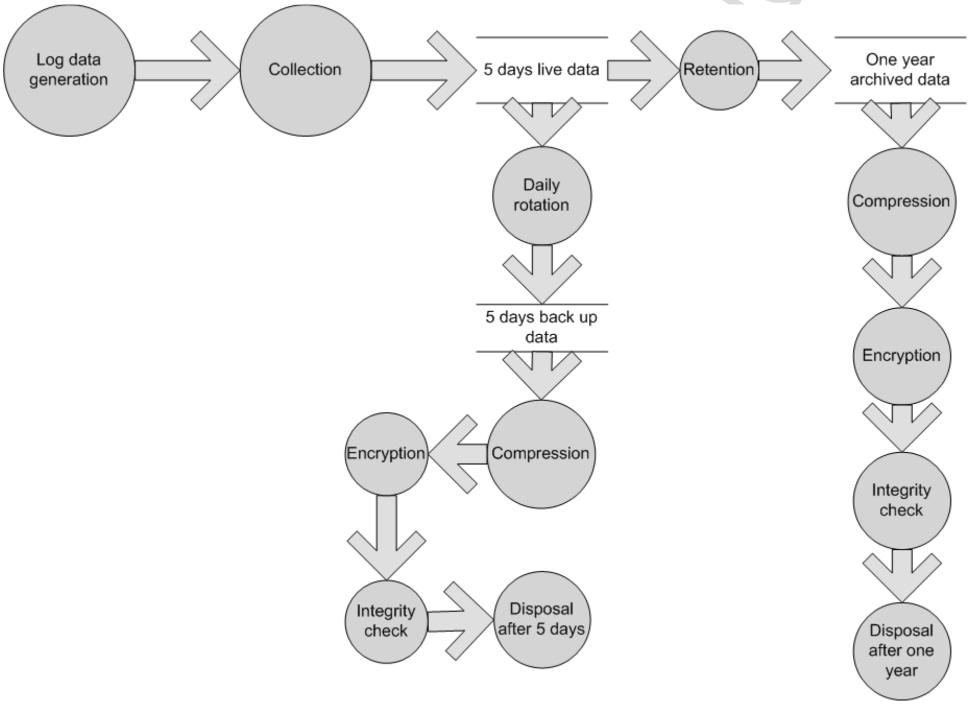


**Figure 20. Log Data Life-cycle Management Process**

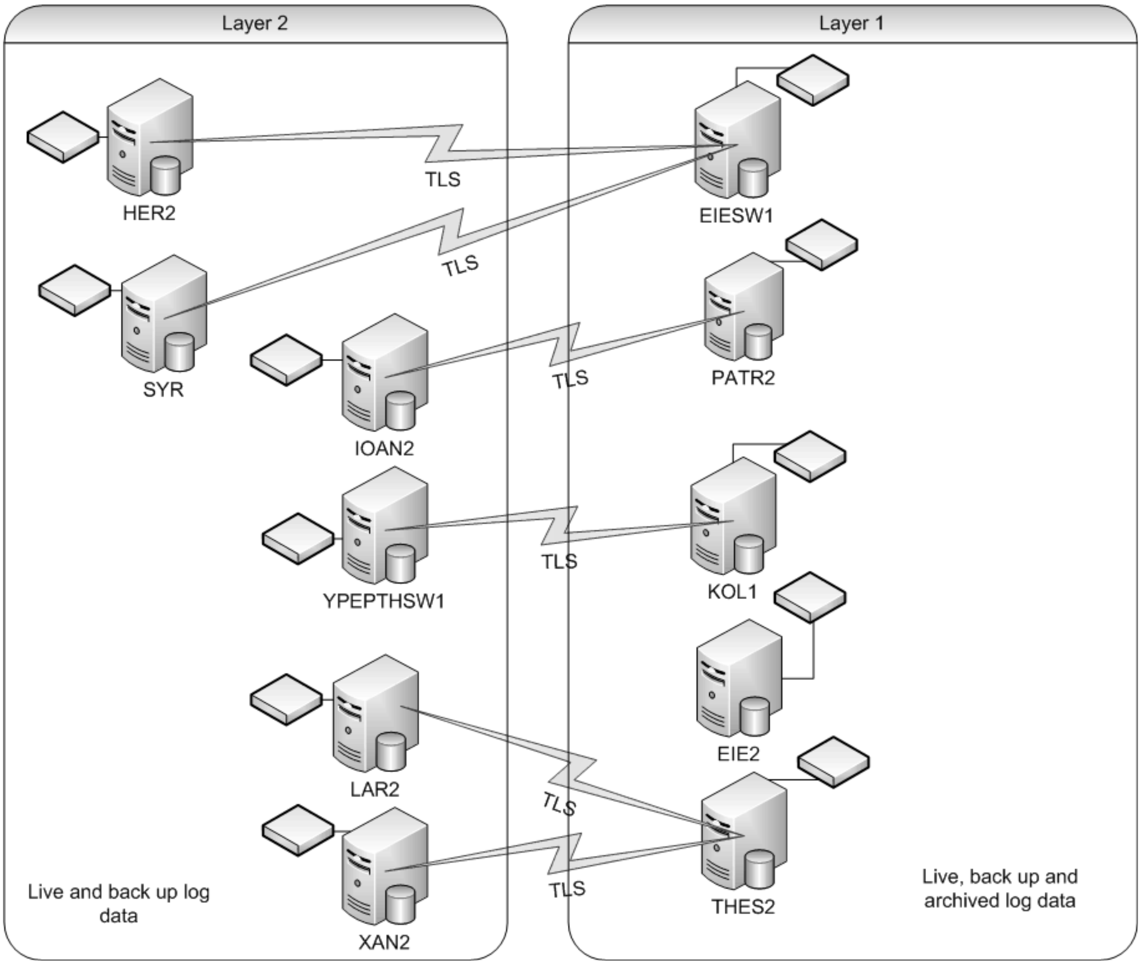


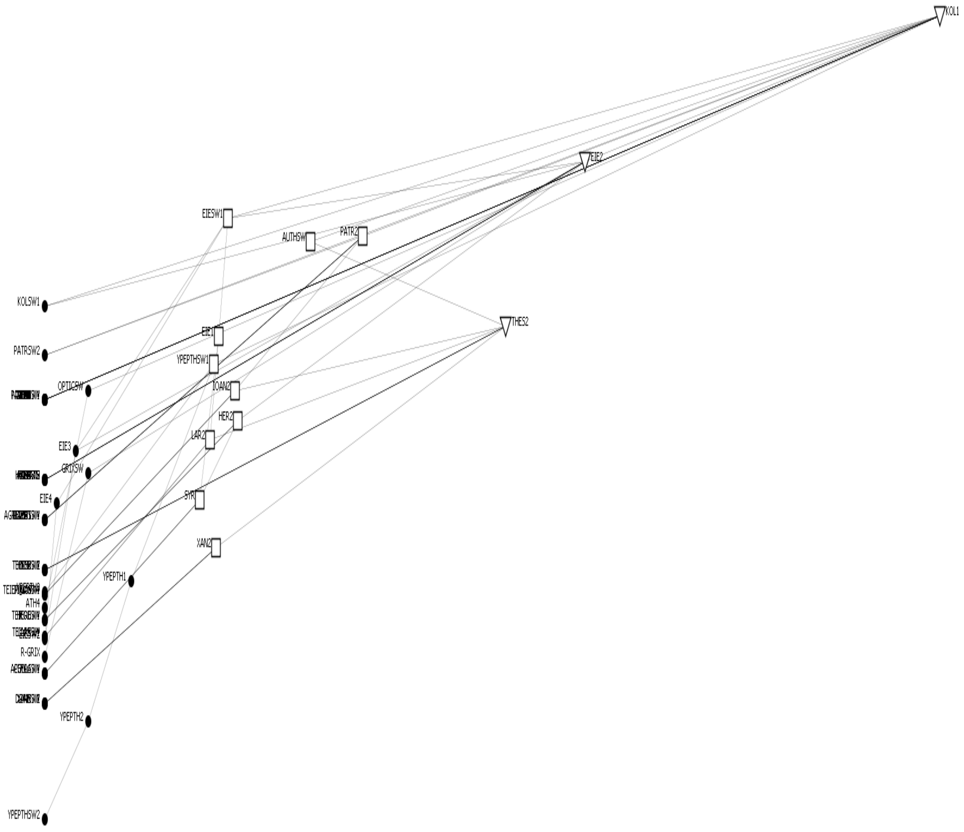
**Figure 21. Log Management Infrastructure Deployment**

Details on the physical security of the GRNET.SA are not available, but it is assumed that the infrastructure is adequately protected. The log data security is included in the life-cycle management process. Integrity, confidentiality and availability are addressed with hash functions, encryption and back ups respectively. The network and system security of the organization does not form part of this methodology and should be addressed in the overall security of the organization.

## 4.8 GRNET.SA Time Synchronization

Having concluded on the collection and storage tier the time synchronization of the devices is addressed. The solution of implementing an NTP server hierarchy is selected as it is supported by both the network devices and the server machines. No hardware equipment needs to be added, adequately configuring the selected devices is enough for the implementation. The hierarchical architecture provides scalability, stability and consistency to the NTP network.

The closeness and betweenness centrality measurements (Appendixes VI and VIII) are sorted in descending order and combined, with the help of ORA software, and visualized in Fig.22. This figure depicts the betweenness centrality in the x-axis and the closeness centrality in the y-axes. The selection of the nodes starts from the top-right moving to the bottom-left of the diagram. The inverse triangle shaped nodes form Stratum 1, the rectangle shaped ones form Stratum 2 and the remaining dot shaped nodes Stratum 3. Their positions on the social network are shown in Fig.23. The NTP servers will be implemented on the network devices, if it is not supported as in node YPEPTHSW (Juniper EX4000) the service will be activated on the *rsyslog* server present at the same location. Of course choosing another adjacent network device would also be an adequate solution.



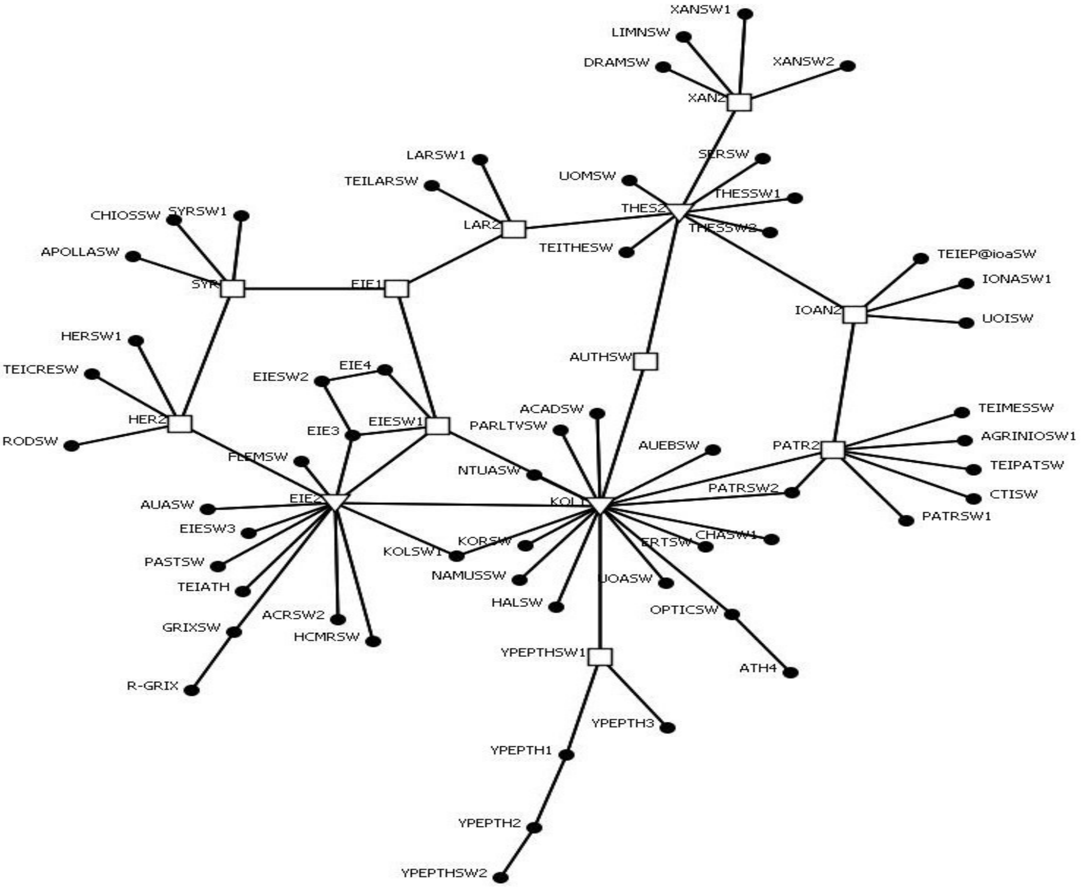


**Figure 23. NTP Strata on the Social Network**

Stratum 1 contains three servers that will reference an external public server. These servers will operate in client/server mode with the external server and in symmetric mode to each other to provide back up. Stratum 2 servers will reference all three Stratum 1 servers in client/server mode and will reference three of their peer servers in symmetric mode. The Stratum 3 servers will reference the three closest servers from Strata 1 and 2 in client/server mode and in symmetric mode with the three closest pier servers.

Concerning the security of the installation, an ACL will be included in the configuration of each NTP server. Each server will be configured to accept requests from specific clients and subnets and the clients will reference specific NTP servers. Using symmetric or asymmetric encryption would induce administrative overhead. The problem of erroneous time values is opposed buy the reference of three time servers from each client and the insanity checks performed by the protocol it self. In a case a server is compromised and false data is provided, the client can be synchronized according to its peer servers.

## 4.9 Preprocessing

At this case study no preprocessing is performed at the log generators. The Layer 1 and 2 collectors are configured to remove duplicate records and aggregate multiple ones, while the data are on transit. At the Layer 1 collectors, the log data are filtered and stored based on the type of device (the Cisco router logs separately from the Juniper routers and so on). This preprocessing results in reduced log records removing the repeating and redundant log data, stored per device, ready to be parsed from any analytical or security product.

## 4.10 Scalability

The GRNET.SA log management infrastructure design is evaluated against the five scalability dimensions. A long term increase of the events volume can be oposed with the addition of more *rsyslog* servers in the locations were the increase occurred. To cope with temporary peaks, *rsyslog* has the feature of configuring the buffer size, thus events can be buffered in the RAM or in the hard disk drive; the latter lacks in performance but preserves them even after a system reboot. A hybrid configuration is also possible, as well as enabling the buffer limits during certain time periods. The scalability of the assets inventory depends on the way of generating and maintaining it. In this case study it is static as the data were manually collected from the web site of the organization. Installing an inventory software solution would provide the desired scalability. Expanding to a new location would not be a scalability problem if the bandwidth of the new link were sufficient. Depending on the log volume, they could be forwarded to an already installed collector. A new collector could be installed to the new location following the distributed architecture. The infrastructure capacity is also scalable. The archived and back up data are stored into external media, so adding storage is of low cost and technically trivial. The live data are stored into MySQL tables, so if the EPS raised too high making the database the infrastructure's bottle-neck, then a database cluster would add the necessary scalability, with the disadvantage of the addition of administrative overhead. The log correlation and analysis is not addressed in the proposed methodology, thus it is not evaluated in terms of scalability.

## 4.11 Performance Measurement

At this point all the design and implementation aspects of the log management infrastructure have been addressed. After the infrastructure has been implemented, its performance needs to be continuously monitored to identify if the goals of its implementation are achieved, to identify reasons of poor performance and proceed to corrective actions. The measurements/metrics development process starts with the definition of goals and objectives:

- **Goal of the log management infrastructure:** To provide real-time security monitoring of the organization's network.
- **Objective-1:** Reliable log generation.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

- **Objective-2:** Reliable delivery of the log data to the distributed central locations.
- **Objective-3:** Monitor the functional state of the log generators.
- **Objective-4**: Maintain the quality of the log data.

Based on the defined objectives the measurements/metrics to monitor their performance are developed as follows (they are detailed in Appendix XIII):

- **Objective-1**:
  - Measurement 1-1: Successful generation of log data.
  - Measurement 1-2: Log record fields validation.
- **Objective-2**:
  - Measurement 2-1: Events successfully delivered.
- **Objective-3**:
  - Measurement 3-1: # of functional devices.
  - Measurement 3-2: Ratio of up and down time.
- **Objective-4**:
  - Measurement 4-1: Log data format validation.
  - Measurement 4-2: Log data time field validation.

The measures development continues with the benchmarking or definition of targets for each one, the necessary data for the measurement and the method of collection (Appendix XIII). The program review/refinement step is not applicable the first time the program is ran. It needs historical data to verify the suitability of the measurement in use.

Since no measurements are available for this case study, they will be assumed to demonstrate the application of the measurements/metrics implementation process. The gap analysis of the measurement 4-1 "Log data format validation" resulted that a Juniper MX960 router was not time-stamping the generated events with the desired format. The corrective action to alleviate the problem was to reconfigure the device to use the correct format. The cause of this misconfiguration was investigated and was found to be the lack of certified personnel for this type of equipment. A business case was compiled to assure the resources for the training of the technical personnel. The task of reconfiguration is assigned to the technical personnel and the management is informed for the necessity of the training.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Chapter 5

## Conclusions and Future Work

In this master thesis a methodology for implementing a log management infrastructure is proposed. Existent methodologies are integrated and adjusted where applicable, while the social network analysis algorithms are used to document the processes and the decisions taken. The result is a step-by-step methodology targeting at large networks. Apart from the infrastructure implementation additional issues are addressed, such as the scalability and the measurement of its performance. The application of the proposed methodology is presented through a case study. Publicly available data of the GRNET.SA network are used, while data that are not available are assumed. The data used and the measurements performed are included either in the body of the thesis or in appendixes, detailing the estimation and calculations performed where it was needed.

The conclusion that is raised through the performed analysis, is that the task of implementing a log management infrastructure is a challenging one. Most related work covers specific aspects of the problem, either tailored to a vendor's product or too abstract, limiting to guidelines. The value of the log data increases every day, as it is the most effective way to "listen" to what our systems are "telling" us. Apart from the organization's interest in leveraging knowledge from them, compliance to standards is often obligatory for certain commercial activities. The process of designing such an infrastructure is complex. Historical data prove to be useful in many cases, though they can lead only to estimations, not accurate predictions. The use of social network analysis is beneficial where applied as most methodologies are based on best practices and experience. The social network analysis, apart from mathematically documenting the performed tasks or decisions facilitates the analyst, allowing him/her to identify the network properties and characteristics. Using the relative software, the analysis if performed fast and accurately with informative visualizations.

The proposed methodology deals with the implementation of a log management infrastructure from the log source till the log data are collected and stored in a central point. The log data analysis and visualization is left as future work, along with the possible integration with a SIEM product. The implementation of the infrastructure in the cloud can also been examined, probably following an Infrastructure as a Service (IaaS) approach. In the proposed methodology security protocols and mechanisms are employed, thus examining a key management scheme could also be added. A key aspect of the infrastructure, that is not currently analyzed, is the assignment of roles and responsibilities to the personnel as well as the definition of the necessary SOP for the function and maintenance of the infrastructure. In addition, a project management process could be adjusted to be integrated with the implementation of the infrastructure and the measurements program. Social network analysis of other aspects of

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

the network, such as the flow of information or the spread of a malware infection, would also be useful. The evolution of the network can also by studied and prediction of possible new nodes and links can be performed. Finally, though the network used for the case study is real and some of the data also, the designed infrastructure needs to be implemented and evaluated for its performance in a real operational environment.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# References

[1] (2012, May).*Log and Event Management Survey Results (SANS Eighth Annual)*. Available: https://www.sans.org/reading-room/analysts-program/SortingThruNoise

[2] *Guide to Computer Security Log Management*, National Institute of Standards and Technolog, SP 800-92, 2006

[3] (2007).*Best Practices in Log Management for Security and Compliance*. Available: http://www.complytec.com/pdf/enVision_Best_Practices_Log_Man_Security_Compliance.pdf

[4] *Building an Infrastructure That Enables Log Management Best Practices*. Available: http://www.comprosec.ch/fileadmin/document_archive/Library/RSA_enVision/WPE_Building_an_Infrastructure_That_Enables_Log_Management_Best_Practices___LMBP-SIEM_WP_0907-lowres_cps_dis.pdf

[5] J.Beechey.(2007, October 21).*Log Management SIMetry: A Step by Step Guide to Selecting the Correct Solution*. Available: http://cyber-defense.sans.org/resources/papers/gsec/log-management-simetry-step-step-guide-selecting-correct-solution-109911

[6] D.Frye.(2009, September 21).*Effective Use Case Modeling for Security Information and Event Management*. Available: http://www.sans.org/reading-room/whitepapers/auditing/effective-case-modeling-security-information-event-management-33319?show=effective-case-modeling-security-information-event-management-33319&cat=auditing

[7] *SIEM in the Cloud: Cost-effective Solutions for Taking Control of Data Overload and Scaling Security*. Available: http://www.netforensics.com/

[8] S.Wasserman, K.Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press,1994

[9] W.D.Noou, A.Mrvar, V.Batagelj, *Exploratory Network Analysis with Pajek*, Cambridge University Press,2005

[10] S.P.Borgatti," The Key Player Problem", Dynamic Social Network Modeling and Analysis National Academy of Sciences Press, 2003

[11] S.Borgatti, "Identifying sets of key players in a social network",Springer Science, pp.21-34, 2006

[12] (2006, June 19).*A Guide to Security Metrics*. Available: http://www.sans.org/reading-room/whitepapers/auditing/guide-security-metrics-55

[13] *Performance Measurement Guide for Information Security*, NIST, SP 800-55 rev.1, 2008

[14] *The Critical Security Controls, Twenty Critical Security Controls for Effective Cyber Defense-Versions 4.1*. Available: http://www.sans.org/critical-security-controls/

[15] *The 6 Categories of Critical Log Information version 3.01*. Available: http://www.sans.edu/research/security-laboratory /article/ sixtoplogcategories

[16] *Guide for Conducting Risk Assessments*, National Institute of Standards and Technology, SP 800-30, 2012

[17] L.Hutcheson.(2011, August 21).*Logs-The Foundation of Good Security Monitoring ver-*

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

*sion1*.Available:https://isc.sans.edu/diary/Logs+The+Foundation+of+Good+Security+Monitoring/11410

[18] D.Miller, S.Harris, A.Harper, S.VanDyke, C.Blask, *Security Information and Event Management (SIEM) Implementation*, McGraw-Hill,2011

[19] *RFC5424 The Syslog Protocol*, IETF, RFC5424, 2009

[20] R.Gerhards.(2008, April 2).*On the (un)reliability of plain tcp syslog*. Available: http://blog.gerhards.net/2008/04/on-unreliability-of-plain-tcp-syslog.html

[21] *RFC3195 Reliable Delivery for syslog*, IETF, RFC3195, 2001

[22] *Cisco Building Scalable Syslog Management Solutions*, , , 2011

[23] J.M.Butler.(2009, February).*Benchmarking Security Information Event Management (SIEM)*.Available: http://www.sans.org/reading-room/analysts-program/even-Mgt-Feb09

[24] P.Matulis.(2009, September).*Centralized Logging with rsyslog*. Available: http://www.canonical.com/sites/default/files/active/Whitepaper-CentralisedLogging-v1.pdf

[25] *ORA: Organization Risk Analyzer, CASOS Technical Report*, Carnegie Mellon University, CMU-ISRI-04-10, 2004

[26] Freeman, L.C, *Centrality in Social Networks I: Conceptual Clarification*,1979

[27] Frantz TL," Annual Tools/Computational Approaches/Methods Conference", 2008

[28] P.Bonacich P, "Power and centrality: A family of measures", American Journal of Sociology 92, pp.1170–1182, 1987

[29] A.Clauset, M. E. J. Newman, Cristopher," Finding community structure in very large networks", Physical Review E, 2004

[30] R.Gerhards.(2008, January 31).*Handling a massive syslog database insert rate with rsyslog*. Available: http://www.rsyslog.com/doc/rsyslog_high_database_rate.html

[31] *RFC5905 Network Time Protocol Version 4: Protocol and Algorithms Specification*, IETF, RFC5905, 2010

[32] *Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE, IEEE 1588-2008, 2008

[33] (2008, December 17).*Network Time Protocol: Best Practices White Paper (Document ID:19643)*.Available:http://www.cisco.com/en/US/tech/tk869/tk769/technologies_white_paper09186a0080117070.shtml#ntpoverview

[34] L.Carroll.(2012,May 14).*NTP Security Analysis*. Available: http://www.eecis.udel.edu/mills/security.html

[35] *Scalability in Log Management (Research 010-021609-02)*. Available: https://www.nd-m.net/siem/pdf/ArcSight%20Whitepaper-%20Log%20Scalability.pdf

[36] *Network Operation Center*. Available: https://www.noc.grnet.gr/en

[37] *Rsyslog project*. Available: http://www.rsyslog.com/

[38] *Cisco*. Available: http://www.cisco.com/en/US/hmpgs/index.html

[39] *Juniper Networks*. Available: http://www.juniper.net/us/en/

[40] *Extreme networks*. Available: http://www.extremenetworks.com/

[41] *Center for Computational Analysis of Social and Organizational Systems*. Available:

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

http://www.casos.cs.cmu.edu/

[42] *The Apache Software Foundation*. Available: http://www.apache.org/

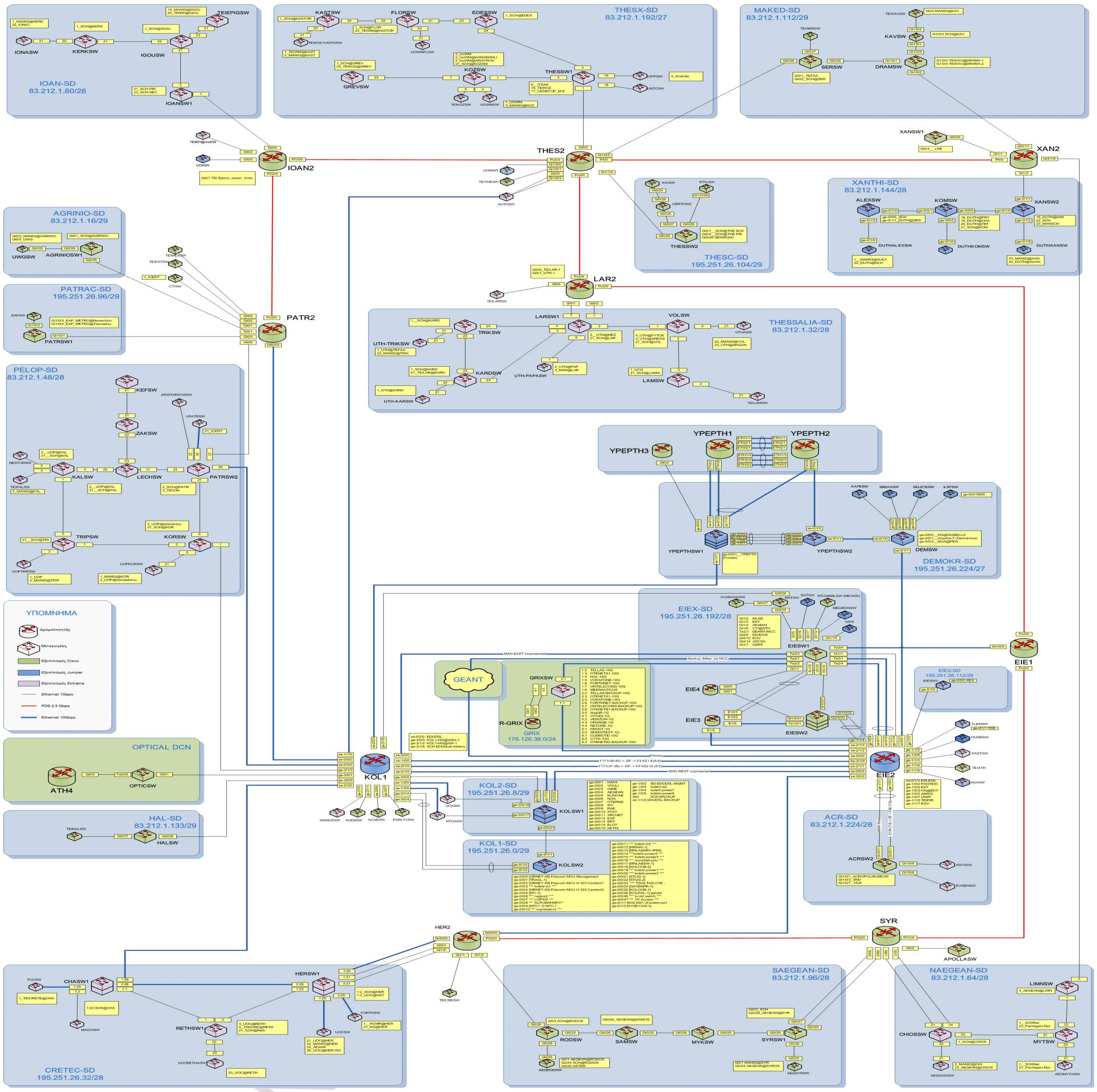
[43] *amba*. Available: http://www.samba.org/samba/

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Appendixes

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

# Appendix I: Network Topology Diagram (L2/L3), version of 2013-07-29



## Appendix II: Network Links Bandwidth

| No | Source Node | Destination Node | Nominal Bandwidth (Gbps) | Nominal Bandwidth (Mbps) | Maximum Observed Bandwidth (Mbps) | Average Observed Bandwidth (Mbps) |
|---|---|---|---|---|---|---|
| 1 | XAN2 | THES2 | 2.5 | 2560.0 | 219.41 | 118.78 |
| 2 | THES2 | IOAN2 | 2.5 | 2560.0 | 594.51 | 50.52 |
| 3 | THES2 | LAR2 | 2.5 | 2560.0 | 681.44 | 328.59 |
| 4 | IOAN2 | PATR2 | 2.5 | 2560.0 | 888.74 | 305.16 |
| 5 | PATR2 | KOL1 | 10.0 | 10240.0 | 1146.88 | 470.89 |
| 6 | LAR2 | EIE1 | 2.5 | 2560.0 | 937.21 | 450.46 |
| 7 | YPEPTH1 | YPEPTH2 | 10.0 | 10240.0 | 112.04 | 17.71 |
| 8 | EIE1 | SYR | 2.5 | 2560.0 | 130.05 | 53.83 |
| 9 | EIE2 | KOL1 | 10.0 | 10240.0 | 2447.36 | 1607.68 |
| 10 | HER2 | SYR | 25.0 | 25600.0 | 154.07 | 60.95 |
| 11 | HER2 | EIE2 | 10.0 | 10240.0 | 875.89 | 134.06 |
| 12 | EIE2 | EIE3 | 10.0 | 10240.0 | 2314.24 | 1075.2 |
| 13 | XAN2 | DRAMSW | 1.0 | 1024.0 | 79.72 | 35.92 |
| 14 | XAN2 | XANSW1 | 1.0 | 1024.0 | 0.00835 | 8.22 |
| 15 | XAN2 | XANSW2 | 1.0 | 1024.0 | 173.01 | 97.95 |
| 16 | XAN2 | LIMNSW | 1.0 | 1024.0 | 107.95 | 2.13 |
| 17 | THES2 | THESSW1 | 1.0 | 1024.0 | 253.8 | 115.82 |
| 18 | THES2 | THESSW2 | 1.0 | 1024.0 | 171 | 62.36 |
| 19 | THES2 | SERSW | 1.0 | 1024.0 | 87.63 | 42.96 |
| 20 | THES2 | AUTHSW | 1.0 | 1024.0 | 339.73 | 14.85 |
| 21 | THES2 | TEITHESW | 1.0 | 1024.0 | 38.72 | 19.36 |
| 22 | THES2 | UOMSW | 1.0 | 1024.0 | 115.07 | 32.76 |
| 23 | LAR2 | LARSW1 | 1.0 | 1024.0 | 539.46 | 119.34 |
| 24 | LAR2 | TEILARSW | 1.0 | 1024.0 | 803.97 | 742.04 |
| 25 | IOAN2 | UOISW | 1.0 | 1024.0 | 444.68 | 246.18 |
| 26 | IOAN2 | TEIEP@ioaSW | 1.0 | 1024.0 | 91.28 | 89.24 |
| 27 | IOAN2 | IONASW1 | 1.0 | 1024.0 | 49.5 | 19.22 |
| 28 | PATR2 | PATRSW2 | 1.0 | 1024.0 | 651.71 | 66.53 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| No | Source Node | Destination Node | Nominal Bandwidth (Gbps) | Nominal Bandwidth (Mbps) | Maximum Observed Bandwidth (Mbps) | Average Observed Bandwidth (Mbps) |
|---|---|---|---|---|---|---|
| 29 | PATR2 | AGRINIOSW1 | 1.0 | 1024.0 | 12.23 | 3.08 |
| 30 | PATR2 | CTISW | 1.0 | 1024.0 | 170.03 | 86.47 |
| 31 | PATR2 | TEIPATSW | 1.0 | 1024.0 | 76.61 | 34.73 |
| 32 | PATR2 | TEIMESSW | 1.0 | 1024.0 | 14.84 | 3.44 |
| 33 | PATR2 | PATRSW1 | 1.0 | 1024.0 | 54.33 | 14.59 |
| 34 | YPEPTH1 | YPEPTHSW1 | 10.0 | 10240.0 | 518,05 | 388.84 |
| 35 | YPEPTH2 | YPEPTHSW2 | 10.0 | 10240.0 | 1116.16 | 694.55 |
| 36 | YPEPTH3 | YPEPTHSW1 | 1.0 | 1024.0 | 19.47 | 9.19 |
| 37 | ATH4 | OPTICSW | 1.0 | 1024.0 | 0.1131 | 0.05107 |
| 38 | KOL1 | YPEPTHSW1 | 10.0 | 10240.0 | 0.40661 | 0.00544 |
| 39 | KOL1 | ERTSW | 1.0 | 1024.0 | 0.39184 | 0.00107 |
| 40 | KOL1 | EIESW1 | 10.0 | 10240.0 | 2826.24 | 19.48 |
| 41 | KOL1 | KOLSW1 | 10.0 | 10240.0 | 84.71 | 39.06 |
| 42 | KOL1 | UOASW | 10.0 | 10240.0 | 643.71 | 329.83 |
| 43 | KOL1 | NTUASW | 10.0 | 10240.0 | 1054.72 | 635.74 |
| 44 | KOL1 | PARLTVSW | 1.0 | 1024.0 | 9.8 | 6.16 |
| 45 | KOL1 | ACADSW | 1.0 | 1024.0 | 6.45 | 1.01 |
| 46 | KOL1 | AUEBSW | 1.0 | 1024.0 | 234.25 | 98.99 |
| 47 | KOL1 | NAMUSSW | 1.0 | 1024.0 | 13.36 | 0.10916 |
| 48 | KOL1 | CHASW1 | 10.0 | 10240.0 | 6.16 | 335.5 |
| 49 | KOL1 | HALSW | 1.0 | 1024.0 | 21.57 | 3.5 |
| 50 | KOL1 | OPTICSW | 1.0 | 1024.0 | 0.72012 | 0.04735 |
| 51 | KOL1 | KORSW | 1.0 | 1024.0 | 454.46 | 20.06 |
| 52 | KOL1 | PATRSW2 | 10.0 | 10240.0 | 842.19 | 566.74 |
| 53 | KOL1 | AUTHSW | 10.0 | 10240.0 | 362.73 | 207.68 |
| 54 | R-GRIX | GRIXSW | 1.0 | 1024.0 | 231.4 | 107.5 |
| 55 | EIE1 | EIESW1 | 1.0 | 1024.0 | 522.24 | 518.03 |
| 56 | EIE2 | EIESW1 | 10.0 | 10240.0 | 833.21 | 316.96 |
| 57 | EIE2 | EIESW3 | 1.0 | 1024.0 | 0.00142 | 0.00032 |
| 58 | EIE2 | FLEMSW | 1.0 | 1024.0 | 64.62 | 15.95 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| No | Source Node | Destination Node | Nominal Bandwidth (Gbps) | Nominal Bandwidth (Mbps) | Maximum Observed Bandwidth (Mbps) | Average Observed Bandwidth (Mbps) |
|---|---|---|---|---|---|---|
| 59 | EIE2 | HCMRSW | 1.0 | 1024.0 | 18.93 | 5.26 |
| 60 | EIE2 | PASTSW | 1.0 | 1024.0 | 14.79 | 2.07 |
| 61 | EIE2 | TEIATH | 1.0 | 1024.0 | 130.36 | 49.9 |
| 62 | EIE2 | AUASW | 1.0 | 1024.0 | 37.57 | 14.2 |
| 63 | EIE2 | ACRSW2 | 1.0 | 1024.0 | 60.15 | 16.51 |
| 64 | EIE2 | KOLSW1 | 10.0 | 10240.0 | 92.47 | 10.68 |
| 65 | EIE2 | GRIXSW | 10.0 | 10240.0 | 2344.96 | 1617.92 |
| 66 | EIE3 | EIESW2 | 10.0 | 10240.0 | 704.61 | 50.35 |
| 67 | EIE3 | EIESW1 | 1.0 | 1024.0 | 62.43 | 2.16 |
| 68 | EIE4 | EIESW1 | 1.0 | 1024.0 | 665.6 | 0.0011 |
| 69 | EIE4 | EIESW2 | 1.0 | 1024.0 | 0.08648 | 0.03328 |
| 70 | HER2 | RODSW | 1.0 | 1024.0 | 8.7 | 1.9 |
| 71 | HER2 | TEICRESW | 1.0 | 1024.0 | 95.63 | 44.23 |
| 72 | HER2 | HERSW1 | 10.0 | 10240.0 | 783.02 | 78.62 |
| 73 | SYR | APOLLASW | 1.0 | 1024.0 | 0.00112 | 0.00105 |
| 74 | SYR | CHIOSSW | 1.0 | 1024.0 | 126.06 | 53.98 |
| 75 | SYR | SYRSW1 | 1.0 | 1024.0 | 135.9 | 66.23 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix III: Output of TDBUMO Bottom Up Step

| **Specific version of system** | **Debian 7.1 64bit** | | |
|---|---|---|---|
| | | **Log file name** | **Log file description** |
| **Specific type of log** | System [19] | auth.log | usage of authorization systems, the mechanisms for authorizing users which prompt for user passwords |
| | System | Daemon.log | contains information about running system and application daemons |
| | System | debug | detailed debug messages from the system and applications which log to syslogd at the DEBUG level |
| | System | Kern.log | detailed log of messages from the kernel |
| | System | messages | contains informational messages from applications, and system facilities |
| | System | syslog | may contain information other logs do not about the system |
| | System | Dpkg.log | Contains information that are logged when a package is installed or removed using dpkg command |
| | Apache [42] | access.log | records of every page served and every file loaded by the web server |
| | Apache | error.log | records of all error conditions reported by the HTTP server |
| | Samba [43] | log.nmbd | messages related to Samba's NETBIOS over IP functionality |
| | Samba | log.smbd | messages related to Samba's SMB/CIFS functionality |
| | Samba | log.[IP_ADDRESS] | messages related to requests for services from the IP address contained in the log file name |
| **Log file location** | System | /var/log/ | |
| | Apache2 | /var/log/apache2/ | |
| | Samba | /var/log/samba/ | |
| **Log levels** | System | Emergency, Alert, | |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| | | | |
|---|---|---|---|
| | | Critical, Error, Warning, Notice, Informational, Debug | |
| | Apache2 | Emerg, alert, crit, error, warn, notice, info, debug | |
| | Samba | Levels 0-10<br>0: critical errors, serious warnings<br>1: small amount of info<br>2-3: debug for Administrators<br>>3: for developers | |
| **Log format** | System | Rsyslog format: TIMESTAMP, HOSTNAME, APP-NAME, PROCID, MSGID, STRUCTURED-DATA, SD-ELEMENT, SD-ID, SD-PARAM, MSG | RFC5424 (obsoletes FRC3164) |
| | Apache2 | Custom format or Common Log Format :<br>Remote host, Remote login name, Remote user, Time the request was received, First line of request, status of the *original* request, Size of response in bytes | |
| | Samba | Timestamp, user, IP address,  message | |
| **Application version** | System | rsyslogd 5.8.11 | |
| | Apache2 | Apache/2.2.22 (Debian) | |
| | Samba | Samba 2:3.6.6-6 | |
| **Storage** | System | Local raw files and MySQL database | |
| | Apache2 | Local raw files | |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| | | | |
|---|---|---|---|
| | Samba | Local raw files | |
| **Access/retrieval** | System | syslog<br>agent | Rsyslog: Transmission over UDP/TCP on port 514, over TLS and over RELP<br>agent: software product dependent |
| | Apache2 | rsyslog (error.log)<br>rsyslog and logger (access.log)<br>agent | |
| | Samba | rsyslog<br>agent | |
| **Log size** | | | |
| **Rotation frequency** | | | |

| **Specific version of system** | **Cisco CAT3750, Cisco GSR12416, Juniper MX960, Juniper EX4200, ExtremeX450** | | |
|---|---|---|---|
| **Specific log files** | | | |
| **Log file location** | internal buffer | | |
| **Log levels** | Cisco | Emergency, Alert, Critical, Error, Warning, Notification, Informational, Debugging | |
| | Juniper | Any, none, emergency, alert, critical, error, warning, notice, info | |
| | Extreme | Critical, warning, informational, debug | |
| **Log format** | Cisco | Facility, severity, hostname, timestamp, message(%FACILITY-SEVERITY-MNEMONIC: Message-text ) | RFC5424 (obsoletes RFC3164) |
| | Juniper | <priority code>version timestamp hostname process processID TAG [junos@2636.platform variable-value-pairs] message-text | RFC5424 (obsoletes RFC3164) |
| | Extreme | Timestamp, username, fault level | |
| **Application version** | Cisco | Cisco IOS | |
| | Juniper | Junos | |
| | Extreme | ExtremeXOS | |
| **Storage** | internal buffer<br>remote server | | |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| | | | |
|---|---|---|---|
| **Access/retrieval** | console<br>syslog | syslog: transmission over UDP 514 | |
| **Log size** | | | |
| **Rotation frequency** | | | |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix IV: TDBUMO: Output of Middle-Out Step

| Specific system | | Log file | Requirement | | | |
|---|---|---|---|---|---|---|
| | | | Log individual user access to protected information | Log the execution of privileged actions | Log the invalid access attempts | Log suspicious network activity |
| Debian 7.1 64bit | system | auth.log | | Informational/ RFC5424 | Informational/ RFC5424 | |
| | system | Daemon.log | | | | |
| | system | debug | | | | |
| | system | Kern.log | | | | |
| | system | messages | | | | |
| | system | syslog | | Informational/RFC5424 | Informational/RFC5424 | Informational /RFC5424 |
| | system | Dpkg.log | | Informational/RFC5424 | | |
| | apache2 | access.log | Info/CLF | | Info/CLF | |
| | apache2 | error.log | Info/CLF | | | |
| | samba | log.nmbd | Level 1/proprietary | | Level 1/proprietary | |
| | samba | log.smbd | Level 1/proprietary | | Level 1/proprietary | |
| | samba | log.[IP_ADDRESS] | Level 1/proprietary | | Level 1/proprietary | |
| Router | Cisco | | | | | Warning/ RFC5424 |
| | Juniper | | | | | Warning/ RFC5424 |
| Switch | Cisco | | | | | Warning/ RFC5424 |
| | Juniper | | | | | Warning/ RFC5424 |
| | Extreme | | | | | Warning/ RFC5424 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix V: Total Degree Centrality

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.269 | 18.000 |
| 2 | EIE2 | 0.194 | 13.000 |
| 3 | THES2 | 0.134 | 9.000 |
| 4 | PATR2 | 0.119 | 8.000 |
| 5 | XAN2 | 0.075 | 5.000 |
| 6 | IOAN2 | 0.075 | 5.000 |
| 7 | SYR | 0.075 | 5.000 |
| 8 | HER2 | 0.075 | 5.000 |
| 9 | EIESW1 | 0.075 | 5.000 |
| 10 | LAR2 | 0.060 | 4.000 |
| 11 | EIE1 | 0.045 | 3.000 |
| 12 | EIE3 | 0.045 | 3.000 |
| 13 | YPEPTHSW1 | 0.045 | 3.000 |
| 14 | YPEPTH1 | 0.030 | 2.000 |
| 15 | YPEPTH2 | 0.030 | 2.000 |
| 16 | EIE4 | 0.030 | 2.000 |
| 17 | AUTHSW | 0.030 | 2.000 |
| 18 | PATRSW2 | 0.030 | 2.000 |
| 19 | OPTICSW | 0.030 | 2.000 |
| 20 | KOLSW1 | 0.030 | 2.000 |
| 21 | EIESW2 | 0.030 | 2.000 |
| 22 | GRIXSW | 0.030 | 2.000 |
| 23 | YPEPTH3 | 0.015 | 1.000 |
| 24 | R-GRIX | 0.015 | 1.000 |
| 25 | ATH4 | 0.015 | 1.000 |
| 26 | IONASW1 | 0.015 | 1.000 |
| 27 | TEIEP@ioaSW | 0.015 | 1.000 |
| 28 | UOISW | 0.015 | 1.000 |
| 29 | THESSW1 | 0.015 | 1.000 |
| 30 | SERSW | 0.015 | 1.000 |
| 31 | THESSW2 | 0.015 | 1.000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 32 | TEITHESW | 0.015 | 1.000 |
| 33 | UOMSW | 0.015 | 1.000 |
| 34 | DRAMSW | 0.015 | 1.000 |
| 35 | XANSW1 | 0.015 | 1.000 |
| 36 | XANSW2 | 0.015 | 1.000 |
| 37 | TEILARSW | 0.015 | 1.000 |
| 38 | LARSW1 | 0.015 | 1.000 |
| 39 | TEIMESSW | 0.015 | 1.000 |
| 40 | TEIPATSW | 0.015 | 1.000 |
| 41 | CTISW | 0.015 | 1.000 |
| 42 | AGRINIOSW1 | 0.015 | 1.000 |
| 43 | PATRSW1 | 0.015 | 1.000 |
| 44 | YPEPTHSW2 | 0.015 | 1.000 |
| 45 | HALSW | 0.015 | 1.000 |
| 46 | NAMUSSW | 0.015 | 1.000 |
| 47 | AUEBSW | 0.015 | 1.000 |
| 48 | ACADSW | 0.015 | 1.000 |
| 49 | PARLTVSW | 0.015 | 1.000 |
| 50 | UOASW | 0.015 | 1.000 |
| 51 | NTUASW | 0.015 | 1.000 |
| 52 | ERTSW | 0.015 | 1.000 |
| 53 | EIESW3 | 0.015 | 1.000 |
| 54 | FLEMSW | 0.015 | 1.000 |
| 55 | HCMRSW | 0.015 | 1.000 |
| 56 | PASTSW | 0.015 | 1.000 |
| 57 | TEIATH | 0.015 | 1.000 |
| 58 | AUASW | 0.015 | 1.000 |
| 59 | ACRSW2 | 0.015 | 1.000 |
| 60 | RODSW | 0.015 | 1.000 |
| 61 | TEICRESW | 0.015 | 1.000 |
| 62 | HERSW1 | 0.015 | 1.000 |
| 63 | CHASW1 | 0.015 | 1.000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 64 | APOLLASW | 0.015 | 1.000 |
| 65 | CHIOSSW | 0.015 | 1.000 |
| 66 | LIMNSW | 0.015 | 1.000 |
| 67 | SYRSW1 | 0.015 | 1.000 |
| 68 | KORSW | 0.015 | 1.000 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix VI: Closeness Centrality

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.438 | 0.007 |
| 2 | EIE2 | 0.385 | 0.006 |
| 3 | EIESW1 | 0.366 | 0.005 |
| 4 | PATR2 | 0.360 | 0.005 |
| 5 | AUTHSW | 0.358 | 0.005 |
| 6 | KOLSW1 | 0.337 | 0.005 |
| 7 | THES2 | 0.330 | 0.005 |
| 8 | EIE1 | 0.327 | 0.005 |
| 9 | PATRSW2 | 0.321 | 0.005 |
| 10 | YPEPTHSW1 | 0.318 | 0.005 |
| 11 | IOAN2 | 0.316 | 0.005 |
| 12 | OPTICSW | 0.309 | 0.005 |
| 13 | HALSW | 0.306 | 0.005 |
| 14 | NAMUSSW | 0.306 | 0.005 |
| 15 | AUEBSW | 0.306 | 0.005 |
| 16 | ACADSW | 0.306 | 0.005 |
| 17 | PARLTVSW | 0.306 | 0.005 |
| 18 | UOASW | 0.306 | 0.005 |
| 19 | NTUASW | 0.306 | 0.005 |
| 20 | ERTSW | 0.306 | 0.005 |
| 21 | CHASW1 | 0.306 | 0.005 |
| 22 | KORSW | 0.306 | 0.005 |
| 23 | HER2 | 0.302 | 0.005 |
| 24 | LAR2 | 0.293 | 0.004 |
| 25 | EIE3 | 0.289 | 0.004 |
| 26 | GRIXSW | 0.282 | 0.004 |
| 27 | EIESW3 | 0.279 | 0.004 |
| 28 | FLEMSW | 0.279 | 0.004 |
| 29 | HCMRSW | 0.279 | 0.004 |
| 30 | PASTSW | 0.279 | 0.004 |
| 31 | TEIATH | 0.279 | 0.004 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 32 | AUASW | 0.279 | 0.004 |
| 33 | ACRSW2 | 0.279 | 0.004 |
| 34 | SYR | 0.272 | 0.004 |
| 35 | EIE4 | 0.271 | 0.004 |
| 36 | TEIMESSW | 0.266 | 0.004 |
| 37 | TEIPATSW | 0.266 | 0.004 |
| 38 | CTISW | 0.266 | 0.004 |
| 39 | AGRINIOSW1 | 0.266 | 0.004 |
| 40 | PATRSW1 | 0.266 | 0.004 |
| 41 | XAN2 | 0.257 | 0.004 |
| 42 | THESSW1 | 0.249 | 0.004 |
| 43 | SERSW | 0.249 | 0.004 |
| 44 | THESSW2 | 0.249 | 0.004 |
| 45 | TEITHESW | 0.249 | 0.004 |
| 46 | UOMSW | 0.249 | 0.004 |
| 47 | YPEPTH1 | 0.245 | 0.004 |
| 48 | YPEPTH3 | 0.242 | 0.004 |
| 49 | IONASW1 | 0.241 | 0.004 |
| 50 | TEIEP@ioaSW | 0.241 | 0.004 |
| 51 | UOISW | 0.241 | 0.004 |
| 52 | ATH4 | 0.237 | 0.004 |
| 53 | RODSW | 0.233 | 0.003 |
| 54 | TEICRESW | 0.233 | 0.003 |
| 55 | HERSW1 | 0.233 | 0.003 |
| 56 | TEILARSW | 0.227 | 0.003 |
| 57 | LARSW1 | 0.227 | 0.003 |
| 58 | EIESW2 | 0.226 | 0.003 |
| 59 | R-GRIX | 0.220 | 0.003 |
| 60 | APOLLASW | 0.215 | 0.003 |
| 61 | CHIOSSW | 0.215 | 0.003 |
| 62 | SYRSW1 | 0.215 | 0.003 |
| 63 | DRAMSW | 0.205 | 0.003 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 64 | XANSW1 | 0.205 | 0.003 |
| 65 | XANSW2 | 0.205 | 0.003 |
| 66 | LIMNSW | 0.205 | 0.003 |
| 67 | YPEPTH2 | 0.199 | 0.003 |
| 68 | YPEPTHSW2 | 0.166 | 0.002 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix VII: Eigenvector Centrality

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.823 | 0.582 |
| 2 | EIE2 | 0.571 | 0.404 |
| 3 | EIESW1 | 0.367 | 0.260 |
| 4 | PATR2 | 0.299 | 0.211 |
| 5 | KOLSW1 | 0.287 | 0.203 |
| 6 | PATRSW2 | 0.231 | 0.163 |
| 7 | EIE3 | 0.206 | 0.145 |
| 8 | AUTHSW | 0.188 | 0.133 |
| 9 | YPEPTHSW1 | 0.185 | 0.131 |
| 10 | OPTICSW | 0.177 | 0.125 |
| 11 | HALSW | 0.169 | 0.120 |
| 12 | NAMUSSW | 0.169 | 0.120 |
| 13 | AUEBSW | 0.169 | 0.120 |
| 14 | ACADSW | 0.169 | 0.120 |
| 15 | PARLTVSW | 0.169 | 0.120 |
| 16 | UOASW | 0.169 | 0.120 |
| 17 | NTUASW | 0.169 | 0.120 |
| 18 | ERTSW | 0.169 | 0.120 |
| 19 | CHASW1 | 0.169 | 0.120 |
| 20 | KORSW | 0.169 | 0.120 |
| 21 | HER2 | 0.148 | 0.105 |
| 22 | GRIXSW | 0.123 | 0.087 |
| 23 | EIESW3 | 0.118 | 0.083 |
| 24 | FLEMSW | 0.118 | 0.083 |
| 25 | HCMRSW | 0.118 | 0.083 |
| 26 | PASTSW | 0.118 | 0.083 |
| 27 | TEIATH | 0.118 | 0.083 |
| 28 | AUASW | 0.118 | 0.083 |
| 29 | ACRSW2 | 0.118 | 0.083 |
| 30 | EIE1 | 0.096 | 0.068 |
| 31 | IOAN2 | 0.092 | 0.065 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 32 | THES2 | 0.090 | 0.063 |
| 33 | EIE4 | 0.088 | 0.062 |
| 34 | TEIMESSW | 0.062 | 0.044 |
| 35 | TEIPATSW | 0.062 | 0.044 |
| 36 | CTISW | 0.062 | 0.044 |
| 37 | AGRINIOSW1 | 0.062 | 0.044 |
| 38 | PATRSW1 | 0.062 | 0.044 |
| 39 | EIESW2 | 0.060 | 0.043 |
| 40 | SYR | 0.058 | 0.041 |
| 41 | LAR2 | 0.042 | 0.030 |
| 42 | YPEPTH1 | 0.040 | 0.028 |
| 43 | YPEPTH3 | 0.038 | 0.027 |
| 44 | ATH4 | 0.036 | 0.026 |
| 45 | RODSW | 0.031 | 0.022 |
| 46 | TEICRESW | 0.031 | 0.022 |
| 47 | HERSW1 | 0.031 | 0.022 |
| 48 | R-GRIX | 0.025 | 0.018 |
| 49 | XAN2 | 0.022 | 0.016 |
| 50 | IONASW1 | 0.019 | 0.013 |
| 51 | TEIEP@ioaSW | 0.019 | 0.013 |
| 52 | UOISW | 0.019 | 0.013 |
| 53 | THESSW1 | 0.018 | 0.013 |
| 54 | SERSW | 0.018 | 0.013 |
| 55 | THESSW2 | 0.018 | 0.013 |
| 56 | TEITHESW | 0.018 | 0.013 |
| 57 | UOMSW | 0.018 | 0.013 |
| 58 | APOLLASW | 0.012 | 0.008 |
| 59 | CHIOSSW | 0.012 | 0.008 |
| 60 | SYRSW1 | 0.012 | 0.008 |
| 61 | TEILARSW | 0.009 | 0.006 |
| 62 | LARSW1 | 0.009 | 0.006 |
| 63 | YPEPTH2 | 0.009 | 0.006 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 64 | DRAMSW | 0.005 | 0.003 |
| 65 | XANSW1 | 0.005 | 0.003 |
| 66 | XANSW2 | 0.005 | 0.003 |
| 67 | LIMNSW | 0.005 | 0.003 |
| 68 | YPEPTHSW2 | 0.002 | 0.001 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix VIII: Betweenness Centrality

| Rank | Agent | Value | Unscaled |
|---|---|---|---|
| 1 | KOL1 | 0.645 | 1427.100 |
| 2 | EIE2 | 0.368 | 813.033 |
| 3 | THES2 | 0.314 | 693.833 |
| 4 | PATR2 | 0.216 | 478.500 |
| 5 | AUTHSW | 0.181 | 400.433 |
| 6 | HER2 | 0.133 | 293.667 |
| 7 | IOAN2 | 0.129 | 285.500 |
| 8 | EIESW1 | 0.125 | 276.133 |
| 9 | EIE1 | 0.119 | 262.400 |
| 10 | XAN2 | 0.117 | 258.000 |
| 11 | YPEPTHSW1 | 0.115 | 255.000 |
| 12 | LAR2 | 0.113 | 248.900 |
| 13 | SYR | 0.105 | 233.000 |
| 14 | YPEPTH1 | 0.059 | 130.000 |
| 15 | YPEPTH2 | 0.030 | 66.000 |
| 16 | OPTICSW | 0.030 | 66.000 |
| 17 | GRIXSW | 0.030 | 66.000 |
| 18 | EIE3 | 0.021 | 46.767 |
| 19 | EIE4 | 0.008 | 18.233 |
| 20 | EIESW2 | 0.000 | 0.500 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix IX: GRNET.SA Node Set

| Node ID | Node Title | Device Model | Device Type | Vendor |
|---|---|---|---|---|
| R-1 | XAN2 | Cisco GSR12416 | Router | Cisco |
| R-2 | THES2 | Cisco GSR12416 | Router | Cisco |
| R-3 | IOAN2 | Cisco GSR12416 | Router | Cisco |
| R-4 | LAR2 | Cisco GSR12416 | Router | Cisco |
| R-5 | PATR2 | Cisco GSR12416 | Router | Cisco |
| R-6 | YPEPTH1 | Cisco GSR12416 | Router | Cisco |
| R-7 | YPEPTH2 | Cisco GSR12416 | Router | Cisco |
| R-8 | YPEPTH3 | Cisco GSR12416 | Router | Cisco |
| R-9 | EIE1 | Cisco GSR12416 | Router | Cisco |
| R-10 | EIE2 | Juniper MX960 | Router | Juniper |
| R-11 | EIE3 | Cisco GSR12416 | Router | Cisco |
| R-12 | EIE4 | Cisco GSR12416 | Router | Cisco |
| R-13 | R-GRIX | Cisco GSR12416 | Router | Cisco |
| R-14 | KOL1 | Juniper MX960 | Router | Juniper |
| R-15 | ATH4 | Cisco GSR12416 | Router | Cisco |
| R-16 | SYR | Cisco GSR12416 | Router | Cisco |
| R-17 | HER2 | Cisco GSR12416 | Router | Cisco |
| SW-1 | IONASW1 | Extreme X450 | Switch | Extreme |
| SW-2 | TEIEP@ioaSW | Extreme X450 | Switch | Extreme |
| SW-3 | UOISW | Juniper EX4200 | Switch | Juniper |
| SW-4 | THESSW1 | Extreme X450 | Switch | Extreme |
| SW-5 | SERSW | Cisco CAT3750 | Switch | Cisco |
| SW-6 | THESSW2 | Cisco CAT3750 | Switch | Cisco |
| SW-7 | AUTHSW | Extreme X450 | Switch | Extreme |
| SW-8 | TEITHESW | Cisco CAT3750 | Switch | Cisco |
| SW-9 | UOMSW | Juniper EX4200 | Switch | Juniper |
| SW-10 | DRAMSW | Cisco CAT3750 | Switch | Cisco |
| SW-11 | XANSW1 | Cisco CAT3750 | Switch | Cisco |
| SW-12 | XANSW2 | Juniper EX4200 | Switch | Juniper |
| SW-13 | TEILARSW | Extreme X450 | Switch | Extreme |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Node ID | Node Title | Device Model | Device Type | Vendor |
|---|---|---|---|---|
| SW-14 | LARSW1 | Extreme X450 | Switch | Extreme |
| SW-15 | TEIMESSW | Cisco CAT3750 | Switch | Cisco |
| SW-16 | TEIPATSW | Cisco CAT3750 | Switch | Cisco |
| SW-17 | CTISW | Cisco CAT3750 | Switch | Cisco |
| SW-18 | AGRINIOSW1 | Cisco CAT3750 | Switch | Cisco |
| SW-19 | PATRSW1 | Cisco CAT3750 | Switch | Cisco |
| SW-20 | PATRSW2 | Extreme X450 | Switch | Extreme |
| SW-21 | YPEPTHSW1 | Juniper EX4200 | Switch | Juniper |
| SW-22 | YPEPTHSW2 | Juniper EX4200 | Switch | Juniper |
| SW-23 | OPTICSW | Cisco CAT3750 | Switch | Cisco |
| SW-24 | HALSW | Cisco CAT3750 | Switch | Cisco |
| SW-25 | NAMUSSW | Extreme X450 | Switch | Extreme |
| SW-26 | AUEBSW | Cisco CAT3750 | Switch | Cisco |
| SW-27 | ACADSW | Cisco CAT3750 | Switch | Cisco |
| SW-28 | PARLTVSW | Cisco CAT3750 | Switch | Cisco |
| SW-29 | UOASW | Extreme X450 | Switch | Extreme |
| SW-30 | NTUASW | Extreme X450 | Switch | Extreme |
| SW-31 | ERTSW | Cisco CAT3750 | Switch | Cisco |
| SW-32 | EIESW1 | Cisco CAT3750 | Switch | Cisco |
| SW-33 | KOLSW1 | Juniper EX4200 | Switch | Juniper |
| SW-34 | EIESW3 | Juniper EX4200 | Switch | Juniper |
| SW-35 | FLEMSW | Juniper EX4200 | Switch | Juniper |
| SW-36 | HCMRSW | Juniper EX4200 | Switch | Juniper |
| SW-37 | PASTSW | Extreme X450 | Switch | Extreme |
| SW-38 | TEIATH | Cisco CAT3750 | Switch | Cisco |
| SW-39 | AUASW | Juniper EX4200 | Switch | Juniper |
| SW-40 | ACRSW2 | Cisco CAT3750 | Switch | Cisco |
| SW-41 | EIESW2 | Cisco CAT3750 | Switch | Cisco |
| SW-42 | RODSW | Cisco CAT3750 | Switch | Cisco |
| SW-43 | TEICRESW | Cisco CAT3750 | Switch | Cisco |
| SW-44 | HERSW1 | Extreme X450 | Switch | Extreme |
| SW-45 | CHASW1 | Extreme X450 | Switch | Extreme |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Node ID | Node Title | Device Model | Device Type | Vendor |
|---|---|---|---|---|
| SW-46 | APOLLASW | Cisco CAT3750 | Switch | Cisco |
| SW-47 | CHIOSSW | Extreme X450 | Switch | Extreme |
| SW-48 | LIMNSW | Extreme X450 | Switch | Extreme |
| SW-49 | SYRSW1 | Cisco CAT3750 | Switch | Cisco |
| SW-50 | GRIXSW | Extreme X450 | Switch | Extreme |
| SW-51 | KORSW | Extreme X450 | Switch | Extreme |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix X: Recurring Top Ranked Nodes

| Rank | Betweenness centrality | Closeness centrality | Eigenvector centrality | Total degree centrality |
|---|---|---|---|---|
| 1 | KOL1 | KOL1 | KOL1 | KOL1 |
| 2 | EIE2 | EIE2 | EIE2 | EIE2 |
| 3 | THES2 | EIESW1 | EIESW1 | THES2 |
| 4 | PATR2 | PATR2 | PATR2 | PATR2 |
| 5 | AUTHSW | AUTHSW | KOLSW1 | XAN2 |
| 6 | HER2 | KOLSW1 | PATRSW2 | IOAN2 |
| 7 | IOAN2 | THES2 | EIE3 | SYR |
| 8 | EIESW1 | EIE1 | AUTHSW | HER2 |
| 9 | EIE1 | PATRSW2 | YPEPTHSW1 | EIESW1 |
| 10 | XAN2 | YPEPTHSW1 | OPTICSW | LAR2 |
| 11 | YPEPTHSW1 | IOAN2 | HALSW | EIE1 |
| 12 | LAR2 | OPTICSW | NAMUSSW | EIE3 |
| 13 | SYR | HALSW | AUEBSW | YPEPTHSW1 |
| 14 | YPEPTH1 | NAMUSSW | ACADSW | YPEPTH1 |
| 15 | YPEPTH2 | AUEBSW | PARLTVSW | YPEPTH2 |
| 16 | OPTICSW | ACADSW | UOASW | EIE4 |
| 17 | GRIXSW | PARLTVSW | NTUASW | AUTHSW |
| 18 | EIE3 | UOASW | ERTSW | PATRSW2 |
| 19 | EIE4 | NTUASW | CHASW1 | OPTICSW |
| 20 | EIESW2 | ERTSW | KORSW | KOLSW1 |
| 21 | YPEPTH3 | CHASW1 | HER2 | EIESW2 |
| 22 | R-GRIX | KORSW | GRIXSW | GRIXSW |
| 23 | ATH4 | HER2 | EIESW3 | YPEPTH3 |
| 24 | IONASW1 | LAR2 | FLEMSW | R-GRIX |
| 25 | TEIEP@ioaSW | EIE3 | HCMRSW | ATH4 |
| 26 | UOISW | GRIXSW | PASTSW | IONASW1 |
| 27 | THESSW1 | EIESW3 | TEIATH | TEIEP@ioaSW |
| 28 | SERSW | FLEMSW | AUASW | UOISW |
| 29 | THESSW2 | HCMRSW | ACRSW2 | THESSW1 |
| 30 | TEITHESW | PASTSW | EIE1 | SERSW |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Betweenness centrality | Closeness centrality | Eigenvector centrality | Total degree centrality |
|---|---|---|---|---|
| 31 | UOMSW | TEIATH | IOAN2 | THESSW2 |
| 32 | DRAMSW | AUASW | THES2 | TEITHESW |
| 33 | XANSW1 | ACRSW2 | EIE4 | UOMSW |
| 34 | XANSW2 | SYR | TEIMESSW | DRAMSW |
| 35 | TEILARSW | EIE4 | TEIPATSW | XANSW1 |
| 36 | LARSW1 | TEIMESSW | CTISW | XANSW2 |
| 37 | TEIMESSW | TEIPATSW | AGRINIOSW1 | TEILARSW |
| 38 | TEIPATSW | CTISW | PATRSW1 | LARSW1 |
| 39 | CTISW | AGRINIOSW1 | EIESW2 | TEIMESSW |
| 40 | AGRINIOSW1 | PATRSW1 | SYR | TEIPATSW |
| 41 | PATRSW1 | XAN2 | LAR2 | CTISW |
| 42 | PATRSW2 | THESSW1 | YPEPTH1 | AGRINIOSW1 |
| 43 | YPEPTHSW2 | SERSW | YPEPTH3 | PATRSW1 |
| 44 | HALSW | THESSW2 | ATH4 | YPEPTHSW2 |
| 45 | NAMUSSW | TEITHESW | RODSW | HALSW |
| 46 | AUEBSW | UOMSW | TEICRESW | NAMUSSW |
| 47 | ACADSW | YPEPTH1 | HERSW1 | AUEBSW |
| 48 | PARLTVSW | YPEPTH3 | R-GRIX | ACADSW |
| 49 | UOASW | IONASW1 | XAN2 | PARLTVSW |
| 50 | NTUASW | TEIEP@ioaSW | IONASW1 | UOASW |
| 51 | ERTSW | UOISW | TEIEP@ioaSW | NTUASW |
| 52 | KOLSW1 | ATH4 | UOISW | ERTSW |
| 53 | EIESW3 | RODSW | THESSW1 | EIESW3 |
| 54 | FLEMSW | TEICRESW | SERSW | FLEMSW |
| 55 | HCMRSW | HERSW1 | THESSW2 | HCMRSW |
| 56 | PASTSW | TEILARSW | TEITHESW | PASTSW |
| 57 | TEIATH | LARSW1 | UOMSW | TEIATH |
| 58 | AUASW | EIESW2 | APOLLASW | AUASW |
| 59 | ACRSW2 | R-GRIX | CHIOSSW | ACRSW2 |
| 60 | RODSW | APOLLASW | SYRSW1 | RODSW |
| 61 | TEICRESW | CHIOSSW | TEILARSW | TEICRESW |
| 62 | HERSW1 | SYRSW1 | LARSW1 | HERSW1 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

| Rank | Betweenness centrality | Closeness centrality | Eigenvector centrality | Total degree centrality |
|---|---|---|---|---|
| 63 | CHASW1 | DRAMSW | YPEPTH2 | CHASW1 |
| 64 | APOLLASW | XANSW1 | DRAMSW | APOLLASW |
| 65 | CHIOSSW | XANSW2 | XANSW1 | CHIOSSW |
| 66 | LIMNSW | LIMNSW | XANSW2 | LIMNSW |
| 67 | SYRSW1 | YPEPTH2 | LIMNSW | SYRSW1 |
| 68 | KORSW | YPEPTHSW2 | YPEPTHSW2 | KORSW |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix XI: Log Collection (rsyslog) Sizing

| Layer | Collector | Routers | | | | Switches | | | | Servers | | | | From Lower Layer | | | Server Requirements | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | # | EPS | Event Size (bytes) | Bandwidth | # | EPS | Event Size (bytes) | Bandwidth | # | EPS | Event Size (bytes) | Bandwidth | Collector | EPS | Bandwidth | Total EPS | Total Bandwidth (Mbps) | Memory-RAM (bytes) |
| Layer 2 | HER2 | 1 | 10 | 1,024 | 0.0819 | 3 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 125 | 0.4149 | 30,360 |
| | SYR | 1 | 10 | 1,024 | 0.0819 | 3 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 125 | 0.4149 | 30,360 |
| | LAR2 | 2 | 10 | 1,024 | 0.0819 | 2 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 130 | 0.4558 | 30,360 |
| | XAN2 | 1 | 10 | 1,024 | 0.0819 | 4 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 130 | 0.4559 | 30,360 |
| | IOAN2 | 1 | 10 | 1,024 | 0.0819 | 3 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 125 | 0.4149 | 30,360 |
| | YPEPTHSW1 | 3 | 10 | 1,024 | 0.0819 | 2 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 140 | 0.5377 | 30,360 |
| Layer 1 | KOL1 | 2 | 10 | 1,024 | 0.0819 | 12 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | YPEPTHSW1 | 140 | 0.5377 | 320 | 1.4035 | 60,720 |
| | EIE2 | 2 | 10 | 1,024 | 0.0819 | 8 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | | | | 160 | 0.7018 | 30,360 |
| | THES2 | 1 | 10 | 1,024 | 0.0819 | 6 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | LAR2, XAN2 | 260 | 0.9117 | 400 | 1.4496 | 91,080 |
| | PATR2 | 1 | 10 | 1,024 | 0.0819 | 6 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | IOAN2 | 125 | 0.4149 | 265 | 0.9528 | 60,720 |
| | EIESW1 | 2 | 10 | 1,024 | 0.0819 | 2 | 5 | 1,024 | 0.041 | 1 | 100 | 150 | 0.21 | HER2, SYR | 250 | 0.8298 | 380 | 1.2856 | 91,080 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix XII: Log Storage Sizing

| Layer | Collector | Routers | | | Switches | | | Servers | | | From Lower Layer | Server Requirements (Gbytes) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | # | EPS | Event Size (bytes) | # | EPS | Event Size (bytes) | # | EPS | Event Size (bytes) | Collector | 1 Day | Live (5 days) | Back up (5 days) | Archived (1 year) |
| Layer 2 | HER2 | 1 | 10 | 1,024 | 3 | 5 | 1,024 | 1 | 100 | 150 | | 3.2669305801 | 16.3346529007 | 16.3346529007 | 1192.4296617508 |
| | SYR | 1 | 10 | 1,024 | 3 | 5 | 1,024 | 1 | 100 | 150 | | 3.2669305801 | 16.3346529007 | 16.3346529007 | 1192.4296617508 |
| | LAR2 | 2 | 10 | 1,024 | 2 | 5 | 1,024 | 1 | 100 | 150 | | 3.6789178848 | 18.3945894241 | 18.3945894241 | 1342.8050279617 |
| | XAN2 | 1 | 10 | 1,024 | 4 | 5 | 1,024 | 1 | 100 | 150 | | 3.6789178848 | 18.3945894241 | 18.3945894241 | 1342.8050279617 |
| | IOAN2 | 1 | 10 | 1,024 | 3 | 5 | 1,024 | 1 | 100 | 150 | | 3.2669305801 | 16.3346529007 | 16.3346529007 | 1192.4296617508 |
| | YPEPTHSW1 | 3 | 10 | 1,024 | 2 | 5 | 1,024 | 1 | 100 | 150 | | 4.5028924942 | 22.514462471 | 22.514462471 | 1643.5557603836 |
| Layer 1 | KOL1 | 2 | 10 | 1,024 | 12 | 5 | 1,024 | 1 | 100 | 150 | YPEPTHSW1 | 12.3016834259 | 61.5084171295 | 61.5084171295 | 4490.1144504547 |
| | EIE2 | 2 | 10 | 1,024 | 8 | 5 | 1,024 | 1 | 100 | 150 | | 6.150841713 | 30.7542085648 | 30.7542085648 | 2245.0572252274 |
| | THES2 | 1 | 10 | 1,024 | 6 | 5 | 1,024 | 1 | 100 | 150 | LAR2,XAN2 | 11.8607282639 | 59.3036413193 | 59.3036413193 | 4329.1658163071 |
| | PATR2 | 1 | 10 | 1,024 | 6 | 5 | 1,024 | 1 | 100 | 150 | IOAN2 | 7.7698230743 | 38.8491153717 | 38.8491153717 | 2835.9854221344 |
| | EIESW1 | 2 | 10 | 1,024 | 2 | 5 | 1,024 | 1 | 100 | 150 | HER2,SYR | 10.2127790451 | 51.0638952255 | 51.0638952255 | 3727.6643514633 |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

## Appendix XIII: GRNET.SA Measurements and Metrics

| Measure ID | Measurement 1-1 | Measurement 1-2 | Measurement 2-1 | Measurement 3-1 | Measurement 3-2 | Measurement 4-1 | Measurement 4-2 |
|---|---|---|---|---|---|---|---|
| **Description** | Successful generation of log data | Log record fields validation. | Events successfully delivered | # of functional devices. | Ratio of up and down time | Log data format validation. | Log data time field validation |
| **Goal** | Objective-1 | Objective-1 | Objective-2 | Objective-3 | Objective-3 | Objective-4 | Objective-4 |
| **Measure** | Percentage | Percentage | Percentage | Percentage | Percentage | Percentage | Percentage |
| **Type of measurement** | Performance | Performance | Performance | Performance | Performance | Performance | Performance |
| **Formula** | # of events generated / # of events that should be generated | None. | # of events generated/ # of events delivered to the central points | # of functional devices/ total # of devices | Downtime / uptime | # of events successfully formated/ # of total events inspected | # of accurate events / # of total events inspected |
| **Method of measurement** | Feed the devices with traffic that should generate log records. | Force the generation of a log data and validate the content and the time data. | Count the number of events generated and the number of events centrally stored | Identify the malfunctioning devices and update the assets inventory. | Specify the up and down time of the devices | Inspect a sample of the collected log data and verify they are in the desired format. | Inspect a sample of the collected log data and verify the accuracy of the time field. |
| **Target** | 100% | 100% | 100% | 100% | 0% | 100% | 100% |
| **Frequency of measurement** | Every month | Every month | Every month | Daily | Daily | Weekly | Weekly |
| **Responsible Parties** | Security personnel | Security personnel | Security personnel | IT personnel | IT personnel | Security personnel | Security personnel |
| **Data source** | A sample of the log sources | A sample of the log files | A sample of the log sources | Inventory and network monitoring software | Network monitoring software | A sample of the log files | A sample of the log files |
| **Reporting format** | Bar chart | Bar chart | Bar chart | Bar chart | Pie chart | Pie chart | Pie chart |

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ