

UNIVERZITET SINGIDUNUM

**Departman za poslediplomske studije
Master studije**

**Marcel Banić
INCIDENT REPORTING AND
NETWORK EVENT ANALYSIS
Master rad**

Beograd, 2012.

UNIVERZITET SINGIDUNUM

**Departman za poslediplomske studije
Master studije**



**Marcel Banić
INCIDENT REPORTING AND
NETWORK EVENT ANALYSIS
Master rad**

**Mentor:
Doc. dr Gojko Grubor**

**Student:
Marcel Banić
Br. Indeksa 410497/2011**

Beograd, 2012.

UNIVERZITET SINGIDUNUM

Departman za poslediplomske studije Master studije



DEPARTMAN ZA POSLEDIPLOMSKE STUDIJE

MASTER STUDIJE

STUDIJSKI PROGRAM: Savremene informacione tehnologije

Kandidat: Marcel Banić

Broj indeksa: 410497/2011

Tema: INCIDENT REPORTING AND NETWORK EVENT ANALYSIS

Zadatak: The objective is to describe, events and event detection, logging infrastructure, regulatory requirements and the relationship between the regulatory requirements and standards of good practice in log management infrastructure.

Mentor:

Doc. dr Gojko Grubor

Beograd, 2012.

ABSTRACT

Internet is increasingly used in all aspects of our daily lives, the realization has emerged that application level security, in many situations, cannot provide the required level of security, therefore traffic analysis can easily help uncover information about the intrusion. Consequently, design question is whether to monitor applications and systems or combine application and system monitoring and perform traffic monitoring to correlate events. Tracing traffic at multiple levels and points could potentially provide more information about the intrusion characteristics. With the wide deployment of log monitoring, the analysis of the data collected from those deployments has become an important research area. It has been shown that data collected in such manner can be used to detect traffic anomalies and improve on intrusion detection. In practice, attacks should be detected as soon as possible in order to take appropriate corrective measures in timely manner.

KEYWORDS

Logging, compliance, network forensics, security event management, intrusion-detection, network flows

Contents

I. SCIENTIFIC RESEARCH METHODOLOGY	9
1.1 INTRODUCTORY REMARKS AND RATIONALE FOR THE RESEARCH	9
1.2 SUBJECT OF THE RESEARCH.....	10
1.3 PURPOSE AND OBJECTIVES OF THE RESEARCH.....	10
1.4 HYPOTHETICAL FRAMEWORK OF THE RESEARCH PAPER	11
1.5 METHODS, TECHNIQUES AND RESEARCH PROCESS	11
INTRODUCTION	13
DIGITAL FORENSICS	13
UBIQUITOUS COMPUTING.....	13
II. INVISIBLE TRAILS.....	14
2.1 ATTACKS.....	14
2.1.1 Types of Attacks.....	14
2.1.2 Types of Attacks.....	14
2.1.3 Interception	14
2.1.4 Interruption	15
2.1.5 Modification	15
2.1.6 Fabrication	15
2.2 ATTACK TAXONOMY.....	15
2.2.1 Threats, Vulnerabilities and Risk	16
2.3 IMPACT.....	16
2.4 CONTROLS.....	17
2.5 IMPLEMENTING VULNERABILITY DATABASE	17
III. IMPLEMENTING INFORMATION SECURITY	18
3.1 SECURITY CONTROLS	19
3.2 CIA TRIAD	20
3.2.1 Confidentiality	20
3.2.2 Integrity	20
3.2.3 Availability.....	20
IV. LOGS AND DATA SOURCES	20
4.1 EVENT	20
4.2 AUDIT.....	21
4.3 ERROR.....	21
4.4 INSTALLATION.....	21
4.5 DEBUGGING.....	21
4.6 ACTIONABLE AND NON- ACTIONABLE LOGS	21
4.6.1 Flagging Actionable Logs	22
4.7 DESIGNING LOG ARCHITECTURE.....	22
4.8 DETAILED ELEMENTS OF EVENT MANAGEMENT.....	23
4.8.1 Event Analysis	23
4.8.2 Event Reporting	23
4.8.3 Event Response	24
4.8.4 Incident Response and Escalation Procedures	24
4.8.5 Log Manager and Event Viewer.....	24
4.8.6 Structural and Behavior Lead Alerts	25
V. EVENT LOGGING ARCHITECTURE	26
5.1 COMPUTER SECURITY LOG MANAGEMENT.....	26

5.1.1 Architecture	26
5.1.2 Log Security	28
5.1.3 Log management infrastructure Functions	28
VI. STANDARDS IN INFORMATION SECURITY	31
6.1 ISO/IEC 27000 FAMILY.....	31
6.2 INDUSTRY GUIDANCE.....	32
6.2.1 COBIT	32
6.2.2 A2 ISO/IEC 20000.....	32
6.2.3 CMMI	32
6.2.4 Balanced Scorecard	33
6.2.5 ITIL.....	33
6.3 STANDARD OF GOOD PRACTICE	34
VII. LEGISLATION	34
7.1 BASEL IMPACT ON RISK MANAGEMENT	34
7.1.1 Use Test and Experience	35
7.1.2 Basel IT Implications	35
7.1.3 Link between Basel and Information Technology.....	36
7.2 SARBANES-OXLEY ACT (SOX)	37
7.2.1 Impact of Sarbanes-Oxley on IT operations.....	37
VIII. RISK AND MEASURES OF SECURITY	38
8.1 PRINCIPLES OF RISK MANAGEMENT	40
8.2 RISK FRAMEWORK.....	41
8.2.1 Risk Framework Objective	41
8.2.2 Components of a Risk Framework.....	42
IX. MODELING SECURITY	43
9.1 SERVICE-ORIENTED MODELING AND ARCHITECTURE	44
9.2 THE CONCEPT OF SERVICE-ORIENTATION	44
9.3 THE BUSINESS DRIVERS FOR A NEW APPROACH.....	45
9.4 SECURITY ENGINEERING AND PROGRAMMING IN SOA.....	45
9.5 APPLICATION BLACK BOX	46
9.6 RISK-BASED SECURITY TESTING: TESTING WITH THREAT MODELING.....	46
9.6.1 Threat Risk Modeling.....	47
9.6.2 The STRIDE Threat Model	48
9.6.3 DREAD Classification	49
9.6.4 Trike Modeling	50
9.6.5 AS/NZS 4360:2004 Risk Management.....	50
9.6.6 CVSS	51
9.6.7 OCTAVE	51
9.6.8 OWASP Guide Project	51
X. DIGITAL FORENSICS AND INTELLIGENCE GATHERING	52
10.1 INTELLIGENCE ANALYSIS.....	52
10.1.1 Strategic Intelligence Analysis	53
10.1.2 Tactical Intelligence Analysis	54
10.1.3 Operational Intelligence Analysis.....	54
10.2.3 Intelligent Distribution Analysis Using Analyzers	55
XI. NETWORK FORENSICS	56
11.1 CATEGORIZING THE DATA.....	56

11.2 STORING LOGGED EVENTS	58
11.3 TIMESTAMPS	59
11.3.1 <i>Analyzing Time Stamps</i>	59
11.3.2 <i>Challenges In Correlating Network Events</i>	59
11.4 LOGS FILTERS AND ARCHIVERS.....	59
11.4.1 <i>What is Event Analyzer?</i>	59
11.4.2 <i>What Problems do Event Analyzers Solve</i>	60
11.4.3 <i>Features of an Event Analyzers</i>	60
11.4.4 <i>Overall Benefits of an Event Analyzer</i>	60
XII. BEYOND LOGGING AND MONITORING	61
12.1 SECURITY EVENT MANAGEMENT SOFTWARE	61
12.2 SECURITY INFORMATION AND EVENT MANAGEMENT SOFTWARE	62
12.3 NETWORK FORENSIC ANALYSIS TOOLS	62
12.4 ACQUIRING NETWORK TRAFFIC DATA.....	63
12.4.1 <i>Technical Issues of Data Acquisition</i>	63
12.5 ALTERNATE ACCESS POINTS AND COVERT CHANNELS	64
12.5.1 <i>Alternate Access Points</i>	64
12.5.2 <i>Covert Channels</i>	65
12.6 COMBINED LOG MANAGEMENT AND SIEM ARCHITECTURE BENEFITS.....	65
CONCLUSION	66
APPENDIX A	67
USING SYSLOG FOR VALIDATING SECURITY POLICY COMPLIANCE	67
1.1 CONFIGURING NETWORK DEVICES.....	67
1.1.1 <i>Microsoft Windows Host as syslog Client</i>	68
1.1.2 <i>Cisco Routers as Syslog Client</i>	69
1.1.3 <i>Linux Hosts as Syslog Clients</i>	70
1.2 ARCHIVING	70
CONCLUSION	71
APPENDIX B	72
NTP, THE ALTERNATIVE APPROACH TO MONITORING	72
1.1 WHAT NTP DOES AND HOW IT WORKS.....	73
1.2. INSTALLING AND RUNNING NTP ON LINUX HOST.....	73
1.2.1 <i>Ubuntu</i>	73
1.2.2 <i>Configuring and Compiling ntp Under RHEL</i>	73
1.2.3 <i>Additional Packages for ntp Functionality Support:</i>	75
1.2.4 <i>Ntp Command Line Configuration</i>	75
1.2.5 <i>Creating a Traffic Filter</i>	76
1.2.6 <i>Suspicious Packet Tracking</i>	76
1.2.7 <i>Refresh Times</i>	77
1.2.8 <i>Setting Interfaces to Work Without Promiscuous Mode</i>	77
1.2.9 <i>Running More Than One Instance of ntp</i>	77
1.3 CONFIGURING PLUGINS AND INTEGRATING SOFTWARE.....	78
1.3.1 <i>Ntp IDS</i>	78
1.3.2 <i>Summary of Aberrant Behavior Detection Algorithm</i>	78
1.3.3 <i>Netflow Plugin</i>	80
1.3.4 <i>Nfdump</i>	81
1.4 CONFIGURING NTP TO USE SYSLOG.....	82
1.5 VIEWING NTP STATS	82

CONCLUSION	85
REFERENCES.....	89

List of Figures

Figure 1 Categories of Attack	14
Figure 2 The Elements of the Risk Treatment Cycle.....	42
Figure 3 Example Configuration of Windows Host	69
Figure 4 Ntop Abberant Behavior Graph.....	79
Figure 5 Ntop Flow Graph.....	80
Figure 6 Ntop Netflow Statistics.....	81
Figure 7 Ntop Global TCP/UDP Protocol Distribution Graphs	83
Figure 8 Ntop Network Load Statistics	84

I. SCIENTIFIC RESEARCH METHODOLOGY

1.1 Introductory Remarks and Rationale for the Research

Management of an organizational network is a challenging task, even more if the network is connected to, or in part, the Internet. One of the reasons is the complexity of the Internet itself, characterized by the interconnection of different devices and device types operated in numerous network domains and managed by diverse network management authorities.

While Internet protocols have been designed to cope with such environments, deliberate attacks against the network infrastructure, unpredicted failures and unintended misconfigurations are reoccurring problems. Therefore, a best-effort network service in the early days of the Internet is not acceptable to the more important functions interconnected networks have these days. As a consequence, managers have to monitor and supervise networks permanently in order to guarantee desired network functions, detect and mitigate degradations of network services in timely manner.

An important function of many complex networks is to inhibit the transmission of unwanted traffic, make resources available, or stop perpetrators from gaining access to information and resources. In part, little attention is given to the individual-level interactions that affect these networks and hence constrain their functions. Systems are needed for linking unique interactions or behavior to network level functions to enable integration of multiple data sources of information to collectively make decisions, and allocate tasks and resources for optimal operation.

In the first part, the basic data about the methodology described in the research project is the subject of the research, objectives and tasks of the research, a hypothetical research framework, methods, techniques and course of the research process.

In the second part, paper describes log management solutions and trails that can be utilized to detect and conduct detailed forensic analyses of cyberattacks, regulatory audits and assist in improvement of network services.

The third part, case studies, reviews open source “syslog” tiered implementation, and “ntop” network traffic monitoring and event correlation tool, its installation, configuration, plugin integration and network behavior configuration to provide network analysts with better overview of network activity

1.2 Subject of the Research

The first part explains the research methodology of the research subject, objectives and tasks of the research, a hypothetical research framework, methods, techniques and course of the research process.

The subject in the first part is risk that organizations embrace resulting from the incorporation of information systems where organizations feel information technology offers the opportunity for greater reward. Organizations seek to minimize identified risks and will attempt to reduce their exposure to risk to least possible level through the implementation of a risk management framework into business processes of an organization.

The subject of the research includes the adaptation of common use of intelligence and information gathering in order to improve defense and consistent approach for information gathering and management processes for information systems used in security services.

Second part researches types of events, how and for what purpose they get recorded, and how are recorded events tied to different log types, typical event management process, recommendations for logging infrastructure, and implementations of Security Event Management systems, and its integration with log management infrastructure.

1.3 Purpose and Objectives of the Research

The main purpose of this paper is to demonstrate the importance of corporate logging infrastructure to support risk mitigation and forensic investigations in response to network-related incidents.

The specific objective is to describe logging infrastructure in accordance with regulatory requirements and the relationship between the regulatory requirements and standards of good practice in log management infrastructure.

The specific objective 2 is to show logging infrastructure, log review process and logged event correlation in order to support forensics of intrusion-detection.

1.4 Hypothetical Framework of the Research Paper

General hypothesis:

Organizations network security and management is usually influenced by the following:

- Regulatory compliance
- Risk, its treatment, and intelligence gathering for future risk mitigation
- Protection of organizations resources for continuity
- Emerging new tools for network-related forensics

While current practices in logging events for infrastructure related events give good insight into possible intrusions, misconfigurations and failures, there is a need for more effective approach to conduct network related forensic investigations.

Working Hypothesis

The working hypothesis is network incident and log analysis to detect attacks and abuse by a perpetrator in order to obtain access to exploit organizational resources for their own benefit, and in what way organizational security systems support detection of network intrusion.

1.5 Methods, Techniques and Research Process

Scientific method is a body of techniques for investigating phenomena, acquiring new knowledge, or correcting and integrating previous knowledge. To be deemed scientific, a method of inquiry must be systematic, critical, controlled and repeatable process of acquiring new skills necessary for identifying, defining and solving scientific problems.

Scientific method, the method employed in exact science consists of:

- Careful and abundant observation and experiment.
- Generalization of the results into formulated statements.

On the basis of the division of scientific research, in this research have been applied mostly fundamental and applied research principles. Fundamental research in order to acquire new knowledge about the area, and it is applied in order to acquire new knowledge and solve hypothesis set in this scientific field.

In order to satisfy the basic methodological requirements in the course of the study this research used different research methods and verification processes in order to collect information about organizations, regulatory compliance, log collection, and intrusion detection; to acquire knowledge and present applied security event management and network-related forensics. First step was collection of literature on the methodology of scientific research and security event and log management from variety of sources; afterwards, the research literature, through the stages of discovery, describes and explains in systematic and structured way the research results.

INTRODUCTION

In modern network analysis, decreasing storage costs and increasing computing capabilities allows many products to generate huge volumes of deep packet data. But in practice, analysts still struggle with translating this raw data into knowledge to guide them in decision making. In large network environments flow of data helps provide a scalable way of seeing the big picture, as well as a streamlined platform for highlighting patterns of malicious behavior over time.

Digital Forensics

Digital forensics is the process of uncovering and interpreting electronic data for use and in the process preserves any evidence in its most original form while performing a structured investigation by collecting, identifying and validating the digital information for the purpose of reconstructing past events.

The evidential nature of digital forensic science requires following of rigorous standards across the spheres during search for a perpetrator of an event.

Ubiquitous Computing

Information security is a concept that becomes ever more present in many aspects of our society, largely as a result of our nearly ubiquitous adoption of computer technology. In our everyday lives, majority of people works with computers for employers, use computers at home, go to school online, buy merchandise over the Internet, take laptops to the nearest restaurant, and check emails from public places via smart phones.

In a digitally recorded world, devices create records of past events, logs of Internet activity accumulates variety of data into a sort of a digital underworld.

Trails of electronic evidence (*e-evidence*) are *left* by workstations, cell phones, PDAs, e-mails, social networks, Web sites, and every wireless on-line activity. The amount of information left traversing through the networks is the basic reason why criminals can exploit someone's presence on-line, but it is also the reason why they get caught.

Digging up data to expose past events is a primary purpose of computer forensics. Devices that people use create and capture large amounts of detail that is stored in more places than realized.

Computer technology enables people to be more productive and allows access to a variety of information, with a click of the mouse, but it also carries with it many security issues. If the information on a system used by employer or a bank is exposed to an attacker, the consequences can be dire, and sensitive data can be lost to the hands of an attacker.

II. INVISIBLE TRAILS

2.1 Attacks

We may face attacks from a wide variety of approaches and angles. When we look at what makes up an attack, we can break it down according to the type of the risk the attack represents, and the controls used to mitigate it.

2.1.2 Types of Attacks

When we look at the types of attacks we might encounter, we can place them into one of four categories: interception, interruption, modification, and fabrication. Each category can affect one or more of the principles of the *Confidentiality Integrity and Availability* (CIA) triad, as shown in figure 1. Additionally, the lines between the categories of attacks and their affects are blurry. Depending on the attack in question it can be concluded if it belongs to one or more categories, or has more than one affect.

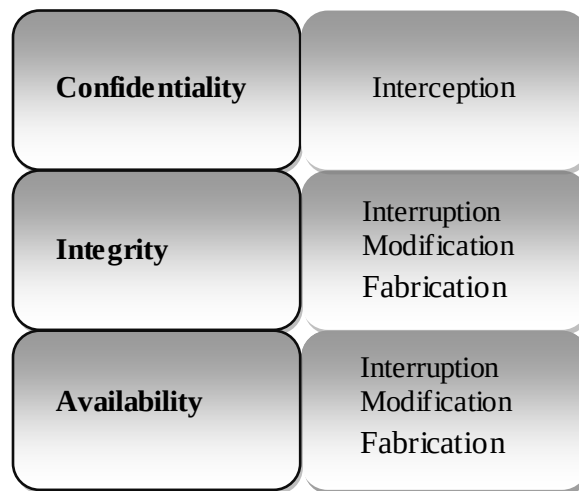


Figure 1 Categories of Attack

2.1.3 Interception

Interception attacks allow unauthorized users to access data, applications, or environments, and are an attack against confidentiality of the information. Interception can take the form of unauthorized file viewing, copying or reading email, and can be conducted against data at rest or at data in motion. Properly executed, interception attacks can be very difficult to detect.

2.1.4 Interruption

Interruption attacks cause assets to become temporary or permanently unusable. Interruption attacks affect availability but can be an attack on integrity of data as well. In the case of a DoS attack on a mail server, we can classify this as an availability attack. In the case of an attacker manipulating the processes on which a database runs in order to prevent access to the data it contains, we might consider this an integrity attack due to the possible loss or corruption of data, or we might consider it a combination of both types of attack. We might also consider such a database attack to be a modification attack instead of an interruption attack.

2.1.5 Modification

Modification attacks involve tampering with assets. These types of attacks are primarily considered as integrity attacks, but could also represent an availability attack. If we are able to access a file and its content in an unauthorized fashion, we have affected the integrity of the data contained in the file. If we consider the case where the file in question is a configuration file managing a particular service, perhaps a service that is acting as Web server, we might affect the availability of that service by changing the contents of the file. If we continue with this concept and say the configuration file we altered is one that alters how the server deals with encrypted connections, we can make this a confidentiality attack.

2.1.6 Fabrication

Fabrication attacks involve generating data, processes, communications, or other activities with the system. Fabrication attacks primarily affect integrity but could also be considered an availability attack as well. If we generate spurious information in a database, this would be considered to be a fabrication attack. Generating e-mail, which is commonly, used method for propagating Malware used to spread a worm. In the sense of an availability attack, besides it being fabrication attack, actions such as generating additional processes, network traffic, e-mail, Web traffic, or nearly anything else that consumes resources, rendering the service that handles such traffic unavailable to legitimate users of the system.

2.2 Attack Taxonomy

System state is to be defined according to the relative impact that an intrusion can have for each type of consequence. As such an approach is very subjective; as such work must be performed using a standard method for attack taxonomy and classification.¹ However the lack of definitions (being a list of terms or matrices-based) that meet the six characteristics of an acceptable taxonomy,² enforces an arbitrary choice. It is possible to use a specific definition or, to rely on an external source such as “unauthorized results.”³

2.2.1 Threats, Vulnerabilities and Risk

In order to be able to speak more specifically about attacks, we need to look at potential for particular attack and the way it affect us, we can speak of it in terms of threats, vulnerabilities, and the associated risks that accompany those attacks.

Threats

When we speak about types of attacks we encounter, we refer to their potential to cause harm to our assets. Ultimately, threat is something that has the potential to cause harm to our assets. Threats tend to be specific to certain environments, particularly in the world of information security. Let examine a virus that might be problematic on a Windows operating system. The same virus will have no effect on a Linux operating system.

Vulnerabilities

Vulnerabilities are weaknesses that can be used to harm our assets. In vulnerabilities are holes that can be exploited by threats in order to cause harm. A vulnerability might be a specific operating system or application that is running, a physical location of our office building, a data center that is populated over capacity of its air conditioning system, a lack of backup generators, or other factors.

Risk

Risk is the likelihood that something bad will happen. In order to have a risk in a specific environment, it is needed to have both a threat and a vulnerability that the specific threat can exploit. Suppose that the structure is made of wood and we set it on fire. Instantly we have both. Threat (the fire) and a vulnerability that matches it (the wood structure). Definitely we have the case of high risk. On the other hand, the same threat of fire with the structure made of concrete will no longer have credible risk. The threat has no vulnerability to exploit.

In computing environments potential risks are attacks that could happen. The best strategy is to spend time mitigating the most likely attacks. If we sink resources into trying to plan for every conceivable attack, we might face the possibility to spread ourselves thin and lack the protection where we need it the most.

2.3 Impact

Organizations such as the *U.S. National Security Agency* (NSA) add an additional factor to the threat/vulnerability/risk equation, in the form of impact. If we consider the value of the asset being threatened to be a factor, it will change whether we consider a risk as being present or not. In general, a backup tape containing encrypted recipe for chocolate cookies may not

actually have a risk. The data being exposed would not cause us a problem. No sensitive information would be revealed and we can make additional backups from the source data. In this case we can say that we do not have a risk.

2.4 Controls

In order to mitigate risk, measures can be put in place to ensure that a given type of threat is accounted for. These measures are referred to as controls. Controls are divided into three categories: physical, logical, and administrative.

- **Physical** controls protect physical environment with our systems, or where our data is stored.
- **Logical** controls, called technical controls, are those that protect the system, network, and environments that process, transmit, and store data. Logical controls can include items such as passwords, encryption, logical access controls, firewalls, and intrusion detection systems. Logical controls enable, in logical sense, prevention of unauthorized activities taking place.
- **Administrative** controls are based on rules, laws, policies, procedures, guidelines, and other items that are “paper” in nature. In essence, administrative controls set behavior rules expected from users and the environment. Depending on the environment and control in question, administrative controls can represent differing levels of authority. One more concept in administrative controls are the abilities to enforce compliance with given controls. If we do not have the ability to ensure that controls are being implemented, then they are useless because they create false sense of security.

2.5 Implementing Vulnerability Database

The vulnerability database holds information about security breaches and insecure behavior that would impact the overall security level or that could be exploited by an attacker in order to perform an intrusion.

The database format makes it possible to include the following types of vulnerabilities:

- **Structural**, vulnerabilities that are internal to specific software such as a buffer overflow, format string, race conditions, and other. This part of the database is the easiest to implement, feed and maintain. The majority of these processes can be scripted from widely available public sources such as software advisories and web sites. However, when using multiple sources for feeds (advisories, web sites, etc.) validation and correlation steps should be performed by a experts team.

- **Functional** vulnerabilities depend on configuration of operational behavior, users, etc. These vulnerabilities differ from the previous as they depend on the environment in which they are implemented. For example, an NFS mount can be considered a functional vulnerability from the fact that an intruder can get into an account or a host allowing them to mount the file system. Therefore, it can be assumed that much vulnerability are present on systems but can be considered as “inactive” as long as at least one of the needed conditions is not satisfied. The hardest part is the definition and formatting of those types of vulnerabilities and feeding them to the database. The need for expert teams in each field such as applications, Operating System, network is obvious.
- **Topology-based** vulnerabilities include intrusion impact on the networking and consequences of such. This part of the database should include network-based vulnerabilities such as sniffing, spoofing as well as the impact of filters on the intrusion path. Network vulnerabilities cannot fit into a vulnerability database unless it supports topology modeling.

III. IMPLEMENTING INFORMATION SECURITY

Information security is defined as “protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction.” In simplest terms, information security means that we want to protect our data and our systems from those who are looking to misuse them.

In general security means protection of assets. This may mean protecting assets from attackers invading networks, theft or vandalism, power failures, adverse environmental conditions, and other undesirable events. Ultimately, security is an attempt to secure assets against most likely forms of attacks, to the best extent we can, given our environment.

In ISO 27001:2005, “Information technology - information security management systems – requirements”, one of the four key process approaches encourages its users to emphasize the importance of implementing and operating controls to manage an organization's information security risks. Information security risks must be managed in the context of the organization's overall business risks. Risk is approached in a number of ways and the approach will be dependent on the type of organization and its willingness to accept levels of risk.

Risk is fundamental to all organizations, and all of them are constantly exposed to, and have to deal with, a range of different types of risk. Organizations embrace risk where they feel it offers the opportunity for greater reward, while other organizations are more conservative and are risk averse. Organizations seek to minimize identified risks and will attempt to reduce their exposure to risk to least possible level.

3.1 Security Controls

Security controls are safeguards or countermeasures to avoid, counteract or minimize security risks. Defining and assessing risk helps review and design controls, they can be classified by several different criteria, for example according to the time that they act, relative to a security incident:

- Before the event, **preventive controls** are intended to prevent an incident from occurring by locking out unauthorized intruders.
- During the event, **detective controls** are intended to identify and characterize an incident in progress by sounding the intruder alarm and alerting the security professionals.
- After the event, **corrective controls** are intended to limit the extent of damage caused by the incident by recovering the organization to normal working status as efficiently as possible.

Security controls can also be categorized according to their nature, for example:

- **Physical controls**: fences, doors, locks and fire extinguishers.
- **Procedural controls**: incident response, management oversight, awareness and training.
- **Technical controls**: user authentication and logical access controls, anti-virus, firewalls.
- **Legal and regulatory or compliance controls**: privacy laws, policies and clauses.

A similar categorization distinguishes control involving people, technology and operations/processes.

Information security controls protect the **confidentiality**, **integrity** and **availability** of information (the so-called *CIA Triad*), depending on how narrowly or broadly the *CIA Triad* is defined.

Risk-aware organizations choose pro actively to specify, design, implement, operate and maintain their security controls, usually by assessing the risks and implementing a security management framework such as ISO/IEC 27002, the Information Security Forum's Standard of Good Practice for Information Security and NIST SP 800-53. Organizations may also demonstrate the adequacy of their information security controls by being independently assessed against certification standards such as ISO/IEC 27001.

3.2 CIA Triad

3.2.1 Confidentiality

Confidentiality is a concept similar to privacy. Confidentiality is a necessary component of privacy, and refers to our ability to protect data from those who are unauthorized to view it. Confidentiality may be implemented at many levels of a process.

3.2.2 Integrity

Integrity is the ability to prevent data from being modified in an unauthorized or undesirable manner. This means that the unauthorized change or deletion of data or portion of data, or it could mean an unauthorized, or undesirable, change or deletion of data. To maintain integrity it is necessary to have the means to prevent unauthorized changes to data, and the ability to reverse authorized changes that need to be undone.

3.2.3 Availability

Availability refers to the ability to access data when needed. Loss of availability can refer to a wide variety of breaks in the chain that facilitates access to data. Such breaks are the result of power loss, operating system or application issues, network attacks, or other problems. When such breaks in the chain of access are caused by an outside party, such as an attacker, they are referred to as *Denial of Service* (DoS) attack.

IV. LOGS AND DATA SOURCES

Most applications (aside some simple ones) record information to logs, or perform some type of logging. An application records logs to an *Operating System* (OS) specific log, event log, a text file, a database, or a proprietary log format. Many applications record different types of events to different logs. Common types of log entries can be divided into following categories:

4.1 Event

Event log entries typically list actions that are performed, date it, and the record results of each action. Examples of recorded events are for example establishing a connection to another system and issuing administrator-level commands. Event log entries might also record information, such as what username was used to perform actions and what status code was returned that provides more information on the result.

4.2 Audit

Audit log entries, known as security entries, are information pertaining to audited activities, such as successful and failed logon attempts, security policy changes, file access, and process execution. Applications may use audit capabilities built into the operating system or work with their own auditing capabilities.

4.3 Error

Many applications create error logs, and record information regarding application errors, usually with timestamps. Error logs help in troubleshooting both operational problems and attacks. Error messages can be used in determining when an event of interest occurred and help in identifying various characteristics of the event.

4.4 Installation

Applications usually create a separate installation log that records information of the initial installation and subsequent updates. Information recorded in an installation log varies from application to application, but most of the time includes the status of phases of the installation, and many times also indicates the source of the installation files, locations of the application components, and options of the application's configuration.

4.5 Debugging

Many applications can be run in a debugging mode, at which time applications log far more information than usual detailing the operation of the application. Debugging records are most of the time cryptic and only have meaning to the software developer, who deciphers error codes and other information captured in the log. When an application has a debugging capability, it is typically enabled by an administrator or developers for specific problem resolution.

4.6 Actionable and Non-actionable Logs

Log messages should be clear and actionable notifications for administrators and staff. Many IT managers try to limit the amount of captured logs because they are non-actionable or because there isn't enough staff to correctly deal with all of the notifications.

4.6.1 Flagging Actionable Logs

To support proactive network and problem management, after any kind of major incident logs need to be reviewed and analyzed to determine if there was a specific pattern of logs to monitor for in the future. In this way, when similar issue occurs, it can be pro-actively detected, and impact will be minimized.

4.7 Designing Log Architecture

To harnessing the full potential of logging architecture and log entries a streamlined process is required supported by an underlying tool structure providing for the collection, analysis, and action on all received log messages. Majority of the recommendations in this type of event management architecture are applied to general event management.

In June 2007, *Information Technology Infrastructure Library V3* (ITIL V3) introduced a new "event management" process that conforms to the centralized log architecture. ITIL V3 describes an event as "any detectable or discernible occurrence that has significance for the management of the IT infrastructure or the delivery of IT services; and evaluation of the impact that a deviation might cause to the services".

It can be summarized as:

- Real-time infrastructure monitoring listening for events that are "bad".
- Event correlation to filter duplicate and combine individual events detecting serious issues.

It addresses the following key elements:

- Event analysis
- Event reporting
- Event remediation
- Event viewer
- Event logging architecture

Event Analysis outlines necessary steps to ensure review and classification of messages.

Event Reporting suggests approaches for managing the initial large volume of messages requiring classification. When complete, the maintenance becomes very manageable.

Event Remediation addresses the critical message response.

Event Viewer outlines the tools necessary to collect, store, and classify messages. There are tools available today that can be customized to the environment providing ticket generation.

Event Logging Architecture addresses how to deploy logging in a small, medium-sized, or large deployment environment.

A good logging platform categorizes log messages, establishes normal operational baseline, and ensures that important messages are recognized and acted upon, organizes messages and ensures that messages of importance are given the required attention.

4.8 Detailed Elements of Event Management

4.8.1 Event Analysis

It is important that only the critical and relevant messages are identified to make any analysis worthwhile. When reviewing an event there are two questions that need to be answered:

- Has this event been seen before?
- Is this an actionable or non-actionable event?

For all incoming events a database should be set up with fields indicating whether or not this is an actionable or non-actionable event. For events marked as non-actionable that have been seen before no further actions need to be taken, but if the event has not been seen before it needs to be further reviewed and, at that time, either marked as non-actionable or acted upon.

With prioritizing incoming events based on the sending source a "metric" can be applied to indicate host or device priority in comparison to another host or device of the same type in a different location. As this kind of implementation matures, fewer non-critical events will be seen by the personnel, and marking only significant events in need of attention.

This type of approach provides support for forensic analysis in the problem management, and for security and non-security-related issues. In the setup as described above the analysis of logs may show some symptomatic causes of the problem. These symptomatic messages can then be used as a list of known logs and relevant actions that can be defined for proactive identification of a problem in the future.

4.8.2 Event Reporting

In network environments today, due to their size, it is almost impossible to view every event coming into the management systems. This is especially true during the initial deployment of the event analysis, since all incoming events will be displayed as critical until they are sorted out and marked appropriately.

The simplest methods to understand the nature of event logs is to produce daily log report that can be used by the management team to eliminate potential problems on the network.

This method can include a mix of reports such as:

- Number of log occurrences sorted by severity
- Top number of log messages by severity
- Top number of devices generating logs by severity

With reports that generate trend reports and problem events utilizing intelligent analysis methods and event correlation it is possible to establish device health metric system and baseline management technique.

These analyses can aide engineers to collaborate by creating a set of dynamic log rules that can be applied to all log messages generated on the network. This type of approach can provide insightful view of network health and network optimization possibilities from a larger base of network expert knowledge.

Custom reports can include:

- Number of log occurrences broken down by a problem.
- Reports on each of the top-n unhealthy devices possibly including problem description and recommended action.

4.8.3 Event Response

Event response consists of problem ticket opening against all critical and actionable messages. The problem management team should be responsible for ticket tracking. Various devices contain details for each of the logged messages containing recommended actions. These messages and recommended actions can be used as a starting point for troubleshooting an issue and identifying the root cause.

4.8.4 Incident Response and Escalation Procedures

Eventually, reacting appropriately to an attack are organizations procedures that need to be enacted by the incident response teams.⁴ Reactions range from passive monitoring for further information through to target system or possibly emergency halt through CERT incident reporting.⁵ Appropriate reaction is determined before an attack takes place and procedures must be validated, then stored and made accessible to supervision teams.

4.8.5 Log Manager and Event Viewer

Log management tool is a major contributor to the success of an effective log management solution. The Event Viewer serves as a "one-stop shop" for correlated event lists, sorted by location and severity for the management team. Some general high-level requirements for this tool should be provided to maximize the tool's scalability and robustness.

Some of the general features are outlined below:

- The tool should be based on a web front-end that allows easy access to all support groups within the organization.
- The tool needs to include a standard user authentication system that be used in the incident or problem ticket generation.
- The tool should be able to identify duplicate (reoccurring) log messages. This feature helps to simplify the *Graphical User Interface* (GUI) and makes it more usable for support groups.
- The tool should incorporate severity reports. These reports are useful in identifying "chatty" devices, which in turn helps provide better reports and a better visual breakdown of the messages.

4.8.6 Structural and Behavior Lead Alerts

The main operations performed that generate alerts are the following: correlation, structural analysis, intrusion path analysis and behavior analysis.

Correlation is a stand-alone operation leading to the creation of contexts against which further analysis will be made, in order to check if they match the characteristics of an intrusion.

Structural analysis is compared to an advanced pattern matching process, used to determine if events stored within a certain context lead to a known intrusion path or attack tree.⁶

Intrusion path analysis is the next step where the output will provide information about the exposure of the target system to the intrusion detected.

Then, the **behavior analysis** will integrate elements of the security policy in order to determine if the intrusion is allowed or not.

The purpose of such operations is to generate alerts that match the structural path of intrusion (scan, fingerprinting, exploiting, back-dooring & cleaning), and take care of the security policy defined, as well as the state of targets systems.

- **Real-time monitoring** interfaces provide raw data in a message form. This allows basic filtering functions in order to isolate specific messages used for debugging, in-depth analysis, of specific events and replay of events.
- **Incident handling** is the internal engine used for generation and follow-up of incident tickets and reaction procedures. It provides alert information as well as numerous debugging data and checkpoints. It is the more complex and it must be within operational performance, advanced filters, research and identification functions. Such an engine is the basics of a timely and appropriate human reaction to intrusions.

- **Statistical analysis** provides raw data of security activity statistics over short, medium and long periods of time. It is mainly used as an under-layer for graphical event representation.

V. EVENT LOGGING ARCHITECTURE

5.1 Computer Security Log Management

A log management infrastructure consists of the hardware, software, networks, and media used to generate, transmit, store, analyze, and dispose of log data. Majority of organizations have one or more log management infrastructure⁷.

The *National Institute of Standards and Technology (NIST)*, *NIST SP800-92* guide, describes typical architecture of a log management infrastructure and its components interaction with each other. It describes the basic functions performed within a log management infrastructure and examines the two major categories of log management software; syslog-based centralized logging software and security information and event management software.⁸

5.1.1 Architecture

A log management infrastructure typically comprises of three tiers:

- **Log Generation Tier.** The first tier contains the hosts that generate the logs. Hosts either run logging client applications or services that make their log data available log servers in the second tier.
- **Log Analysis and Storage Tier.** The second tier is comprised of one or more log servers that receive log data from the hosts in the first tier. The data is transferred to the servers. Servers that receive log data from multiple log generators are sometimes called collectors or aggregators. Log data may be stored on the log servers themselves or on separate database servers.
- **Log Monitoring Tier.** The third tier contains consoles that are used to monitor and review log data, or the results of automated data analysis, and can be used to generate reports. In some log management infrastructures, consoles can be used to provide management for the log servers and clients.

The second tier (log analysis and storage) can differ in complexity and structure. The simplest arrangement is a single log server that handles all log analysis and storage functions. Examples of more complex second tier arrangements can be:

- Multiple log servers that perform specialized functions such as one server performing log collection, analysis, and short-term log storage, and another server performing long-term storage.
- Multiple log servers that each analyze and store logs for select log generators. This type of setup can also provide a degree of redundancy. A log generator can switch between log servers if its primary log server becomes unavailable. At the same time log servers can be configured to share log data with each other, which is also a type of redundancy.
- Design where two levels of log servers (sometimes called log catchers) are setup where the first level of distributed log servers receives logs from the log generators and forwards the log data they receive to a second level of centralized log servers. This can be done to protect the second level of log servers from attacks, and is useful when there are network reliability security concerns between the log generators and the second level of log servers, such as log servers being accessible only over the Intranet. In that case it is desirable to have log caching servers which can then transfer the logs to the second level log servers.

Communication between log management infrastructure components usually occurs over an organization's network because the hosts generating log data are most likely located throughout the organization's networks. In some instances it is desirable to use a physically or logically separate logging network, particularly for getting logs from key devices such as firewalls and network intrusion detection systems or systems that create large amounts of log data, or for transferring log data between log servers. During a network-based attack, regular networks might be unstable or unavailable. One of the motivations for a separate logging network is to protect log data from eavesdropping. If a separate logging network is not used, logging communications on a regular network can be protected using security controls such as data encryption. Benefit of having a separate logging network is protecting components that are only used for log management from attacks.

A log management infrastructure also needs to accommodate hosts with intermittent or low-bandwidth connectivity, such as mobile hosts and hosts connecting through dial-up modems. In log management infrastructure, there might be some log-generating hosts that cannot automatically participate in the infrastructure, such as computers that are not networked, legacy systems, and appliance-based devices that cannot be configured to transfer their logs to the log servers. Their log data needs to be incorporated into the log management infrastructure manually.

An organization can have a single log management infrastructure for the entire enterprise, yet it is more likely to have multiple infrastructures that do not necessarily interoperate. Single log management infrastructure provides a single point for reviewing of all the log data, but for a larger organization the size of such an infrastructure and the volume of data makes it impossible. Larger organizations usually use multiple log management infrastructures. Each infrastructure is dictated by many factors, including the organization's internal structure, system types, log types, and facility locations.

5.1.2 Log Security

As the security of logs is the concern, improvements in the implementations of syslog have been created that include emphasis on security.³⁰ Based on a proposed standard, *Request For Comment* (RFC 3195), which was designed specifically to improve the security of syslog. Implementations can support log confidentiality, integrity, and availability through several features, including the following:

- **Reliable Log Delivery.** Several syslog implementations support the use of *Transmission Control Protocol* (TCP) in addition to *User Datagram Protocol* (UDP). With using TCP, a connection-oriented protocol, reliable delivery of information across networks is insured. Using TCP requires the use of more network bandwidth; also, it takes more time for log entries to reach their destination.
- **Transmission Confidentiality Protection.** RFC 3195 recommends the use of the *Transport Layer Security* (TLS) protocol to protect the confidentiality of transmitted messages only protects the payloads, not the IP headers, which means that the source and destination of transmitted syslog messages can be identified revealing the IP addresses of the syslog servers and log sources. It is possible in some implementations to use *Secure Shell* (SSH) tunnels for passing messages.
- **Transmission Integrity Protection and Authentication.** RFC 3195 recommends that if integrity protection and authentication are desired, that a Message Digest (MD5) algorithm be used. RFC 3195 recommends the use of MD5. Proposed *Secure Hash Standard* (SHS) mentions the use of *Secure Hash Algorithm 3* (SHA-3).

5.1.3 Log management infrastructure Functions

Log management infrastructures typically perform several functions that assist in the storage, analysis, and disposal of log data. These functions are normally performed in a way that do not alter the original logs.

The following items describe common log management infrastructure functions:

General

- Log parsing extracts data from a in a way that the parsed values can be used as input for another logging process. An example of parsing is reading a text-based log file that contains comma-separated values, and extracting the values from each line. Parsing is performed as part of other logging functions, such as log conversion and log viewing.
- Event filtering is the suppression of log entries from analysis, reporting, or long-term storage because their characteristics indicate that they do not contain information of interest. For example, duplicate entries and informational messages are filtered because they do not provide useful information in the log analysts. Typically, filtering does not affect short-term storage of events because it does not alter the original log files.

- In event aggregation, occurrences of the same event are consolidated into a single entry containing a count of occurrences of the event. For example, a thousand entries recording a part of a scan can be aggregated into a single entry indicating number of hosts that were scanned. Aggregation is performed as logs are originally generated, and it can also be performed as part of log reduction or event correlation processes.

Storage

- Log rotation is closing a log file and opening a new one when the first file is considered to be complete. Log rotation is typically performed according to a schedule, or when a log file reaches a certain size. The primary benefits of log rotation are preserving log entries and keeping the size of log files manageable. When a log file is rotated, the preserved log file can be compressed and archived to save space.
- Log archiving is retaining logs for an extended period of time, typically on removable media, a *Storage Area Network* (SAN), or a log archiving appliance or server. Logs often need to be preserved to meet legal or regulatory requirements. There are two types of log archiving: retention and preservation.
 - Log retention is archiving logs on a regular basis as part of standard operational activities.
 - Log preservation is keeping logs that normally would be discarded, because they contain records of activity of no particular interest. Log preservation is performed to support incident handling or investigations.
- Log compression is storing a log file in a way that reduces the amount of storage space needed for the file without altering its contents. Log compression is usually performed when logs are rotated or archived.
- Log reduction is removing unneeded entries from a log to create a new log that is smaller. A similar process is event reduction, which removes unneeded data fields from all log entries. Log and event reductions are performed together with log archiving and only the log entries and data fields of interest are placed into long-term storage.
- Log conversion is parsing a log in one format and storing its entries in a different format. For example, conversion can be done on data from a log in a database and save it in an XML file format. Many log generators convert logs to another format. Log conversion can include actions such as filtering, aggregation, and normalization.
- In log normalization, each log data field is converted to a particular data representation and categorized consistently. Common uses for normalization is storing dates and times in a single format. For example, one log generator might store the event time in a twelve-hour format, while another log generator might store it in twenty-four format. Normalizing the data makes analysis and reporting much easier when same log formats are in use.

- Log file integrity checking involves calculating a message digest for each file and storing the message digest securely to ensure that changes to archived logs are detected. The most commonly used message digest algorithms are MD5 and SHA. If the log file is modified and its message digest is recalculated, it will not match the original message digest, indicating that the file has been altered.⁹

Log Analysis

- Event correlation is finding relationships between log entries. The most common form of correlation is rule-based correlation that matches log entries from a single source or multiple sources based on certain values, such as timestamps, IP addresses, and event types. Event correlation can be performed in other ways, such as using statistical methods or visualization tools. If correlation is performed through automation the result is a new log entry that brings together the information into a single entry. Depending on the nature of that information, it can generate an alert to indicating that the event needs further investigation.
- Log viewing is displaying log entries in a human-readable visual format. Most log generators provide some sort of log viewing capability. Some log viewers provide filtering and aggregation capabilities.
- Log reporting is displaying the results of log analysis. Log reporting is usually performed to summarize significant activity over a period of time or to record detailed information related to an event or series of events.

Log Disposal

Log clearing is removing all entries from a log that precede a certain date and time. Log clearing is performed to remove old log data that is no longer needed and is of no importance, or it has been archived.

A log management infrastructure usually contains most of or all of the functions described. The placement of some of the log functions among the three tiers depends primarily on the log management software that is used.

Log management infrastructures are usually based on one of the two categories of log management software: syslog-based centralized logging software and *Security Information and Event Management* (SIEM) software.

VI. STANDARDS IN INFORMATION SECURITY

6.1 ISO/IEC 27000 Family

ISO/IEC 27001 is a part of the evolving ISO/IEC 27000 family of standards. It is an *Information Security Management System* (ISMS) standard published in October 2005 by the *International Organization for Standardization* (ISO) and the *International Electromechanical Commission* (IEC).

ISO/IEC 27001 formally specifies a management system and a way to bring information security under explicit management control. As a formal specification it mandates specific requirements. Organizations that have adopted ISO/IEC 27001 can be formally audited and certified for compliance with the standard.

Most organizations have a number of information security controls. However, without an ISMS, controls tend to be disorganized and disjoint when implemented as a solutions to a specific situations or as a matter of convention. Security controls in operation typically address certain aspects of data security leaving non-IT information assets less protected.

ISO/IEC 27001 requires that management:

- Systematically examine the organization's information security risks, taking into account threats, vulnerabilities, and impacts.
- Design and implement a coherent and comprehensive suite of information security controls and other forms of risk treatment to address risks that are deemed unacceptable.
- Adopt an overarching management process to ensure that the information security controls continue to meet the organization's information security needs on an ongoing basis.

Organizations that implement a suite of information security controls according to ISO/IEC 27002 are likely to meet many of the requirements of ISO/IEC 27001, but may lack some of the management system elements. In other words, ISO/IEC 27001 compliance certificate provides assurance that the management system for information security is in place, but says little about the state of information security within the organization. Technical security controls such as anti-virus and firewalls are not normally audited in ISO/IEC 27001 certification; it is *presumed* that organization has adopted all information security controls since the ISMS is in place and is considered adequate by the requirements of ISO/IEC 27001. Furthermore, organizations management determines the scope of the ISMS audit for certification purposes, and can limit it to, for example, to a single business unit or specific location. The ISO/IEC 27001 certificate

ultimately does not mean that the rest of the organization, outside the scoped area, has an adequate approach to information security management.

Other standards in the ISO/IEC 27000 family of standards provide additional guidance on certain design, implementation and operating an ISMS, for example on information security risk management (ISO/IEC 27005).

6.2 Industry guidance

6.2.1 COBIT

The COBIT framework is produced and maintained by the *Information Systems Audit and Control Association* (ISACA) and managed by the IT Governance Institute, provides a very useful framework of guidance for IT audit and security personnel.

The current version of COBIT, edition 4, includes 34 High Level Control Objectives

COBIT is primarily aimed at auditors, and has an emphasis on what should be audited and how, rather than including detailed guidance for those who are operating the processes that will be audited.

COBIT provides guidance on how processes should be audited and assessed to determine whether they are operating as intended and giving optimum benefit for the organization.

6.2.2 A2 ISO/IEC 20000

In December 2005 the International Standards Organization launched a formal international standard, ISO/IEC 20000, against which organizations can seek independent accreditation for ITSM. This was preceded by a British Standard, BS15000, which was originally introduced in 2000, and under which some organizations became accredited, but was superseded by ISO/IEC 20000 and accreditations were carried over.

For organizations seeking formal accreditation to ISO/IEC 20000, to get external or international recognition, for the success of their ITSM processes, there will be a significant involvement by Service Operation staff in preparing for and undergoing the formal surveillance necessary to achieve the standard.

6.2.3 CMMI

The Capability Maturity Model Integration (CMMI) is a process improvement approach developed by the Software Engineering Institute (SEI) of Carnegie Mellon University. CMMI provides organizations with the essential elements of effective processes. It can be used to guide process improvement across a project, a division, or an entire organization. CMMI helps

integrate traditionally separate organizational functions, set process improvement goals and priorities, provide guidance for quality processes and provide a point of reference for appraising current processes.

6.2.4 *Balanced Scorecard*

A new approach to strategic management was developed in the early 1990s by Drs. Robert Kaplan (Harvard Business School) and David Norton. They named this system the “Balanced Scorecard.” Recognizing some of the weaknesses and vagueness of previous management approaches, the balanced scorecard approach provides a clear prescription as to what organizations should measure in order to “balance” the financial perspective. The Balanced Scorecard suggests that the organization is viewed from four perspectives, and it is valuable to develop metrics, collect data and analyze it relative to each of these perspectives:

- The Learning and Growth Perspective
- The Business Process Perspective
- The Customer Perspective
- The Financial Perspective

Some organizations may choose to use the Balanced Scorecard method as a way of assessing and reporting their IT quality performance in general and their Service Operation performance in particular.

6.2.5 *ITIL*

When ITIL was first introduced in the 1980s, there was little else available in terms of non-proprietary guidance on ITSM best practice. Today, there are other frameworks or methodologies that have valid contributions to make in this area, that complement and have synergy with ITIL and which can be of assistance to Service Operation.

Since its creation, ITIL has grown to become the most widely accepted approach to IT service management in the world. The ITIL Core consists of five publications. Each provides the guidance necessary for an integrated approach, as required by the ISO/IEC 20000 standard specification:

- Service Strategy
- Service Design
- Service Transition
- Service Operation
- Continual Service Improvement

Whether to a global company, a government department or a small business, ITIL gives access to world-class service management expertise. Essentially, it puts IT services where they belong; at the heart of business operations.

6.3 Standard of Good Practice

The Standard of Good Practice for Information Security, published by the Information Security Forum (ISF), is a business-focused, practical and comprehensive guide to identifying and managing information security risks in organizations and their supply chains.

The 2011 publication includes information security “hot topics” such as consumer devices, critical infrastructure, cybercrime attacks, office equipment, spreadsheets, databases and cloud computing.

The 2011 Standard is aligned with the requirements for an Information Security Management System (ISMS) set out in ISO/IEC 27001, and provides wider coverage of ISO/IEC 27002 control topics, encompassing cloud computing, information leakage, consumer devices and security governance.

In addition to providing a tool to enable ISO 27001 certification, the 2011 Standard provides full coverage of COBIT v4, and offers alignment with other standards and legislation such as PCI DSS and the *Sarbanes Oxley Act*, to enable compliance with these standards.

The Standard of Good Practice is used by *Chief Information Security Officers* (CISO's), information security managers, business managers, IT managers, internal and external auditors, and IT service providers in organizations of all sizes.¹⁰

VII. LEGISLATION

7.1 BASEL Impact on Risk Management

The Basel Framework envisages that, over time, the operational risk discipline will mature and converge towards a narrower band of effective risk management and risk measurement practices. Understanding the current range operational risk management and measurement practices, both within and across geographic regions, contributes significantly to the efforts to establish consistent supervisory expectations. Through the analysis of existing practices, and the publication of this paper, the Committee expects operational risk practices maturation and supports supervisors in developing more consistent regulatory expectations. As part of this process, the Committee has identified practices that fall outside the range of effective and good operational risk practices.

Irrespective of adopted risk management and measurement practices, a bank's operational risk strategy should reflect the nature and source of the bank's operational risks for all *Operational*

Risk Measurement System (ORMS) elements, including regular review of predictive elements against experience. The strategy of operational risk management needs to be current and reflect material changes to the internal and external environment. Risk reporting should provide a clear understanding of the key operational risks related to the effectiveness of the internal controls. The internal reporting framework needs to include relevant information reporting of information at all levels of the bank, be transparent, responsive to changes, appropriate and support the proactive management of operational risk.

7.1.1 Use Test and Experience

A bank may use various approaches to articulate and demonstrate the integrated use of its *Operational Risk Measurement Framework (ORMF)*. The integrated framework should be updated regularly and evolve as the bank gains more experience in the management and quantification of operational risk. The level at which ORMF broader processes and practices have been embedded at all organizational levels across a bank are referred to as “embeddedness.” The requirement is ongoing and banks need to ensure that they have the ability to demonstrate embeddedness and that it is not adversely impacted over time by change.

The internal reporting framework should be transparent, responsive to changes, appropriate, and support proactive operational risk management. Performance management criteria relating to ORMS elements and outputs should be established.

The Basel II Framework requires banks to develop an *operational risk management framework*. The ORMF consists of a bank’s:

- (a) Organizational risk and governance structure
- (b) Policies, procedures and processes
- (c) Bank’s systems used in identifying, measuring, monitoring, controlling and mitigating operational risk
- (d) Operational risk measurement system

7.1.2 Basel IT Implications

Operational risk presents a newer area of risk technology. There is a need to develop sound internal loss-reporting practices supported by an infrastructure of loss database systems that are consistent with the scope of operational losses defined by supervisors and the banking industry.

The other challenges in managing operational risk include:

- Data integrity through strict authorization procedure and clear audit trails.
- Proof of transaction, origination & information delivery that needs to be captured to ensure non-repudiation of genuine transactions by the counter party.

- Business continuity and disaster recovery plan needs to be in place.
- Data on high-frequency/low-impact and low frequency/ high-impact events needs to be collated.

7.1.3 Link between Basel and Information Technology

Banks quickly identified Basel II requirements for product portfolio risk segmentation and operational risk data management that will require them to capture and manage extensive instrument, account level and operational risk event information across all business lines. Banks that have already invested in data warehouses and *Business Intelligence* (BI) solutions should be able to leverage built infrastructure to provide the foundation for delivering Basel II compliance.

Basel II requirements, in majority of cases, are likely to require additional data capture, new data models, enhanced data quality, and new reporting structures. There is also a component of the Base II accord that prescribes intent to ensure that banks are to utilize the same data for financial reporting, market disclosure, and internal daily management. This would require building linkages between the financial reporting structures and the Basel II reporting environments. Likewise, banks will benefit from unified BI architecture consolidating reporting and analysis applications.

From an operational perspective, there are many areas where BI service providers can provide proactive assistance to their banking clients including:

- Supporting development of the gap analysis and operational solutions to assess the additional data requirements that are not met through existing data warehouses
- Identifying internal and external options for data capture
- Determining and developing approaches to support incremental data quality, data auditability, and data storage required by Basel II compliance standards
- Working with clients and banking regulators to develop standardized reporting requirements
- Developing checklists of reconciliation requirements (Basel II data to general ledger, Basel II data to financial reporting, and other)
- Ensuring the Basel II environment is integrated into the overall data warehousing architecture
- Working with clients to develop potential extraction, transformation, and loading solutions where historic information is not available for risk model testing, etc.

From a value added perspective, leveraging current data warehouses, consolidated reporting and analysis applications on a unified BI architecture on the foundation for Basel II compliance, will provide impacted banks with ability to:

- Maximize previous data management investments
- Ensure that banks are moving towards the objective of providing consistent management information across the enterprise.

7.2 Sarbanes-Oxley Act (SOX)

The *Sarbanes-Oxley Act* of 2002 (often shortened as *SOX*) is legislation enacted in response to the high-profile Enron and WorldCom financial scandals to protect shareholders and the general public from fraudulent accounting practices and errors in the enterprise organizations. The SOX act is overseen by the *Securities and Exchange Commission* (SEC), which sets compliance deadlines and publishes rules on requirements. SOX is not a set of business practices and does not specify how a business should store records; rather, it defines which records need to be stored and for how long.

The legislation not only affects the financial side of organizations, but it also affects the IT departments whose job is to store electronic records. The *Sarbanes-Oxley Act* states that all business records, including electronic records and electronic messages, must be saved for "not less than five years." The consequences for non-compliance are fines, imprisonment, or both. IT departments are increasingly faced with the challenge of creating and maintaining records archives in a cost-effective fashion that satisfies the requirements put forth by the legislation.

7.2.1 Impact of Sarbanes-Oxley on IT operations

The following sections of *Sarbanes-Oxley* contain the three rules that affect the management of electronic records. The first rule deals with destruction, alteration, or falsification of records.

Sec. 802(a) "Whoever knowingly alters, destroys, mutilates, conceals, covers up, falsifies, or makes a false entry in any record, document, or tangible object with the intent to impede, obstruct, or influence the investigation or proper administration of any matter within the jurisdiction of any department or agency of the United States or any case filed under title 11, or in relation to or contemplation of any such matter or case, shall be fined under this title, imprisoned not more than 20 years, or both."

The second rule defines the retention period for records storage. Best practices indicate that corporations securely store all business records using the same guidelines set for public accountants.

Sec. 802(a)(1) "Any accountant who conducts an audit of an issuer of securities to which section 10A(a) of the Securities Exchange Act of 1934 (15 U.S.C 78j-1(a)) applies, shall maintain all audit or review workpapers for a period of 5 years from the end of the fiscal period in which the audit or review was concluded."

This third rule refers to the type of business records to be stored, including all business records and communications, including electronic communications.

Sec. 802(a)(2) "The Securities and Exchange Commission shall promulgate, within 180 days, such rules and regulations, as are reasonably necessary, relating to the retention of relevant records such as workpapers, documents that form the basis of an audit or review, memoranda, correspondence, communications, other documents, and records (including electronic records) which are created, sent, or received in connection with an audit or review and contain conclusions, opinions, analyses, or financial data relating to such an audit or review."

VIII. RISK AND MEASURES OF SECURITY

Risk is an important measure for evaluating the quality of the system. In the field of information security management, the loss expectancy over some time period is often used as a quantitative measure of risk associated with a class of recurring event:

loss expectancy rate = rate of occurrence $\times$ single loss expectancy

Where the “*rate of occurrence*” refers to the frequency at which the “*event*” is conducted, or in broader terms how often an adversary might try to steal information, and “*single loss expectancy*” is the “*average*” cost of the damage arising from a single “*event*”. The *Single Loss Expectancy* (SLE) can be estimated as follows:

SLE = $\sum$ i probability of outcome i $\times$ cost of outcome i

Where the “*outcome i*” are the possible results of the “*event*”, in other words failure of an attack and various degrees of success.

Another common approach is the qualitative equivalent of the above:

Where risk, threat and impact are each measured in terms of a gradual symbolic scale (high, medium, low), and the operator is defined in the look-up table.

The approaches are equivalent, where the risk is seen as a simple function of parameters measuring the likelihood and frequency of some undesirable event (threat), and the seriousness

of the consequences of the event (impact). While useful, these measures should not be treated as definitions of risk.

This can be illustrated contrary conclusion drawn from the quantitative variant, that a rare event with catastrophic impact appears to have the same risk as a frequent event with minor consequences if the ratio of their frequency is the inverse of the ratio of their impact. In a more careful analysis, when modeling the threat as a sequence of independent events with outcomes of fixed probability, reveals important differences between the two scenarios. While the average impact of the event over time is the same, the probability distribution of the impact is very different. The frequent events have a well-defined peak and fairly predictable impact, whereas the rare events have a much flatter distribution. Comparatively it is more likely to experience either a low impact or a high impact in a given time interval from the rare event type than the common one. If, for example, the probability of serious impact in a given time interval is used as a measure of risk in place of the average impact, then the rare event type is evaluated as carrying the greater risk.

This shows us that risk is a complex topic, and that trying to capture it in a single measure can yield misleading results. We can adopt a quantitative approach based on conditional probabilities, with impact measured in monetary terms. This is done for two main reasons:

- Probability theory gives a coherent and well understood mathematical framework for expressing, combining and reasoning about uncertainty and risk.
- Having a common numeric measurement unit for different risks, value and cost, allows risk estimates to be compared, combined and aggregated, and rational decisions can be made about acceptance, treatment and displacement of risk.

A benefit argued for qualitative framework approach is easier to input gathering and information display for users and stakeholders. However, it is quite straightforward to map between quantitative and qualitative scales for the purposes of ease of interaction with stakeholders.

Viewing risk as a distribution (a curve relating impact to probability) gives us useful numeric measures that can be derived from this distribution:

- Expectation of impact (mean); approximation of the “standard” quantitative measure of risk.
- Most likely impact; the impact corresponding to the highest peak of the distribution.
- The probability of the impact lying above or below a given value.
- The standard deviation; a measure of how spread out the distribution is, and how certain is the possibility of the impact.

In the probability model there is implicit value that can be assigned to various states or configurations to the system of interest. System are subject to “threat events” that affect the value of the system (the event’s impact), and events are governed by a probability distribution. In the

context of security risk, the impact is usually zero or negative, but in other contexts the impact may be positive and the event can be characterized as an opportunity rather than a threat event. Single event has the potential to generate a number of alternative futures on a probabilistic basis. Risk management is about taking steps to optimize range of future possibilities. Generally it involves increasing the probability of favorable futures and decreasing the probability of unfavorable ones.

Generally speaking, vulnerability can be addressed by technical measures, such as firewalls, access controls, encryption, patch management, etc, as well as management policies and awareness training in the case of the human element. On the other hand, threat levels are governable by measures such as the effect of legal sanctions.

The overall risk depends on:

- The threat level (numbers and types of threats and their associated probability distributions)
- The vulnerability (a probabilistic model of system alteration by different types of threat event)
- Values associated with different system states; this is in principle probabilistic as the impact may easily depend on external factors.

8.1 Principles of Risk Management

It is commonly accepted that there are four main principles of risk management strategy, and they are: co-ordination, credibility, effectiveness and transparency.

These are explained in more detail as follows:

• Co-ordination

Any approach that is adopted should provide consistent risk assessment and management throughout the organization. Where identified risks affect more than one part of the organization, the emphasis will be on meeting the objectives rather than addressing organizational structure issues, and the risk owner will have the authority to operate across all of the relevant areas.

• Credibility

The approach should be, where possible, proactive, proportionate and potential threats and vulnerabilities should be addressed before any impact on the organization. Cost of action should be balanced against the potential impact.

- **Effectiveness**

The approach to risk management that is adopted has to be robust and determined, covering all aspects of the process. It has to stand up to internal and external audit and should be developed from best practice. The appropriate and proportionate use of all methods for the treatment of risk should be encouraged, including that of appropriate risk taking. Those with responsibility should address risk management as an integral part of their job, and those risks that have been identified as significant to the organization must be kept under regular review.

- **Transparency**

For a risk management process to be effective there is the need for transparent approach to the process, and the standard “blame culture” needs to be avoided. Staff should be given access to information, based on the business need, with the risks to the organization that have been identified. The process and the risk information that is produced should assist managers in taking appropriate actions to reduce risk. The risk management process should be inclusive and include regular consulting with all stakeholders. This must be balanced with the need to protect the organization from the negative effects of the processes becoming known to people who do not have a need to know it.

8.2 Risk framework

A risk framework is a tool, which must be both convenient and communicable, that can be used to describe the principles and essential components of the security risk management process of an organization. The framework has to describe how the most significant information security risks can be identified, assessed, monitored and treated. It also needs to describe a number of the measures that can be taken to “treat”, or mitigate, the future risk exposure of the organization. A risk framework provides a common language that is used as a carrier for communicating and improving understanding by all of the stakeholders involved in the process, including members of the board, security and audit staffs and the end user. The risk framework can provide a high-level outline for the way in which an organization implements information security risk management and defines the roles of the key participants in the process.¹¹

8.2.1 Risk Framework Objective

The objective of a risk framework is to develop a structured process that will protect the organization's property as it carries out its normal business. The framework is applied on a continuous cycle of activity that is referred to as the risk treatment cycle. This is shown in Fig 2, together with its mapping to the *Plan Do Check Act* (PDCA) cycle.¹²

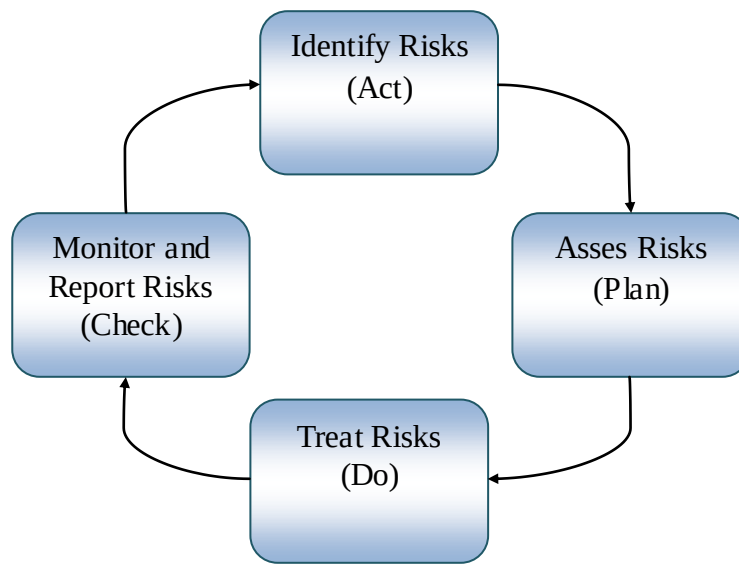


Figure 2 The Elements of the Risk Treatment Cycle

Fig. 2 Taken from: BT Technology Journal • Vol 25 No 1 • January 2007

8.2.2 Components of a Risk Framework

A risk framework is constructed from a number of elements and phases of activity. Phases map to the cycle stages listed below:

- Identify risks:
 - compliance to standards and regulations
 - asset and process identification
 - environment mapping
 - risk identification
 - risk ownership
- Assess risks:
 - risk reduction planning
 - risk modeling
 - testing
- Treat risks:
 - prioritizing risk mitigation
 - risk treatment

- Monitor and report risks:
 - risk monitoring
 - reporting

IX. MODELING SECURITY

Many engineering and scientific disciplines make use of models. A model is an abstract representing a system or artifact. Inevitably, a model is an approximation to the real system it represents. Abstraction helps modelers focus on particular properties of the system of importance and their purpose. If the model is expressed with a formal framework then analysis of the model can make predictions about the real system and in terms can reveal truths about the real system. Meta-model depends on the model purpose and the properties that are studied in the real system. The main uses of the models are interested in constructing and to provide a means of:

- Capturing and agreeing security requirements in the dialogue between the analyst, modeler and system stakeholders; in this role, the model is a medium for communication providing support for making and recording decisions.
- Analysis and refinement of the requirements to produce a model that can be used as a high level design specification for the security aspects of the system; usually the design specifications of secure systems are concerned with integration more-so than development of new code.
- Supporting security management of the system at run time; the model can form high-level baseline for communication between the automated and human elements of the security management system, intelligent interpretation of data, and ultimately intelligent and adaptive security management.

Risk management is concerned with the maximization of the net productive value of the subject system to the uncertain effects of uncertain events. Security can be considered a branch of risk management, in which the events affect the system value through properties such as confidentiality, integrity, availability, privacy and accountability. Typically, increasing security contributes positively to net productive value by reducing the likelihood of events that would decrease that value, but increasing security also contributes negatively by increasing costs and decreasing productivity, usability, and other key feature if the system. In that sense the best amount of security is rarely the highest, but rather amount of security that is the best compromise between risk and cost.

It should be clear from this description that a model that is useful in representing security issues needs to include representation of:

- System stakeholders, the purpose and value-producing activities of the system, and those elements of the system concerned with producing value
- The events of concern that can affect value, their probability of occurrence and factors within and outside the system that affect this probability, and their (probable) consequences
- Elements of the system concerned with managing and implementing security policies, monitoring their correct execution, evaluating their effectiveness, etc.

9.1 Service-oriented modeling and architecture

Service Oriented Architecture (SOA) is an emerging architectural style for designing new generation of enterprise applications. SOA approach strongly reinforces well established, general software architecture principles such as modularization, information hiding, separation of concerns, and it also adds additional approaches as service choreography, service repositories, and others.

SOA is a set of independently running services communicating with each other loosely connected via event-driven messages. Although the concepts behind SOA were established well before Web Services became adopted standard, a service within SOA is completely independent of a Web Service, and current SOA architectures employ them. Today Web Services implement the SOA philosophy by using lightweight protocols based on widely accepted standards.¹³

9.2 The concept of service-orientation

In anticipation of the discovery of new business opportunities or threats, the SOA architectural style aims to provide enterprise business solutions that can extend or change *on demand*. SOA solutions are composed of reusable services, with well-defined, published and standards-compliant interfaces. SOA provides a mechanism for integrating existing legacy applications regardless of their platform or language.

Conceptually, there are three major levels of abstraction within SOA:

- *Operations*: SOA have a specific, structured interface, and return structured responses.
- *Services*: Represent logical groupings of operations.
- *Business Processes*: A long running set of actions or activities performed with specific business goals in mind. In SOA terms, a business process consists of a series of operations or choreography. Typically, choreographed services are invoked in order to respond to business events.

From a modeling standpoint, the resulting challenge is how well-designed operation, service, and process abstractions that can be characterized and constructed systematically.

Advantage is gained by systems that are flexible, by separating the interface, implementation, and binding protocols offered by a SOA, giving the opportunity for the choice of a service provider at any given point in time based on business requirements.

In SOA services can be reused across the internal business units or across the value chains among business partners applied to one tier in architecture, or to multiple tiers across the enterprise architecture, or it can be between business units and business partners within a value chain in a uniform scalable manner.

9.3 The business drivers for a new approach

Improvements in technology accelerated changing customer requirements and business needs to be rapidly adaptive in today's dynamic and competitive environment, whereas the IT infrastructure must enable businesses' ability to adapt.

From 1980's business organizations evolved from vertical isolated business divisions, the horizontal business process focused structures, towards the new dynamic business paradigm. Today business services are fragmented and distributed. The focus is on the extended supply chain, enabling customer and partner access to business services.

Service Oriented Architecture presents an approach for building distributed systems that deliver application functionality as a service; to end-user, applications, or other services. It is comprised of elements that can be categorized into *functional* and *quality of service*.¹⁴

SOA is a natural evolution of client-server systems. Services are readily created and deployed for users by making them available through networking. This is the core of the Internet, creating a service and providing it to users. Although, in most enterprise environments there is a need for control over how users access those services. In enterprise environments SOA is about controlled and shared services that will be helpful in supporting the business, reliable protocols for using them, and how they will continue to operate and be administered over time.

9.4 Security Engineering and Programming in SOA

Security engineering is about building systems to remain dependable in the face of malice, error, or mischance. As a discipline, it focuses on the tools, processes, and methods needed to design, implement, and test complete systems, and to adapt existing systems as their environment evolves.

Security engineering requires cross-disciplinary expertise, ranging from cryptography and computer security through hardware tamper-resistance and formal methods to knowledge of applied psychology, organizational and audit methods and the law. System engineering skills, from business process analysis through software engineering to evaluation and testing, are also important; but they are not sufficient, as they deal only with error and mischance rather than malice.

9.5 Application Black Box

Many applications have been developed in a standalone fashion. Once deployed in an enterprise environment applications cannot function in a standalone manner. Enterprises are increasingly looking for ways to centralize management and administration.

Furthermore, emerging audit and compliance needs means that applications can no longer work in a black box mode. It has become essential for auditors to understand what is going on inside an application, so they can understand the application enforcements of the controls and policies (or lack thereof).

The management and configuration of these policies have ceased being the responsibility of IT professionals, and became the domain of business administrators, requiring the policies to be presented in an appropriate business-level context.

This means that policy logic previously embedded within application code must now be taken out of that code and put into some logical container for administration and audit purpose.¹⁵

Security in the SOA environment involves securing all elements of the SOA network; services, messages, data stores. Due to the decentralized and distributed nature of SOA networks and the use of web services, different services are distributed across different platforms and enterprises which means that data flows in all directions and needs to be protected at all times.¹⁶

Security is frequently looked at as a black art, but in reality the core concepts of security “knowing your assets and designing for failure” are just good engineering practices.

9.6 Risk-based Security Testing: Testing with Threat Modeling

In application design, it is essential to apply threat risk modeling; otherwise resources, time, and money are wasted on useless controls that fail to focus on the real risks.

The method used to assess risk is not nearly as important as actually performing a structured threat risk modeling.

9.6.1 Threat Risk Modeling

Threat risk modeling is an essential process for secure application development. It allows organizations to determine the correct controls and to produce effective countermeasures within budget.

Threat modeling has two distinct, but related meanings in computer security. The first is a description of the security issues the designer cares about. This is the sense of the question, "What is the threat model for DNSSec?" In the second sense, a threat model is a description of a set of security aspects; that is, when looking at a piece of software (or any computer system), one can define a threat model by defining a set of possible attacks to consider. It is often useful to define many separate threat models for one computer system. Each model defines a narrow set of possible attacks to focus on. A threat model can help to assess the probability, the potential harm, the priority etc., of attacks, and help minimize or eradicate the threats. More recently, threat modeling has become an integral part of Microsoft's SDL (Security Development Lifecycle) process. The two senses derive from common military uses in the United States and the United Kingdom.¹⁷

Threat modeling has changed in recent years to take on a more defensive perspective rather than an adversarial perspective. The problem with an adversarial perspective is that it is reactive.

With adopting an adversarial perspective the approach is to examine software applications, or any system, by trying to find holes in it and ways they might be exploited. Techniques that are often used in an adversarial approach are penetration testing (white box and black box), and code review. While these are valuable techniques to discover potential problems, the flaw is that they can only be used once the software has been written.

Current threat modeling takes on a defender's perspective. This means that threats are examined and countermeasures, or security services, are identified at the design state of the application before any code is written. This way the defensive mechanisms are built into the code as it is written rather than patched in later. This is much more cost effective and has the added benefit of increasing security awareness in the development team. However, the disadvantage is that all threats cannot be identified unless the code is trivially simple and often threat modeling on a defender's perspective will cause the development team to falsely believe that the code is secure.¹⁸

General high level overviews of common steps in the defensive perspective threat modeling are:

- Define the application requirements:
 - Identify business objectives
 - Identify user roles that will interact with the application
 - Identify the data the application will manipulate
 - Identify the use cases for operating on that data that the application will facilitate

- Model the application architecture
 - Model the components of the application
 - Model the service roles that the components will act under
 - Model any external dependencies
 - Model the calls from roles, to components and eventually to the data store for each use case as identified above
- Identify any threats to the confidentiality, availability and integrity of the data and the application based on the data access control matrix that your application should be enforcing
- Assign risk values and determine the risk responses
- Determine the countermeasures to implement based on your chosen risk responses
- Continually update the threat model based on the emerging security landscape.

9.6.2 The STRIDE Threat Model

Microsoft uses a specific model to categorize software security threats. It is called **STRIDE**. The six letters in STRIDE each represent a particular risk. These are used in security bulletins to describe the nature of security vulnerability.¹⁹

The simplest way to understand TRIDE model is to consider how each of the threats in the model affects each component and each of its connections or relationships with other application components. Essentially, a look at each part of the application determines whether any threats exist and fall into the STRIDE categories for that component or process.

STRIDE model acronym is derived from the following six threat categories:

- **Spoofing identity.** Illegally accessing and then using another user's authentication information, such as user-name and password.
- **Tampering with data.** Data tampering involves the malicious modification of data. Examples include unauthorized changes made to persistent data, such as that held in a database, and the alteration of data as it flows between two computers over an open network, such as the Internet.
- **Repudiation.** Repudiation threats are associated with users who deny performing an action; a user performs an illegal operation in a system that lacks the ability to trace the prohibited operations.
- **Nonrepudiation** refers to the ability of a system to counter repudiation threats.
- **Information disclosure.** Information disclosure threats involve the exposure of information to individuals who are not supposed to have access to it
- **Denial of service.** Denials of service (DoS) attacks deny service to valid users, by making a server temporarily unavailable or unusable.
- **Elevation of privilege.** In this type of threat, an unprivileged user gains privileged access and thereby has sufficient access to compromise or destroy the entire system.

9.6.3 DREAD Classification

DREAD is a classification scheme used at Microsoft for quantifying, comparing and prioritizing the amount of risk presented by each evaluated threat.

DREAD modeling influences the thinking behind setting the risk rating, and is used directly to sort the risks. The DREAD algorithm, below, is used to calculate a risk value, which is an average of all five categories.

Risk_DREAD = (DAMAGE + REPRODUCIBILITY + EXPLOITABILITY + AFFECTED USERS + DISCOVERABILITY) / 5

The calculation always produces a number between 0 and 10; whereas the higher the number, the more serious the risk.

When a given threat is assessed using DREAD, each category is given a rating. For example, 3 for high, 2 for medium, 1 for low and 0 for none. The sum of all ratings for a given exploit can be used to prioritize among different exploits.

Here are some examples of how to quantify the DREAD categories.

Damage Potential

- If a threat exploit occurs, how much damage will be caused?
 - 0 = Nothing
 - 5 = Individual user data is compromised or affected.
 - 10 = Complete system or data destruction

Reproducibility

- How easy is it to reproduce the threat exploit?
 - 0 = Very hard or impossible, even for administrators of the application.
 - 5 = One or two steps required, may need to be an authorized user.
 - 10 = Just a web browser and the address bar is sufficient, without authentication.

Exploitability

- What is needed to exploit this threat?
 - 0 = Advanced programming and networking knowledge
 - 5 = Malware exists on the Internet, or an exploit is easily performed
 - 10 = Just a web browser

Affected Users

- How many users will be affected?
 - 0 = None
 - 5 = Some users, but not all
 - 10 = All users

Discoverability

- How easy is it to discover this threat?
 - 0 = Very hard to impossible; requires source code or administrative access.
 - 5 = Can figure it out by guessing or by monitoring network traces.
 - 9 = Details of faults like this are already in the public domain and can be easily discovered using a search engine.
 - 10 = The information is visible in the web browser address bar or in a form.

9.6.4 Trike Modeling

Trike is a threat modeling framework with similarities to the Microsoft threat modeling processes. However, Trike differs because it uses a risk based approach with distinct implementation, threat, and risk models, instead of using the STRIDE/DREAD aggregated threat model (attacks, threats, and weaknesses). From the Trike paper, Trike's goals are:

- With assistance from the system stakeholders, to ensure that the risk this system entails to each asset is acceptable to all stakeholders.
- Be able to tell whether we have done this.
- Communicate what we've done and its effects to the stakeholders.
- Empower stakeholders to understand and reduce the risks to them and other stakeholders implied by their actions within their domains.

9.6.5 AS/NZS 4360:2004 Risk Management

The Australian/New Zealand Standard AS/NZS 4360, is the world's first formal standard for documenting and managing risk and is still one of the few formal standards for managing it. The standard's approach is simple (it's only 28 pages long), flexible, and iterative. Furthermore, it does not lock organizations into a particular risk management methodology, provided the methodology fulfils the AS/NZS 4360 five steps. It also provides several sets of risk tables as examples, and allows organizations to freely develop and adopt their own.

The five steps of the AS/NZS 4360 process are:

- *Establish Context*: Establish the risk domain, i.e., which assets/systems are important?
- *Identify the Risks*: Within the risk domain, what specific risks are apparent?
- *Analyze the Risks*: Look at the risks, determine if there are supporting controls in place.
- *Evaluate the Risks*: Determine the residual risk.

- *Treat the Risks:* Describe the method to treat the risks so that risks selected by the business will be mitigated.

AS/NZS 4360 assumes that risk will be managed by an *operational risk group*, and that the organization has skills and risk management resources to identify, analyze, and treat the risks.

9.6.6 CVSS

The US Department of Homeland Security (DHS) established the NIAC Vulnerability Disclosure Working Group, which incorporates input from Cisco Systems, Symantec, ISS, Qualys, Microsoft, CERT/CC, and eBay. One of the group's outputs is the *Common Vulnerability Scoring System (CVSS)*.

9.6.7 OCTAVE

OCTAVE is a risk methodology approach originating from Carnegie Mellon University's Software Engineering Institute (SEI) in collaboration with CERT. OCTAVE focuses on organizational risk, not technical risk. OCTAVE comes in two versions: Full OCTAVE, for large organizations, and OCTAVE-S for small organizations, both of which have specific catalogs of practices, profiles, and worksheets to document the modeling outcomes.

9.6.8 OWASP Guide Project

Unlike other forms of security (such as firewalls and secure lockdowns), web applications have the ability to make a skilled attacker rich, or make the life of a victim a complete misery. At this highest level of the OSI software map, traditional firewalls and other controls simply do not help. The application itself must be self-defending, and The Development Guide can help there. The Development Guide has been written to cover all forms of web application security issues, from SQL Injection, through modern concerns such as AJAX, phishing, credit card handling, session fixation, cross-site request forgeries, compliance, and privacy issues.

The Development Guide is aimed at architects, developers, consultants and auditors and is a comprehensive manual for designing, developing and deploying secure Web Applications and Web Services. The original OWASP Development Guide has become a staple diet for many web security professionals. Since 2002, the initial version was downloaded over 2 million times. Today, the Development Guide is referenced by many leading government, financial, and corporate standards and is the Gold standard for Web Application and Web Service security.

X. DIGITAL FORENSICS AND INTELLIGENCE GATHERING

The true power of any report lies in its ability to let users analyze data visually and effortlessly, involving the least amount of manual calculation. To support such a level of visual analysis, the main ingredient of the reporting is visualization and info graphics. In simple terms, a reporting platform should be very rich in charting and graphing abilities. Rich features are the right kind of graph authoring capabilities required for various kinds of analysis like out-lier analysis, correlation analysis, frequency distribution analysis and other forms of analysis.

The difficulties associated with intelligence analysis are often attributed to the inadequacy of available information. The US Intelligence Community invests more on improved intelligence collection, while managers of analysis argue that comparatively small sums devoted to enhancing analytical resources, improving analytical methods, or gaining better understanding of the cognitive analytical processes are involved in making better judgments. There is often-implicit assumption that lack of information is the principal obstacle to accurate intelligence judgments.²⁰

The relationship between the amounts of information available to the experts, the accuracy of judgments they make based on this information and the experts' confidence in the accuracy of these judgments is unclear. The word "information," as used in the context and refers to the material an analyst has available to work with in making a judgment.

10.1 Intelligence Analysis

Variety of technologies communicating and cooperating with one another and advantages made by increased information flow that brought with it far-reaching consequences are still barely understood. Growth and sophistication of information systems has seen an enormous increase in dependence on these systems. Information and communication technologies have been embraced enthusiastically but with little attention to vulnerabilities. Poor information security could lead to the compromise of privacy, confidentiality, integrity and disruption of critical infrastructure. The benefits accrued by a more connected infrastructure can be realized only through information infrastructure security.

Protecting and trusting information is not unique to the computer world. Solutions designed for the Internet are not always appropriate for securing the infrastructure, which has a different set of priorities and communication needs. Any viable information security solution must address unique challenges posed by IT infrastructure.

Intelligence analysis requires many of the same analytical methods and techniques, irrespective of domain. From this perspective, intelligence analysis for Internet security is no different from any other area of national and international security. The commonality is the concern with those who want to inflict harm and the vulnerabilities they can exploit to do this. Threats can come from many sources: the military activities of hostile nations, economic strategies pursued by adversarial; the activities of terrorist groups intent on inflicting harm on society in order to achieve political objectives, transnational criminal organizations pursuing the acquisition of large profits through illegal activities, violence, and corruption, and individuals with the ability and desire to distort, disrupt, degrade or destroy information and communications systems. Whatever their source, threats are inextricably linked to vulnerabilities; absolute security could only be achieved if there were no vulnerabilities to be exploited. It is crucial, therefore, to identify vulnerabilities and understand how hostile groups or individuals might exploit them. Strictly speaking, real security threats only come from those with a combination of hostile intent and the capacity to inflict harm.

A newly discovered interesting vulnerability exploit in an application can increase interest in other vulnerabilities in that application and an intensive search for similar vulnerabilities in other applications. This result is waves of discovered vulnerabilities that impact both specific vendors and the community. At the same time, the recognition that there are measurable trends in vulnerability exploitation is at least a temporary advantage to intelligence analysis efforts. From a tactical perspective, the recognition of a trend in intrusion is going to allow for a short-term predictive analysis and limited warning.

Obtaining prior knowledge of threats and vulnerabilities, as well as sensitivity to possible opportunities to exploit the vulnerabilities, is essential. Intelligence analysis, of course, operates at different levels, ranging from the specific to the general, and from short-term incidents and operations to long term patterns and challenges. Each form of analysis is crucial, and complements and supplements the others. Nevertheless, it is important to distinguish them from one another and to be clear at which level the activities take place. It is also important to recognize that the most critical insights will be obtained from fusion efforts that combine these different levels. The several complementary levels of intelligence analysis are *Strategic Analysis*, *Tactical Analysis* and *Operational Analysis*. In practice, these categories shade into one another and are not always differentiated, and differing definitions for these terms exist in the intelligence community. Nevertheless, they offer a useful framework within which intelligence tasks and requirements can initially be delineated.

10.1.1 Strategic Intelligence Analysis

Strategic analysis is perhaps the most important form of intelligence. It provides a framework within which other forms of intelligence collection and analysis take place. It offers overall assessment from the top down rather than from the bottom up, and helps to provide a basis for policy formulation, strategic planning, and resource allocation. Its weakness is its focus on macro-level issues rather than more detailed levels of analysis.

Analytical initiatives that fall in the realm of strategic intelligence analysis include:

- *Overall threat assessments.*
- *Sector threat assessments.*
- *Trend Analysis*
- *Potential Damage Assessments*
- *Categorizing and differentiating attacks and attackers*
- *Anomaly identification*
- *Future Net Environments analysis*

10.1.2 Tactical Intelligence Analysis

Tactical analysis is a necessary and important complement to work done at the strategic level. It is the natural linking element between macro-level analysis and the micro-level focus on individual cases. The kinds of studies that are used in tactical analysis would be:

Cluster and pattern analysis
Stimulus-response analysis
Network environmental analysis

In physical space, geography helps to differentiate tactical from strategic, but on the Internet the absence of geography makes the distinction between strategic and tactical somewhat more difficult than in other domains. At the same time, the strategic and tactical levels are mutually reinforcing: although strategic intelligence analysis provides the framework for tactical assessments, these assessments in turn help feed strategic analysis. With a highly dynamic inter-relationship among the various levels of analysis, each level is strengthened by and strengthens the others.

10.1.3 Operational Intelligence Analysis

Operational intelligence analysis overlaps with investigation and is usually single case oriented. It involves technological assessments of the methods used for the intrusion and specific investigations of the intruders. An important component of operational analysis is identifying the particular vulnerabilities that have been exploited to provide guidance on how can they be minimized or eliminated. Another component of operational analysis will be the potential ability to provide attribution during, or after an intrusion. The data collected during an incident, combined with the profiling efforts provided at the tactical level could lead to attribution. While there are significant policy and legal questions associated with such a capability, its importance to the overall analytical process cannot be overemphasized.

Cyber defense combines protection, detection and response. Systems are protected by eliminating vulnerabilities that an attacker can exploit. While protection of a computer system must always be the first line of defense, experience suggests that it is never perfect and therefore security

professionals should not rely on it at the expense of excluding other defense mechanisms.²¹ Preparations should always be made to detect and respond to any attack that succeeded in circumventing a system's protection.

A good defense reacts to intrusions by containing them, undoing their damage and preparing protection against similar attacks in the future.

10.2.3 Intelligent Distribution Analysis Using Analyzers

Business intelligence reporting tools should be capable of analyzing massive amounts of quantitative data through categorization into distributed frequencies and groups. The intelligence expected is facilitating analysis of data density distribution across various logically-related or unrelated groups, identifying outliers, quality control, identifying non-linear relationships between different business parameters, and so on. This corresponds to a branch of statistical analysis known as distribution analysis.

For example, business assignments that are tracked using various project management metrics have *Cost Performance Index* (CPI) and *Schedule Performance Index* (SPI) as two of the tracked metrics. A large organization could have several hundred projects running simultaneously. To analyze how the business is performing on cost against schedule, data of all projects is categorized into clusters based on CPI versus SPI, and analysis can be done on clusters to derive the relationship between these parameters as well as performance of projects under different clusters.

Any intelligent distribution analysis starts with the study of higher level composition. In this analysis recipe a capable reporting tool can help with an interesting and intelligent distribution analysis, and extract insights.

Intelligence in any form of analysis sits in the brain of the individual analyzing the data. With analyzers one can leave a reasonable level of onus of representing data intelligently on the tool itself, to analyze data using intelligent and interactive visualization suited for different forms of analysis.

Companies need constant evaluation for their security measures and can use a number of tools to do so from a network perspective. It is important to understand that such tools often do not fall cleanly across the lines of network security, operating system security, and application security, but usually encompass one or more of these aspects. This reflects the need to ensure security in all these ways, and that these categories overlap heavily.

XI. NETWORK FORENSICS

Computer forensics is a new field in the computer business; and network forensics is in its infancy. Two changes have ignited the field of network forensics: Network forensics technology and its methods are now well understood by more than just network administrators, and storage device costs are affordable. Terabytes of data can now be stored on a network without breaking the storage bank. Networks are high-volume traffic connections, which makes network forensic investigations challenging. Finding the right network forensic tool for specific situation may be difficult, but it's not impossible with the right guidance. Working with network forensic tools is a complex process, but they make the job easier by automating most data acquisition tasks. Just as computer forensics has its roots in data recovery, network forensics is rooted in network security and intrusion detection. Network forensics deals with data that changes from millisecond to millisecond. Investigations of cyber attacks or intrusions are network forensic investigations. The major challenge faced by investigators is how to contain the intrusion while preserving the evidence for later study or use in court.

Even though most systems (such as an IDS or IPS) track and record network data, still there are extra steps needed to add a forensic component to a network system. A forensic component to any system adds depth to your investigation; forensic tools can be used, such as a timeline analysis, e-mail reconstruction, metadata analysis, packet/frame analysis.

The second and less-well-known area of network forensics is the subset of forensic software, which applies these concepts to a network by way of static forensic technology. In simple terms: You can conduct a computer forensic investigation over a network connection and not necessarily have physical access to the suspect computer.

The front lines of any network forensic system consist of the agents deployed across the network. Most software agents are 200 kilobytes (K) or smaller and work in stealth mode to avoid detection and capture majority of data, including agents not usually visible to the operating system.

11.1 Categorizing the Data

The type of data found on a network forensic installation ranges from hard drive images to logs of perimeter routers. The type of data that can be collected is determined mainly by the focus on needs; whether the system in place is designed to manage internal threats, such as industrial espionage or theft, or designed to track information from the network itself looking for external threat or break-in.

Data can be collected from the following devices:

Host: Regular computer forensic acquisitions, such as storage device images, RAM contents, and any static evidence physically located within agents reach, can be transmitted over the network to the forensic server. Hosts are more than workstations, they are also the servers on the network, and those can be e-mail, file, print, and database servers.

Router: A router that is primarily designed to move data between networks, so keeping track of data from these devices is somewhat low priority. The type of information that can be found on a router is related more to traffic logs (ingress or egress) rather than detailed network conversations. Router logs in many cases contain routing process errors, router component status, such as the interfaces, or in some cases even suspicious activity, depending on logging configuration.

Firewall: Firewalls keep detailed logs of network activity, or activity occurring on its system. Firewalls keep logs on activity such as regular network traffic, recognized attacks, dropped or rejected packets, applications that are allowed in or out the firewall boundary, and sources of suspicious activity such as IPs, keeping track of which protocols or services tried to break in.

Switch: Information on network switches can be found in the *Content Addressable Memory* (CAM), where the mapping of a MAC address to a specific port is located in addition to information about the *Virtual Local Area Network* (VLAN).

Intrusion Detection System (IDS): An IDS logs everything that's considered suspicious. One of the purposes of IDS is to log events for further study to keep that event from recurring.

IDS can log the following (not all inclusive list):

- Port scans
- Traffic coming in on uncommon ports or protocols
- Recognized threats, such as worms or viruses attempting to enter the network
- Anonymous FTP attempts or other services on the network
- Originating IP addresses of attacks
- Bandwidth usage

IDSs are designed to be passive watchers and are considered to be the burglar alarms of the information systems.

Intrusion Prevention System (IPS): An IPS works by blocking or shutting down perceived threats on the network. IPS can log the same events that an IDS does, but main difference is analysis of data on the network in *real-time* and scanning for threats.

Network printer: Usually overlooked as a network device has the capability to store information, printers usually have logs showing print jobs with the associated metadata. Modern network printers use Linux and or other NIX operating systems, so it is possible to put an agent on a network printer and capture its data.

Network copier: A network printer relative keeps logs and has evolved in a way printers did and can handle agent installs.

Wireless Access Point (WAP): WAP logs everything similarly to a normal router, with the addition of wireless-specific information, authentication, SSIDs and other incoming connections.

11.2 Storing Logged Events

The amount of data crossing even a small network is substantial in terms of the storage requirements and the allocation of bandwidth required to transport data to the storage. From a logistical point of view, the amount of data security professionals can store is limited by network transport and storage limitations. The answer to this question determines whether system uses agents and sensors to analyze network data in real-time and save only suspect data, or system archives every single bit for later in-depth analysis. If the choice is limited by storage, the option of real-time analysis to extract suspect data saves room because data storage requirements are somewhat small. In this scenario, the agents alert on suspicious activity and forward only limited data that reduces the storage load on the system. If ability to set up large storage areas exist, the following storage options can be taken into consideration:

Storage Area Network

Storage Area Networks (SAN) are a separate network consisting of devices dedicated to data storage. The SAN concept is simple, but the implementation of SAN systems is sometimes complex because these systems can rival the size of the entire network.

Network Attached Storage

A *Network Attached Storage* (NAS) system connects to the network with file level protocols such as NFS or Samba. NAS system can be striped down to be a server dedicated to nothing more than storage access, like a normal file server but with even less general-purpose functionality.

Direct Attached Storage

Unlike a SAN or NAS, a *Direct Attached Storage* (DAS) system is non-networked storage. A DAS system connects to the server that's entering or extracting the data. DAS just extends the storage capability of the server by attaching another computer that's solely dedicated to storage. DAS systems are extremely fast because they have no network structure to contend with, but suffer from not being able to share storage space with other servers except with its directly connected host.

11.3 Timestamps

11.3.1 Analyzing Time Stamps

In any network investigation correlating the time stamps from network devices is an important step. If timestamps on various systems are not synchronized it is impossible to establish a baseline from which to compare data time stamps. Establishing a time stamp can become complicated very quickly. A way to synchronize all devices on a network and eliminate the headaches of manually setting time function is to use *Network Time Protocol* (NTP). The basic function of NTP is to keep all network devices accurate within milliseconds of *Coordinated Universal Time* (UTC). In the case of network devices, the UTC time stamp ensures that a network device is accurate within milliseconds of any other network device. This accuracy is necessary because network communication relies on accurate time stamps to function correctly.

11.3.2 Challenges In Correlating Network Events

- High amount of data
 - Many repeating events (TCP-SYNs,...)
 - Huge amount of data
- Selection of suitable classification rules
 - Separate known events from new/interesting packets
 - Feature selection difficult
 - Features of interest may change
 - High analysis effort
 - Detection of different events requires various metrics

11.4 Logs Filters and Archivers

11.4.1 What is Event Analyzer?

Event analyzer is real time, agent-less, agent-based, event log and application log monitoring and management software. The event analyzer software collects, analyzes, reports, and archives from distributed Windows hosts, distributed Unix hosts, Routers, Switches, and other Syslog devices, The event analyzer application generates graphs and reports that help in analyzing system problems with minimal impact on network performance.

11.4.2 What Problems do Event Analyzers Solve

Event analyzers help monitoring internal threats to the enterprise IT resources and tighten security policies in the enterprise. The event analyzer software generates reports to comply with various regulations such as:

- *Basel III*
- *Sarbanes-Oxley Act (SOX)*
- *Payment Card Industry Data Security Standards (PCI-DSS)*

Archived logs are used for the purpose of network compliance auditing and forensic analysis. The event analyzer software also reduces system downtime, and increases network performance in the organization's environment. It helps administrators in troubleshooting problems on hosts, various applications, and the network.

Event management software and logs also help organizations meet host-based *Security Information Event Management (SIEM)* objectives.

11.4.3 Features of an Event Analyzers

- Centralized event log management
- Compliance reporting
- Automatic alerting
- Historical trending
- Security analysis
- Host grouping
- Pre-built event reports
- Customizable report profiles
- Report scheduling
- Multiple report formats

11.4.4 Overall Benefits of an Event Analyzer

- Network wide visibility and control
- Managing the following types of logs:
 - Windows event logs
 - Unix/Linux syslogs
 - SQL Server audit logs
 - W3C logs
- Log data collection at single location
- Automatic and secured log data archive
- Useful reports that serve the purposes of compliance and security

- Configuration to address different compliance needs
- Support for new compliance reports
- Report scheduling and distribution
- Real time alerts
- Filtering capacity
- Data integration and normalization
- Important events separated from a pile of events
- Continuous watch
- Customizable requirements solution
- Monitoring network performance
- Customized administrative dashboard view

XII. BEYOND LOGGING AND MONITORING

SEM, SIEM and NFAT Monitoring

Network monitoring software observes network traffic and gathers statistics on it. It records information on traffic flows for a particular network or a network segment, such as consumption of bandwidth by various protocols, network activity capturing the payload size and the source and destination IP addresses and ports of each packet.

12.1 Security Event Management Software

Security Event Management (SEM)* imports security event information from network traffic-related security event data sources such as *Intrusion Detection Systems* (IDS), firewall logs, and correlates events among the sources.²² SEM works by receiving copies of logs from data sources over secure channels, normalizing the logs into a standard format and identifying related events by matching IP addresses, timestamps, and other details. SEM does not generate event data; rather, they generate metaevents based on received event data. SEM can not only identify malicious activity, such as attacks and virus infections, but can detect misuse of systems and networks. SEM is helpful in making many sources of network traffic information accessible in a single interface.

SEM can handle almost any security event data source, such as OS logs, antivirus alerts, and physical security logs. SEM contains a variety of information regarding events. Because most data sources record information in different formats, SEM typically normalizes data converting

* Another common term for *Security Event Management* is *Security Information Management* (SIM).

each data field to a standard format. SEM typically does not alter the original data source, so data accuracy can also be verified by other tools.

12.2 Security Information and Event Management Software

Security Information and Event Management (SIEM) software³⁴ is a relatively new type of centralized logging software compared to syslog.³⁵ SIEM can have one or more log servers that perform log analysis, and one or more database servers that store the logs.³⁶ Majority of SIEM products support two ways of collecting logs from log generators:

- **Agentless.** SIEM server receives data from log hosts without the need any special software installed on those hosts. Some servers pull logs from the hosts, which is done by the server authenticating to the host and for log retrieval. In other cases, the hosts push their logs to the server, which also involves authentication (in this case to the server) for log transfer. Whether the logs are pulled or pushed is a question of design, then, the server performs event filtering, aggregation, log normalization and analysis on the collected logs.
- **Agent-Based.** An agent program is installed on the log host to perform event filtering, aggregation and log normalization for a particular log, then it transmit the normalized log data to designated SIEM server, in most cases on a real-time or near-real-time basis for analysis and storage. In some cases when a host has multiple types of logs it might be necessary to install multiple agents. Some SIEM products offer generic agents for formats such as syslog or *Simple Network Management Protocol* (SNMP). A generic agent is used to get log data from a source for which a format-specific agent or an agentless method is not available.

Advantages and disadvantages exist for each method. The primary advantage of the agentless approach is that agents do not need to be installed, configured, and maintained on each logging host.

SIEM products include support for many types of log sources, such as OS, security software, application servers, physical security control devices such as badge readers. For most of the supported log sources, except for generic formats such as syslog SIEM products can categorize the most important logged fields. This feature improves the normalization, analysis, and correlation of log data in comparison to software with a less detailed understanding of specific log sources and formats.

12.3 Network Forensic Analysis Tools

Network Forensic Analysis Tools (NFAT) provide the same functionality as packet sniffers, protocol analyzers, and SEM in a single product.²³ Whereas, SEM software is designed to correlate events among existing data sources, NFAT is focused on collecting and analyzing

network traffic. NFAT software offers additional features that further support network data analysis, such as:

- Reconstruction of events by replaying the traffic within the tool, ranging from an individual session to all sessions during a particular time period.
- The speed of the replaying can be adjusted as needed.
- Visualizing the traffic flows and the relationships among hosts. Some tools can tie IP addresses, domain names, or other data to physical locations and produce a geographic map of the activity.
- Building profiles of typical activity and identifying aberrant behavior.
- Searching application content for keywords.

12.4 Acquiring Network Traffic Data

Organizations typically have network traffic data recorded in many places and use the same data recording mechanisms to acquire additional data when investigating incidents or troubleshooting problems.

Network traffic data usually gets recorded in a log or is stored in a packet capture file. Mostly, acquiring the data is acquiring copies of the logs or packet capture files. If data is not stored in a file, raw format, but rather displayed graphically, screen captures may be needed.

Although most network traffic data occurs as part of regular operations, it can be acquired as part of troubleshooting or incident handling. In the case of troubleshooting or incident handling, it is important to follow outlined processes and document all performed actions. Majority of organizations have policies that clearly explain what types of monitoring can and cannot be performed without prior approval, and have procedures that detail the request and approval process.

12.4.1 Technical Issues of Data Acquisition

There are several technical issues that can impede the acquisition of data on network traffic.

- **Data Storage.** With large volumes of network particularly during events such as attacks, logs record many events in a short time. If storage is insufficient, information about recent activity can be overwritten or lost. Organizations need an estimate of peak and typical log usage, determining how many hours or days' of log data can be retained, and ensure sufficient storage to meet those needs. Organizations need to provide sufficient data storage to keep logs associated with computer security incidents for a substantially longer time than other logs as needed. For example, *General Records Schedule (GRS) 24, Information Technology Operations and Management Records*, specifies that “computer security incident handling, reporting

and follow-up records” should be destroyed 3 years after all necessary follow-up actions have been completed.”²⁴

- **Encrypted Traffic.** When protocols such as, SSH, *Secure Socket Layer* (SSL) or *Internet Protocol Security* (**IPsec**) are used to encrypt traffic, devices monitoring traffic along the encrypted path can see only the traffic basics, such as source and destination IP addresses. If tunneling techniques are used, the IP addresses are for the tunnel and not the true source and destination of the traffic. To acquire data on the decrypted traffic, monitoring device needs to be positioned where it can capture decrypted activity.
- **Services Running on Unexpected Ports.** IDS and protocol analyzers rely on port numbers to identify service is in use, but most services can be configured to run on any port number. Traffic involving services running on different port numbers cannot be easily captured, monitored, or analyzed properly, making usage of unauthorized services hard to detect. Another motivation is to circumvent perimeter devices that filter based on port numbers. There are several ways to identify unexpected port usage, such as:

Performing traffic flow analysis and identifying unusual traffic flows.

- Configuring IDS sensors to perform protocol analysis and alert on connections that use an unexpected protocol.
- Configuring IDS sensors to alert on connections involving unknown server ports.
- Configuring a protocol analyzer to analyze a particular stream as something else.

12.5 Alternate Access Points and Covert Channels

13.5.1 Alternate Access Points

Attackers can enter networks from an alternate access points to avoid detection by security controls monitoring major access points, such as the Internet gateway. Organizations usually address this by limiting alternate access points, such as modems and wireless access points by ensuring that they are monitored and restricted through firewalls, IDS sensors, and various other controls.

Inevitably, systems and applications always experience failures or outages, for various reasons. In the case of dedicated monitoring systems, as are IDS sensors, using redundant equipment can lessen the impact of monitoring failures. Another approach is to perform multiple levels of monitoring, such as configuring network based and host based firewalls to log connections (Defense in depth, also known as deep or elastic defense).²⁵

12.5.2 Covert Channels

Attackers can use covert channels, using protocols in unintended ways to secretly communicate information. The use of covert channels is usually hard to detect, but one useful method is identifying deviations in expected traffic flows. Another possibility is to use a visualization tool, which presents security event data in a graphical format. These tools are mostly used to represent traffic flows visually, and can be very helpful in troubleshooting operational issues and identifying misuse.

Tools for identifying malicious activity are not completely accurate. They can produce false positives (reporting regular activity as malicious), and false negatives (reporting malicious activity as regular traffic). Validation should be based on examining additional data such as raw packets and data captured by other sources, and past experience with the tool in question. In many cases, in those situations experience is the key, examination of the supporting data is needed to determine if an alert needs further investigation.

12.6 Combined Log Management and SIEM Architecture Benefits

A well designed log management and SIEM deployment can significantly reduce the time to incident identification and enhance overall information security capability. It unifies searching, reporting, alerting and analysis across all types of enterprise log data with capability to analyze and store large amounts of data generated by modern networks.

The log management needs to refine the data set and forward only applicable events for analysis to the correlation engine, SIEM, through intelligent “tagging” of events. Overall data reduction and linear compatibility (compatibility to collect and “read” the logs from various systems) is only a part of the solution; it is also important to ensure the right data is forwarded and evaluated to get overall benefits offered by the SIEM.

CONCLUSION

Current tools are helpful in gathering and presenting traffic data, but those tools have limited analysis abilities and can be effectively used only by well-trained administrators or users. In addition to understanding the tools, administrators need to have strong background in networking, common network and application protocols, security products, and network-based threats and attack methods.

Besides knowledge of networking, protocols and applications, it is important that administrators have good knowledge of the organization's environment. Administrators need to understand the organization's normal operating baseline, such as typical patterns of usage on systems and networks; with that knowledge their performance would reflect with good IT performance. Administrators need to have strong understanding of the network traffic data sources, as well as supporting materials for IT operations.

Appendix A

USING SYSLOG FOR VALIDATING SECURITY POLICY COMPLIANCE

Organizations routinely undertake the expensive task of auditing event trails from electronic and physical systems. Most audits are performed to identify problems, adjust capacity issues, or to maintain compliance with regulatory laws. Every log may contain critical information about prior or ongoing events. Examples of these events include electronic events such as logon, connect, and write, or physical events such as building access or equipment malfunction alerts. Electronic events reflect status, threats, and other changes in the environment that help an enterprise in maintaining situational and informational awareness. Today, there is no standard for representing and describing these events in logs. This is a significant data management problem, since enterprise wide situational awareness depends on the ability to process and analyze event data.

There are several design interests when implementing logging infrastructure to capture the log data from every device on the network at all times. Logging into one central place where all the information would converge, is to be able to correlate events across multiple devices. For this reason a common time source and log formats are desirable. Logs need to have integrity in case the logs were required for use in audit or as evidence. This demands well designed archiving processes and handling procedures.

From a usability perspective the log data needs to be easily reported and manipulated by common database tools. A final requirement is to have the system immediately alert administrators whenever events of interest appear in the logs.

Syslog is most feature rich product in this area and majority of features and functionality is on its native OS UNIX.

1.1 Configuring Network Devices

Before configuring devices to log into syslog server it is important to understand how the Syslog protocol itself is structured in order to create efficient design to use.

Each Syslog message includes a priority value which is made up of two parts. The value is expressed as **facility.severity** where facility is a type of category for the message and severity is its relative importance.

Some of the facilities are assigned for static purposes and others are user definable. User definable facilities need to be considered in design. They are facilities named local0 through local7.

When configuring devices to use Syslog there is an option to determine which facility and sometimes which severity the device would use. This helps with organizing the incoming log data in a way that would make it easier to manipulate both in an immediate and an historical context.

Facility Name Purpose Example:

Local0 Perimeter devices
Local1 Firewalls
Local2 Interior networked servers
Local3 Interior routers & switches
Local4 Workstations
Local5 Peripheral devices
Local6 unassigned
Local7 unassigned

Configuring devices according to this scheme has many applications:

- Applying different rules to the different facilities
- This approach allows focus on high-risk devices
- Messages from perimeter hosts can be isolated from the rest of the traffic for increased scrutiny

1.1.1 Microsoft Windows Host as syslog Client

When using syslog agent for Windows messages from log files and event log groups are transferred to the central syslog server using TCP connections. That way Windows and UNIX devices are integrated into the same logging infrastructure.

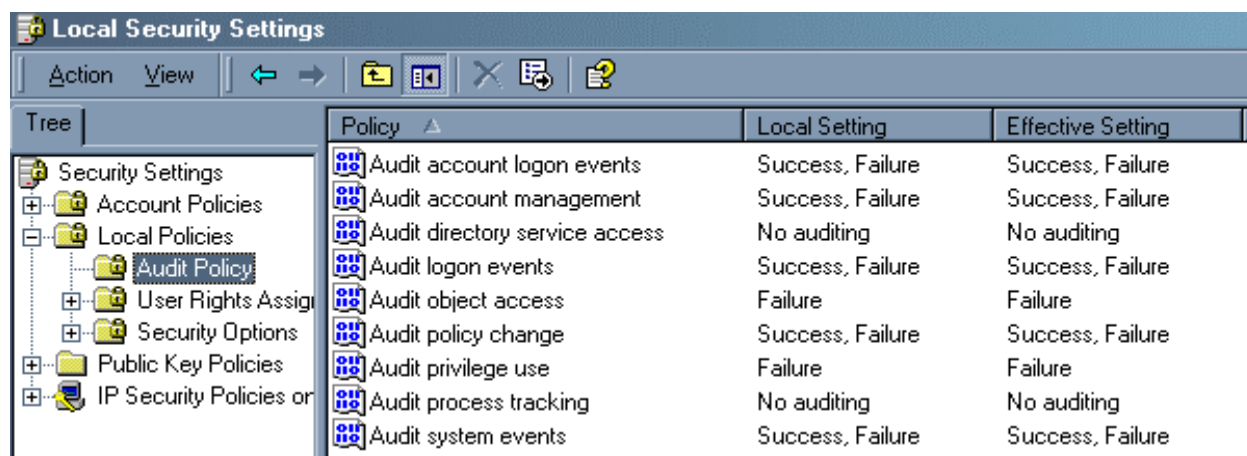
Many programs exist for the purpose of logging Windows messages to a syslog server. Some are more feature rich and robust than others. Particular important feature is the ability to specify the syslog facility for each Windows system to use. Forehand mentioned features are present in BackLog7 from Intersect Alliance.

Auditing policies in Windows system need to be enabled because BackLog will only forward messages if they get written to the local event log. Attention needs to be given to configuration of the server information that is considered important. This configuration is done within the operating system itself.

Example configuration of Windows host is as follows:

In User Manager. Pull down the Policies menu from the tool bar and select Audit. Choose the settings that comply with organizations security policy.

Following are possible settings that can be used:



The screenshot shows the 'Local Security Settings' window. In the left-hand 'Tree' pane, the hierarchy is: Security Settings > Local Policies > Audit Policy. The main pane displays a list of audit policies with their current 'Local Setting' and 'Effective Setting'.

Policy	Local Setting	Effective Setting
Audit account logon events	Success, Failure	Success, Failure
Audit account management	Success, Failure	Success, Failure
Audit directory service access	No auditing	No auditing
Audit logon events	Success, Failure	Success, Failure
Audit object access	Failure	Failure
Audit policy change	Success, Failure	Success, Failure
Audit privilege use	Failure	Failure
Audit process tracking	No auditing	No auditing
Audit system events	Success, Failure	Success, Failure

Figure 3 Example Configuration of Windows Host

1.1.2 Cisco Routers as Syslog Client

Cisco routers are easy to configure because they natively support syslog. A few simple commands are needed to get Cisco devices logging to syslog. The following are the configuration steps that can be followed:

```
Router# conf t
Router(config)# logging on
Router(config)# logging 192.168.1.1 (address of syslog host)
Router(config)# logging trap informational
Router(config)# logging facility local7
```

Additionally commands can be used to fine tune the device, nevertheless this is a basic configuration that can be utilized as a good starting point.

There is a certain amount of overhead consumed by the logging process on a router. It is generally minimal but logging debug messages can consume a lot of processing cycles.

Logging access lists on a Cisco router can be achieved by adding the keyword “log” to the end of an access list statement the router, which will tell the router to log any packet that meets the criteria of that statement. One of the useful places to employ this function is in a “deny” statement at the end of an access list.

For example:

access-list 101 deny ip any any log

Above, access list would log any IP traffic that is trying to traverse the interface. This would obviously be excessive, but it is an example of proper syntax. The best application of this rule is determined by the needs of each network.²⁶

1.1.3 Linux Hosts as Syslog Clients

For example, on a Mandrake Linux system (Red Hat variant), edit the /etc/sysconfig/syslog file, and change the SYSLOGD_OPTIONS parameter to the following:

SYSLOGD_OPTIONS="-r -m 0"

Next, restart the syslog service. You should also ensure that the firewall on that machine allows access to UDP port 514 from the machines that will be sending it logs.

On the system for which you wish to send log entries, modify the /etc/syslog.conf file, and add something similar to the following at the very bottom:

***.info @loghost.mydomain.com**

This tells syslog to send all *.info level log entries to the host loghost.mydomain.com. You can change which facilities you wish to remotely log, but *.info is generally sufficient. Restart syslog on this machine as well, and ensure that the firewall allows sending from the local host to the remote machine on UDP port 514.

Log entries from the one host should now appear on the remote host, mixed in with that host's own logs. For instance, your log files may now look like this:

Jan 8 13:23:22 loghost fam[3627]: connect: Connection refused

Jan 8 13:23:24 remote.mydomain.com su(pam_unix)[3166]: session closed for user root

1.2 Archiving

One of the issues that needs to be addressed is how to maintain the integrity of the log data. Procedures for archiving and handling the log files are needed if the information within is to be found trustworthy.

In mature IT operations, particularly in regulated environments, it is not uncommon for an organization's security policy to require controls on the use of privileged ID's. Especially, the use of the "root" or "administrator" accounts that do not provide individual accountability. A policy requirement might be to require the use of "sudo" or equivalent access control to allow administrators to perform their jobs without losing individual accountability.

A good way to ensure policy effectiveness is to analyze logs. If it is a policy exception that an administrator has "root" access, then perusing logs for root access occurrence is a good way of verifying policy adherence. For example root login on Linux might look like this:

Aug 7 17:22:00 www2 sshd[54036]: Accepted keyboard-interactive/pam for root from 127.0.0.1 port 50496 ssh2

Building a query can be as simple as "grep root /var/log/messages", or can be performed in any number of log analyzers. Of course, root would have the ability to remove the log evidence, which is a very good reason to relay logs to a central log server. It is imperative that administrators with access to root accounts on systems do not also have administrative access to the central log server.

Pulling evidence of root logins out of the central log server which administrators do not have access to is a good control to determine whether administrators are following policy. This solution allows the administrators to retain the root access in the event of an emergency or maintenance. The policy should require reconciliation of any root logins found with known, documented issues. Use of the password without a valid reason will show evidence of policy violation.

CONCLUSION

To standardize event logs from electronic systems summarized in the centralized event log, the core of the centralized log architecture needs to enable collaborative efforts in the creation of an open, practical, and inter-operable system.

This core provides the architecture of the core components including the data dictionary, syntax specifications, and event taxonomies. This is a collection of documents and specifications. The combination of the documents and specifications provides the necessary pieces to create a complete event log standard, which can be mapped against the transport, syntax, taxonomy, and log recommendations.

APPENDIX B

NTOP, THE ALTERNATIVE APPROACH TO MONITORING

Simple alternatives to network monitoring are packet tracers and decoders, often called network sniffers. Examples are tcpdump (Linux)²⁷ and snoop (Sun)²⁸. These tools are used for packet capture and require off-line analysis tools to correlate data and identify network flows. Sniffers usually provide details on packet activity but lack information on the network as a whole.²⁹ Protocol analyzers, like Ethereal³⁰, focus on the content of single network packets and not on global network activities. These solutions lack support for high-level management activities.

More advanced alternatives include RMON (*Remote Network Monitoring*) management platforms.³¹ Those platforms promote separation between probes and managers. Probes are devices that collect data and managers are applications that provide higher level information for operators. RMON managers are seen as data analyzers, but they also configure probes and retrieve relevant data via SNMP³².

In RMON, flexibility is in modularity and standardization. RMON's standards define the way in which a manager retrieve information from probes and which pieces of information are available. In an RMON configuration, a manager collects data from several probes. Therefore, it is possible to monitor many subnets from a central manager, a feature not present in ntop. Ntop runs as a single application, including the probe and the analyzer, and consequently, is not able to monitor traffic collected from different probing points; traffic from different subnets.

As the RMON standards specify a communication mechanism between manager and probes, it is possible to create managers for different purposes, for instance, a security failure manager, or to use different applications for data analysis. With ntop, specialized functions are implemented via new versions and optional plug-ins.

RMON based solutions are powerful, but unfortunately need sophisticated SNMP managers. Due to the complexity and costs of RMON solutions are basically used by network managers in large companies.

Tools for network monitoring such as NeTraMet³³ and NFR³⁴ offer advanced programming languages for analyzing network flows and building statistical event records, but these languages are only useful to experienced network operators, and are not present in ntop for the sake of simplicity.

Ntop has proven to be a valuable tool for quick access to network monitoring, with integrated web interface, minimal requirements and lightweight CPU utilization. It is available for network administrators with minimal (installing, learning) effort and cost, as opposed to expensive and complex management platforms.

1.1 What ntop Does and How it Works

Ntop shows the current network usage. It displays a list of hosts that are currently using the network and reports information concerning the IP and non-IP traffic generated and received by each host. Ntop can operate as a front-end collector with sFlow and/or netFlow plugins or as a stand-alone collector or correlation program. A web browser is needed to access the information captured by the ntop program.

Ntop is a hybrid layer 2 / layer 3 network monitor, that is by default it uses the layer 2 *Media Access Control* (MAC) addresses and the layer 3 TCP/IP addresses. Ntop is capable of associating the two, so that ip and non-ip traffic (e.g. arp, rarp) are combined for a complete picture of network activity.³⁵

1.2. Installing and Running Ntop on Linux Host

1.2.1 Ubuntu

In Ubuntu Debian package installed via “apt-get install ntop” installs the package and libraries, configures and runs ntop without any need for configuration. Default configuration enables most of the functionality that ntop supports, on the other hand for some the plugins like cPacket, ICMPwatch, NetFlow, RRD and sFlow require manual configuration and enabling is needed.

1.2.2 Configuring and Compiling ntop Under RHEL

Download latest ntop:

```
# wget http://freshmeat.net/redir/ntop/7279/url_tgz/ntop-3.3.6.tar.gz
```

Untar tar ball:

```
# tar -zxvf ntop-3.3.6.tar.gz
```

Type the following commands to compile and install ntop:

```
# cd ntop
```

```
# ./autogen.sh
```

Type: make to compile ntop:

```
# make
```

Type: make install to install ntop:

```
# make install
# make install-data-as
Create ntop user:
# useradd -M -s /sbin/nologin -r ntop

Setup directory permissions:
    # chown ntop:root /usr/local/var/ntop/
# chown ntop:ntop /usr/local/share/ntop/

Setup ntop user admin password:
# ntop -A
```

Sample output:

```
Thu Aug 16 11:23:34 2012 NOTE: Interface merge enabled by default
Thu Aug 16 11:23:34 2012 Initializing gdbm databases
ntop startup - waiting for user response!
Please enter the password for the admin user:
Please enter the password again:
Thu Aug 16 11:23:34 2012 Admin user password has been set
```

Starting ntop From Command Line:

Type the following command to start ntop:

```
# /usr/local/bin/ntop -d -L -u ntop -P /usr/local/var/ntop --skip-version-check --use-
syslog=daemon
```

Sample output:

```
Thu Aug 16 11:23:34 2012 NOTE: Interface merge enabled by default
Thu Aug 16 11:23:34 2012 Initializing gdbm databases
```

If you have multiple interfaces (eth0, eth1 and so on), start ntop as follows:

```
# /usr/local/bin/ntop -i "eth0,eth1" -d -L -u ntop -P /usr/local/var/ntop --skip-version-check --use-
syslog=daemon
```

Where,

- i "eth0,eth1" : Specifies tnetwork interfaces to be used by ntop for network monitoring.
- d : Run ntop as a daemon.
- L : Send all log messages to the system log (/var/log/messages) instead of screen.
- u ntop : Start ntop as ntop user
- P /usr/local/var/ntop : Specify where ntop stores database files.

--skip-version-check : By default, ntop accesses a remote file to periodically check if the most current version is running. This option disables that check.
--use-syslog=daemon : Use syslog daemon.

1.2.3 Additional Packages for ntop Functionality Support:

rrdtool - time-series data storage and display system (programs)
rrdtool-dbg - Time-series data storage and display system (debugging symbols)
rrdtool-tcl - time-series data storage and display system (Tcl interface)
bindgraph - DNS statistics RRDtool frontend for BIND9
libcap-dev - development libraries and header files for libcap2
libcap-ng-dev - Development and header files for libcap-ng
libcap-ng0 - An alternate POSIX capabilities library
libcap2 - support for getting/setting POSIX.1e capabilities
libcap-ng-utils - Utilities for analyzing and setting file capabilities
libcap2-bin - basic utility programs for using capabilities
libcapi20-3 - libraries for C API support
libcapi20-dev - libraries for C API support
libcapsinetwork-dev - C++ network server library, development files
libcapsinetwork0c2a - C++ network server library
libcaptcha-recaptcha-perl - perl implementation of the reCAPTCHA API
libcapture-tiny-perl - module to capture STDOUT and STDERR
python-cap-ng - Python bindings for libcap-ng

1.2.4 Ntop Command Line Configuration

Beginning with 3.1, many command-line options may also be set via the web browser interface. These changes take effect on the next run of and on each subsequent run until changed.

ntop -w 3000 -W 0 (this is the default setting) HTTP requests on port 3000 and no HTTPS.
ntop -w 80 -W 443 Both HTTP and HTTPS have been enabled on their most common ports.
ntop -w 0 -W 443 HTTP disabled, HTTPS enabled on the common port.

Sensitive, configuration pages of the ntop web server are protected by a userid/password. By default, these are the user/URL administration, filter, shutdown and reset stats are password protected and are accessible initially only to user admin with a password set during the first run of ntop.

Users can modify/add/delete users/URLs using ntop itself - see the Admin tab.

1.2.5 Creating a Traffic Filter

The filter expression is set at run time by this parameter, but it may be changed during the ntop run on the Admin | Change Filter web page.

The basic format is **-B filter** , where the quotes are REQUIRED

The syntax of the filter expression uses the same *Berkeley Packet Filter* (BPF) expressions used by other packages such as tcpdump

For instance, suppose you are interested only in the traffic generated/received by the host centurion.bnlan.net.

ntop can then be started with the following filter:

```
ntop -B src host centurion.bnlan.net or dst host centurion.bnlan.net
```

or in shorthand:

```
ntop -B host centurion.bnlan.net or host centurion.bnlan.net
```

1.2.6 Suspicious Packet Tracking

Command Line Option:

```
-q --create-suspicious-packets
```

This parameter tells ntop to create a dump file of suspicious packets. The following is not all inclusive list, there are many occurrences that cause a packet to be labeled as 'suspicious', including:

- Detected ICMP fragment
- Detected Land Attack against host
- Detected overlapping/tiny packet fragment
- Detected traffic on a diagnostic port
- Host performed ACK/FIN/NULL scan
- Host rejected TCP session
- HTTP/FTP/SMTP/SSH detected at wrong port
- Malformed TCP/UDP/ICMP packet (packet too short)
- Packet # %u too long
- Received a ICMP protocol Unreachable from host
- Sent ICMP Administratively Prohibited packet to host
- Smurf packet detected for host
- TCP connection with no data exchanged
- TCP session reset without completing 3-way handshake
- Two MAC addresses found for the same IP address
- UDP data to a closed port

Unknown protocol (no HTTP/FTP/SMTP/SSH) detected (on port 80/21/25/22)
Unusual ICMP options

When this parameter is used, one file is created for each network interface where suspicious packets are found. The file is in tcpdump (pcap) format and is named:

<path>/ntop-suspicious-pkts.<device>.pcap

Where:

<path> is defined by the -O | --output-packet-path parameter.

The intrusion detection mechanism is defined using a rule configuration file specifying the -F switch, and it is also possible to force ntop to log all suspicious packets into one file for each device with the -q switch.

1.2.7 Refresh Times

-r --refresh-time

Specifies the delay (in seconds) between automatic screen updates for those generated HTML pages which support them. This parameter allows you to leave your browser window open and have it always displaying nearly real-time data from **ntop**.

The default is 3 seconds. Please note that if the delay is very short (1 second for instance), **ntop** might not be able to process all of the network traffic.

1.2.8 Setting Interfaces to Work Without Promiscuous Mode

-s --no-promiscuous

Using this parameter prevents setting the interface(s) into promiscuous mode.

1.2.9 Running More Than One Instance of ntop

--instance

It is possible to run multiple instances of ntop simultaneously by specifying different -P values (typically through separate ntop.conf files). If a value for this parameter is set (available only on the command line), it displays the 'instance' name on every web page, and it is possible to alter the log prefix from "NTOP" to a chosen value.

Note (UNIX): To run completely different versions of the ntop binary, it is need to compile and install into a different library (using `./configure --prefix`) and then specify the `LD_LIBRARY_PATH` before invoking, e.g.

`LD_LIBRARY_PATH=/devel/lib/ntop/:... /devel/bin/ntop ...args...`

1.3 Configuring Plugins and Integrating Software

1.3.1 Ntop IDS

Ntop and snort are both about the best passive open source intrusion detection tools that can be found. They work in different ways, so a combination of the two is useful, but it can generate duplicates of an event.

Snort is able to detect a wide range of network activity including different kinds of port scans and malformed HTTP requests.

Ntop can be configured to detect aberrant behavior within the rrd plugin configuration.

1.3.2 Summary of Aberrant Behavior Detection Algorithm

Aberrant behavior detection is decomposed into three pieces, each building on its predecessor:

- (1) An algorithm for predicting the values of a time series one time step into the future.
- (2) A measure of deviation between the predicted values and the observed values.
- (3) A mechanism to decides if and when an observed value or sequence of observed values is ‘too deviant’ from the predicted value.

The rrd tool plugin uses the Holt-Winters Time Series Forecasting Algorithm, or incremental, algorithm that adaptively predicts future observations in a time series. It’s forecast is the sum of three components:

- a baseline (or intercept)
- a linear trend over time (or slope)
- and a seasonal coefficient (a periodic effect, such as a daily cycle)

There is one seasonal coefficient for each time point in the period (cycle). After a value is observed, each of these components is updated via exponential smoothing. The algorithm learns from past values and uses them to predict the future. The rate of adaptation is governed by 3 parameters:

- alpha (intercept)

- beta (slope)
- and gamma (seasonal)

The prediction can also be viewed as a ‘smoothed’ value for the time series.

The measure of deviation is a seasonal weighted absolute deviation. The term ‘seasonal’ means deviation is measured separately for each time point in the seasonal cycle. As with Holt-Winters Forecasting, deviation is predicted using the measure computed from past values (but only at that point in the seasonal cycle). After the value is observed, the algorithm learns from the observed value via exponential smoothing. Scaling the sequence of predicted deviation values for the observed time series generates confidence bands (we usually think of the sequences as continuous lines rather than as a set of discrete points).

Aberrant behavior (a potential ‘failure’) is reported whenever the number of times the observed value violates the confidence bands, meets or exceeds a specified threshold within a specified temporal window (i.e. 5 violations during the past 45 minutes with a value observed every 5 minutes).

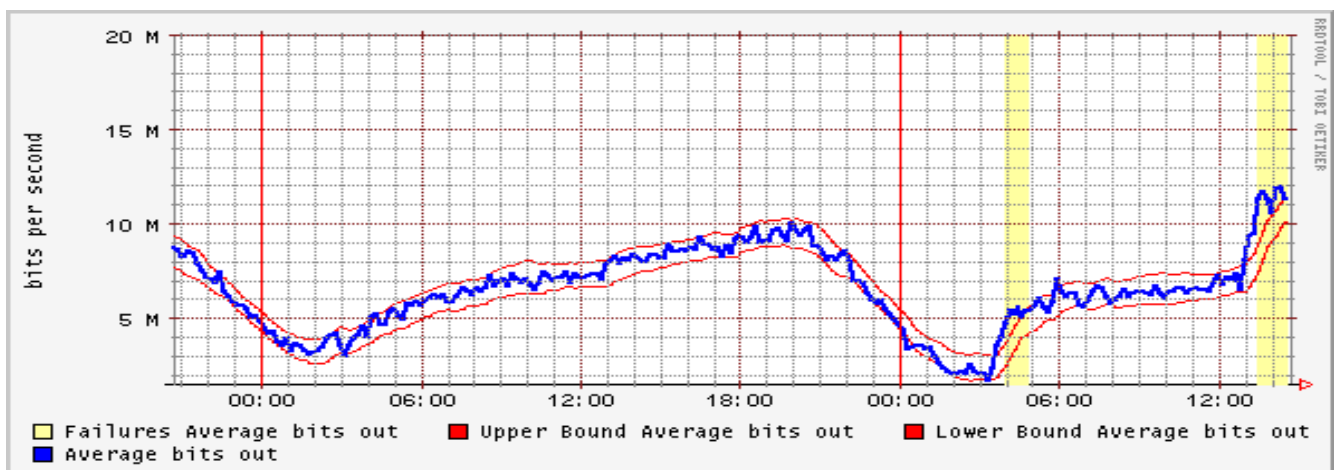


Figure 4 Ntop Abberant Behavior Graph

The thin red lines are the confidence bands and the yellow bars represent failures (actually multiple failures in sequence).

The graph suggests that bandwidth on this outgoing link is increasing faster than expected by the model (past history). It is up to the network administrator to decide if it represents aberrant behavior. One approach might be to view the time series for this router interface over a longer time period.

1.3.3 Netflow Plugin

Netflow is the protocol developed by Cisco to manage data about IP traffic. Using Netflow it is possible to collect data about all IP data sent or received on multiple Cisco, Linux, BSD or Juniper hosts and send it to central Netflow collector that will display graphs. It helps to visualize a complete picture of what data was sent and received on monitored hosts including destination and source IP, port, bytes transferred, in and out the interfaces etc. Nfdump is Netflow collector. Nfsen is a graphical tool for generating graphs and querying Nfdump for historical traffic reports.

Netflow probe nfcapd is required to collect IP traffic data on Linux hosts. In general nfcapd sits in the background, stores network activity on network interfaces and then sends collected data to Netflow collector nfdump. Nfcapd is a simple application that just collects data.

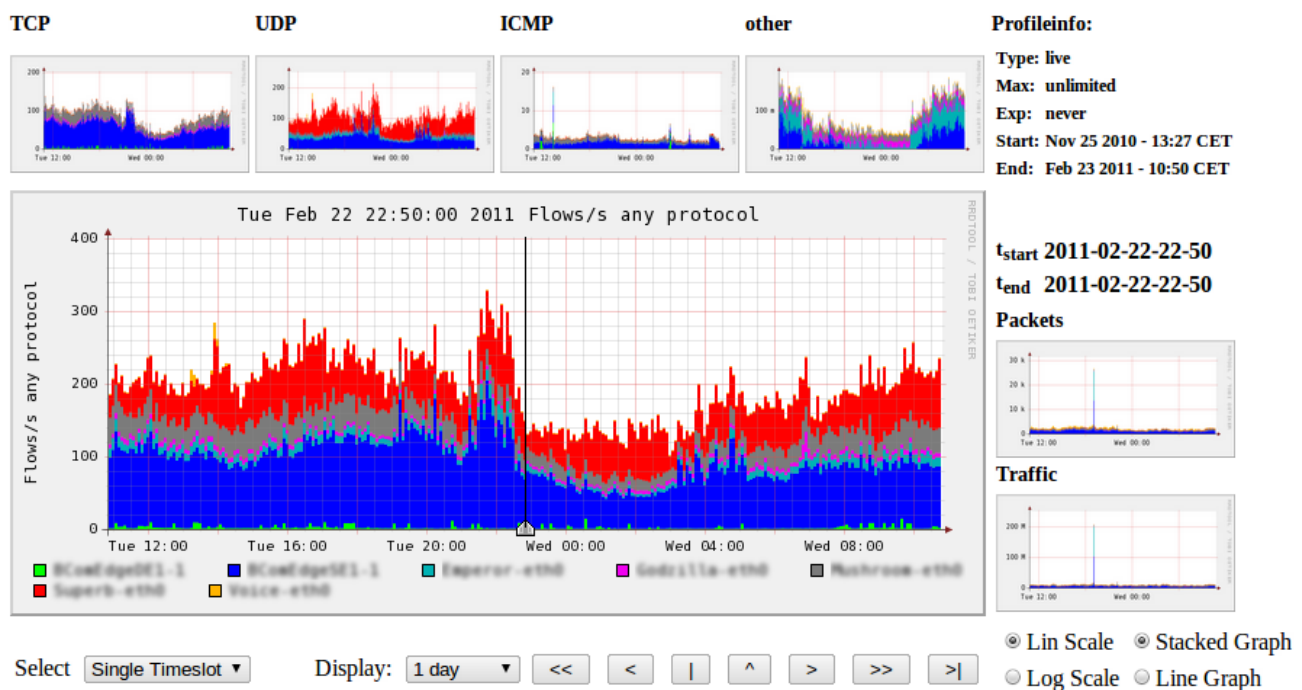


Figure 5 Ntop Flow Graph

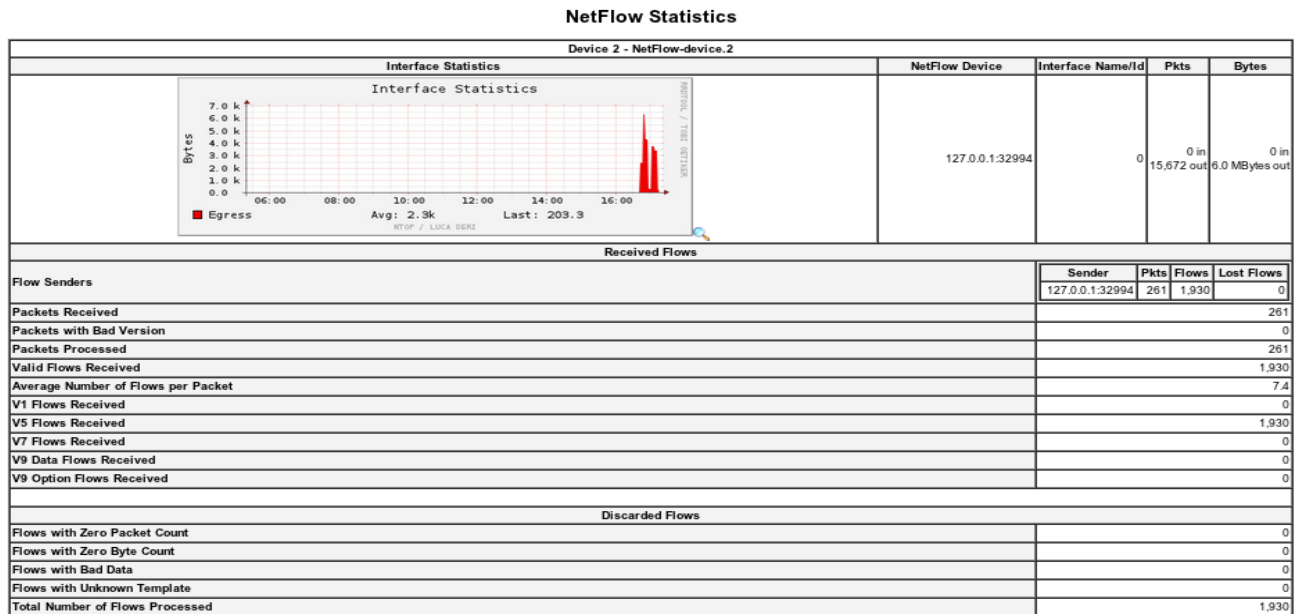


Figure 6 Ntop Netflow Statistics

1.3.4 Nfdump

Netflow analysis tools Nfdump and Nfsen are the current Netflow tools of choice. Nfdump is the Netflow display and analyzing program of the nfdump tool set. It reads the Netflow data from files stored by nfcapd and processes the flows according to given options. The filter syntax is comparable to tcpdump and extended for Netflow data. Nfdump can display many different flow element statistics. The weakness with the tools has been on the web side. The command line tools are rich, but the graphical side is lacking.

The alternative to this problem is to use nProbe_utility from Ntop. Connecting a promiscuous ethernet port to a span port on a switch, nProbe captures packets, evaluates them, and forwards Netflows to the Netflow capture utility. NProbe is useful in VOIP networks as it evaluates RTP streams and forwards helpful statistics on per flow basis.

Cisco Netflow Configuration Example:

```
telnet 10.0.0.254
```

```
conf t
```

```
ip flow-cache timeout inactive 10
```

```
ip flow-cache timeout active 5
```

```
ip flow-export version 5
ip flow-export destination 10.0.0.1 3456
```

```
interface Vlan 1
ip route-cache flow
```

```
end
copy running startup
```

```
# Test
```

```
show ip flow export
```

Ntop must be configured to receive each flow.

Ntop will handle as many flows as they are setup. Computer running ntop must be more powerful as more flows are added.

1.4 Configuring ntop to Use Syslog

-L | --use-syslog | --use-syslog=xxxx -- By default, ntop writes it's messages to stdout (the terminal).

Running ntop as a daemon (--daemon parameter), the stdout (terminal) does not exist and so messages will be dropped. It is necessary to use -L | --use-syslog parameter to save the messages into the system log (/var/log/messages).

Thus a typical startup for running ntop as a daemon is:

--daemon --use-syslog

For directing the messages to another file, for example to use "local3" to keep ntop messages separate, the entry in /etc/syslog.conf:

local3.* /var/log/ntop.log

In this case, when starting ntop it has to be started with the following option:

--use-syslog=local3 NOTE: The = is REQUIRED and no spaces are permitted.

1.5 Viewing Ntop Stats

By default ntop listens on port 3000. To view ntop stats on local machine the following url needs to be used:

http://localhost:3000/ OR http://server-IP:3000/

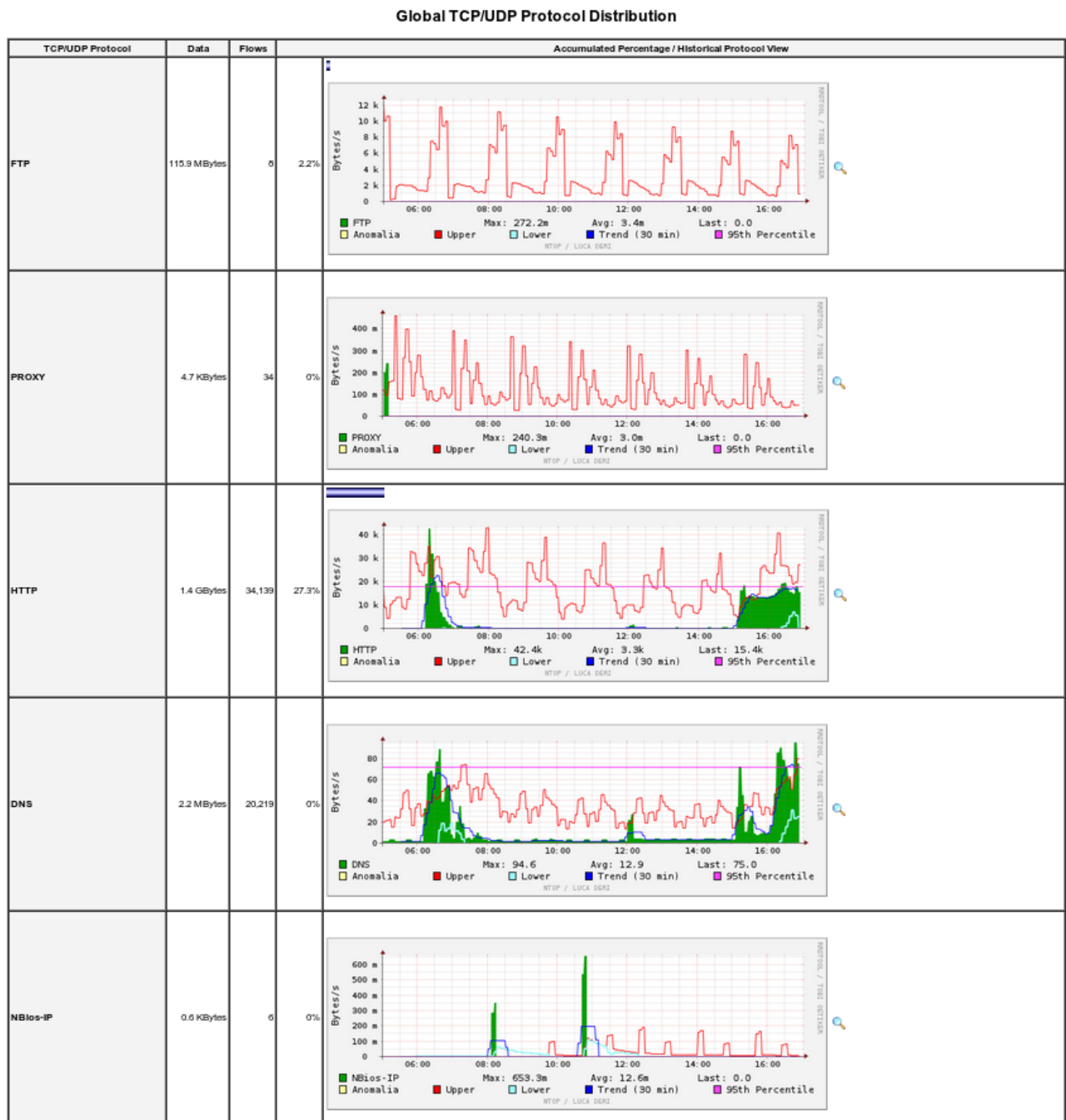


Figure 7 Ntop Global TCP/UDP Protocol Distribution Graphs

Network Load Statistics

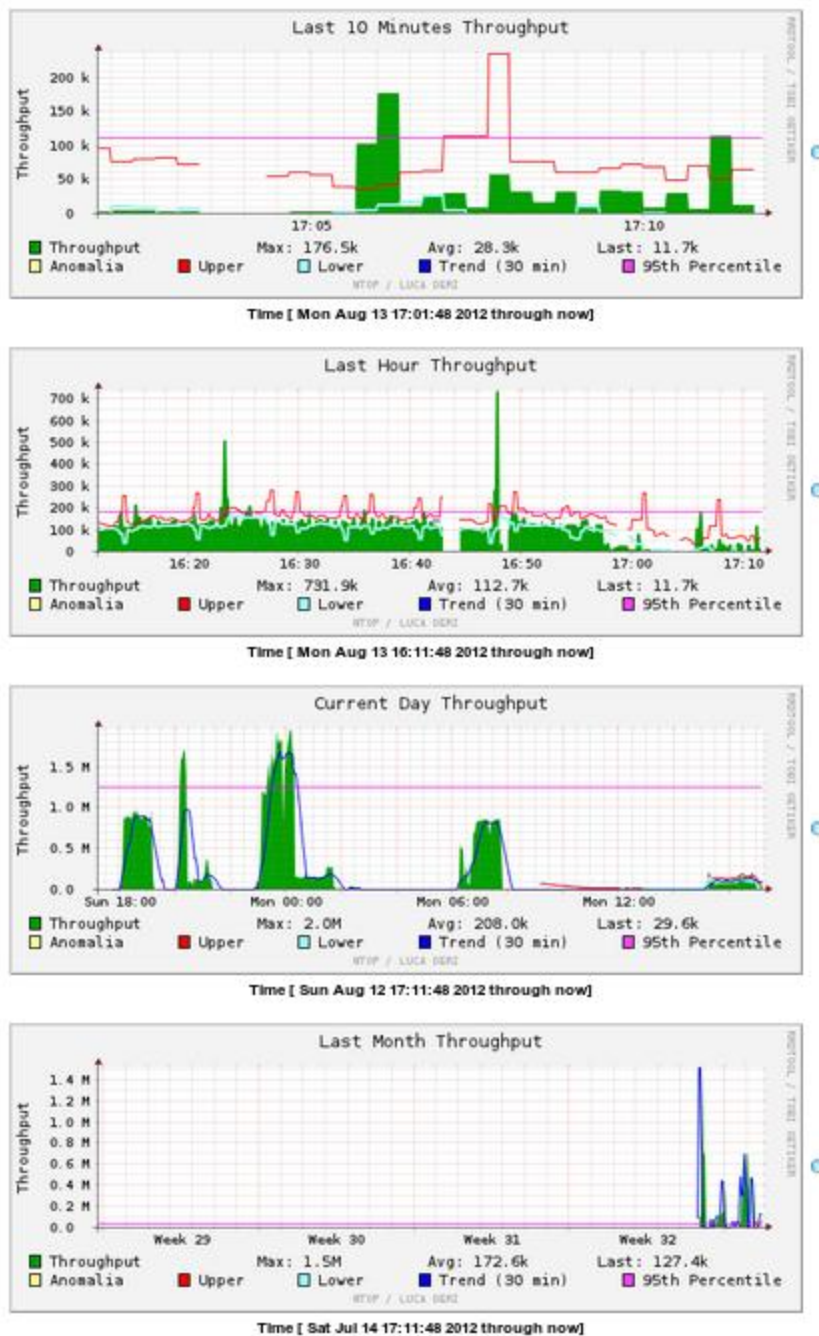


Figure 8 Ntop Network Load Statistics

CONCLUSION

There are many solutions for detecting various network activities as are specific service usages, virus alerts or surfing.

Ntop has a simple configuration structure and is easy to set up with all of its plugins. The reason for the ease of use is the fact that its data collection architecture is not client-server based which makes it robust, and it is not dependent on a working network. Simple interface minimizes redundant information, making it easy to manage, and preventing mistakes from occurring due to simple configuration interface.

Predicting aberrant behavior can alleviate future problems with access to information about how the available resources are used. Thus plans can be made to keep the availability of services. Trend analysis can also be used to get evidence of things occurring in the past.

Ntop is flexible and an easily scalable tool that is easy to install and configure. It is especially easy to install on the Debian Linux distribution where installation and package dependencies are checked automatically.

While some old school administrators might argue that particular software is poorly designed when only using the GUI, and could have been done better; what is needed is a clean GUI that enables what most of the administrators have to cope with. If there is a need for advanced capability that is rarely used by a small portion of administrators, it can be made command line and keep the GUI simple and clean.

Glossary:

Ajax is a group of interrelated web development techniques used on the client-side to create asynchronous web applications. Ajax is not a single technology, but a group of technologies.

BSD Berkeley Software Distribution (BSD), called Berkeley Unix) is a Unix OS.

BIND *Berkeley Internet Name Daemon*, named is the most widely used DNS software on the Internet. On Unix-like operating systems it is the de facto standard.

Black-box testing is a method of software testing that tests the functionality of an application as opposed to its internal structures or workings. This method of test can be applied to all levels of software testing: unit, integration, system and acceptance.

Libpcap is the library component of tcpdump. They have extracted it from the application part so other products can use the tcpdump capabilities without the need for executing a program. It has become very popular and all programs mentioned in this thesis that listen on network packets use this library. The libpcap library makes it possible for the tool (if implemented correctly) to read or write tcpdump data files.

POSIX an acronym for "*Portable Operating System Interface*" is a family of standards specified by the IEEE for maintaining compatibility between operating systems.

Tcpdump is a command line tool for network packet collection from a network interface. Normally it prints out the packet headers that match a specific Boolean expression but you can also instruct it to display the entire packet in different output formats.

TCL (originally from "*Tool Command Language*", but conventionally spelled "**Tcl**" rather than "**TCL**"; pronounced as "tickle" or "tee-see-ell") is a scripting language.

White-box testing is a method of testing software that tests internal structures or workings of an application, as opposed to its functionality (i.e. black-box testing). In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases.

Abbreviations:

API	Application Program Interface
ARP	Address Resolution Protocol
BI	Business Intelligence
BPF	Berkeley Packet Filter
CAM	Content Addressable Memory
CIA triad	Confidentiality Integrity and Availability
CISO	Chief Information Security Officers

CMMI	Capability Maturity Model Integration
COBIT	Control Objectives for Information and Related Technology
CPI	Cost Performance Index
CVSS	Common Vulnerability Scoring System
DAS	Direct Attached Storage
DHS	Department of Homeland Security
DoS	Denial of Service
GRS	General Records Schedule
GUI	Graphical User Interface
ICMP	Internet Control Message Protocol
IDS	Intrusion Detection System
IEC	International Electrotechnical Commission
IP	Internet Protocol
IPS	Intrusion Prevention System
IPsec	Internet Protocol Security
ISACA	Information Systems Audit and Control Association
ISF	Information Security Forum
ISMS	Information Security Management System
ISO	International Standardization Organization
IT	Information Technology
ITIL	Information Technology Infrastructure Library
MAC	Media Access Control
MD	Message Digest
NAS	Network attached Storage
NFAT	Network Forensic Analysis Tools
NFS	Network File System
NIST	National Institute of Standards and Technology
NSA	U.S. National Security Agency
NTP	Network Time Protocol
ORMF	Operational Risk Measurement Framework
ORMS	Operational Risk Measurement System
OS	Operating System
PCI-DSS	Payment Card Industry Data Security Standards
PDA	Personal Data Appliance
PDCA	Plan Do Check Act
PHP	Personal Home Page, server side scripting language
RAM	Random Access Memory
RARP	Reverse Address Resolution Protocol
RFC	Request For Comment
RMON	Remote Network Monitoring
SAN	Storage Area Network
SDL	Security Development Lifecycle
SEC	Securities and Exchange Commission
SEI	Software Engineering Institute
SEM	Security Event Management
SHA	Secure Hash Algorithm

SHS	Secure Hash Standard
SIEM	Security Information and Event Management
SLE	Single Loss Expectancy
SNMP	Simple Network Management Protocol
SOX	Sarbanes Oxley Act
SPI	Schedule Performance Index
SQL	Structured Query Language
SSID	Service Set Identification
SSL	Secure Socket Layer
stderr	Standard Error
stdout	Standard Output
TCP	Transmission Control Protocol
TLS	Transport Layer Security
UDP	User Datagram Protocol
UTC	Coordinated Universal Time
VLAN	Virtual Local Area Network
W3C	World Wide Web Consortium
WAP	Wireless Access Point
XML	Extensible Markup Language

Web sources used:

Dave Dittrich, Basic Steps in Forensic Analysis of Unix Systems, Available at:
<http://staff.washington.edu/dittrich/misc/forensics/>
 NIST SP 800-61, Computer Security Incident Handling Guide, Available at:
<http://csrc.nist.gov/publications/nistpubs/index.html>
 [RFC 3227] Guidelines for Evidence Collection and Archiving, Available at:
<http://www.ietf.org/rfc/rfc3227.txt>
 [OpenSource] The Open Source Page: <http://www.opensource.org/>
 [FSF] Free Software Foundation. GNU General Public License: <http://www.gnu.org/copyleft/gpl.html>
 [Sun] Sun Microsystems: www.sun.com
 [IETF] Internet Engineering Task Force. RFC Archive: <http://www.ietf.org/rfc.html>
 [The U.S. Intelligence Community (IC)] <http://www.intelligence.gov>

REFERENCES

- ¹ John Radatz, The IEEE Standard Dictionary of Electrical and Electronics Terms, Ed. IEEE, 1996
- ² Edward G. Amoroso, Fundamentals of Computer Security Technology, Prentice Hall, 1994
- ³ A John D. Howard, Thomas A. Longstaff, Common Language for Computer Security Incidents, Sandia Report, SAND98-8667, Sandia National Laboratories, Albuquerque, New Mexico, 1998
- ⁴ R. J. Ellison, D. A. Fisher, R. C. Linger, H. F. Lipson, T. Longstaff, N. R. Mead, Survivable Networks Systems: An Emergency Discipline, CERT Technical Report, Carnegie Mellon University, Pittsburgh, Pennsylvania, 1999
- ⁵ Incident Reporting Guidelines, CERT Coordination Center, available at <http://www.qcert.org/EN/ReportIncidents/Pages/IncidentReportingGuidelines.aspx>
- ⁶ Bruce Schneier. Attack trees: Modeling security threats. Dr. Dobbs's journal, December 1999.
- ⁷ Karen Kent, Murugiah Souppaya, NIST SP800-92, Guide to Computer Security Log Management, Gaithersburg, 2006
- ⁸ Karen Kent, Murugiah Souppaya, NIST SP800-92, Guide to Computer Security Log Management, Gaithersburg, 2006
- ⁹ FIPS 180-2, Secure Hash Standard, available at www.itl.nist.gov/fipspubs/fip180-1.htm
- ¹⁰ Info Security Forum, Standard of Good Practices, pg 162, March 2003
- ¹¹ BT Technology Journal, pg2, Vol 25 No 1, January 2007
- ¹² Jones A., A Framework for the Management of Information Security Risks, BT Technology Journal, Vol 25 No 1, January 2007
- ¹³ Mark Colan, IBM developerWorks, Service-Oriented Architecture, April 2004, Available at www.ibm.com/developerworks/library/ws-soaintro/index.html
- ¹⁴ Mark Endrei, Jenny Ang, Ali Arsanjani, Sook Chua, Philippe Comte, Pål Krogdahl, Dr Min Luo Tony Newling. Patterns: Service-Oriented Architecture and Web Services, IBM Redbooks, Jul 2004
- ¹⁵ Ross Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, Wiley, 2008
- ¹⁶ Cecilia Phan. Service Oriented Architecture security challenges and mitigation strategies. Military Communications Conference (MILCOM 2007), pages 1–7, IEEE Computer Society, October 2007
- ¹⁷ Michael Howard, A Look Inside the Security Development Lifecycle at Microsoft, MSDN, November Issue 2005
- ¹⁸ Chris Wysopal, Lucas Nelson, Dino Dai Zovi, Elfriede Dustin, The Art of Software Security Testing: Identifying Software Security Flaws, Symantec Publishing, 2006
- ¹⁹ Microsoft Developers Network, <http://msdn.microsoft.com/en-US/library/ee823878%28v=cs.20%29.aspx>
- ²⁰ Center for the Study of Intelligence, Psychology of Intelligence Analysis, <https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/books-and-monographs/psychology-of-intelligence-analysis/art8.html#rft52>
- ²¹ B. Blakely, The emperor's old armor. In New Security Paradigms Workshop, pages 2,16, Sept. 1996.
- ²² Common Vulnerabilities and Exposures (CVE), available at: <http://www.cve.mitre.org/compatible/product.html>
- ²³ Talisker Security Wizardry Portal, <http://www.networkintrusion.co.uk/forntetools.htm>
- ²⁴ National Archives and Records Administration, General Records Schedule 24, Information Technology Operations and Management Records, Available at: <http://www.archives.gov/records-mgmt/grs/grs24.html>
- ²⁵ Defense in Depth, A practical strategy for achieving Information Assurance in today's highly networked environments, Available at: http://www.nsa.gov/ia/_files/support/defenseindepth.pdf
- ²⁶ Dan Rathbun, Using Syslog in a Microsoft & Cisco Environment, SANS Institute Reading Room, June 27, 2003
- ²⁷ Jacobson, V., Leres, C., and McCanne, S. tcpdump, Lawrence Berkeley National Labs, Available at <ftp://ftp.ee.lbl.gov/>
- ²⁸ Sun Microsystems Inc., Snoop UNIX man pages (SunOS 5.6).
- ²⁹ Deri, L. and Suin, S. Ntop: beyond Ping and Traceroute. In Proceedings of the DSOM'99, Zürich, Switzerland, October 1999
- ³⁰ Combs, G., The Ethereal Network Analyzer, Available at: <http://www.wireshark.org/>
- ³¹ Waldbusser, S., Remote Network Monitoring Management Information Base, IETF STD 0059, May 2000.
- ³² Stallings W., SNMP, SNMPv2, SNMPv2 and RMON 1 and 2, Third Edition, Addison Wesley, Sept. 1999.
- ³³ Brownlee N., NeTraMet 4.2 Users' Guide, Information Technology Systems & Services, The University of Auckland, New Zealand, August 1998
- ³⁴ Network Flight Recorder Inc., Network Flight Recorder. Available at <http://www.nfr.net>
- ³⁵ Deri, L., Suin, S. and Carbone, R. Ntop - Network Top, Available at: <http://www.ntop.org/>