Master’s Thesis

Intrusion detection for grid and cloud computing

By

Vamsi Krishna Popuri

LiTH-ISY-EX--11/4456--SE

2011-08-25
Master’s Thesis

Intrusion detection for grid and cloud computing

By

Vamsi Krishna Popuri

LiTH-ISY-EX--11/4456--SE

2011-08-25

Examiner & Supervisor
Dr Viiveke Fåk
Linköping Institute of Technology
In today’s life providing security has become more cumbersome because of all the malicious possibilities in data transmission, so we need a system which makes data transmission more secure beyond encryption, passwords and digital signatures. The system that we are discussing in this thesis is an Intrusion Detection System, which is a platform that provides security in the distributed systems.

This paper also attempts to explain the drawbacks in conventional system designs, which results in low performance due to network congestion and less data efficiency. We consider cloud and grid computing systems to improve the performance of the system. Cloud systems are characterized by a main server and other connected servers which provide certain services. Cloud systems, especially public cloud systems are prone to intrusions and care must be taken to secure the system. The emphasis in this thesis is to make cloud systems secure using intrusion detection system. Intrusion detection can be performed using either behaviour based or knowledge-based techniques or both. We use UML as a tool to design the system, which helps in reducing the design complexity.

Keywords
Intrusion detection system, Grid computing, Cloud computing, IAAS, SAAS.
Acknowledgements

Firstly, I would like to thank my parents for their love and continuous support and the strength they have given to me for my life.

I would like to thank Dr. Viiveke Fäk for giving me this opportunity to perform my master thesis at ISY Department and also for her guidance, support, encouragement and valuable suggestions during the semester period. Her assistance helped me in achieving each and every task in the thesis. She was very kind hearted who always answered my questions even when she was busy in her work, helped us in gaining the knowledge and completion of the thesis.

I would like to thank Viiveke Fäk, Siva subramanyam, Bharath suri, Saif K. Mohamed, T V K chaitanya, Anvesh Mondem, Raj Kumar Kallem and Sampath Kumar for reviewing the document.

Your’s
Vamsi Krishna popuri
Abstract

In today’s life providing security has become more cumbersome because of all the malicious possibilities in data transmission, so we need a system which makes data transmission more secure beyond encryption, passwords and digital signatures. The system that we are discussing in this thesis is an Intrusion Detection System, which is a platform that provides security in the distributed systems.

This paper also attempts to explain the drawbacks in conventional system designs, which results in low performance due to network congestion and less data efficiency. We consider cloud and grid computing systems to improve the performance of the system. Cloud systems are characterized by a main server and other connected servers which provide certain services. Cloud systems, especially public cloud systems are prone to intrusions and care must be taken to secure the system. The emphasis in this thesis is to make cloud systems secure using intrusion detection system. Intrusion detection can be performed using either behaviour based or knowledge based techniques or both. We use UML as a tool to design the system, which helps in reducing the design complexity.
# Table of contents

Acknowledgements .......................................................................................................................... 5

Abstract ........................................................................................................................................... 7

List of figures .................................................................................................................................... 12

1. Introduction ................................................................................................................................... 15
   1.1 IDS introduction ............................................................................................................................. 15
   1.2 Project task .................................................................................................................................... 15
   1.3 IDS survey ..................................................................................................................................... 15
   1.4 Target Audience ............................................................................................................................. 16
   1.5 Organization of the thesis .............................................................................................................. 16

2. Computer Security Background ................................................................................................. 17
   2.1 Computer security ........................................................................................................................ 17
   2.2 Components .................................................................................................................................. 17
   2.3 Threats .......................................................................................................................................... 17
   2.4 Types of threats against computer security ................................................................................... 17
   2.5 IP spoofing attacks & DNS attacks ............................................................................................... 18
   2.6 Examples of well known, successful attacks against major services ............................................ 19
   2.7 Security counter-measures ............................................................................................................ 19
   2.8 policy and mechanism ................................................................................................................... 20
   2.9 Cybercrimes in Sweden/India ........................................................................................................ 20

3. Intrusion Detection System Background ...................................................................................... 23
   3.1 IDS Overview ................................................................................................................................ 23
   3.2 Signature/pattern Based IDS ......................................................................................................... 23
   3.3 Anomaly/Heuristic Based IDS ...................................................................................................... 24
   3.4 Network Based IDS ...................................................................................................................... 24
   3.5 Host Based IDS ............................................................................................................................. 24
   3.6 Intrusion detection architecture .................................................................................................... 25
List of figures

Figure 1: Norton cyber-crime report in India ................................................................. 21
Figure 2: Architecture of grid and cloud computing intrusion detection ......................... 25
Figure 3: IDS installed in the network ............................................................................... 33
Figure 4: Cloud computing architecture .......................................................................... 34
Figure 5: Grid computing architecture ............................................................................ 34
Figure 6: Use case diagram related to the project ............................................................. 38
Figure 7: Class diagram related to the project .................................................................. 38
Figure 8: User login page .................................................................................................. 41
Figure 9: File search is similar to behaviour based ........................................................... 42
Figure 10: File upload is similar to knowledge based ......................................................... 44
1. Introduction

1.1 IDS introduction

Intrusion Detection systems is a fast growing technology in the computer security field. It has mainly three components [1].

- Information Source.
- Analysis Engine.
- Alert.

An IDS needs an information source for monitoring the traffic in the network. The information source contains packets, log files, I/O operations etc. [1]

The analysis engine is the brain of the IDS. The main function of an analysis engine is to detect the intrusions in the network. It replies to the attack and alerts the network for similar events.

When it detects an intrusion, an IDS alert sends a message to the network or system administrator [1].

1.2 Project task

The task for this thesis was to study security in grid and cloud computing and to construct a system using IDS to get better security. The implementation was to be developed for Windows XP, a 32 bit operating system

*Note: In addition to Windows there can be other OS’s, but a 32 bit operating system is compulsory.*

We focused on three main systems such as, grid systems to increase the system performance, cloud systems to increase the data efficiency and IDS methods i.e. (behavior based and knowledge based) to provide better security. Refer to the “IDS chapter” for further information.

In order to start the thesis, we studied a lot of journals that are related to IDS, grid and cloud systems etc. Refer to the “Bibliography” chapter for further information about the journals. We learned UML language, Java swings, Apache Tomcat and MS-SQL server 2000.

1.3 IDS survey

The aim of this survey is to compare firewalls and antivirus with what IDS can achieve in terms of security with the hope that it is better. To confirm this we have taken personal interviews from three people. One purpose of this interview is to create awareness for IDS. This interview has helped during our documentation work. Refer to “Appendix” for further information.
According to Cisco, a firewall is: “a system or a group of systems that enforces an access control policy between two networks” [30]. Firewalls can monitor incoming and outgoing traffic. It is free of cost or inexpensive for personal use. Latest firewalls have good human system interaction i.e. are user friendly. Firewalls are good in access control points, but the main disadvantage of firewalls is that they are not good enough in intrusion prevention. In this case, most of the organizations are keeping an IDS behind the firewall or at the host. Refer to the “Solutions” chapter for further information.

Antivirus software is used to detect, prevent and remove the malware in the system [31]. The main disadvantage of a good antivirus software is that it is much more expensive compared to IDS and also that we have to register for its subscription either monthly or yearly. Thus, users have to spend more money for good antivirus software. A few antivirus software’s are automatically updated every day. As a result, the system performance and internet connection might get slowed down. Please refer to the “signature/pattern based IDS” section for further information.

An IDS is like an alarm on your computer. It is possible to block the unauthorized persons who are accessing the data. In IDS we are discussing a few tools like signature/pattern and anomaly/heuristic. The signature based IDS cannot detect new attacks and is therefore complemented with anomaly based IDS. It can however discover deviations from acceptable use. As a result it identifies privilege abuse. These are the best tools when we compare to antivirus, firewall etc. An IDS is a system which can overcome all the above issues. IDS systems also support all the environments, no daily updates and are more effective than firewalls. Refer to the “IDS” chapter for further information [4].

1.4 Target Audience

This paper will be useful for people who are interested in the cloud computing and security field.

1.5 Organization of the thesis

The thesis is divided into nine sections. The first section explains the introduction of IDS, project task and IDS survey. Second section provides the information about computer security, their components, threats, goals, cyber-attacks in India/Sweden and common attacks in IDS etc. Third section explains the IDS features and terminology. Fourth section, explains the introduction of cloud computing and their vulnerabilities etc. Fifth section deals with the problems in the previously used techniques and gives the solution with a better compatibility issue for that problem. Sixth section provides the pictorial representation of UML diagrams and the results of my thought execution. Finally conclusion about the thesis and references are presented.
2. Computer Security Background

2.1 Computer security

Computer security is meant to prevent data access by unauthorized users and to prevent other damages like destruction of data, DoS etc. Security can be broken by threat agents such as intruders, crackers and third party users”.

2.2 Components

Computer security normally has three basic requirements

- **Confidentiality**: Only legitimate users can get the data.
- **Integrity**: Only legitimate users can modify the data.
- **Availability**: Legitimate users can always use the data.

2.3 Threats

A threat is a possible, unwanted event [26]. Threats are divided into four types

- **Disclosure**: Illegitimate users use the data.
- **Deception**: A legitimate user accepts incorrect data.
- **Disruption**: The expected system functions are disrupted.
- **Usurpation**: Illegitimate users act as a supervisor in a part of the system [20].

2.4 Types of threats against computer security

2.4.1 Snooping

Illegitimate users access the information via any communication.

As an example, let us consider two people who are talking to each other through the land line (land phone). An intruder listens to the information by using wiretapping. This threat comes under the category of Confidentiality.

2.4.2 Modification/Alteration

Illegitimate user modifies the information via any communication.

Example: Man-in-the -middle attack ‘A’ sends a message to ‘C’. In between an intruder reads the message and modifies the message sent to ‘C’. ‘A’ and ‘C’ do not know about the intruder. This threat comes under the category of Integrity.
2.4.3 Spoofing/Masquerading

The system has falsely accepted data as coming from the authorized users. This threat comes under the category of Integrity.

Example: 'A' wants to download “A.doc” file from the server, but the cracker gives a different file to 'A'. This is also known as spoofing.

2.4.4 Repudiation of origin

The user denies having sent information.

Example: A customer sends a letter to the dealer about the agreement of payment for the product. A dealer sends the product and asks for payment to the customer. In this case the customer denies having sent information. If the dealer cannot prove that the letter came from the customer, then attacks succeed.

2.4.5 Denial of Reception

The user denies having received information.

Example: A customer buys a product through online payment method. The customer made a payment for the product. The dealer sends the product and the customer keeps on asking the dealer as to when will he get his product. If the customer has already received the product then the query represents denial of receipt attack.

2.5 IP spoofing attacks & DNS attacks

In IP spoofing the intruder creates some packets with the help of duplicate IP address. The two attacks discussed below come under this category.

2.5.1 TCP sequence number prediction

TCP means Transmission Control Protocol or Transfer Control Protocol. Client and server have a sequence number that is allocated by TCP. If the intruder predicts the TCP sequence number, he will create data packets with duplicate IP address and steal the TCP connection [17].

2.5.2 DNS poisoning through sequence prediction

DNS means Domain Name System. DNS servers will "recursively" resolve DNS names. At first the client requests an information from the DNS server and in the next step the DNS server itself acts as a client and requests information from the next DNS server in the recursive chain. The sequence numbers are predictable. So, an intruder could come in between the client and the DNS server and forge the information and send to the client and request information from the DNS server as a client and shares the information to satisfy the intruder clients [32].
2.5.3 DNS cache poisoning

Basic DNS operation
A user types a website name into the browser. The browser asks the DNS server at the address and takes the user to the desired website.

DNS cache operation
A local DNS server makes this faster. They have cache addresses. So the request doesn’t go to the internet every time. If the request is not in the cache, the local DNS server forwards the request to the internet DNS.

A hacker sends a request to a local DNS, the query is then forwarded to the internet DNS, and the hacker floods the local DNS with fake responses. The local DNS server finds the malicious sites in its cache and forwards the user to the malicious sites. Such attacks are called pharming attacks. They send people to websites that contain tools for stealing information or infecting systems [33].

2.6 Examples of well known, successful attacks against major services

Apache and IIS (Internet information server) are web servers, which have been susceptible to attacks. For example, Code Red is for IIS and Linux. Slapper is for Apache [17].

IIS/Linux have been susceptible to Code Red worm and the Apache to the Slapper worm. Initially, Slapper worm attacks one system and then it spreads to other systems. In the meantime these worms have codes that will create peer-to-peer attacks in the network. Hence, the infected systems can launch DDOS attacks [18].

Code Red worm is a computer worm. It attacks the running IIS web servers. Chinese were the first to hack these kinds of attacks [19].

2.7 Security counter-measures

2.7.1 Prevention

Intruders fail in their attempts.

Example: An intruder trying to download something on someone else’s system. But he is not allowed access to the internet. As a result, the intruder fails to attack the system.

2.7.2 Detection

Intruders succeed to attack the system but the attack is detected. The detection mechanism is useful for analysis of the attacks as it reports to the system for further possible actions.

Example: Let us assume an intruder is trying to download something on someone else’s system and he has access to the internet too. As a result, the intruder succeeds to attack the system. Therefore by using detection mechanism we can detect the attacks in the system.
2.7.3 Recovery

Stop the attacks and repair the system when data is altered/deleted.

Example: Let us assume the same scenario as above, if the intruder modifies the data, by using recovery method we can recover the original data.

2.8 policy and mechanism

Security policy means a policy is set by the system owner, which is supposed to be followed by developers as well as users.

Example: In a company, the manager assigns a deadline for the project. Here, deadline refers to the policy.

Security mechanism is a tool for keeping the data security.

Example: After a project is completed; the manager feels to secure the project using some tools, where those tools refer to security mechanism. It can be simply said that IDS is the tool for securing the data.

2.9 Cybercrimes in Sweden/India

These are just single examples of cybercrimes.

Cybercrimes in Sweden

On August 23\textsuperscript{rd}, 2010 political parties’ official site were hacked by intruders. According to the investigation it was found to be a DDoS attack (Distributed Denial of Service means authorized users are unable to access their resources). After that incident, the intruders cracked the Swedish Social Democratic Youth official site and data was stolen from the website [14].

Cybercrimes in India

On March, 2010 Income tax department official sites were hacked by intruders. According to the investigation it was a phishing attack. One research on cybercrime, tells phishing attack is common in India. An attacker can steal any confidential information such as bank passwords, CVN numbers, credit card expiry dates etc[15].

For concluding this concept, according to the government news- they have not mentioned which tool that has been used for security purposes for the data in the network. Therefore in my view, they have not used IDS for providing security in the network via any communication because an IDS requires a knowledgeable manager, which many companies lack.

The hacked official sites in Sweden/India might have used firewalls and antivirus software’s for providing security. Hence, all network traffic doesn’t go via firewall. For example there may be a modem on a system. So risks are high with a firewall. On the other hand, antivirus
software’s will not be able to detect new threats. Refer to the concept “Signature based IDS” for further information.

**Norton Cybercrime Reports: The Human Impact**

This survey was conducted in India by the “Norton” group. They released cybercrime report on the Norton official website. In this survey more than 14 countries and ten thousand people were participated. Full statistics are given in the picture below.

![Norton Cybercrime Report: The Human Impact](image_url)

**Figure 1: Norton cyber-crime report in India**
3. Intrusion Detection System Background

3.1 IDS Overview

Nowadays everyone has to accept the fact that the percentage of intruders is increasing day by day as technology advances. But at the same time we have a lot of chances to keep the data in security. We have a lot of free software on the internet to protect data from intruders such as antivirus software etc.

Coming to this thesis, Intrusion Detection System (IDS) is the tool for monitoring traffic in the network. Whenever it identifies vulnerabilities in the network, it sends a report to the system or network administrator. IDS is like an alarm on your computer. It makes it possible to block the unauthorized persons from accessing data [3].

IDS and IPS both are software but they have different terminologies. IDS controls the Intrusion Detection process while IPS has all the features from IDS as well as tools to prevent the attacks from intruders [3].

The intrusion detection system is classified into two methods namely, behavior analysis and knowledge analysis. Behavior analysis method is done by comparing the previous behavior with the present one. It detects the intrusion by using heuristic/anomaly. Knowledge analysis method detects intrusions by using signatures/patterns. We are free to delete and modify rules at will in a knowledge based system. We refer to the next section “Signature/pattern & heuristic/anomaly” for further information [4].

*How to detect:* The two IDS tools which we are going to discuss related to the project are Signature/pattern Based IDS and Heuristic/anomaly based IDS.

3.2 Signature/pattern Based IDS

*Signature: A signature is preconfigured to identify network exploits* [5].

“Signature/pattern based IDS” is also known as “Knowledge based IDS”. It is similar to antivirus software for detecting intrusions in the system. The aim of the antivirus is to detect viruses in the system. For detecting viruses, antivirus has a database, in that virus signatures are stored. And if data match one of the signatures in the database then the system decides there is a possible virus. If an antivirus fails to detect, this means no signatures are available in the database for the current virus attack or the database is out-of-date. IDS can be placed on a network to monitor the network attacks and can also be placed on host.

*Advantages:* For signature systems, the database contains known signature patterns. If the intruder attacks the system, the database signature and the intruder signature both are compared and both signatures are the same. Hence, the system decides there is a possible intrusion [6].

False alarms ratio is less when we compare to heuristic/anomaly based IDS. For further information about what is an alarm refer to the “terminologies” chapter.
Disadvantages: There is a delay between discovering a new threat and the signature for detecting that threat is being applied to the IDS. Therefore, throughout that delay IDS will not be able to detect new threats [6].

This tool is concentrated on knowledge about the attacks, in which it depends up on the operating system and platform etc. Thus, IDS is almost connected to a given environment.

3.3 Anomaly/Heuristic Based IDS

“Heuristic /anomaly based IDS” is also known as “Behavior based IDS”. Anomaly based IDS monitors the system activities and detects the attack. Events are classified into either normal or anomalous based on the rules or heuristic instead of signatures. Anomalous means abnormal behavior or deviating from what is normal. Heuristic is nothing but a practical technique for learning, discovering and problem solving etc. Therefore by using these methods, the process moves quickly for finding a better solution and also detects incorrect system activities [7].

In the system, we have to build a profile for each user group. These profiles can be created either automatically or manually, but this procedure is not significant. Therefore, each profile can define the features of the user or users groups in the network. These profiles can be used as a baseline and it defines normal user activity. Let us assume that if the network activity is deviated too far from this baseline then a signal gives the information to the system about the attacks from intruders [34].

Advantage

This tool will identify any possible attack.

Disadvantage

This tool requires a knowledgeable person to figure out what triggered an alarm [34].

Where to detect: A detector can be placed on a network or on a host because it helps to finds the intrusion on a network or on a host.

3.4 Network Based IDS

An event that monitors, all incoming and outgoing packets on the system, when it identifies attacks, it sends a report to the system. An example is Snort

3.5 Host Based IDS

Host Based IDS monitors the packets and identifies the network attacks and crimes in a specific host computer, for example at nodes. When an attack is detected it sends a message about the event and makes the system safe from future attacks via some security enhancement. An example is OSSEC.
3.6 Intrusion detection architecture

Before reading this chapter we have to clarify one thing that, we are intending to protect the “cloud system”. Refer to the “Cloud core technologies” chapter for information about vulnerabilities in cloud computing and how IDS can prevent data from intruders in the cloud system.

The below explanation is all about the IDS parts and IDS functions.

The elements that are participating in the architecture are nodes, service, event auditor and storage devices [4].

*A node has resources which are accessed homogeneously through middleware. The middleware sets the access-control policies and supports a service-oriented environment* [4].

*In the environment a service provides its functionalities via middleware, which facilitates communication* [4].

In the architecture, the event auditor plays an important role in the system. Initially, it captures the information from different sources such as the log system, nodes, and services. After capturing the data the IDS service analyzes the data based on the intrusion detection techniques such as knowledge based and behavior based. If an intrusion is detected in the system then the IDS service uses middleware communication mechanisms for sending alerts to the other nodes. The middleware synchronizes the knowledge based and behavior based databases [4].

The storage service holds data that should be analyzed by the IDS service. Because all nodes have to access the same data in the environment. So, the middleware must transparently create a virtualization in the homogenous environment [4].

![Figure 2: Architecture of grid and cloud computing intrusion detection](image-url)
The client sends a request to the server for getting a service. In the “conventional client server” system, the client communicates with the end server directly, due to which traffic congestion or data loss etc might take place. So to overcome this issue we have implemented a proxy server, which extends the functionality of the main cloud server and is the mediator between your web browser and the end server. Initially, your web browser sends a request to the proxy server, after which the proxy server forwards the request to the end server. The end server then gives acknowledgment to the proxy server. Finally, the proxy server replies to the browser. Therefore, there is no direct communication between the user and the end server. So, HTTP request is originated from the intermediate proxy server. As a result the client computer’s IP address will be in hidden state and illegitimate users cannot access the client computer’s IP address. This type of proxy server is also known as anonymous proxy server [35] [36].

Let us assume that user wants to get a file from the main cloud server. Initially, the user sends a service request to the proxy server. Here, the proxy server checks the service request by using filtering rules based on the traffic such as IP address or protocols. If the request is validated by the filter then it forwards the service request to the main cloud server [35].

The optional choice for the proxy servers are that it can change user’s request or main cloud server acknowledgments and also send the response without contacting the end server. This is because, it ‘caches’ responses from the remote server, and returns subsequent requests for the same content directly [38].

A client communicates to the end server by using internet. So in this case a proxy server can be installed in several points anywhere between the client and the end server.

Reverse proxies: Client sends a request to the reverse proxy by using internet and then the reverse proxy forwards the request to the web server in an internal network. Hence,

Load-balancing: The reverse proxy server shares the traffic to various web servers. In this case every web server has its individual application area. As a result, the reverse proxy server may have to alter the URL’s in each web page. By that we can achieve load balancing in the network [38].

*Intrusion detection system service* helps to increase the security in the cloud system by using two methods i.e. behavior based and knowledge based service. We refer to the previous section for further information. The audited data is sent to the IDS service core, which analyzes the behavior using artificial intelligence to detect deviations. It has two sub systems namely analyzer and alert systems.

*Analyzer system* the analyzer gets audit data and examines whether a heuristic in the database is being broken, after which it sends the outcome to the IDS service. For these outcomes, IDS estimates the attack probability and if probability ratio is high then it alerts the other nodes [4].

*Alert system* in the cloud system if any one node is harmed by the intruder then alert system will alert the remaining nodes in the network regarding the attack.

[4]*Storage service* is a database. When the node receives a request or an acknowledgment, then the analyzer system will compare the node information in the storage service. Storage service
has two services namely behavior and knowledge service.

Knowledge service we used audit information for the communication system and the logging system for evaluating the knowledge service. Moreover, we are free to delete and modify rules at will in a knowledge service.

Behavior service it compares recent user actions to the usual behavior. It is divided into two types i.e. user behavior and node behavior.

User behavior is nothing but analyzing the user’s behavior. By using this method we can identify expected behavior or a severe behavior deviation [4].

Node behavior Refer to the concept “Anomaly based IDS “for further information.

Event auditor has two components for detecting an intrusion in the network as data is exchanged between the nodes and environment states. In the first component, when the data is exchanged between the nodes, audit information about the communication between the nodes is being captured. Therefore, audit data captures only the node information, but not network data. The second component is logging system, for each event in the node, a log entry is created and they have action types such as error, alert or warning. By using this approach we can easily find ongoing intrusions [4].

3.7 Terminologies

Alert/Alarm A signal that gives the information to the system about the attacks from intruders.

True Positive System gets an alert when an attack has taken place.

False Positive System gets an alert when no attacks have taken place.

True Negative There is no attacks from intruders as well as no alert from alarm.

False Negative Attacks have taken place in the system, but IDS failed to detect them.

Noise Information that causes a false positive.

Site policy IDS rules within the company.

Confidence value Value placed on IDS which is based on previous analysis to get information about how efficient it is to identify attacks.

Alarm filtering IDS is the system to detect intrusions in the system. After detecting intrusions, alarm filtering helps to analyze these attacks and compare them from false positives to actual attacks.

Attacker The unauthorized user accessing the data without having permissions.
**Masquerader** A user who does not have permissions to access the system, but still accesses the data as an authorized user.

**Misfeason** A legitimate user who uses his permissions for incorrect purposes

**Clandestine User** A user who can secretly use any authority.
4. Cloud computing technologies

4.1 Cloud computing

Cloud computing is a technology which works only on the internet; central remote servers are used to maintain data and applications. Cloud computing allows the users to use applications without installing software’s. The users can access the internet and send messages anywhere in the world. Cloud computing allows more efficient computing by centralized storage, memory, processing and bandwidth [42]. The best example is Google mail. For this, the users need not install any software or a server to use a Google mail account. The user can access internet, through which he sends messages. Therefore the servers and email software are all present on the cloud i.e. internet and these software’s and servers are managed by the cloud service provider i.e. Google [42]. Cloud computing is divided into three layers, infrastructure, software and platform. Refer to the paragraph below for further information.

In this thesis we are going to discuss IAAS (Infrastructure as a service) and SAAS (Software as a service).

IAAS: An organization or a company which provides a service. Those services may include much essential functionality such as computer networking, information storage, servers and virtualization [9].

SAAS: This is also provided by an organization. They provide software, in which you can use these services. An example is Facebook.

Before going to the cloud characteristic, even today and ongoing tomorrow there are so many existing services such as infrastructure, platform and software that are provided in the context of IAAS, SAAS. But we do not think that, these services are not consistently cloud technologies and can be provided in a number of ways [9].

4.2 NIST (National Institute of Standards and Technology) Cloud computing characteristics

Vulnerability:

Vulnerability is a prominent factor of risk. ISO 27005 defines risk as” the potential that a given threat will exploit vulnerabilities of an asset or group of assets and thereby cause harm of the organization” [10].

NIST is advancing the measurements of science, standards and technology. It is located in United States of America. The below are the most essential characteristics that are given by the NIST. These characteristics are very useful for the project work.

1. Users can manage their accounts by themselves. An example is Linköping email account. We can send emails to our colleagues without human interaction.
2. The cloud computing services are using protocols and methods. These services are accessed through the internet.

3. Cloud computing resources are utilized to provide cloud services, in which these services are using homogenous infrastructure and that are shared to all the service clients [10].

4. The resources can be increased or decreased rapidly or elastically.

4.3 Cloud Specific Vulnerabilities

1. A vulnerability is cloud specific if it is prevalent in established state-of-the-art cloud offerings [10].

2. A vulnerability is cloud specific if it is intrinsic to or prevalent in a core cloud computing technology [10].

3. A vulnerability is cloud specific if its main cause is in NIST’s cloud characteristics [10].

These are the three types of cloud specific vulnerabilities that are related to the project. The above mentioned three cloud specific vulnerability types are explained below in brief detail.

NIST characteristics Vulnerabilities

Internet protocol Vulnerability: The cloud computing services use protocols and methods. These services are accessed through the internet. Internet is the unsecure environment. Internet protocol vulnerabilities allow “Man-in-the-middle” attacks.

Man-in-the-middle: 'A' sends a message to ‘C’. In between an intruder reads the message and modifies the message sent to 'C'. 'A' and 'C' do not know about the intruder.

Very common Vulnerabilities in state-of-the-art cloud offerings

Even though cloud computing is a new system in the market, in spite of this fact, currently thousands of cloud offerings are in the market. So we are going to discuss two important vulnerabilities that are related to our project. The two main vulnerabilities are injection vulnerabilities and weak authentication schemes.

Injection vulnerabilities are attack techniques, in which the attacker exploits by editing the services or by user inputs to perform, execute and oppose the user’s aim. The below three examples are injection vulnerabilities.

1. In SQL injection, the user input has SQL code in which the code is performed incorrectly in the database backend.
2. In command injection the user input has commands, in which the commands are performed incorrectly through the operating system.

3. In cross-site scripting the user input has JavaScript, in which the JavaScript code is performed incorrectly by the infected browser.

Authentication Mechanism: The main root cause of weaknesses in current authentication mechanism are usernames and passwords, because the users are using the same passwords in different websites and weak passwords (123, 999) etc.

In the present era, most of the web applications in state-of-the-art-cloud services are using usernames and passwords as an authentication mechanism.

Core cloud computing technology Vulnerabilities

In today’s world if the cloud system develops, then the core cloud computing technologies will elaborate. Web applications and services and virtualization IAAS offerings are the cloud computing technologies. The below two are the best examples of vulnerabilities in cloud computing technology that are related to our project.

The possibility that an intruder gets away successfully from a virtualized environment lies in virtualization’s very nature. Virtual machine escape is an exploit; an attacker can execute arbitrary code on a virtual machine. So in this case the operating system executes within itself to break out and interact directly with the hypervisor. As a result, an attacker can gain access to the host operating system and all other virtual machines running on that host [39]. In this case we can imagine that this type of vulnerability is a high risk in cloud systems.

Note: Hypervisor is the mediator between the virtual machines and the host operating system.

Before moving to Session Hijacking concept, we have to know about cookies. “Cookies” are not software and they do not detect malware. But they are used by spyware; it tracks the user’s browsing actions via cookies.

Magic cookie refers to a packet of data that a program gets and then forwards again without modifying the data or we can say simply, a short packet of data passed between communicating programs [40].

HTTP cookie or cookie An origin website can send information to a user’s browser which they can returns the information to the origin website. Here information refers to authentication and identification of a user’s session etc. [41].

Session hijacking is an attack technique. A user does not have permission to access the system, but he gains the services in the system. The attackers steal a magic cookie that can be used to authenticate a client to a remote server. The fact is that these types of attacks are most common in websites because the web applications run over HTTP from a web server to a web browser. Therefore, HTTP cookies are used to maintain a session on so many websites. So in this case the attacker can steal cookies easily by using an intermediate system and they can access the user’s web account [42]. Session hijacking produces anomalies in the network traffic. Therefore, by using IDS we can detect the attack.
5. Architecture for the created system

5.1 Problems
Suppose that two users, Client1 and Client2 use the “conventional client-server” system and both are simultaneously sending multiple requests to the same server. In this case, acknowledgment time is increasing from the server to users, due to heavy traffic in the network. If the acknowledgment time is increasing, perhaps data will be lost. The second problem is, multiple users are requesting the same data at the same time in server. So server might not give equal performance for all the systems. The third problem is, there is no security mechanism. Security mechanism is the main topic in this thesis.

5.2 Solution

![Figure 3: IDS installed in the network](image)

The main cloud server is connected to all the proxy servers and it maintains the index of the data information by all the proxy servers. At first, the user sends the request to the main cloud server about the “hi.doc” file. Then the main cloud server will forward the request to the exact proxy. The proxy server gives back the information to the main cloud server. Finally, the proxy server will send the file to the user. In this case the main cloud server can handle multiple user requests. The proxy server helps to decrease the acknowledgment time and gives the reply to the users in a proper manner. A proxy server can be the other way around: a server which receives and queues requests, makes format changes and packet divisions etc for information from just one single information carrying server. The network administrator cannot monitor all the clients, so it will depend on the IDS which gives the alerts to the remaining “cloud server”. In this case we can hide the data from the intruders. In cloud computing, there is more possibility to hack the data. So for preventing data from hackers. We are providing a security mechanism called IDS. For further information about IDS refer to chapter 3. An IDS can be placed on a network for monitoring and detecting the intrusions and can be placed on a host. An IDS is placed as shown in figure 3. An IDS is placed in the proxy server. If the detects then alert the other proxy servers and the main server. Refer to “figure 4” for further information.
The client sends a request to the main server and it forwards the request to all the proxy servers for processing. All the proxy servers give acknowledgement to the main server and it replies to the client. This is working procedure for grid computing. Let us consider an example; initially the client triggers a request at time “n”, the server accepts the request at time “n+1” and it equally divides the task (15000 iterations) and assign them to proxy servers. Let us consider in this case there are three proxy servers and each will process 5000 iterations parallel and get back to the main server at time “n+10”. After getting processed output from all the proxy server’s main server provides output to the client at the time “n+12”. If the same experiment is carried out in conventional “client-server” system the client will trigger the request at time “n”. Afterwards main server starts processing the task (15000 iterations) and it get back to the client at the time “n+31”. Hence, grid system has processed the task (15000 iterations) at time n+12 seconds, where as conventional “client-server” system has processed the task (15000 iterations) at n+31 seconds. Therefore, for comparing these two systems we can say, the grid computing has less computation time than the conventional “client-server” system.
6. Unified Modeling Language Background

6.1 UML introduction

UML is a language for reducing task complexity. In this project we have used UML for designing the view of the system. Refer to “class diagram & use case diagram” subsection for further information.

A software development method includes a process and a modeling language. The software development process is a highly complex process due to many disciplines such as code generation, testing, deployment, maintenance, implementation etc. So to reduce the complexity in the process, we need a language which makes the process efficient of good quality and flexible. A language what we are talking about is UML (Unified Modeling language). It is called a modeling language, but not a method. UML plays a vital role to control the complexity in the software development process. UML is a standard language as well as an easy language; everyone can easily understand the graphical design with a little knowledge about UML. It is used to visualize, specify, construct and document the artifacts of a software intensive system [20].

Visualize draw a picture for the task that you have in your mind.

Specify a complete and correct procedure for the task such as analysis, design, implementation procedure etc.

Construct combining all the tasks information.

Document official information on paper such as design, source code, prototypes, architecture requirements etc.

The UML language means a complete plan to accomplish the task or blueprint or the complete procedure for the task like where to start, how to implement, what are the methods, where to end the task etc. UML supports software development methods and principles, but that does not mean it specifies the process. We are using UML in a different perspective, for example database management system, business process modeling, web services, banking services, transportation etc. In the below paragraph we discuss two examples [21].

In database projects, before starting the work we have to visualize EER (Enhanced entity relationship) diagrams for improving the efficiency and reducing the complexity of the task. We need for example ER (Entity relationship), EER diagrams.

In business modeling, UML helps to analyze and improve the process. So in this case quality will be improved. An example is thesis time plan.

In this thesis we need the UML language for designing the concepts. Concept designing plays a vital role in every project. By using this language we can achieve efficiency, flexibility and complexity reduction in the tasks. As we said earlier, the UML language looks like a graphical design. Hence, everyone can get a clear idea of how the system looks and its functionalities. Coming to the subject, a cloud system is a very sensitive and tricky part. Therefore for reducing the complexity and to improve the quality in the process we have used
the UML language. It has two specific implementation languages: forward and reverse engineering. Forward engineering refers to design of a new system. Reverse engineering refers to redesign of an existing one. In our project forward engineering is used, because we are designing a new system. We started the task with visualization then we specified the correct procedure and later we combined all the information. Refer to figures 6 & 7 for further information.

6.2 Diagrams in the UML

UML consists of class diagrams, object diagrams, use case diagrams, sequence diagrams, collaboration diagrams, state chart diagrams, activity diagrams, component diagrams and deployment diagrams. But we will be focusing only on the class diagram and use case diagram.

We shall discuss about a few basic functions before moving on to UML diagram.

**Class** A class is a set of objects sharing similar constraints. A class is represented in the form of a rectangle which contains attributes and operations [22].

**Interface** An interface is a set of operations that defines a class. An interface is represented in a circular shape.

**Collaboration** It is a cluster of things, which work together to provide some functionality within the system. Collaboration is represented in the form of an ellipse with dashed line, which contains a name.

**Use cases** the system performs a set of actions to achieve a specific result. A use case is represented by an ellipse with solid lines.

**Active classes** an active class is just like a class, but here the objects have a few threads, so that they can start an event with control activity. An active class is represented by a rectangle with dark lines, which contains attributes and operations [22].

**Components** A component refers to the physical and replaceable part in a system. A component is represented by a rectangle with tabs, which contain a name.

**Nodes** A node refers to a physical element that exists at run time. A node is represented by a cube, it contains a name [22].

**Messages** The messages are switched between the elements for getting a specific result. A message is represented by a directed line with a name on it.

**States** The state is nothing but the sequence of states of an object. The state is represented in the form of rounded rectangle which contains a name.

**Packages** It is a collection of structural and behavior things. A package is represented by a tabbed folder, it contains a name or content.
Notes A note is just like a comment. A note is represented by a dog-earned corner [22].

Dependencies Dependency is defined as the case where two elements have a relationship between each other. Change in one element affects the other one. A dependency is represented in the form of a dashed line.

Associations An association refers to a set of links that connect elements of a UML model. An association is represented in the form of a solid line.

Generalizations A generalization is defined as an inheritance relationship within objects. A generalization is represented by a solid line with a hollow arrow head pointing to the parent.

Realizations A realization is defined as the relationship between two elements. Both the elements are sharing the work. A realization is represented by a dashed line with a hollow arrow head pointing to the parent.

6.2.1 Use case diagram

The user sends a request to the system, to which it responds. It is used to visualize, specify and document the actions of an event. Use case diagrams include actors, use cases and their relationship (dependency, generalization etc.).

Use case Modeling techniques are:

Modeling the context of a system every system consists of response (inside) and request (outside). For example in a student identification card, the inside things include card validation process, unauthorized user access detection system and much more. Whereas the outside things are the user and the users interactions with the system context. Context refers to the environment of the system where the system works.

Modeling the requirements of a system Requirements are nothing but a design of a system such as what are the requirements needed for the system, the system behavior, how the system performs and how it responds etc.

Forward engineering Visualizing, specifying and constructing the new system is known as forward engineering.

Reverse engineering The concept starts with an existing system, understanding and re-designing the system is known as reverse engineering.
The below use case diagram is related to the project.

![Use case diagram](image)

**Figure 6: Use case diagram related to the project**

### 6.2.2 Class diagram

A class diagram addresses the static design view of a system. It is for visualizing, specifying, documenting and also constructing via forward engineering and software reverse engineering. It includes classes, interfaces and collaborations and their relationships (dependency, association, realization etc.).

The below class diagram is related to the project.

Note: We already have given the explanation about the diagram in the “solutions” chapter.

![Class diagram](image)

**Figure 7: Class diagram related to the project**
7. Results

The task was to find intrusions in the host or network by using tools both behavior based and knowledge based. After studying the background, a system as described here was created. Before moving to the results screenshots, we give a brief explanation about the main functions in the sections given below. Initially, we have to run main servers and proxy servers, then after successfully running the main server and all proxy servers, we switch to functions. The main functions are: User login helps to secure the system from unauthorized users. Refer to “figure 8” for further information.

IDS main frame page, which handles knowledge based and behavior based functions. In this project, behavior based is used for to file search and knowledge based is used for to file upload. We are going to discuss briefly how they relate to each other in the paragraphs given below.

The working principle of behavior based is similar to anomaly based concept. Refer to “section 3.3” for further information about the anomaly based concept. According to the anomaly based concept we have to create profiles for each user group in the system. These profiles can act as a baseline. If the network activity deviates too far from the baseline, then it alerts the system. We have followed the same procedure, creating profiles in the system. However, here we are searching for a file in the system. If the file is in the database then we have permission to download or else a pop up shows that there are no such files in the database, and if the user acts abnormally, then the system decides there might be an intrusion. Refer to “figure 9” for further information about abnormal activity. Therefore, in this project we have used the concept called file search/behavior based. File search’s general function is, to search a file from ‘X’ location. Refer to “figure 9” for further information about, where to search a file and where it will be downloaded etc.

The working principle of knowledge based is similar to antivirus software. It has a database, in which virus signatures are stored and if the uploaded contents matches one of the signatures in the database then system decides that there is a possible virus. Therefore, in this project we have used the concept called file upload/knowledge based. File upload is normal function is, to upload a file in ‘X’ location. After successfully uploading, we have to click the send button. Here, the send button checks the signatures in the database. If the file matches, then the system detects an intrusion. Refer to “figure 10” for further information.

These results are all of my thought’s execution. Initially, we have to run the main server.

In this project we use three proxy servers. After successfully running the main server we have to run the first proxy server, because the main server is connected to all the proxy servers and
it maintains the index information for all the proxy servers. The user sends the request to the single server about the “hi.doc” file. Then the main server will forward the request to the exact proxy. The proxy server gives back the information to the main server. Finally, the server will send the file to the user.

Then we have to run the second proxy server.

Then we have to run the third proxy server.

After successfully running the main server and all the proxy servers, we have to run user login.

After successfully running the user login command, we will get a user login window. Here we can enter a username and password; the login details are already stored in the database. If we are not saving user id and password details in the database, then we can never enter into the detection system.
The below figure shows the Intrusion detection system’s main frame page. In this page we can see three buttons i.e. file search, file upload and logout.

First, behavior based method. In this method it is done by comparing the previous behavior with the present one. In the picture given below, we can see four fields. The user IP address and downloading folder are specified in the code. We have to specify the current network IP address. IP address, file searching and downloading folder are explained in detail given below.

Let us assume only one system is connected to the network

User IP address: User IP address is nothing but the network’s IP address
File name: You can enter one file name in the system (that folder should be specified in the code as explained below).
File download: It will download the file in the system (as we set the path in the system where it should be downloaded).
We should search the files from “C:\Download” folder only because as we said we have specified that in the code.

1. Enter the “file name” and click “search” button. If the file matches in the “C:\Download” folder, then we can download that file and it will be stored inside the “project\downloads” folder.

Here we are searching for Linkoping.txt file which is located in c:\download. If the searched file is there in the folder then it is downloaded to the specified download path, alternatively if the file is not found the error message is displayed.
If we are searching the same file for more than 5 times continuously, then the system assumes it is an intrusion and blocks the user.

Second, we are free to delete and modify rules at will in a knowledge based method.
We browse the files in the “C:\Download” folder and then click “send” button.
1. If the file that is browsed has the same content to the file that is in the database then an intrusion is detected.

For example: suppose there is a file “a.txt” which contains “Hej” content. In the database we have many files, but only one file named “b.txt” contains “Hej” content. In this case both files are different but content is the same. If the content is the same then it is detected as an intrusion.

Figure 10: File upload is similar to knowledge based

Here we are browsing “Linkoping.text” file.
After browsing the file, click the send button. If the file content matches with the database file content, it detects the intrusion.
The below picture is the second example in the knowledge based method. We follow the same procedure here, browse the file and then click the send button. If the file content does not match the database’s file content, it is sent successfully, otherwise an intrusion is detected.
The third button is logout button. Logout is to exit from the system.

The concept is SAAS (Software as a service). Initially we have to start apache software. Then we will get the below web page. Here, we are using only the java path to create a webpage. In this case there is no need to use the java editor to compile the code.

First we have to create an account. If you already have an account then you can sign-in in the homepage directly.
After successfully signing in, we have to create one java file
In the blank space shown in the figure, we can write a JAVA program and save the file.

We have written a small program in the blank space and it is saved as “sample.java”. Note: all the files should be saved with “.java” extension
We have created the file successfully, and then the next step is to compile the program. Compile is among other things checking for bugs.
While compiling the program, we have to enter the file name and then click the “compile” button. After compiling it detects the bugs in the program. Here the bug is in the println statement. After fixing the errors in the program, we have to save the program again with the same name or with a different one.
The program was saved successfully.

Then, once again we have to compile the program.
It was compiled successfully with no errors.
8. Conclusion

Everything starts with an idea, and then we have to analyze the idea step by step. Based on the implementation procedure we will get results. We have also followed the same procedure in our project. My organization idea came with an IEEE journal (authors are Kleber Vieira, Alexandre Schulter, Carlos Becker Westphall and Carla Merkle Westphall). Before starting the practical part, a research was done on lot of journals, textbooks, online links etc. Those sources helped during the work. The authors have given crystal clear ideas on how to overcome the drawbacks in the existing system. Our research concludes that those techniques and methods are the best ones in the current security field. Finally, with those ideas, we have designed the architecture.

This thesis is mainly concentrated on three phases, i.e. security, data efficiency, system performance. Coming to the conclusion these features gives better results in the real time as well as these are the best techniques to overcome the drawbacks in the “conventional client-server” system. For providing security, we have implemented IDS. It is more secure beyond passwords, digital signatures, and confidentiality. IDS cannot substitute any of those mentioned mechanisms. But it can enhance a system where those mechanisms are not enough. For data efficiency, we have implemented a proxy server. The proxy server extends the functionality for a single server. Hence proxy servers are more efficient than a single server. For system performance, we have implemented a grid system. Grid system was constructed to provide equal performance for all the systems. For each node we have analyzed the system separately, by which we have achieved the complexity reduction in the system.
9. Bibliography

9.1 Journals
Towards a Taxonomy of Intrusion Detection Systems
A Security Architecture for Computational Grids
Intrusion Detection for Computational Grids
Grid-M: Middleware to Integrate Mobile Devices, Sensors and Grid
Artificial Intelligence Techniques Applied to Intrusion Detection Responses
Improvements in the Model for Interoperability of Intrusion Detection Compatible IDWG with the Model
4. Intrusion detection for grid and cloud computing
10. Understanding for cloud computing Vulnerabilities.

9.2 Text Books

20. Addison Wesley- The UML User Guide

9.3 Online Links
2. Text book
4. IEEE paper: Intrusion detection for grid and cloud computing
10. Literature
20. Text book
30. Literature
33. http://www.youtube.com/watch?v=1d1tUefYn4U

Note:

- We used “Wikipedia” source is only for the definition purpose.
- Reference number 33 is the YouTube link. We refereed the lecture on DNS cache poisoning.
9.4 Figures

Figure 1: Norton cybercrime report in India

Figure 2: Architecture of grid and cloud computing intrusion detection
   IEEE journal: Intrusion detection for grid and cloud computing

Figure 3: IDS placed in the network
   http://www.slidefinder.net/I/IDS/ids/27132832

Figure 4: Use case diagram related to the project

Figure 5: Class diagram related to the project
Appendix A - IDS Survey

IDS Survey

Author       Vamsi Krishna Popuri
Examiner     Dr Viiveke Fåk

1. In your point of view what is an intrusion detection system?
2. How are you managing security threats in the network? Which type of security system are you using?
3. If you are using firewall in the network, do you need an IDS?
4. What type of threats are risky? i.e. inside or outside
5. Do you get false positive or false negative alarm in your network?
6. Why most of the organizations are not using IDS? Please justify

IMP: Send the answers to vampo650@student.liu.se
Appendix B- Answers for the IDS survey

Dr Viiveke Fåk (Vice head of department, ISY)

1. In your point of view what is an intrusion detection system?

A system which analyzes all communications passing a specific point in a network or into a specific network connected computer. The analysis is done with the aim to find out if the system is attacked (or probed for a possible future attack) via specific messages sent to the system. An IDS only detects and raises alarms, to human users or to other tools. It does not contain any prevention part unless that is specifically specified, and in that case it is usually called an IPS.

2. How are you managing security threats in the network? Which type of security system are you using?

I am not a manager, but my system uses firewalls and malware detectors, not any specific IDS.

3. If you are using firewall in the network, do you need an IDS?

Yes, in larger systems definitely. You must allow so much, that there are too many possibilities for attacks. For home users it is more doubtful, provided that they can keep a very close set of rules for their firewall.

4. What type of threats are risky? I.e. inside or outside

Outside risks are by far the most numerous, but insiders can do more harm once they turn against the system.

5. Do you get false positive or false negative alarm in your network?

Not at my level, which is behind several nodes with firewalls etc.

6. Why most of the organizations are not using IDS? Please justify

Because they feel that an IDS doesn’t capture everything anyway and a firewall does most of the work. An IDS requires a knowledgeable manager, which many companies lack.
Mr Jens Ogniewski (Ph.D student, ISY department)

1. In your point of view what is an intrusion detection system?

An intrusion detection system is software that monitors a computer system and raises an alarm if it detects a possible illegal action (like installation of a virus etc.).

2. How are you managing security threats in the network? Which type of security system are you using?

@home:
I am using a (passive) firewall, encryption for any kind of transfer of sensible information (passwords, bank transaction etc.). I configured my web browser to delete any kind of information when I close it, like cookies, super cookies, flash cookies etc. Apart from that (and some common sense) I am currently not using any security measures, but would do so if more threats existed for Linux or if I would use a different operating system. I just remembered that my web browser warns me every time a website wants to launch an application or install something on my computer, which might be treated as kind of an (very stupid) IDS.

@work:
I have a central administrated PC running Linux and quite frankly I don't know if they are running some kind of IDS. I set up my web browser to the same settings as above, and also use encryption for all important actions. And as far as I know they automatically encrypt my home directory.

3. If you are using firewall in the network, do you need an IDS?

@home:
Depends. A firewall and IDS are working differently, so just using a firewall won't render IDS superfluous. As said, I only don't use IDS right now since its benefits for Linux systems seems limited. However, if I would be running windows I would probably using different IDS to detect viruses, Trojans, spyware etc.

4. What type of threats are risky? (I.e. internal or external)

@home:
I am not quite sure what you mean with that. Since only I am using my network there seems to be little internal risk. And then again, luckily there haven't been much threads on Linux systems (yet).

@work:
Personally, I don't see many risks. However there is a risk of personal which is uneducated in security measures, thus introducing certain internal risks. Furthermore, if these employees share data between their work and home-pc there is a thread that malware can enter from an unsecured private PC, which does not benefit from any security measure used in the company.

5. Do you get false or negative alarm in your network?
@home:
No, since no IDS is used => no alarms (except the afore mentioned warnings of my web browser, than there are a lot, but since I expect them they are just a minor necessary annoyance). I remember however a Windows PC that I administrated ones where I got a few false alarms every now and then.

@work:
None that I noticed.

6. Why most of the organizations are not using IDS? Please justify.

Probably because they annoy people and most of them don't even know how to handle warnings. Thus most of them would be ignored anyway. Furthermore, they cost money, both for the software and for maintaining them. These are the only reason I can think of and quite frankly I think this is more than a little bit shortsighted. And all companies I worked for actually used IDS, at least at the departments I was working.
Kristina Arkad (IT manager, ISY department)

1. In your point of view what is an intrusion detection system?  
   A system that helps to discover attempts to unauthorized access to systems.

2. How are you managing security threats in the network? Which type of security system are you using?  
   Several

3. If you are using firewall in the network, do you need an IDS?  
   Yes

4. What type of threats are risky? i.e. internal or external  
   Both

5. Do you get false positive or false negative alarm in your network?  
   Yes

6. Why most of the organizations are not using IDS? Please justify  
   I don’t know
### Appendix C - Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IDS</td>
<td>Intrusion Detection System</td>
</tr>
<tr>
<td>IAAS</td>
<td>Infrastructure as a Service</td>
</tr>
<tr>
<td>SAAS</td>
<td>Software as a Service</td>
</tr>
<tr>
<td>PAAS</td>
<td>Platform as a Service</td>
</tr>
<tr>
<td>NIDS</td>
<td>Network based Intrusion Detection System</td>
</tr>
<tr>
<td>HIDS</td>
<td>Host based Intrusion Detection System</td>
</tr>
<tr>
<td>OSSEC</td>
<td>Open Source Security</td>
</tr>
<tr>
<td>NIST</td>
<td>National Institute of Standards and Technology</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>DDoS</td>
<td>Distributed Denial of Service</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>IPS</td>
<td>Intrusion Prevention System</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hyper Text Transfer Protocol</td>
</tr>
<tr>
<td>DNS</td>
<td>Domain Name System</td>
</tr>
<tr>
<td>SQL</td>
<td>Structured Query Language</td>
</tr>
<tr>
<td>UML</td>
<td>Unified Modeling Language</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>OS</td>
<td>Operating System</td>
</tr>
</tbody>
</table>