Detection and Cure of attacks/threats in Grid Environment

Undertaken By:

Mehmoon Anwar
417-FBAS/MSCS/S08

Supervised by:

Mr. Sajjad Asghar
Scientific Officer
Advanced Scientific Computing (Head)
National Centre for Physics (NCP)
Islamabad

Co Supervised by:

Mr. Shehzad Ashraf Ch.
International Islamic University Islamabad

Department of Computer Science

Faculty of Basic Applied Sciences

International Islamic University Islamabad

April 2010

Department of Computer Sciences
International Islamic University Islamabad

Date 29-4-2010

Final Approval

It is certified that we have examined the thesis report submitted by Mr. Mehmoon Anwar, Registration number 417-FBAS/MSCS/S08, and it is our judgment that this thesis is of sufficient standard to warrant its acceptance by the International Islamic University, Islamabad for the MS Degree in Computer Science.

Committee:

External Examiner

Dr. Arfan Jaffer
Assistant Professor,
FAST-NU
Islamabad.

Internal Examiner

Mr. Syed M. Saqlain
Assistant Professor,
Department of Computer Science,
Faculty of Applied Sciences,
International Islamic University,
Islamabad.

Supervisor

Mr. Sajjad Asghar
Head Grid Computing,
National Centre for Physics,
Islamabad.

Scientific Officer
National Centre for Physics
Islamabad.

Co-Supervisor

Mr. Shehzad Ashraf Chaudhry
Lecturer,
Department of Computer Science,
Faculty of Applied Sciences,
International Islamic University,
Islamabad.

**A dissertation submitted to the
Department of Computer Science,
International Islamic University, Islamabad
as a partial fulfillment of the requirements
for the award of the degree of
MS in Computer Science**

Declaration

I hereby declare that this thesis neither as a whole nor as a part has been copied out from any other source. It is further declared that no portion of the work presented in this thesis has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning.

Mehmoon Anwar

Acknowledgments

I like to thank my Merciful, Beneficent and Almighty ALLAH Who deserves all the Respect and Praise, who gave me the Ability and opportunity to write this Thesis of MS (CS) and who made my Efforts Fruit Full.

I would also like to thank my Supervisors Mr. Sajjad Asghar and Mr. Shehzad Ashraf Ch. from the core of my heart for guiding, supporting, motivating and boosting my confidence throughout this project to the best of their knowledge and abilities.

I would also like to thank my truly and most sincere and loving parents, wife, sisters, brothers and friends who always remembered me in their prayers.

List of Tables

List of Figures

Abstract

National Center of Physics, Islamabad maintains a grid computing infrastructure. The IT department of the center is trying to come up with a *Self Healing System* for grid computing at the center, which can provide an expert security system that will monitor and fix the problems and threats encountered in the grid infrastructure.

Objective or goal of this research is to develop an expert security system/Intrusion Detection System that will monitor and fix the attacks and threats that are found in Grid Environment and that can remove the limitations of previously proposed intrusion detection system/security systems. In case of new or undiscovered threats the system would be intelligent enough to get direction from the system administrator.

To find out current proposed IDSs (Intrusion detection systems) and the limitations in current proposed IDSs a literature survey has been carried out. During literature survey it has been found that current Intrusion detection systems are not comprehensive in terms of providing cure against detected intrusions. So after examining current proposed IDSs and the limitations in current proposed IDSs a model for Detection and Cure of Attacks in Grid environment has been proposed, moreover its Prototype has been given.

Now the problems which were unsolvable previously have been solved because of these huge computing architectures. As Grid system is a very fast processing tool and includes a very high speed network, therefore any usage of the resources in Grid for attack purposes can be very harmful. As Grid contains many resources and if theses resources are not secured properly then the results could be very unpleasant.

1.2. Attacks on Grid Computing

There are many different kind of attacks which an intruder can launch in a grid environment. Out of many, I have discussed some of them, which are.

i) Malicious code and viruses

Grid includes a very fast and high speed network. Therefore, because of the structure of the grid, an attacker could penetrate a virus or a malicious code easily and at a very high speed. But we can't stop this attack manually because of the slow speed of manual process. Grid middleware's flaws can help an intruder to launch these attacks. The possible outcomes of these attacks could be from back doors creation, to loss and damage of precious data. So a comprehensive security system could prevent from such kinds of Attacks. [5]

ii) Malicious usage by Resource's Exploitation

In Grid computing a job or a problem takes very little time to get solved because of a very high performance. This feature of Grid can be exploited by an intruder in order to process his malicious code e.g. if an intruder wants to crack a password by means of brute force attack, then he can use some computers in the Grid by ill legal access, these computers will perform the cracking as fast as no other computer can do, and now there is no need for an intruder to wait for months for cracking the password. Although Grid computing is not for these purposes, but It can be used by intruders for their benefits. The number of system affected by an intruder can be very high. But we can't stop this attack manually because of the slow speed of manual process. So a comprehensive security system is needed to prevent from such kind of Attacks. [6]

iii) Data destruction

The data produced by Data and Computational Grids is very large in amount e.g data produced per year by LCG (LHC computing Grid) is in peta bytes. LCG generated this data by using a very high CPU power. As many resources are used for generating experimental data, therefore data has very worth in Grid environment, and therefore security of data is required. Huge amount of data is also used by Weather forecasting grids. Large amount of data is also used to be processed and produced by some of the weather forecasting grids. The security of this important data is also required. A very insufficient support is provided for data security by Grid's middleware. As in Grid, sometimes the data is replicated on more than one machine therefore changing/tempering of that data by an intruder can't be detected easily. A very large time and CPU power is needed to check data integrity. [6]

iv) DDOS and DOS

In grid computing those computers can be used to launch DOS attacks which have a very high speed and bandwidth access to the internet. In Linux and Solaris there are some known vulnerabilities, intruders make use of those vulnerabilities to get root access to the computers in grid.[7].

When some of these very fast and well connected computers in grid are accessed by an intruder then they become a very powerful tool to launch a DOS and DDOS(Distributed Denial of Service) attack . Now system should be intelligent so that it can detect that either it is working as a zombie and not for an intruder, It is not easy to prepare a system to tackle this kind of threat.

The valid user's authentication is also delayed in this kind of attack or threat so that they are unable to access resources for those applications which are very critical in time. numbers of computer that got affected will be very high if any one of the previously defined attacks is launched. It will be very inefficient to use simple traditional procedures for removing such affected computers.

It is very difficult and time consuming task to remove DOS, DDOS intrusions from only one computer (i.e. single server) and now we are dealing with such situation in Grid that there may be thousands of affected machines from such attacks.

In Grid the effect of DOS, DDOS might be range from thousands of computers to whole site unlike in simple environment (which are not grid) in which very small number of systems becomes unavailable due to these attacks.

1.3. Need For Grid Security

Because of all the above kind of attacks that could be launched on grid, Grid security is very much important. That's why there is a need to develop a comprehensive intrusion detection system for grid that could save grid from any kind of unpleasant results.

1.4. Intrusion Detection Systems

Intrusions/attacks are detected by intrusion detection systems. There are different categories of intrusion detection systems, they are.

i) Host-based IDSs

Host-based IDSs are used to detect host specific intrusions. Systems logs or application logs are examined by Host-based IDSs to detect host specific intrusions.

ii) Network-based IDSs

Network-based IDSs are used to detect network specific attacks/intrusions by analyzing captured network packets.

1.5. Approaches used by IDSs

Following are the approaches which could be used by an intrusion detection system for detecting intrusions/attacks.

i) Misuse Detection Approach

In this approach known attack patterns which are also called signatures or rules are compared with different patterns in audit data to find out intrusions/attacks.

ii) Anomaly Detection Approach

In this approach behavior of a user is compared with user's normal behaviors/profiles stored in databases to find out intrusions/attacks.

1.5.1. False Positive

When an ID considers an activity which is actually normal as malicious then it is called False Positive.

1.5.2. False Negative

Similarly when an intrusion/attack is considered a normal activity by IDS and no alerts are generated then it is called False Negative.

Needs of a variety of different projects (scientific collaborations, engineering projects) of multi-companies and multi-institutional research groups are met by Grids. World Wide LHC Computing Grid Project (WLCG) is one such project. Purpose of WLCG was the simulation, processing and analysis of the data of the Large Hadron Collider (LHC) experiments. The LHC is being constructed at European Laboratory for Particle Physics (CERN). LHC will be the world's largest and most powerful particle accelerator.

1.6. CERN (European Research Centre)

An European Organization for Nuclear Research is known as CERN. **20 European Member States** run CERN. There is also involvement of some non-European countries in different ways. Funding agencies and Physicists in both non-Member and Member States have the Responsibility for financing, construction and operation of the experiments.

1.6.1 Worldwide LHC Computing Grid Project (WLCG)

The name of new project of CERN is WLCG (**Worldwide LHC Computing Grid Project**). This new project deals with LHC and all computations involved in it. There are different sites (domain administration) of WLCG (GRID sites) in different CERN Member States.

1.6.1.1. LAYOUT OF WLCG INFRASTRUCTURE

Building, developing and maintaining a computing infrastructure for storage and analysis of the LHC data is the mission of Worldwide LHC Computing Grid Project (**WLCG**). A four -tiered model will be used to distribute the data around the globe.

Large Hadrons Collider (**LHC**) is currently build by European research centre CERN. In a number of tiers CERN computing sites are arranged. One Tier-0 site is serving as CERN. One Tier-0 site will collect and distribute data. The data will be collected and distributed by Tier-0 site to 11 Tier-1 sites. The data will be processed by 140 Tier-2 sites linked by Tier-1 sites.

1.6.1.2. Large Hadron Collider (LHC)

LHC is a particle accelerator and collider. **LHC** located at CERN, near Geneva, Switzerland (46°14′N, 6°03′E). Funds are provided by over two thousand physicists from thirty-four countries, universities and laboratories for building LHC. Two beams of protons for the new machine were proposed to be used for reaching the highest possible collision energies and intensities. The project was called the Large Hadron Collider (LHC) where the 'hadrons' in LHC means matter particles such as protons.

1.7. Details of GRID infrastructure in National Centre For Physics (NCP)

1.7.1. COMPONENTS OF WLCG

The grid infrastructure in National Centre For Physics (NCP) has different nodes, which are.

i) User Interface (UI) Nodes

ii) Computing element (CE) nodes

iii) Storage elements (SE) Nodes

iv) Resource Broker (RB) Nodes

Different services run at each node e.g. different services that run at Computing element (CE) node are gate keeper service, batch server service, batch client service etc.

World wide LHC Computing grid consists of many components or nodes. They are as follows:

i) USER INTERFACE (UI)

A user accesses grid and its other components by means of User Interface (UI). Personal account and certificates of users are installed on this machine. Command line interface tools to perform some basic operations of grid are also provided by UI. Similarly access to other functionalities offered by Information, Workload and Data management systems is also provided by a UI. development in grid enabled application can occur because of WLCG Application Program Interface (API) which is also available on the UI.

ii) COMPUTING ELEMENT (CE)

Set of computing resource localized at a site is called Computing Element (CE).

CE include **Grid Gate (GG), LRMS (Local Resource Management System)** also called batch system and **WN (Worker Nodes)** where the jobs are run. Jobs are accepted by GG whereas dispatching of jobs to worker nodes for execution is done by LRMS. There are different types of batch systems but the batch systems which Grid infrastructure in National centre for physics is using is called torque. Torque is an enhanced version of Portable batch systems (PBS). In a batch system CE node is called batch server and worker nodes are called batch clients. CE node divides the execution of job among the worker nodes for load balancing.

iii) STORAGE ELEMENT (SE)

Uniform access to all storage resources is provided by a Storage Element (SE). Each WLCG site provides at least one SE. Simple disk servers, tape-based Mass Storage Systems (MSS) or large disk arrays may be controlled by SE.

GSIFTP (A protocol for whole file transfer) **and RFIO** (A Protocol for local or remote file access) **are** different data access protocols and interfaces **which are supported by SE.**

Storage Resource Manager (SRM)

SRM manages some storage resources. Transparent file transfer from disk to tape, space reservation and file pinning etc. are provided by this middleware module (SRM). different versions of SRM are supported by different SE, because of this their capabilities also differ from each other. SRM interface are always provided by mass storage systems and large disk arrays.

iv) RESOURCE BROKER (RB)

Job Management Service is called a Resource Broker (RB). A job is submitted to a RB by a user. Then a Computing Element is selected by a RB. CE will use and manage job over its lifetime and store the output of the job for the user to retrieve.

WORKLOAD MANAGEMENT (WMS)

RB machine runs Workload Management Service (WMS).

WMS purpose is to.

1) Accept Job.

2) Accept Job and then finding most appropriate CE for job execution.

3) Recording status and retrieving output of the job.

Job Description Language (JDL) describes Job to be submitted. Executable to run and the input files moved to and from worker nodes are specified by JDL.

To find best possible CE for the execution of job a process called **Match Making** is done

Chapter # 2
Literature Review

Literature Review

"A review of prior, relevant literature is an essential feature of any academic project. An effective review creates a firm foundation for advancing knowledge. It facilitates theory development, closes areas where a plethora of research exists, and uncovers areas where research is needed" [8].

2.1. Papers Selected

Following are the papers selected for literature review.

P1→ Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang "A Performance-Based Grid Intrusion Detection System," Computer Software and Applications Conference (COMPSAC), Vol. 2, Page(s): 525 – 530, July 2005.

P2→ Sanjeev Rana, Rajneesh Gujral and Manpreet Singh "Securing Grid Using Intrusion Detection System" Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007) RIMT-IET, Mandi Gobindgarh. March 23, 2007.

P3→ Andrea Bosin, Nicoletta Dessì, Barbara Pes and Dipartimento di Matematica e Informatica "A Service Based Approach to a New Generation of Intrusion Detection Systems," Sixth European Conference on Web Services (ecows), pp.215-224, 2008

P4→ Fang-Yie Leu, Ming-Chang Li, Jia-Chun Lin "Intrusion Detection based on Grid," International Multi-Conference on Computing in the Global Information Technology (ICCGI), Volume, Page(s):62 – 62, Aug. 2006

P5→ Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang and Po-Chi Shih "Integrating Grid with Intrusion Detection," 19th International Conference on Advanced Information Networking and Applications (AINA), Volume 1, Page(s): 304 – 309, March 2005.

P6→ Alexandre Schulter, Júlio Albuquerque Reis, Fernando Koch and Carlos Becker Westphall "A Grid-based Intrusion Detection System," Proceedings of the International Conference on Networking, International Conference on Systems and

International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL’06),_Volume, Page(s): 187 – 18, April 2006.

P7→ PEI-YOU ZHU, JI GAO1, BO-OU JIANG and HUI SONG “A NEW FLEXIBLE MULTI-AGENT APPROACH TO INTRUSION DETECTION FOR GRID,” International Conference on Machine Learning and Cybernetics, page(s): 7-12, Aug 2006.

P8→ Kleber Vieira, Alexandre Schulter, Carlos Westphall and Carla Westphall “Intrusion Detection Techniques in Grid and Cloud Computing Environment,” IEEE computer Society Digital Library IT Professional, 26 Aug. 2009. http://doi.ieeecomputersociety._org/10.1109/MITP.2009.89

P9→ Desheng Fu, Shu Zhou, Ping Guo “The Design and Implementation of a Distributed Network Intrusion Detection System Based on Data Mining,” World Congress on Software Engineering (wcse), vol. 3, pp.446-450, 2009

P10→ Yu-Fang Zhang, Zhong-Yang Xiong, Xiu-Qiong Wang “Distributed Intrusion Detection System Based on Clustering,” Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005

P11→ Steven R. Snupp, James Brentano, Gihan V. Dias, Terrance L. Goan, Tim Grance, L. Todd Heberlein, Che-Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Douglass L. Mansur, Kenneth L. Pon Stephen E. Smahd “A System for Distributed Intrusion Detection,” In COMPCOM Spring '91 Digest of Papers, pages 170-176, February/March 1991.

P12→ Sattarova Feruza Yusufovna “Integrating Intrusion Detection System and Data Mining,” International Symposium on Ubiquitous Multimedia Computing (umc), pp.256-259, 2008

P13→ Yi Hu, Brajendra Panda “A Data Mining Approach for Database Intrusion Detection,” Symposium on Applied Computing Proceedings of the ACM, Pages: 711 – 716, 2004

2.2. Salient Features of selected studies

Papers studied	**Important Features**
PGIDS [9]	Propsed PGIDS has removed SNIDS's limitations by effectively detecting DDoS, DoS and logical attacks. Moreover Capability and detection effectiveness of PGIDS is higher as compared to SNIDS and RGIDS
GIDA [10]	This article has proposed Grid Intrusion Detection System(GIDA) to detect grid intrusions by applying anomaly detection (to detect user behavior deviation from normal behavior) approach to intrusion detection.
Andrea Bosin [11]	This paper proposed an ID system for Grid environment, ID processes corresponding to ID tasks in ID system are deployed in the form of ID services.
FGIDS [12]	This paper proposed a Fault-tolerant Grid Intrusion Detection System (FGIDS) for detecting network specific intrusions.
GIDS [13]	Proposed Grid Intrusion Detection System (GIDS) is used to detect intrusion packets in grid environment for detecting network specific attacks.
Alexandre Schulter[14]	Proposed "*distributed grid-based intrusion detection system architecture*" has detected network, host and grid specific attacks.
MAIDG[15]	Proposed MAIDG is used for detecting grid specific intrusions.
Kleber Vieira [16]	Proposed "*solution for intrusion detection in grid and cloud computing environment*" has been used for detecting grid specific intrusions.
Desheng Fu[17]	Proposed IDS detected network specific intrusions using a data mining technique.
Yu-Fang Zhang[18]	Proposed DIDS detected intrusions using a modified clustering algorithm.
Steven R. Snupp [19]	Proposed System for Distributed Intrusion Detection is used for detecting network intrusions in distributed environment.
Sattarova	This paper has focused on several data mining techniques/approaches that

Feruza [20]	can aid in the process of intrusion detection.
Yi Hu [21]	In this paper a data mining approach for detecting malicious transactions in a Database System has been proposed

Table 2.1: Salient Features of Selected Studies

2.3. Detailed Summary of Selected Studies

The studies selected in section 5 are examined in detail.

P1→ Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang "A Performance-Based Grid Intrusion Detection System," Computer Software and Applications Conference (COMPSAC), Vol. 2, Page(s): 525 – 530, July 2005.

Goals of the Paper:

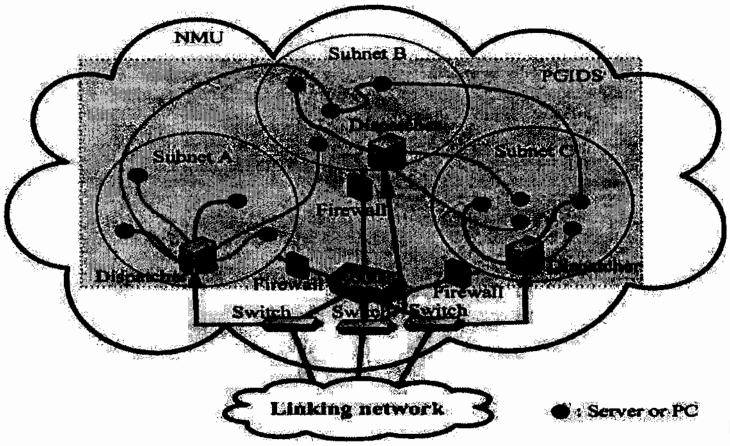
Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang proposed Performance based Grid Intrusion Detection System (PGIDS) to detect intrusions in Grid environment. PGIDS has removed the limitations of SNIDS because using PGIDS, DDoS, DoS and logical attacks have been effectively detected.

Assumptions considered:

1) There are autonomous network units called NMUs(e.g. campus network & enterprise's Intranet) in internet

2) PGIDS has been deployed in each NMU for detecting transmitted packets.

3) PGIDS's members have been placed at different NMU's subnets which may include PC or server as shown in figure on the next page.



[9]

Figure 2.1: Overview of PGIDS

4) PGIDS comprises of

i) Dispatcher:

Dispatcher gathered network packets. Network flows and jobs have also been distributed by it.

ii) Scheduler:

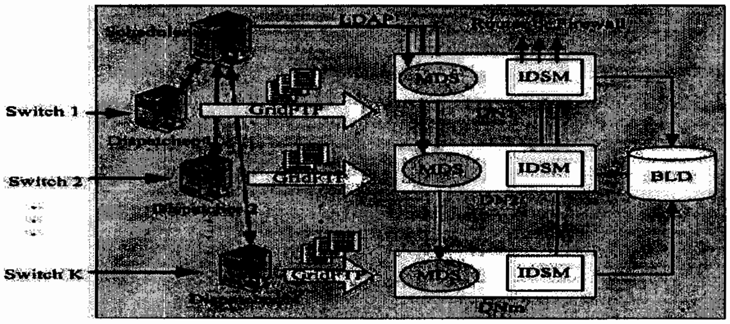
Network traffic has been predicted by Scheduler. DN allocation to dispatcher and scoring DN has also been done by Scheduler.

iii) DN:

DoS, DDoS and logical attacks are detected by DN which inspects network packets . MDS (Monitoring and Directory Service) component is there in each DN. DN's feature informations (FI), (processor grade, processor load, memory size, memory load & bandwidth) are collected by MDS

iv) BLD:

Intruder's and detection information is held by BLD pool.



[9]

Figure 2.2: Architecture of PGIDS

Claims of Paper:

i) logical, DoS and DDoS attacks which are the limitation of SNIDS are efficiently detected by PGIDS.

ii) Detection capability of PGIDS is higher as compared to SNIDS and RGIDS during a high volume attack.

Experimental Results and Analysis:

ART = Average response time

AWT = Average waiting time

Max/Min PL = Maximum and Minimum processor load

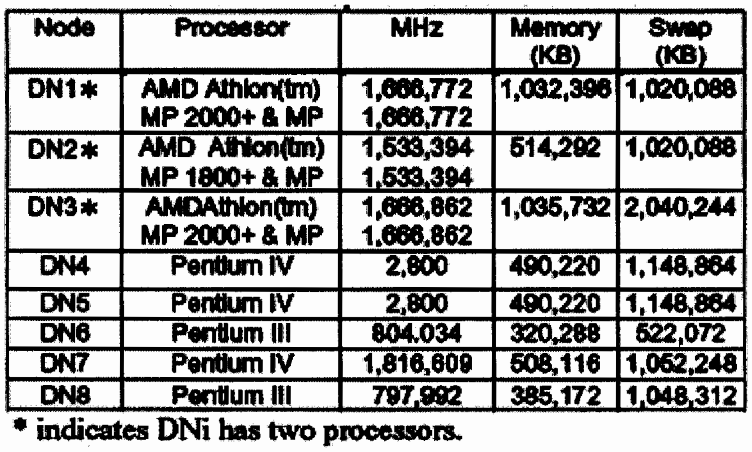
APL = Average processor load

Max/Min ML = Maximum and Minimum memory load

AML = Average memory load

NI = Nodes involved

For experiments resources from Tunghai University has been used. In the experimental analysis eight DNs and two Dispatchers for two subnets are used as shown in Table 2.2 below.



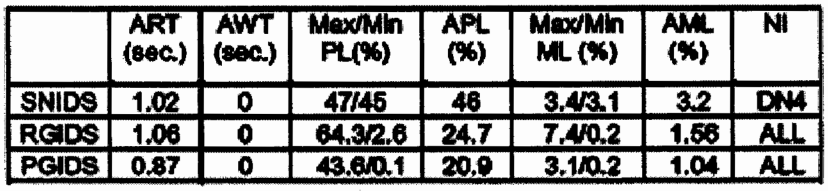
Node	Processor	MHz	Memory (KB)	Swap (KB)
DN1*	AMD Athlon(tm) MP 2000+ & MP	1,666,772 1,666,772	1,032,396	1,020,088
DN2*	AMD Athlon(tm) MP 1800+ & MP	1,533,394 1,533,394	514,292	1,020,088
DN3*	AMDAthlon(tm) MP 2000+ & MP	1,666,862 1,666,862	1,035,732	2,040,244
DN4	Pentium IV	2,800	490,220	1,148,864
DN5	Pentium IV	2,800	490,220	1,148,864
DN6	Pentium III	804.034	320,288	522,072
DN7	Pentium IV	1,816,609	508,116	1,052,248
DN8	Pentium III	797,992	385,172	1,048,312

* indicates DNi has two processors.

[9]

Table 2.2: The specifications of DNs

During Normal Network traffic detection results of PGIDS, SNIDS and Run-Robin GIDS (RGIDS) are shown in table 2.3 on the next page.



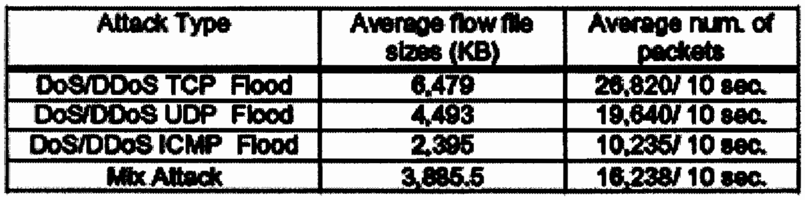
	ART (sec.)	AWT (sec.)	Max/Min PL(%)	APL (%)	Max/Min ML (%)	AML (%)	NI
SNIDS	1.02	0	47/45	46	3.4/3.1	3.2	DN4
RGIDS	1.06	0	64.3/2.6	24.7	7.4/0.2	1.56	ALL
PGIDS	0.87	0	43.6/0.1	20.9	3.1/0.2	1.04	ALL

[9]

Table 2.3: Detection results on normal networktraffic

Less computing resources has been used by PGIDS as compared to SNIDS and RGIDS moreover PGIDS has ART (0.87 sec) which is shortest as compared to SNIDS and RGIDS.

DoS/DDoS UDP flood, DoS/DdoS ICMP flood DoS/DDoS TCP flood, and mix attacks has been launched by Intrusion tools. Table 2.4 below shows the details.

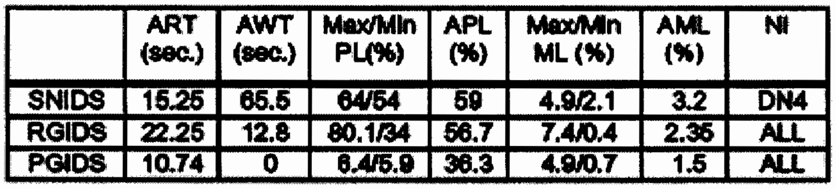


Attack Type	Average flow file sizes (KB)	Average num. of packets
DoS/DDoS TCP Flood	6,479	26,820/ 10 sec.
DoS/DDoS UDP Flood	4,493	19,640/ 10 sec.
DoS/DDoS ICMP Flood	2,395	10,235/ 10 sec.
Mix Attack	3,885.5	16,238/ 10 sec.

[9]

Table 2.4: Information about attacks

DoS/DDoS TCP flood attack's detection results have been shown in table 2.5 below .



	ART (sec.)	AWT (sec.)	Max/Min PL(%)	APL (%)	Max/Min ML (%)	AML (%)	NI
SNIDS	15.25	65.5	64/54	59	4.9/2.1	3.2	DN4
RGIDS	22.25	12.8	80.1/34	56.7	7.4/0.4	2.35	ALL
PGIDS	10.74	0	6.4/5.9	36.3	4.9/0.7	1.5	ALL

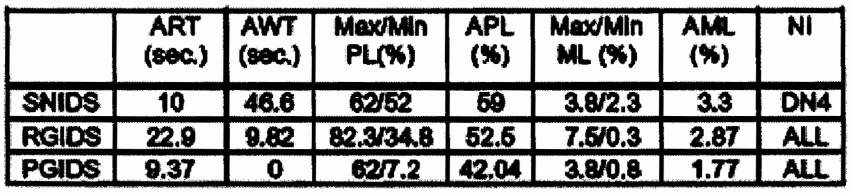
[9]

Table 2.5: Detection Results on DOS/DDOS TCP flood

Results show that PGIDS is more faster in detecting DoS/DDoS TCP flood attack as compared to SNIDS and RGIDS.

Under DoS/DDoS UDP flood attack. ART of PGIDS's are always the lowest. But SNIDS waits 46.6 seconds before an FF can be processed, as shown in Table 11. PGIDS has the best performance.

DoS/DDoS UDP flood attack's detection results have been shown in table 2.6 on the next page.



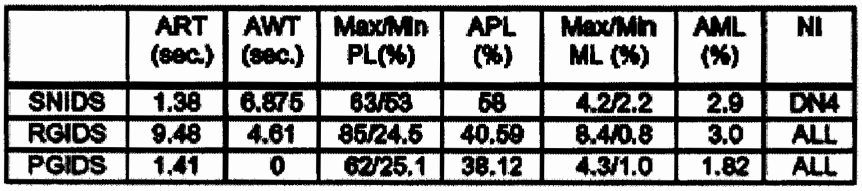
	ART (sec.)	AWT (sec.)	Max/Min PL(%)	APL (%)	Max/Min ML (%)	AML (%)	NI
SNIDS	10	46.6	62/52	59	3.8/2.3	3.3	DN4
RGIDS	22.9	9.82	82.3/34.8	52.5	7.5/0.3	2.87	ALL
PGIDS	9.37	0	62/7.2	42.04	3.8/0.8	1.77	ALL

[9]

Table 2.6: Detection Results on DOS/DDOS UDP flood

Performance of PGIDS is best.

DoS/DDoS ICMP flood attack's detection results have been shown in table 2.7 below .



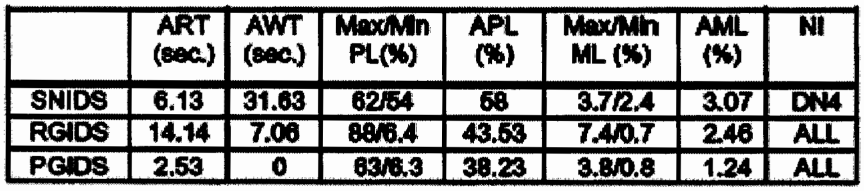
	ART (sec.)	AWT (sec.)	Max/Min PL(%)	APL (%)	Max/Min ML (%)	AML (%)	NI
SNIDS	1.38	6.875	63/53	58	4.2/2.2	2.9	DN4
RGIDS	9.48	4.61	85/24.5	40.59	8.4/0.8	3.0	ALL
PGIDS	1.41	0	62/25.1	38.12	4.3/1.0	1.82	ALL

[9]

Table 2.7: Detection Results on DOS/DDOS ICMP flood

PGIDS is better as compared to SNIDS and RGIDS because it waits for 0 second.

Detection results on Mix attacks have been shown in table 2.8 below.



	ART (sec.)	AWT (sec.)	Max/Min PL(%)	APL (%)	Max/Min ML (%)	AML (%)	NI
SNIDS	6.13	31.63	62/54	58	3.7/2.4	3.07	DN4
RGIDS	14.14	7.06	88/6.4	43.53	7.4/0.7	2.46	ALL
PGIDS	2.53	0	63/6.3	38.23	3.8/0.8	1.24	ALL

[9]

Table 2.8: Detection Results on Mix Attacks

PGIDS has better performance.

Future Work:

Not mentioned

Limitations:

But this paper has only detected network specific attacks/intrusions; it has neither detected host nor Grid specific attacks/intrusions. Moreover this paper has only detected but not provided cure against attacks/intrusions.

P2→ Sanjeev Rana, Rajneesh Gujral and Manpreet Singh "Securing Grid Using Intrusion Detection System" Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007) RIMT-IET, Mandi Gobindgarh. March 23, 2007.

Goals:

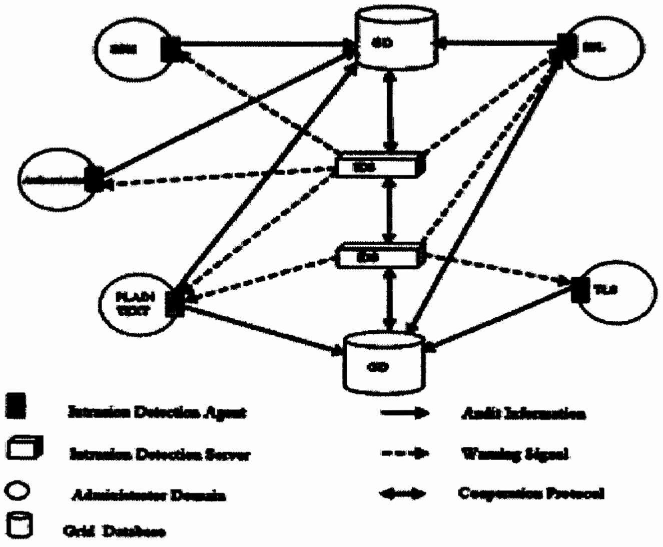
Sanjeev Rana, Rajneesh Gujral and Manpreet Singh proposed a Grid Intrusion Detection System(GIDA) to detect intrusions in grid environment by applying anomaly detection (to detect user behavior deviation from normal behavior) approach to intrusion detection.

Assumptions:

GIDA consist of

1) **Intrusion Detection Agent (IDA):** Information is gathered by IDA. Each IDA is assigned to a single domain to gather data from its domain. "*IDA will register with one or more IDSs which will analyze the gathered data*". IDA is designed in such a way that any class of resources can be handled by it.
2) **Intrusion Detection Server (IDS):** Gathered information is analyzed by IDS. Different IDS's cooperate with each other for intrusion detection.

GIDA architecture is shown in figure 2.3 below.



[10]

Figure 2.3: Proposed Grid intrusion detection architecture

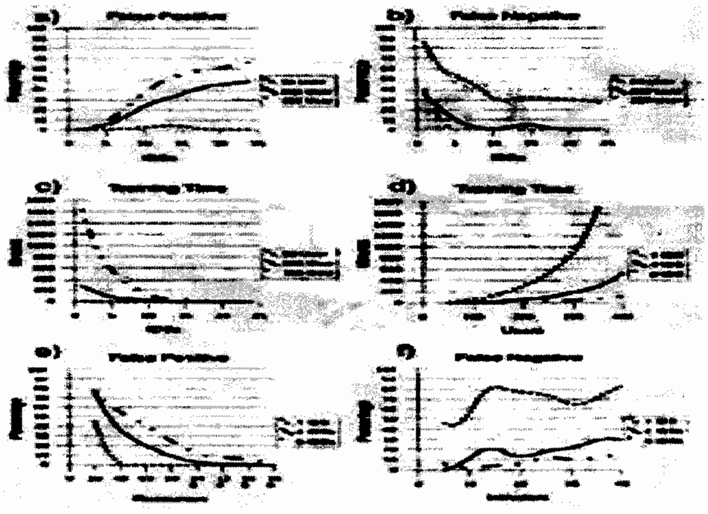
Claims:

1) Host and Network based IDS's have been deployed in some Grid environments to improve Grid's security. But deploying Host and Network based IDS's in some Grid environments can't detect grid specific intrusions. In this paper a Grid based intrusion detection architecture have been proposed that solved this limitation.

2) Some IDS (Intrusion detection system) perform behavior analysis by assuming that normal use is different from the malicious use. Some new unique challenges have been imposed for detecting such difference in behavior because of computational Grid's special characteristics and requirement. These new challenges can't be solved by traditional intrusion detection system. In this paper a Grid based intrusion detection architecture have been proposed that solved this limitation.

Analysis of Result:

When at each IDSs there is less user's behavior information then false positive percentage increases, as shown in figure 2.4.a. But less user's behavior information at each IDSs also resulted in decreased false negative percentage, as shown in figure 2.4.b. Increased number of IDSs resulted on reduced training time as shown in figure 2.4.c. Increased number of user resulted an increase in training time as shown in figure 2.4.d. Increased number of resources resulted a reduction in false positive percentage. as shown in figure 2.4.e. When number of intruders are increased then it resulted a slightly increase in the false negative's percentage as shown in figure 2.4.f.



[10]

Figure 2.4: Some Experimental Results

Limitation:

This paper only provides the detection but not the cure of intrusions.

Future Work:

There is not any future work mentioned.

P3→ Andrea Bosin, Nicoletta Dessì, Barbara Pes and Dipartimento di Matematica e Informatica "A Service Based Approach to a New Generation of Intrusion Detection Systems," Sixth European Conference on Web Services (ecows), pp.215-224, 2008

Goals:

Andrea Bosin, Nicoletta Dessì, Barbara Pes and Dipartimento di Matematica e Informatica proposed an intrusion detection system for grid environment. The paper first proposed the ID processes (which corresponds to ID tasks) necessary for detecting intrusions in grid, then it suggested that theses ID processes should be decomposed into ID tasks and ID subtasks which should be deployed in GRID in the form of ID services, then it proposed SOA reference architecture for showing that how ID process can be deployed in the form of ID services in the grid environment. This paper proposed that it would use technique of workflows for the service's composition .This paper then proposed an infrastructure which is used for removing the issues and problem related to ID service's composition. This paper also suggested some implementation hints for proposed ID system.

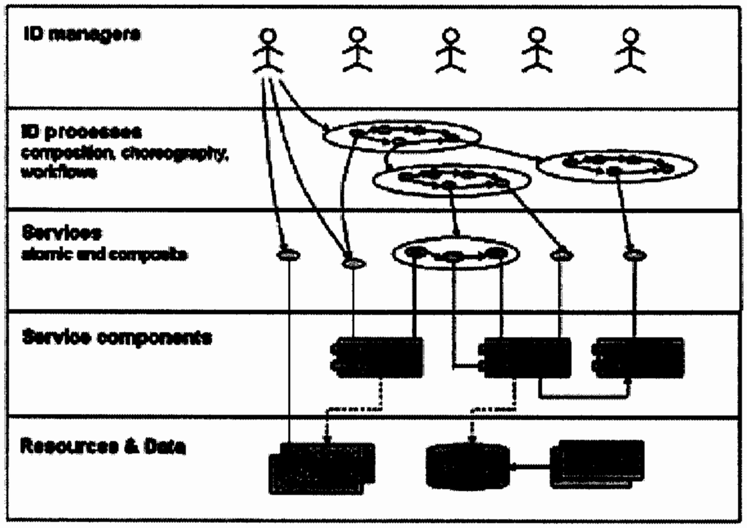
Claims:

1) The proposed ID system removed the limitations of all the current IDS, because it can detect all kinds of intrusions which other could not.
2) In grid there are many networked resources, the proposed ID system is scalable with all those networked resources.

Assumptions:

1) Three Main ID process that corresponds to ID tasks are
   a) Data management processes: It is used to convert raw audit data into meaningful form.
   b) Classification processes: Theses processes find Patterns from data by applying statistic and data mining methods and construct predictive models using those patterns.
   c) Monitoring processes: Theses processes at execution time dynamically add new attack signatures for security analysis.

2) "*All the above processes can be decomposed into smaller highly specialized tasks and sub-tasks*". These tasks are proposed to be implemented in the form of services which are autonomous, distributed and interacting with one another.
3) This paper proposed SOA architecture for deploying different ID tasks in GRID environment in the form of services. As shown in Fig. 2.5 below.



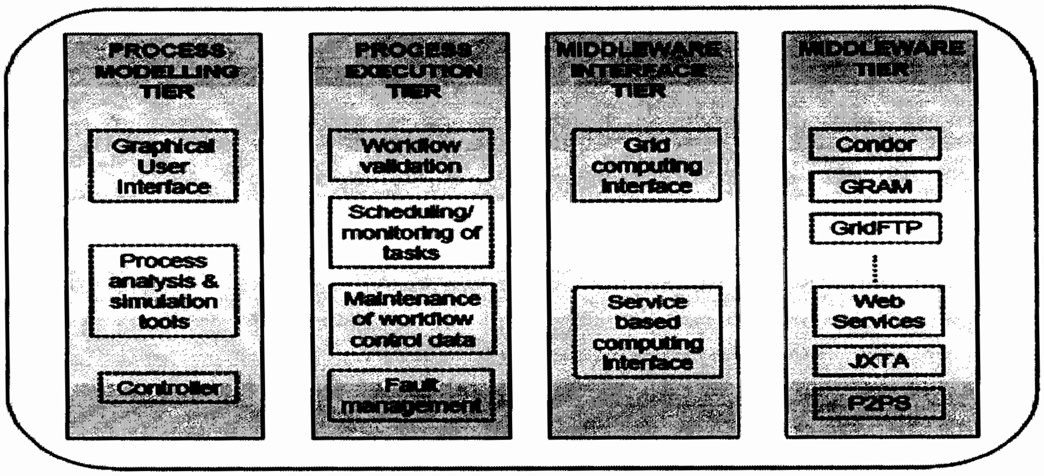
[11]

Figure 2.5: SOA reference architecture

a) At the service component layer reusable components are defined. The components may include IDS elements, toolkits and other software tools.
b) At the services layer, some services depends upon the reusable components at service component layer.

c) ID managers at the ID managers layer are used to perform ID activities at network nodes.

4) Issues related to service's composition are solved by an infrastructure shown in figure 2.6 below.



[11]

Figure 2.6: Proposed Infrastructure

a) **Process modeling tier:** functional components for workflow definition are specified here.

b) **Process execution tier:** It is used to perform workflow enactment.

c) **Middleware interface tier:** Uniform access to different types of middleware is provided by this tier. Workflow instance's composition services are also invoked and located through this tier.

d) **Middleware tier:** It *"masks the underlying resources spread over the network*".

Result Analysis:

Prototype of the proposed system is not given and results are not analyzed.

Limitation:

But this paper only provided the detection of intrusions and not the cure of intrusions. Moreover it has only suggested some implementation hints instead of giving a prototype of the proposed IDS.

Future work:

The authors have planed "*to further specify possible ID services, to implement additional user facilities and to explore the characteristics of ID grid workflows*".

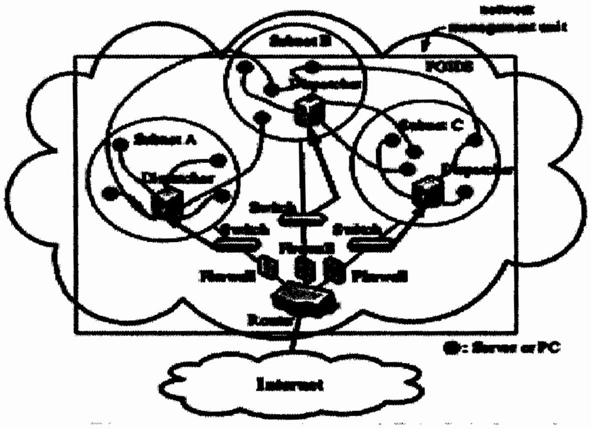
P4→ Fang-Yie Leu, Ming-Chang Li, Jia-Chun Lin "Intrusion Detection based on Grid," International Multi-Conference on Computing in the Global Information Technology (ICCGI), Volume, Page(s):62 – 62, Aug. 2006

Goals:

Fang-Yie Leu, Ming-Chang Li, Jia-Chun Lin proposed a *Fault-tolerant Grid Intrusion Detection System (FGIDS)* which is used to detect network specific intrusions in grid environment. The word "fault-tolerant" in **FGIDS** means that if a detector is overloaded by extra tasks, crashes or its detection and analytical performance becomes unbearably low then some detectors can join the FGID for improving FGID performance. But the proposed method could detect only network specific intrusions instead of detecting all the network, host and Grid specific intrusions. Moreover it has only detected and not provided cure of intrusions.

Assumptions:

There are many autonomous network management units (NMU) i.e. enterprise's intranets and campus networks in the internet. FGIDS is deployed in each NMU for its security. FGID is shown in figure 2.7 below.



[12]

Figure 2.7: Overview of FGIDS

Following are the components included in FGIDS.

i) Mirror port:

Mirror port of a switch is used to gather inbound packets of a subnet; Attacks have been detected from mirror port's traffic by the detector.

ii) Detector:

DDoS, DoS, and logical attacks have been detected by a detector

iii) Dispatcher:

Packets from mirror port have been received by dispatcher to predict next second's traffic volumes.

iv) Scheduler:

Capability of each detector is measured by a Scheduler.

v) Backup broker:

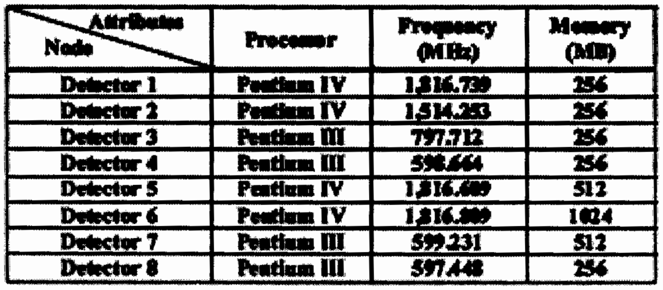
Network traffic is also stored by Backup broker so that if detector fails then Network traffic could be forwarded to any other detector.

vi) Block list database:

Information about intrusions is stored in Block list database.

Analysis of Result:

In the experimental analysis Tunghai University's resources have been used. Eight detectors and two dispatchers for gathering two subnet's traffics have been used. Detectors are shown in table 2.9 below.



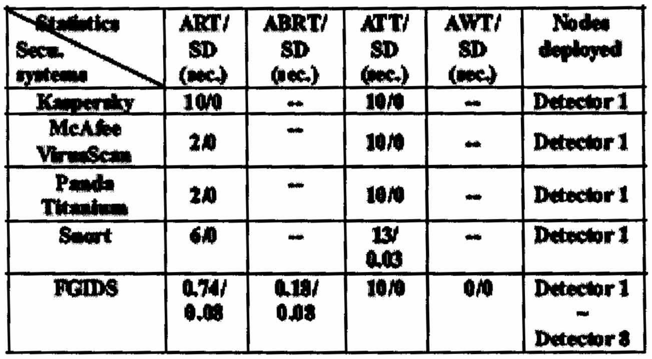
Attributes Node	Processor	Frequency (MHz)	Memory (MB)
Detector 1	Pentium IV	1,816.739	256
Detector 2	Pentium IV	1,514.253	256
Detector 3	Pentium III	797.712	256
Detector 4	Pentium III	598.664	256
Detector 5	Pentium IV	1,816.689	512
Detector 6	Pentium IV	1,816.809	1024
Detector 7	Pentium III	599.231	512
Detector 8	Pentium III	597.448	256

[12]

Table 2.9: The Detector's specifications

Kaspersky, Snort, Panda Titanium , FGIDS and McAfee Virus Scan security systems have been tested.

Tcp-syn-flood resource consumption attack's detection results are shown below.

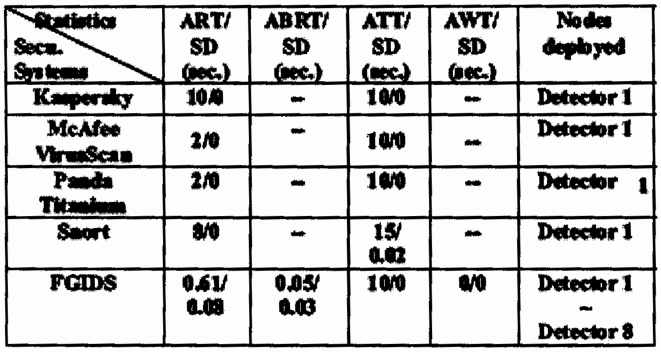


Statistics / Secu. systems	ART/ SD (sec.)	ABRT/ SD (sec.)	ATT/ SD (sec.)	AWT/ SD (sec.)	Nodes deployed
Kaspersky	10/0	--	10/0	--	Detector 1
McAfee VirusScan	2/0	--	10/0	--	Detector 1
Panda Titanium	2/0	--	10/0	--	Detector 1
Snort	6/0	--	13/ 0.03	--	Detector 1
FGIDS	0.74/ 0.08	0.18/ 0.08	10/0	0/0	Detector 1 – Detector 8

[12]

Table 2.10: Detection results of tcp-syn-flood resource consumption attack at 9,160 ANP/sec

icmp-flood resource consumption attack's detection results are shown below in table 2.11

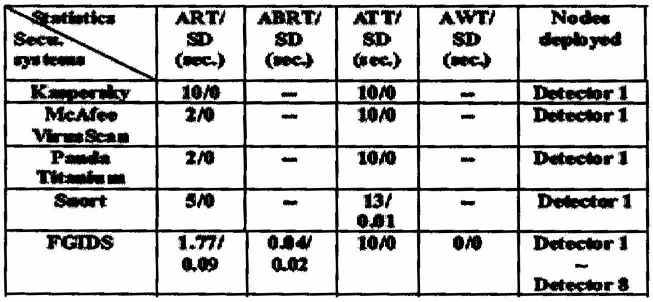


Statistics / Secu. Systems	ART/ SD (sec.)	ABRT/ SD (sec.)	ATT/ SD (sec.)	AWT/ SD (sec.)	Nodes deployed
Kaspersky	10/0	--	10/0	--	Detector 1
McAfee VirusScan	2/0	--	10/0	--	Detector 1
Panda Titanium	2/0	--	10/0	--	Detector 1
Snort	8/0	--	15/ 0.02	--	Detector 1
FGIDS	0.61/ 0.08	0.05/ 0.03	10/0	0/0	Detector 1 – Detector 8

[12]

Table 2.11: Detection results of icmp-flood resource consumption attack at 705 ANP/sec

udp-flood bandwidth consumption attack's detection results are shown below in table 2.12.

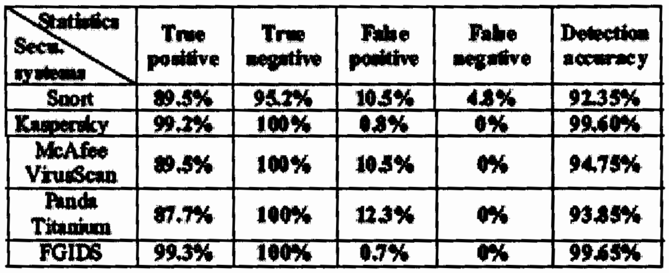


Statistics / Secu. systems	ART/ SD (sec.)	ABRT/ SD (sec.)	ATT/ SD (sec.)	AWT/ SD (sec.)	Nodes deployed
Kaspersky	10/0	--	10/0	--	Detector 1
McAfee VirusScan	2/0	--	10/0	--	Detector 1
Panda Titanium	2/0	--	10/0	--	Detector 1
Snort	5/0	--	13/ 0.01	--	Detector 1
FGIDS	1.77/ 0.09	0.04/ 0.02	10/0	0/0	Detector 1 – Detector 8

[12]

Table 2.12: Detection results of udp-flood bandwidth consumption attack at 704 ANP/sec

Resource/bandwidth consumption attack's detection results are shown below in table 2.13.

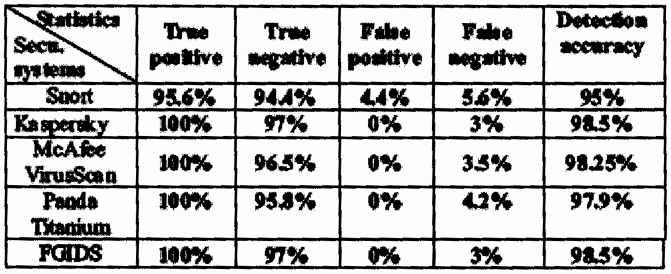


Statistics Secu. systems	True positive	True negative	False positive	False negative	Detection accuracy
Snort	89.5%	95.2%	10.5%	4.8%	92.35%
Kaspersky	99.2%	100%	0.8%	0%	99.60%
McAfee VirusScan	89.5%	100%	10.5%	0%	94.75%
Panda Titanium	87.7%	100%	12.3%	0%	93.85%
FGIDS	99.3%	100%	0.7%	0%	99.65%

[12]

Table 2.13: Detection accuracy of resource/bandwidth consumption attacks

Accuracy of logical attack's detection is shown below in table 2.14.



Statistics Secu. systems	True positive	True negative	False positive	False negative	Detection accuracy
Snort	95.6%	94.4%	4.4%	5.6%	95%
Kaspersky	100%	97%	0%	3%	98.5%
McAfee VirusScan	100%	96.5%	0%	3.5%	98.25%
Panda Titanium	100%	95.8%	0%	4.2%	97.9%
FGIDS	100%	97%	0%	3%	98.5%

[12]

Table 2.14 Detection accuracy of logical attacks

As compared to other systems logical, resource and bandwidth consumption attacks have been effectively detected by FGIDS, which is clear from the above mentioned results.

Limitaion:

But the proposed method could detect only network specific intrusions instead of detecting all the network, host and Grid specific intrusions. Moreover it has only detected and not provided cure of intrusions.

Future work:

Future work is not mentioned.

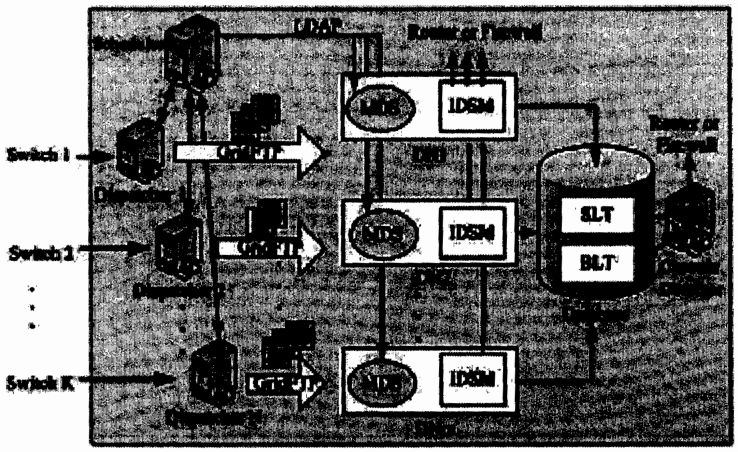
P5→ Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang and Po-Chi Shih "Integrating Grid with Intrusion Detection," 19th International Conference on Advanced Information Networking and Applications (AINA), Volume 1, Page(s): 304 – 309, March 2005.

Goals:

Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang and Po-Chi Shih proposed a Grid Intrusion Detection System (GIDS) which is used to detect intrusion packets in grid environment for detecting network specific attacks by using Grid computing resources.

Assumption:

There are many autonomous network management units (NMU) i.e. enterprise's intranets and campus networks in the internet. GIDS is deployed in each NMU for its security. GIDS is shown in figure 2.8 below.



[13]

Figure 2.8: The architecture and process of GIDS

Following are the components included in GIDS.

i) Dispatcher:

Distribution of jobs and network flows is handled by Dispatcher.

ii) Scheduler:

Computing resources are scored, collected and allocated by scheduler.

iii) Detection node:

Momentary and logical attacks are detected by detection node (DN) after network's packet inspection.

iv) Database:

Information of intrusions detected and the information about intruders is stored in Database.

v) Chronic Detector:

Chronic attacks have been detected by Chronic Detector(CD).

Experiment Results:

In the experimental analysis Tunghai University's resources have been used. Six Grid nodes are used in experimental analysis. Different attacks are launched by intrusions tools. Attack's details are shown in table 2.15 below, where.

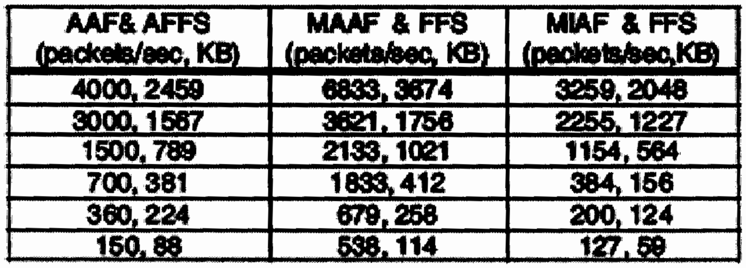
AAF = Average attack frequencies

AFFS = Average flow file size

MAAF = Maximal attack frequency

FFS = Flow file size

MIAF = Minimal attack frequency



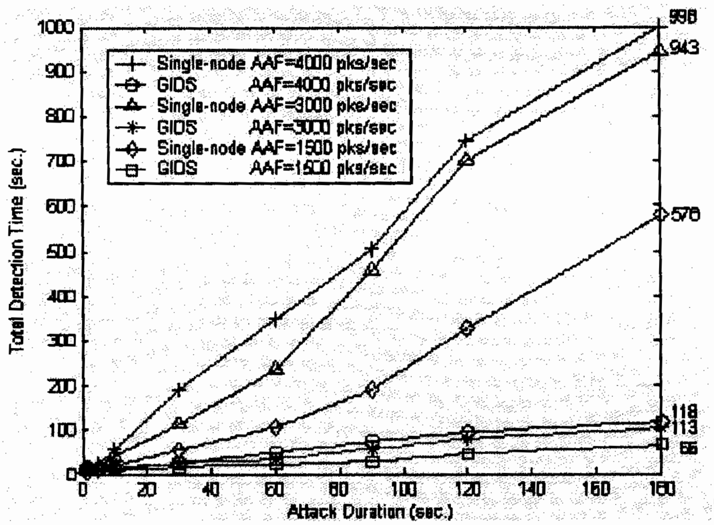
AAF& AFFS (packets/sec, KB)	MAAF & FFS (packets/sec, KB)	MIAF & FFS (packets/sec,KB)
4000, 2459	6833, 3674	3259, 2048
3000, 1567	3621, 1756	2255, 1227
1500, 789	2133, 1021	1154, 564
700, 381	1833, 412	384, 156
360, 224	679, 258	200, 124
150, 88	538, 114	127, 59

[13]

Table 2.15: Detail information about attacks

Processor load and detection time of single node and GIDS is shown on the next page in figures. Figures show that AAF is kept at 4000, 3000 and 1500.

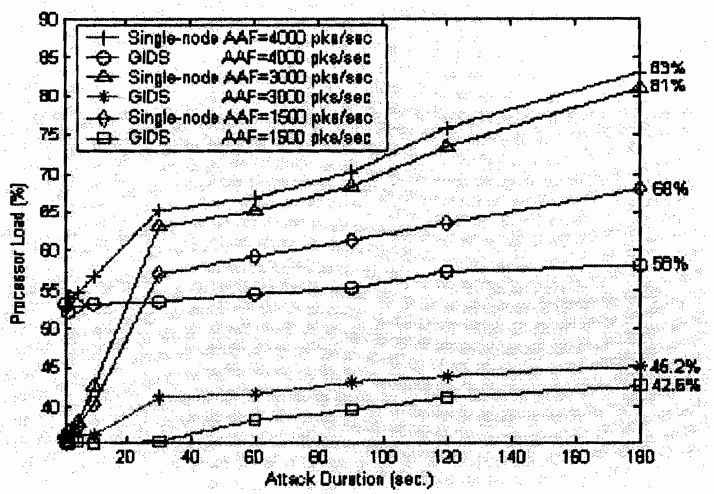
Detection time:



[13]

Figure 2.9: Total detection time of single-node and GIDS when AAFs are 4000, 3000 and 1500 packets/sec

Processor load:

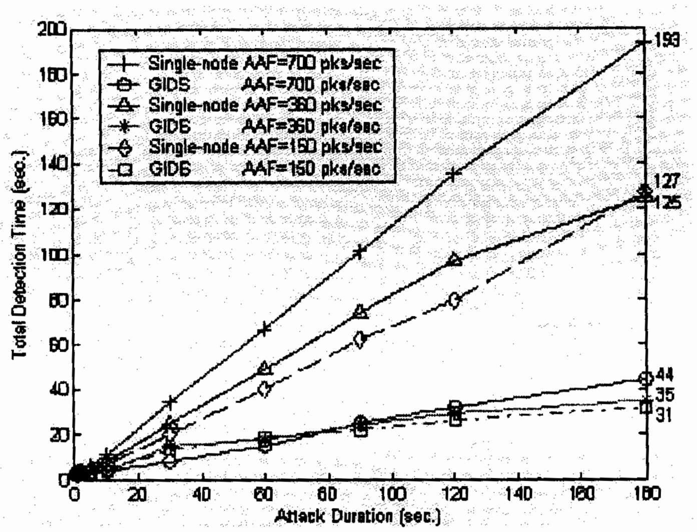


[13]

Figure 2.10: Processor load of single-node and GIDS when AAFs are 4000, 3000 and 1500 packets/sec

Similarly Processor load and detection time of single node and GIDS when AAF is kept at 700, 360 and 150 packets/sec is depicted in figures on the next page.

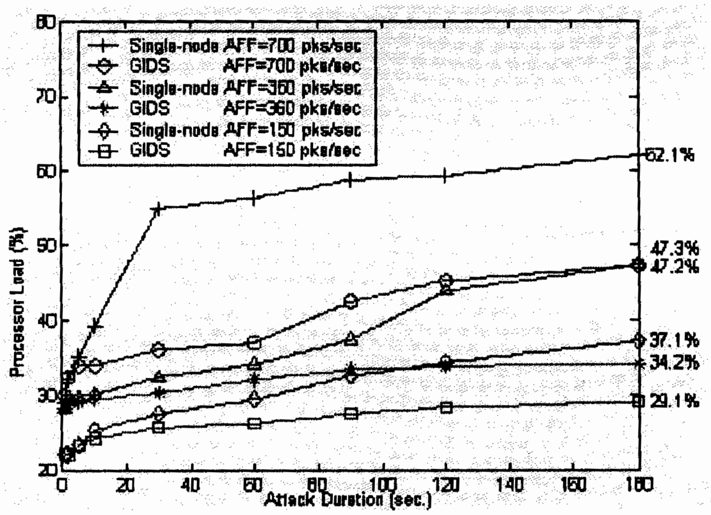
Detection time:



[13]

Figure 2.11: Total detection time of single-node and GIDS when AAFs are 700, 360 and 150 packets/sec

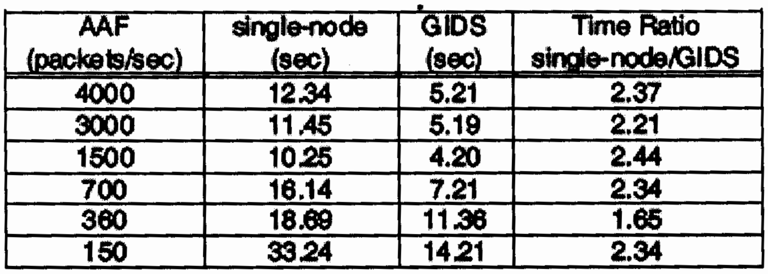
Processor Load:



[13]

Figure 2.12: Processor load of single-node and GIDS when AAFs are 700, 360 and 150 packets/sec

GIDS and Singe-node's average response times for different AAFs are shown in table on the next page. Attack's duration is 10 sec, moreover attack types include UDP, TCP and ICMP flood. 1.65 to 2.44 is the time ratio range of single-node over GIDS.



AAF (packets/sec)	single-node (sec)	GIDS (sec)	Time Ratio single-node/GIDS
4000	12.34	5.21	2.37
3000	11.45	5.19	2.21
1500	10.25	4.20	2.44
700	16.14	7.21	2.34
360	18.69	11.36	1.65
150	33.24	14.21	2.34

Table 2.16: Average response times of single-node and GIDS on different attack frequencies

Limitations:

But proposed GIDS has detected only Network specific attacks instead of detecting both network and host specific attacks. Moreover proposed GIDS detects intrusions but has not provided the cure of intrusions.

Future Work:

Data mining and neural network techniques will be used in the future to increase the intrusion detection accuracy.

P6→ Alexandre Schulter, Júlio Albuquerque Reis, Fernando Koch and Carlos Becker Westphall "A Grid-based Intrusion Detection System," Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06),_Volume, Page(s): 187 – 18, April 2006.

Goals:

Alexandre Schulter, Júlio Albuquerque Reis, Fernando Koch and Carlos Becker Westphall have proposed a distributed grid-based intrusion detection system architecture for the detection of network, host and grid specific attacks means of HIDS, NIDS and GIDS respectively. The proposed architecture has removed the limitations of those proposed architectures which could not detect all the host, network and grid specific attacks.

Assumptions:

In this paper GIDS has been Proposed. In grid Network there are many network intrusion detection systems. Network audit data is captured by network intrusion detection systems, after capturing network audit data network intrusion detection system is used to check attack trails and anomalies in network packets.

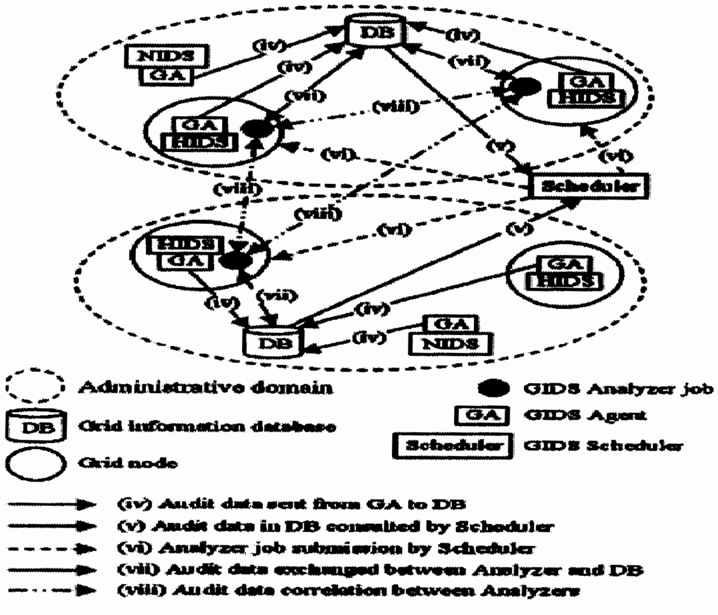
Audit data used by IDSs is also used by GIDS for behavior based analysis. Alert are send to security manager whenever IDSs send them or whenever GIDS detects intrusions. Following are the components of GIDS

i) Agents

ii) Analyzers

iii) Scheduler

Host and network IDS share their audit information with Agents of GIDS. Grid Information Databases is used to store this audit information. User profile is checked by GIDS Schedulers at user's access to GRID environment, and then Analyzer jobs which may be one or more are submitted to nodes which have required resources for analysis. For user behavior analysis, data is exchanged with databases by analyzer jobs. Other responsibilities of analyzer also include, identifying the exploits by correlating the audit data.


DB
NIDS
GA
HIDS
Scheduler
(iv)
(v)
(vi)
(vii)
(viii)
Administrative domain
Grid information database
Grid node
GIDS Analyzer job
GIDS Agent
GIDS Scheduler
(iv) Audit data sent from GA to DB
(v) Audit data in DB consulted by Scheduler
(vi) Analyzer job submission by Scheduler
(vii) Audit data exchanged between Analyzer and DB
(viii) Audit data correlation between Analyzers

[14]

Figure 2.13: GIDS architecture

Analysis of results:

Result Analysis is not peformed

Limitations:

The limitation of this paper is that this paper does not give any prototype of the proposed architecture, Moreover this paper tells us only about the detection and not about the cure of intrusions.

Future Work:

Topics like fault tolerance, performance, timeliness and accuracy of intrusion detection are included in Future Work

P7→ PEI-YOU ZHU, JI GAO1, BO-OU JIANG and HUI SONG "A NEW FLEXIBLE MULTI-AGENT APPROACH TO INTRUSION DETECTION FOR GRID," International Conference on Machine Learning and Cybernetics, page(s): 7-12, Aug 2006.

Goals:

PEI-YOU ZHU, JI GAO1, BO-OU JIANG and HUI SONG proposed a multi-agent approach to intrusion detection for Grid (MAIDG). MAIDG used agent technology along with the Globus Toolkit4.0 (GT4)'s data management components for detecting intrusions in Grid environment. There is a lack of flexibility in conventional IDS which is necessary for Grid environment and therefore conventional IDS's structure in Grid Computing applications could not be dynamically adjusted. Some proposed agent method to IDS have not resolved the above problems due to the heterogeneity and dynamic nature of Grid. The proposed distributed grid-based intrusion detection system architecture has solved the above limitations.

Assumptions:

MAIDG includes three layers.

i) Global Monitoring Agent (GMA) Layer:

GMAs in Global Monitoring Agent (GMA) layer perform following operations.

1) Checks for required resources and nodes required for Grid applications for sending MAs to the searched nodes.

2) Grid application's global monitoring

3) Data provided by MAs is analyzed by GMAs in coordinated fashion..

4) MAs receives executive instructions from GMAs .

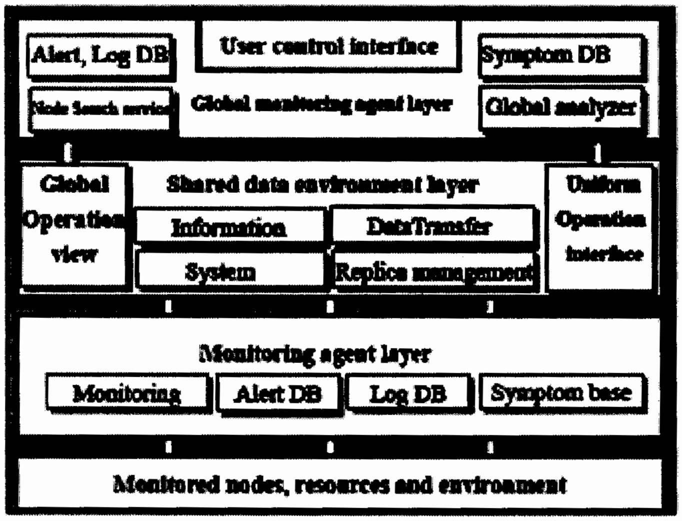
5) MAs are also killed by GMAs if they are no more needed.

ii) Shared data environment (SDE) layer:

Shared data environment (SDE) layer is middle layer. Different services are provided by SDE. Virtual data interface to monitoring resources is provided by SDE.

iii) Monitoring agent layer:

Monitoring agent layer which contains various monitoring agents (MA) is bottom layer. Network log data is analyzed by a MA if it belongs to its own allocated area. After network log data's analysis, alert data is generated, stored in DB and then GMA will access that alert DB using SDE.



[15]

Figure 2.14: The architecture of MAIDG

There are two types of reorganizations in MAIDG

i) System level reorganization:

MAIDG will collapse if GMA will collapse.

When GMA of a system is collapsed then a new GMA is relocated, this is called System level reorganization.

ii) Nodes level reorganization:

Grid application's resources are changeable with the passage of time. Detection nodes in MAIDG are contracted or extended timely along with Grid application's resource change, this is known as nodes level reorganization.

Analysis of Results:

Result analysis is not performed

Limitations:

This paper has not provided an implementation of the proposed architecture in grid environment in detail and moreover the results have not been analyzed. And this paper tells us only about the detection and not about the cure of intrusions.

Future work:

Future work is not mentioned

P8→ Kleber Vieira, Alexandre Schulter, Carlos Westphall and Carla Westphall "Intrusion Detection Techniques in Grid and Cloud Computing Environment," IEEE computer Society Digital Library IT Professional, 26 Aug. 2009. http://doi.ieeecomputersociety._org/10.1109/MITP.2009.89

Goals:

Kleber Vieira, Alexandre Schulter, Carlos Westphall and Carla Westphall proposed a solution for intrusion detection in grid and cloud computing environment for detecting grid specific intrusions. The proposed solution for intrusion detection in grid and cloud computing environment has removed the limitations of host and network-based systems that could not detect all the attacks. In the proposed architecture Behavior and Knowledge based Analysis has been performed on the audit data.

Assumptions:

Every node will detect and alert other nodes of detected intrusions. "These individual IDS will cooperatively participate in the intrusion detection". Following components which are included in IDS for grid and cloud computing environment are shown below.

i) Node:

Resources required for intrusion detection are stored in Node.

ii) Service:

Service gives functionality like communication by means of middleware.

iii) Event auditor:

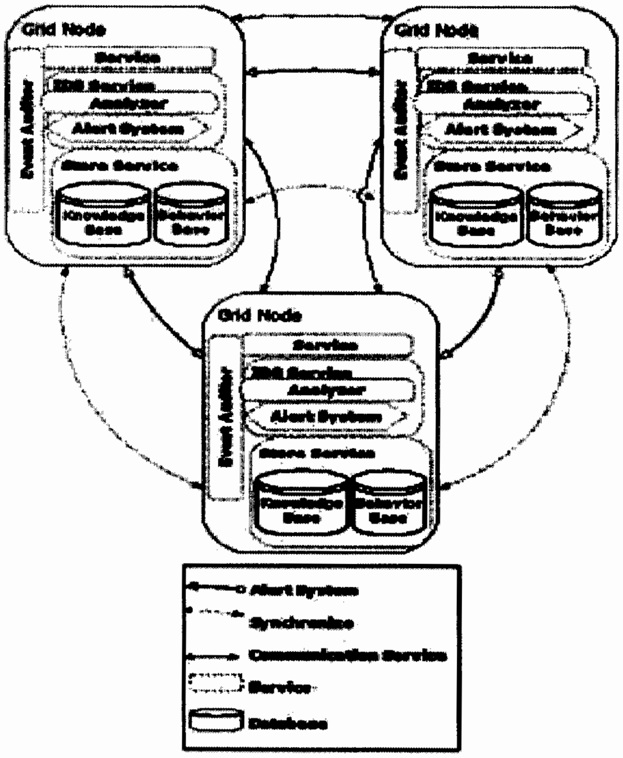
Audit data from log system, node messages and service is captured by Event auditor.

iv) IDS Service:

Knowledge and behavior based Analysis of audit data is done by IDS Service.

v) Storage Service:

IDS service needs audit data to be analyzed. This required audit data is stored by Storage Service.



[16]

Figure 2.15: Architecture of grid and cloud computing intrusion detection

Result Analysis:

Prototype using Grid-M (a middleware) has been developed for architecture's evaluation. Data of three types has been prepared. Data representing legitimate actions, Data representing behavior anomalies anḍ Data representing policy violation

i) Event Auditor

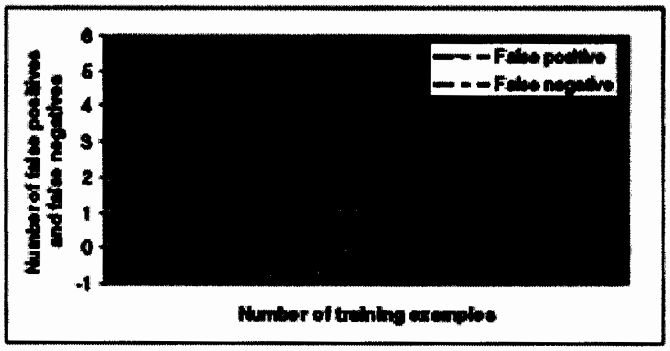
Five legal users and five ill legal(intruders) users have been considered.

Data set which includes 10 days of usage is used as training by neural network. False negatives in high numbers have been resulted by this data. Output also became much better with the increase in sample period.

ii) Behavior

A series of tests were conducted to evaluate the performance and efficiency of the techniques. Analysis technique cost is evaluated by means of performance test. 1 to 100,000 actions were analyzed by means of load test. 0.000271 sec have been taken by an action for processing in our system. "The training time for an input of 30 days of sample behavior took 1.993 seconds. As artificial neural networks are not deterministic, the graph number of false positive and false negative doesn't represent a linear decreasing progression." False positive (legitimate actions marked as attacks) results are shown in fig. 2

Legitimate actions as attacks are not considered by neural network, but same input data give output with higher number of false negatives.



[16]

Figure 2.16: Behavior score results

With the increase in volume of input data the output got better i.e. number of false positive and false negatives becomes low. Neural network has non-deterministic nature because number of false positive will vary for same input data when the experiment will be repeated many times.

iii) Knowledge

Security policies are mentioned in the form of rules that are checked during Knowledge Analysis. 10 to 1,000,000 rules are used in the load test. 0.361sec were taken by hundred thousand rules' comparisons whereas 2.7sec were taken by million rules' comparisons.

Limitation:

But this paper tells us only about the detection and not about the cure of intrusions.

Future Work:

Future Work is not Mentioned

P9→ Desheng Fu, Shu Zhou, Ping Guo "The Design and Implementation of a Distributed Network Intrusion Detection System Based on Data Mining," World Congress on Software Engineering (wcse), vol. 3, pp.446-450, 2009

Goals:

Desheng Fu, Shu Zhou, Ping Guo Proposed a Distributed Network Intrusion Detection System for detecting network intrusions in distributed environment by suing data mining technique. Proposed Distributed Network Intrusion Detection System used an Improved Association analysis algorithm (AAA). This improved AAA was based on FCM network intrusion detection technologies (uses statistical binning) & FP-Growth.

Assumptions:

System model

The structure model of distributed IDS based on data mining is shown below in diagram. Three layers are included in it.

i) Local layer:

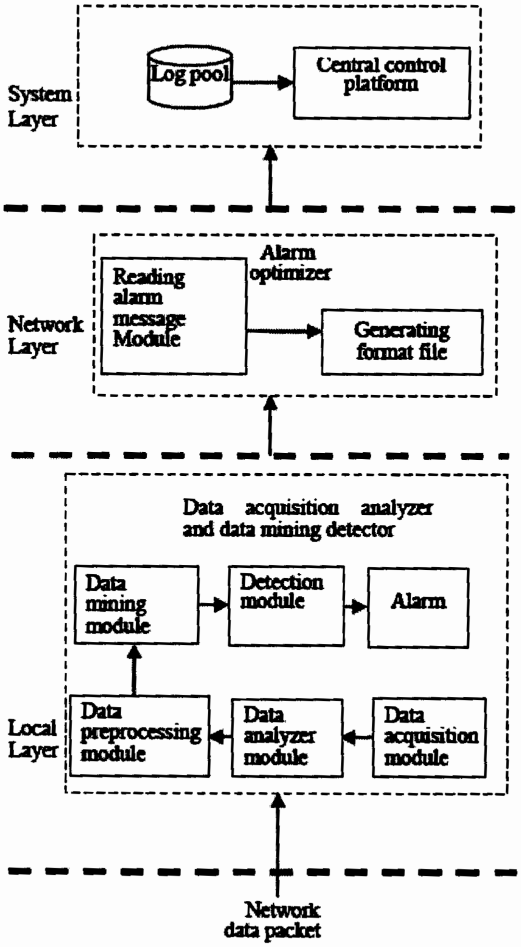
Data mining detector, moreover data acquisition analyzer are included in Local layer. "Data acquisition analyzer and data mining detector have been distributed in the LAN key node". Data of whole network segment is monitored by LAN key node. Intrusion activities are also responded by LAN key node, moreover alarm messages to alarm optimizer are also send by LAN key node.

ii) Network layer:

Alarm optimizer is included in Network layer. Location of alarm optimizer is in WAN. Alarm messages are collected by alarm optimizer. In LAN detection engines are used to send theses alarm messages. Alarm optimizer is also used to store theses alarm messages in logger.

iii) System layer:

Central control platform and logger are included in system layer. System layer is located in central control platform. Friendly and visual control interface is also provided by system layer to the administrator.



[17]

Figure 2.17: System general structure model

Experimental analysis

Dataset chosed is network intrusion detection's dataset in KDDCup99. Four attacks were launched, which were, R2L, U2R, DoS and Probing. Data set included training and testing data sets. Three groups of data were formed in the dataset to show less, medium and more attacking situations. It is shown below in table 2.17

Group	Training /Testing	Totality	Normal	Abnormal			
				DOS	R2L	U2R	Probing
1	Training	2000	1960	24	10	2	4
	Testing	2500	2430	42	18	4	6
2	Training	2500	2310	114	47	10	19
	Testing	3000	2770	138	57	12	23
3	Training	3000	2460	324	135	27	54
	Testing	3500	2870	378	157	31	64

[17]

Table 2.17: Composition of the data set of sample

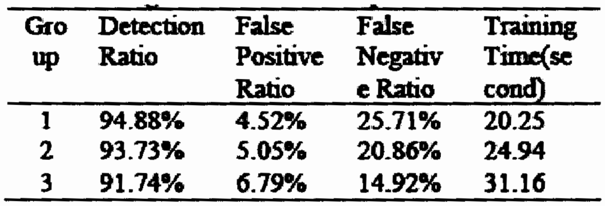
When experiments are performed on testing data sets, then results are given below in Table 2.18

Group	Detection Ratio	False Positive Ratio	False Negative Ratio	Training Time(second)
1	96.01%	3.71%	14.28%	10.31
2	94.93%	4.33%	13.91%	12.69
3	93.34%	5.85%	10.31%	15.02

[17]

Table 2.18: Experimental results of data mining based on the improved FP-Growth algorithm

Experimental results in which there is no improvement are shown on the next page in table 2.19



Group	Detection Ratio	False Positive Ratio	False Negative Ratio	Training Time(second)
1	94.88%	4.52%	25.71%	20.25
2	93.73%	5.05%	20.86%	24.94
3	91.74%	6.79%	14.92%	31.16

[17]

Table 2.19: Experimental results of the FP-Growth algorithm without improvement

From table 2.18 and 2.19 it is clear that by using improved FP-Growth algorithm efficiency of detecting intrusions could be greatly increased. Similarly by using improved FP-Growth algorithm, 1.19% to 1.75% is increase in detection ratio, 30.90% to 44.46% is reduction in false negative ratio and 13.84% to 17.52% is reduction in false positive ratio.

Limitations:

But this proposed IDS only detected Network Intrusions and not Grid and Host specific intrusions. Moreover it provided only detection and not cures of intrusions.

Future work:

No future work is mentioned

P10→ Yu-Fang Zhang, Zhong-Yang Xiong, Xiu-Qiong Wang "Distributed Intrusion Detection System Based on Clustering," Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005

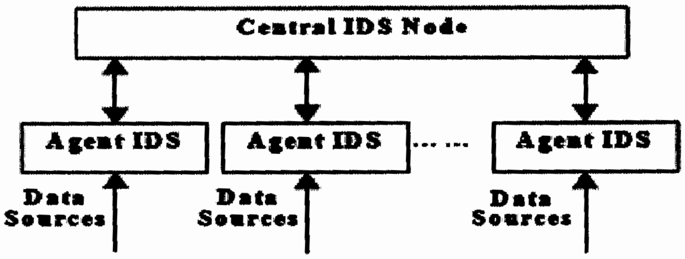
Goals:

In [18] Yu-Fang Zhang, Zhong-Yang Xiong, Xiu-Qiong Wang purposed a Distributed Intrusion Detection System Based on Clustering for detecting intrusions in distributed environment. In this paper a modified clustering algorithm is used for proposed DIDS to improve the efficiency and accuracy of intrusion detection.

Assumptions:

Agent-based DIDS using clustering:

DIDS architecture is shown in fig. below. Agent IDS and central IDS node are logically independent. Although both Agent IDS and central IDS are independent logically but there is a chance that they may reside on the same node. Communication between Agent IDS and central IDS is birectional. One IDS will be running out of many central IDSs on different hosts. If one central IDS will fail then other central IDS will be used.



[18]

Figure 2.18: The simple architecture of DIDS

i) Agent IDS

Collection, normalization, analysis and candidate's anomalies selection of network data is done by Agent IDS. For analysis a modified cluster based algorithm is used by Agent IDS.

ii) Central IDS

Control and co-ordination of all agents is done by central IDS, that's why it is called heart and soul for many Intrusion detection systems. Candidate anomalies are again analyzed by central IDS by using an algorithm.

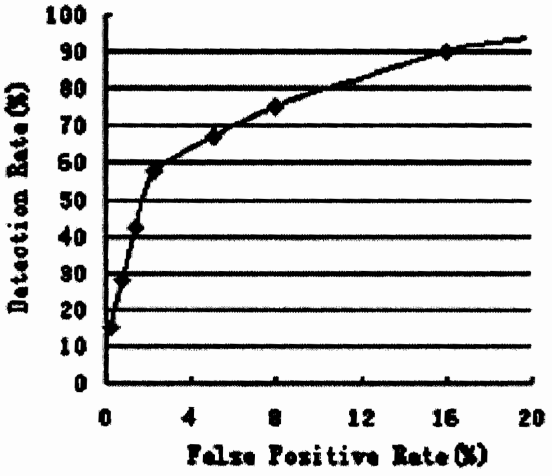
Analysis of Results:

In a military network, intrusions were simulated to obtain an intrusion dataset. "In order to make the data set more realistic, we sampled data sets consisted of 1 to 1.5% attacks and 98.5 to 99% normal instances. For each of data sets, the data is split into two sets: training dataset (40%) and test dataset (60%)." Test data is divided into four data sets of equal size "which simulate the instances collected from different agent IDS". Cluster width is kept 40 and percentage is kept 15% for experiment over training dataset. Similarly for experimental results' performance evaluation K is kept 15.

False positive and detection rates are used for performance's indication. Many datasets with parameters k give the results which are shown in table below.

Using above experimental results Receiver Operating Characteristic (ROC) curve is drawn which is shown in graph below. Graph shows the deviation in false alarm and detection rates at various thresholds.

k	Detection Rate	False Positive Rate
4	15	0.2
6	28	0.8
8	42	1.4
10	58	2.3
12	67	5.1
14	75	8.2
15	90	16



[18]

Figure 2.19: ROC Curve showing

Limitations:

But the limitation of this paper is that it does not tell that how can we deploy this DIDS in grid environment efficiently and moreover it tells only about detection of intrusions and not about the cure of intrusions.

Future Work:

In future homogeneous and heterogeneous distributed systems will be combined with clustering technique.

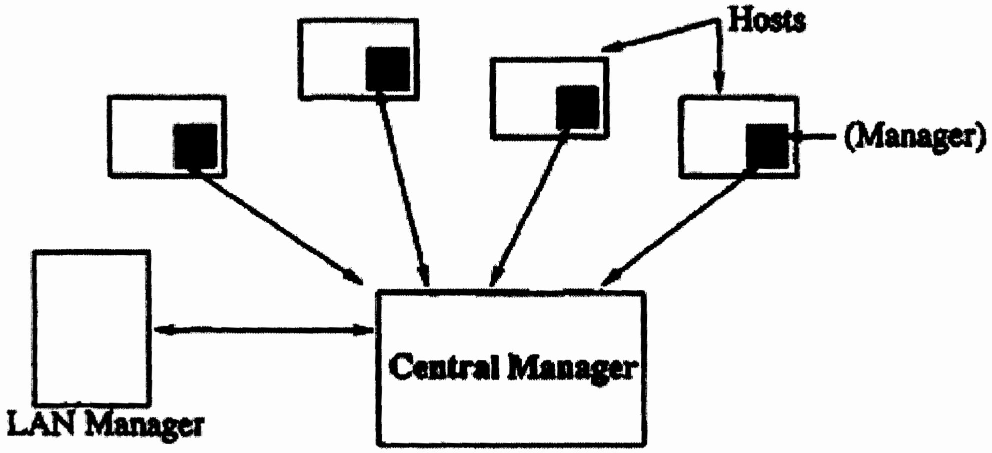
P11→ Steven R. Snupp, James Brentano, Gihan V. Dias, Terrance L. Goan, Tim Grance, L. Todd Heberlein, Che-Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Douglass L. Mansur, Kenneth L. Pon Stephen E. Smahd "A System for Distributed Intrusion Detection," In COMPCOM Spring '91 Digest of Papers, pages 170-176, February/March 1991.

Goals:

Steven R. Snupp, James Brentano, Gihan V. Dias, Terrance L. Goan, Tim Grance, L. Todd Heberlein, Che-Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Douglass L. Mansur, Kenneth L. Pon and Stephen E. Smahd proposed A System for Distributed Intrusion Detection for detecting intrusions in distributed environment. It has removed the limitations of some previous intrusions detection systems which were developed for stand-alone hosts and LAN environments by extending the intrusion detection scope from LAN to WAN environments having arbitrary network topologies.

Assumptions:

Following are the components included in DIDS. LAN manager, host manager and central manager. These components perform different analysis and monitoring functions. Known attack signatures and single events are handled by LAN and Host managers. LAN and Host managers should constantly monitor their allocated domains. LAN and Host managers report events to a central manager. So the duty of central manger is to analyze reported events. Processing of queries and responses and transfer of data is because of the bidirectional communication of the central manager with LAN and host managers. Correlation of LAN and host manager's information is also done by Central manager; moreover user interface is also supported by Central manager. Host manager's presence is transparent to the host's user because in background collection of processes run as host manager. Processes running on single hosts on the network are also called Central and host manager. Data's high level analysis is in the responsibilities of a Central managers. All the managers either they are network, host or central they are logically independent but may physically present at same node. So the division of IDS functionalities among different DIDS's is quite balanced and it is clear from the above discussion. DIDS is shown in fig. below.



[19]

Figure 2.20: Architecture of the Distributed Intrusion Detection System

Result Analysis:

Result analysis is not performed.

Limitations:

But this paper has only provided detection and not cures of intrusions, moreover results are also not analyzed.

Future Work:

Future work is not mentioned.

P12→ Sattarova Feruza Yusufovna "Integrating Intrusion Detection System and Data Mining," International Symposium on Ubiquitous Multimedia Computing (umc), pp.256-259, 2008

Goals:

An IDS model as well as its limitation in determining security violations has been presented by Sattarova Feruza Yusufovna in this paper. Furthermore, this paper has

IDS Limitations:

Following are the limitations in IDS, they are.

1) Known attacks could be detected by current IDS, because current IDS are designed for them , therefore they could not detect malicious attacks

2) Logs could reach million records/day depending upon the type of intrusion tool selected. This is problem of data overloading

3) False positive and false negative rates are high in current IDS.

Data Mining Techniques

Following are the data mining techniques that could be used in intrusion detection.

1) Feature Selection
2) Machine Learning
3) Classification Techniques.
4) Clustering Techniques.
5) Statistical Techniques
6) Predictive Analysis

This paper has suggested that Intrusion detection process could be made better by integrating all the above data mining techniques instead of using a single data mining technique for intrusion detection.

Result analysis:

In this paper neither prototype nor experimental results are given.

Limitations:

But the limitation of this paper is that it does not tell that how can we deploy these data mining approaches for an intrusion detection system in Grid Environment.

Future work:

Future work is not mentioned.

P13→ Yi Hu, Brajendra Panda "A Data Mining Approach for Database Intrusion Detection," Symposium on Applied Computing Proceedings of the ACM, Pages: 711 – 716, 2004

Goals:

A data mining approach for detecting malicious transactions in a Database System has been proposed by Yi Hu and Brajendra Panda. In the paper data dependencies between data items in the database have been mined by means of a data dependency miner. Data dependency miner used database logs to find data correlation (data dependencies) between data item. A transaction is identified as malicious transaction if it is not according to the data correlation found by Data dependency miner.

Assumptions:

Data dependencies in database system could be found by using a data mining approach. By using the database logs classification rules can be deduced. Data dependencies are actually these classification rules. These data dependencies are used to find out anomalous or malicious transactions in such a way that if a transaction is not according to data dependencies mined then transaction is said to be anomalous or malicious. The sensitivity of this approach to user behavior's change is very low, moreover changes in data correlations are very rare and uncommon. An attacker who breaks access control mechanism can launch a malicious transaction, theses malicious transactions could be identified by using the proposed model. The proposed method is for relational database, moreover database should log both read and write operation records because it is the requirement of this proposed method.

Analysis of Results:

Proposed method's performance is checked by many experiments. Proposed method's True positive and false positive rates were checked by using two distinct database logs. One log contains malicious and the other log contains normal user transactions. Baseline setting of experiment is shown in table below.

Baseline setting is shown in table 2.20 on the next page.

support	confidenc e	# of read operations	# of write operations	# of transactions
.15	.75	2	2	2000

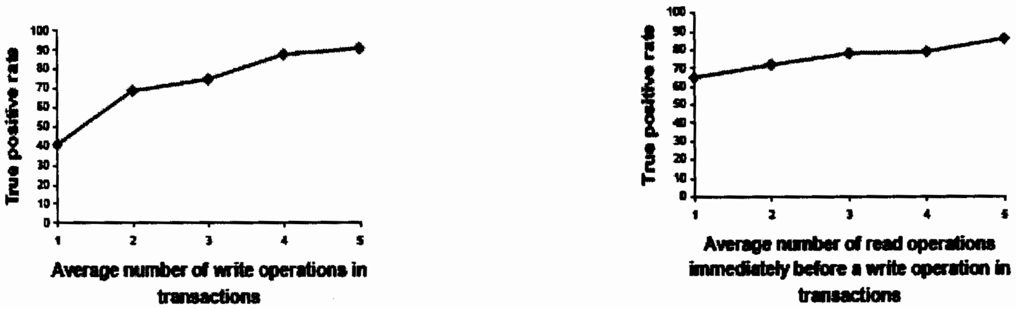
[21]

Table 2.20: Baseline Setting of Experiments

Change in any of these parameters could effect the IDS process. IDS response to change in any of theses parameters is tested.

i) Using log of malicious transactions:

Because of the results of malicious transaction's detection the true positive rates as outcome are shown below in fig. 2.22. Because of the stronger data dependency among data items increase in true positive rate is steady. The change in average number of write operations from 1 to 5 resulted change (increase) in positive rate from 41% to 91%. Similarly the change in "average number of read operations immediately before a write operation in transactions" resulted change (increase) in true positive rate from 65% to 86%. So it is clear form the two graphs that an increase in detection rate will be there if transactions contain more update statements, so sensitivity of true positive rates is higher for average number of write operations



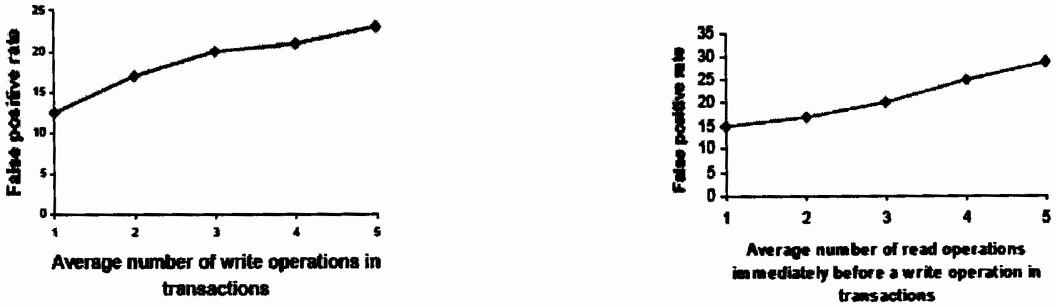
[21]

Figure 2.22: True Positive Rates in Detecting Malicious Transactions

ii) Using log of normal transactions:

Because of the results of normal transaction's detection the false positive rates as outcome are shown below in fig. 2.23 Because of the weak data dependency among data items false positive rates were very low i.e. 12.5%. Similarly "when the average number of read operations immediately before a write operation in transactions" will be 5 then the maximum false positive rate will be 29%.

So from the two figures it is clear that increase in true positive rate is very higher from false positive rates with increase in data dependency.



[21]

Figure 2.23: False Positive Rates in Testing Normal User Transactions

Limitations:

The limitation of this paper is that it does not tell that how can we deploy this data mining approach for detecting malicious transactions in grid environments.

Future Work:

Future work is not mentioned.

2.4. Research Problem Definition

National Center of Physics, Islamabad maintains a grid computing infrastructure. The IT department of the center is trying to come up with a *Self Healing System* for grid computing at the center, which can provide an expert security system that will monitor and fix the problems and threats encountered in the grid infrastructure.

The self healing frame work is a very large scale research problem, covering several research and development scenarios. Our research problem is to design and implement a model for automatic discovery and analysis of attack patterns and attacks from log information and from other means and ways in a grid infrastructure.

Objective or goal of our research is to develop an expert security system that will monitor and fix the attacks and threats that are found and that can remove the limitations of previously proposed intrusion detection system/security systems. In case of new or undiscovered threats the system would be intelligent enough to get direction from the system administrator.

To find out current proposed IDSs (Intrusion detection systems) and the limitations in current proposed IDSs a literature survey has been carried out, whose critical evaluation is given below.

Critical evaluation is arranged as follows. Section 3.1 has critically evaluated those IDS which are designed for Grid whereas Section 3.2 has critically evaluated those IDS which are not designed for Grid environment.

2.4.1. Critical Evaluation of Grid based IDS

The key difference between "PGIDS" proposed by Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang [9] and "Service Based ID system" [11], "GIDA" [10], "solution for intrusion detection in grid and cloud computing environment" [16] is that "PGIDS" detected only network specific intrusions and it could not detect GRID specific intrusions whereas "Service Based ID system" [11], "GIDA" [10], and "solution for intrusion detection in grid and cloud computing environment" [16] detected GRID specific intrusions.

"PGIDS" [9], **FGIDS [12]** and GIDS [13] are almost same because these three proposed architectures could all detect only network specific attacks.

Similarly "PGIDS" [9] has detected only network specific intrusions and not GRID specific intrusions whereas "distributed grid-based intrusion detection system architecture" [14] has provided an approach for detecting network, host and grid specific attacks separately in Grid environment and MAIDG [15] has detected grid specific intrusions. Moreover "PGIDS" [9] has analyzed the results whereas "distributed grid-based intrusion detection system architecture" [14] and MAIDG [15] has neither given the prototypes nor analyzed the results

A key difference between "Service Based ID system" proposed by Andrea Bosin, Nicoletta Dessì, Barbara Pes and Dipartimento di Matematica e Informatica [11] and GIDA [10], **FGIDS [12]**, GIDS [13], "distributed grid-based intrusion detection system architecture" [14] is that "Service Based ID system" [11] has not analyzed the results whereas GIDA [10], **FGIDS [12]**, GIDS [13] and "distributed grid-based intrusion detection system architecture" [14] have analyzed the results.

"Service Based ID system" [11.], "distributed grid-based intrusion detection system architecture" [14] and MAIDG [15] are almost same in term of types of intrusions detected and results analyzed.

The key difference between GIDA [10] proposed by Sanjeev Rana, Rajneesh Gujral and Manpreet Singh and FGIDS [12] is that GIDA [10] detected grid intrusions whereas FGIDS [12] and GIDS [13] detected network specific intrusions.

Similarly GIDA [10] has only used one approach to intrusion detection i.e. anomaly detection approach (to detect user behavior deviation from normal behavior) instead of using both of the misuse approaches ("where predefined patterns known to be malicious are looked for") and anomaly detection approach (to detect user behavior deviation from normal behavior)for intrusion detection whereas "solution for intrusion detection in grid and cloud computing environment" [16] has provided both the approaches for intrusion detection.

Moreover GIDA [10] has provided an approach for detecting grid specific attacks whereas "distributed grid-based intrusion detection system architecture" [14] has provided an approach for detecting network, host and grid specific attacks separately in Grid environment.

Similarly difference between GIDA [10] and MAIDG [15] is that GIDA [10] has analyzed the results whereas MAIDG [15] has neither given the prototype nor analyzed the results.

A key difference between **FGIDS [12]** and "solution for intrusion detection in grid and cloud computing environment" [16] is that **FGIDS [12]** has detected network specific attacks in GRID environment whereas "solution for intrusion detection in grid and cloud computing environment" [16] has detected Grid specific attacks.

Similarly A key difference between **FGIDS [12]** and "distributed grid-based intrusion detection system architecture" [14], MAIDG [15] is that **FGIDS [12]** has detected network specific attacks and not grid specific attacks in GRID environment whereas "distributed grid-based intrusion detection system architecture" [14] and MAIDG [15] has detected Grid specific attacks in GRID environment. Moreover **FGIDS** [12] has analyzed the results whereas "distributed grid-based intrusion detection system architecture" [14] and MAIDG [15] has neither given the prototype nor analyzed the results.

Both **FGIDS** [12] and GIDS [13] are same in terms of types of intrusions detected and results analyzed

Key difference between GIDS [13] proposed Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang and Po-Chi Shih and "solution for intrusion detection in grid and cloud computing environment" [16], MAIDG [15] is that GIDS [13] has detected network specific attacks and not Grid specific attacks in GRID environment whereas "solution for intrusion detection in grid and cloud computing environment" [16] and MAIDG [15] have detected Grid specific attacks in GRID environment. Moreover GIDS [13] has analyzed the results whereas "solution for intrusion detection in grid and cloud computing environment" [16] and MAIDG [15] have neither given the prototype nor analyzed the results.

Similarly key difference between GIDS [13] and "distributed grid-based intrusion detection system architecture" [14] is that GIDS [13] has detected network specific attacks in GRID environment whereas "distributed grid-based intrusion detection system architecture" [14] has detected Grid ,network and host specific attacks in GRID environment. Moreover GIDS [13] has analyzed the results whereas "distributed grid-based intrusion detection system architecture" [14] has neither given the prototype nor analyzed the results.

Difference between "solution for intrusion detection in grid and cloud computing environment" [16] proposed by Kleber Vieira, Alexandre Schulter, Carlos Westphall and Carla Westphall and "distributed grid-based intrusion detection system architecture" [14] Is that "solution for intrusion detection in grid and cloud computing environment" [16] has detected grid specific attacks whereas "distributed grid-based intrusion detection system architecture" [14] has provided an approach for detecting network, host and grid specific attacks in Grid environment. Another Difference in “solution for intrusion detection in grid and cloud computing environment” [16] results are analyzed whereas “distributed grid-based intrusion detection system architecture” [14] has neither given the prototype nor analyzed the results.

Similarly difference between “solution for intrusion detection in grid and cloud computing environment” [16] and MAIDG [15] is that in “solution for intrusion detection in grid and cloud computing environment” [16] results are analyzed whereas in MAIDG [15] results are not analyzed.

A key difference between “distributed grid-based intrusion detection system architecture” [14] proposed by Alexandre Schulter, Júlio Albuquerque Reis, Fernando Koch and Carlos Becker Westphall and MAIDG [15] is that “distributed grid-based intrusion detection system architecture” [14] has detected network, host and grid specific attacks separately in GRID environment whereas MAIDG [15] has detected Grid specific attacks in GRID environment.

Summary of the above critical evaluation is shown in table 2.21 and table 2.22 below.

Paper studied	Types of Intrusions Detected separately in Grid Environment			Analysis		
	Host specific	Network specific	Grid specific	Analysis Performed	Comprehensive	Unsatisfactory / Non comprehensive
PGIDS [9]	No	Yes	No	Yes	Yes	No
GIDA [10]	No	No	Yes	Yes	Yes	No
Andrea Bosin [11.]	No	No	Yes	No	-----	-----
FGIDS [12]	No	Yes	No	Yes	Yes	No
GIDS [13]	No	Yes	No	Yes	Yes	No

Alexandre Schulter[14]	Yes	Yes	Yes	Yes	No	Yes
MAIDG[15]	No	No	Yes	No	-----	-----
Kleber Vieira [16]	No	No	Yes	Yes	Yes	No

Table 2.21: Summary of Critical Evaluation of Grid based IDS

Paper studied	Cure Against intrusions	Approach Used for Detecting Grid Specific attacks		
		Anomaly/behavior based	Misuse/signature based	not mentioned
PGIDS [9]	No	-----	-----	-----
GIDA [10]	No	Yes	No	No
Andrea Bosin [11.]	No	No	No	Yes
FGIDS [12]	No	-----	-----	-----
GIDS [13]	No	-----	-----	-----
Alexandre Schulter[14]	No	No	No	Yes
MAIDG[15]	No	No	No	Yes
Kleber Vieira [16]	No	Yes	Yes	No

Table 2.22: Summary of Critical Evaluation of Grid based IDS

2.4.2. Critical Evaluation of Non-Grid based IDS

A key difference between "The Design and Implementation of a Distributed Network Intrusion Detection System Based on Data Mining" [17] and "DISTRIBUTED INTRUSION DETECTION System BASED ON CLUSTERING" [18] is that Distributed Network Intrusion Detection System [17] has detected only network attacks whereas "DISTRIBUTED INTRUSION DETECTION System BASED ON CLUSTERING" [18] detected network as well as host attacks.

Similarly in Distributed Network Intrusion Detection System [17] analysis of result is performed whereas in data mining based IDS [20] neither analysis of result is performed nor prototype is given.

Distributed Network Intrusion Detection System [17] is designed for distributed environment whereas Database Intrusion Detection approach [21] is designed to be deployed in any node either the node is alone or is in grid or in distributed environment i.e. Database Intrusion Detection approach [21] is designed for general environment.

A key difference between "Distributed Intrusion Detection System Based on Clustering" [18] proposed by Yu-Fang Zhang, Zhong-Yang Xiong, Xiu-Qiong and A System for Distributed Intrusion Detection [19] is that "Distributed Intrusion Detection System Based on Clustering" could detect host and network intrusions whereas A System for Distributed Intrusion Detection [19] could detect only network intrusions.

Similarly IDS based on clustering [18] is designed for distributed environment whereas A Data Mining Approach for Database Intrusion Detection [21] is designed to be deployed in any node either the node is alone or is in grid or in distributed environment i.e. it is designed for general environment.

Simlarly in IDS based on clustering [18] analysis is performed whereas in Integrating Intrusion Detection System and Data Mining [20] analysis of results is not performed.

A key difference between A System for Distributed Intrusion Detection [19] and Integrating Intrusion Detection System and Data Mining [20], Data Mining Approach for Database IDS [21] is that System for Distributed Intrusion Detection [19] is designed for distributed environment whereas Integrating Intrusion Detection System and Data Mining [20] and Data Mining Approach for Database IDS [21] are designed for general environment.

Difference between Integrating IDS and Data Mining [20] and data Mining Approach for Database Intrusion Detection [21] is that in Integrating IDS and Data Mining [20] result analysis is not done whereas in data Mining Approach for Database IDS [21] analysis of result is performed.

Summary of the above critical evaluation is shown in table 2.23 and table 2.24.

Papers Studied	**Types of intrusions detected**		**Analysis of results**		
	Host	**Network**	**Analysis of results Performed**	**Comprehensive**	**Unsatisfactory/ Non comprehensive**
Desheng Fu[17]	No	Yes	Yes	Yes	No
Yu-Fang Zhang[18]	Yes	Yes	Yes	Yes	No
Steven R. Snupp [19]	No	Yes	No	-----	-----
Sattarova Feruza [20]	Yes	Yes	No	-----	-----
Yi Hu [21]	Yes	No	Yes	Yes	No

Table 2.23: Summary of Critical Evaluation of non-Grid based IDS

Papers Studied	**Environment for which IDS is designed**		**Cure Against intrusions**
	Distributed	**General**	
Desheng Fu[17]	Yes	No	No
Yu-Fang Zhang[18]	Yes	No	No
Steven R. Snupp [19]	Yes	No	No
Sattarova Feruza [20]	No	Yes	No
Yi Hu [21]	No	Yes	No

Table 2.24: Summary of Critical Evaluation of non-Grid based IDS

So during the above literature survey and critical evaluation it has been found that current Intrusion detection systems are not comprehensive in terms of providing cure against detected intrusions. So after examining current proposed IDSs and the limitations in current proposed IDSs a model for Detection and Cure of Attacks in Grid environment has been proposed, moreover its Prototype has been given in the next coming chapters.

Chapter #3 Proposed Solutions

Proposed Solution

3.1. Architectural Design:

The idea is to develop a security system that could act like a human internal defense system against the diseases. Each node in an administrative domain will contain an ID (intrusion detection) service. Moreover each administrative domain will also contain a centralized database server. Centralized database server contains knowledge database, Error database and Local cure database.

An ID service consist of other services like HID (host intrusion detection) service, NID (network intrusion detection) service and GID (grid intrusion detection service). This ID service will also use some of the services of Grid middleware (Globus, glite).

An ID service that works at each node is shown in diagram.

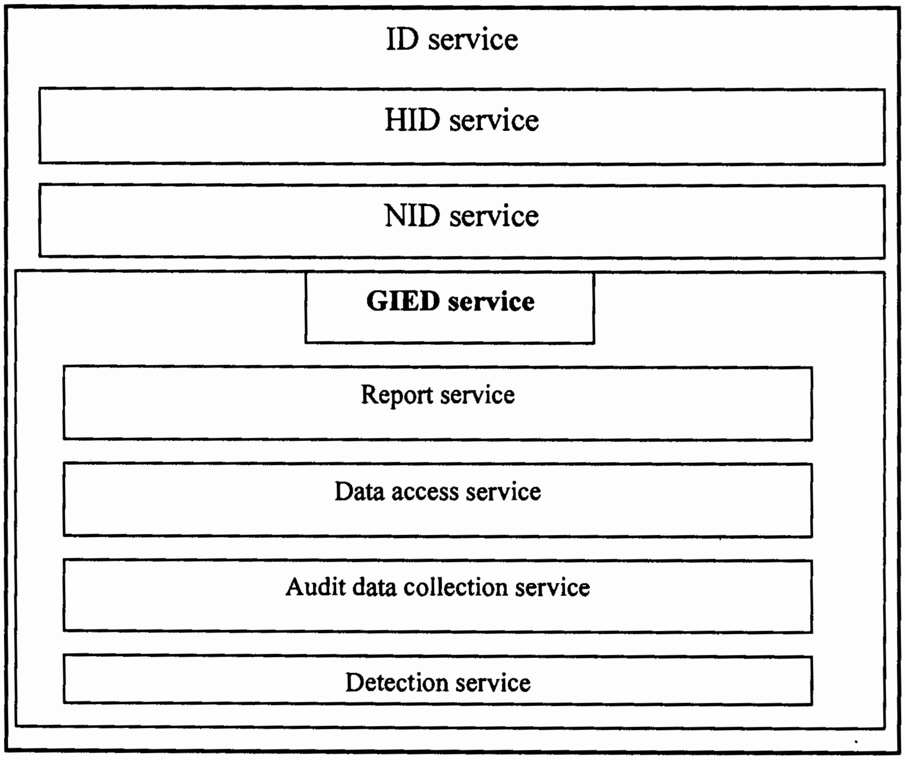


Figure 3.1: ID service

1. **HID (host intrusion detection) service:** This service is used to detect host specific intrusions/attacks. This service will use the commercially available best

Host intrusion detection systems (either in software form or hardware form) to detect host specific intrusions.

2. **NID (network intrusion detection) service:** This service is used to detect network specific intrusions/attacks. Just like HID service this service will also use commercially available best Network intrusion detection systems (either in software form or hardware form) to detect network specific intrusions.

3. **GIED (grid intrusion and error detection) service:** This service is used to detect and provide cure against grid specific intrusions and errors. GID service in turn consists of following services.

   a) **Report service:** Report service is used by the GIED service to report the domain administrator that either a cure against intrusion or system error has been provided or not.

   b) **Data access service:** This service is used by GIED service to access data in local cure, knowledge and error database.

   c) **Audit data collection service:** This service is used by GIED service to collect audit data i.e. logs. This audit data is then used by detection service.

   d) **Detection service:** This service is used by GIED service to compare 'patterns' in audit data (i.e. data from logs) with 'attack patterns' and 'error patterns' in 'knowledge and error database' to detect grid specific intrusions and errors.

3.2. Architecture for Intrusion Detection in Grid Environment

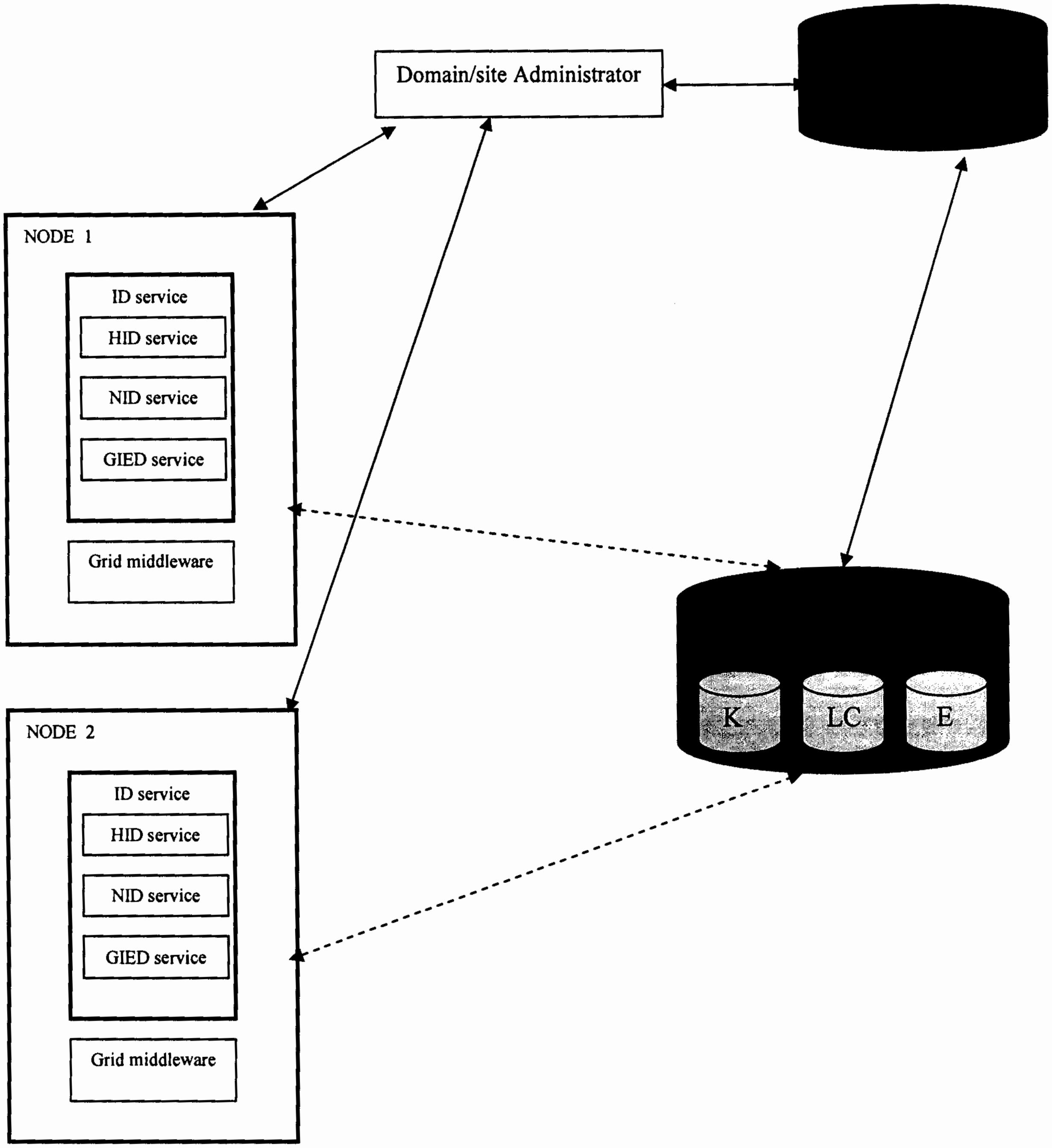


Figure 3.2: Architecture for Intrusion Detection in Grid Environment

i) Node:

Node contains the services and resources necessary for intrusion detection.

ii) ID service:

ID service is used for detecting host, network and Grid specific intrusions at a node. ID service and its components have been described above in detail.

iii) Grid Middleware:

ID service also uses some of the services or functionalities of Grid Middleware.

iv) Local Cure Database:

Local cure database stores the strategies(i.e. cure) that should be taken against intrusions and errors detected by local nodes.

v) Knowledge and Error Databases:

Knowledge and Error Database is used for Knowledge based Analysis. GID service use Knowledge and Error Database to detect intrusions and errors. Attack and error patterns stored in knowledge database are used to compare with patterns in audit data for detecting intrusions and errors.

vi) Global Cure Database:

Global cure database is accessed by different domain administrators to access information about newly detected intrusion and newly detected errors. Moreover Knowledge and Error databases of each domain are synchronized with Global Cure Database i.e. when any new intrusion or error is discovered in any domain then it is also stored in Global Cure Database which can be used in future by any other domain administrator for future reference.

vii) Domain Administrator:

Domain Administrator helps the local nodes in the domain for taking actions against newly detected intrusions and newly detected errors by updating the local cure, knowledge and error databases in centralized database servers.

3.3. Cure Strategy against Intrusions and Errors:

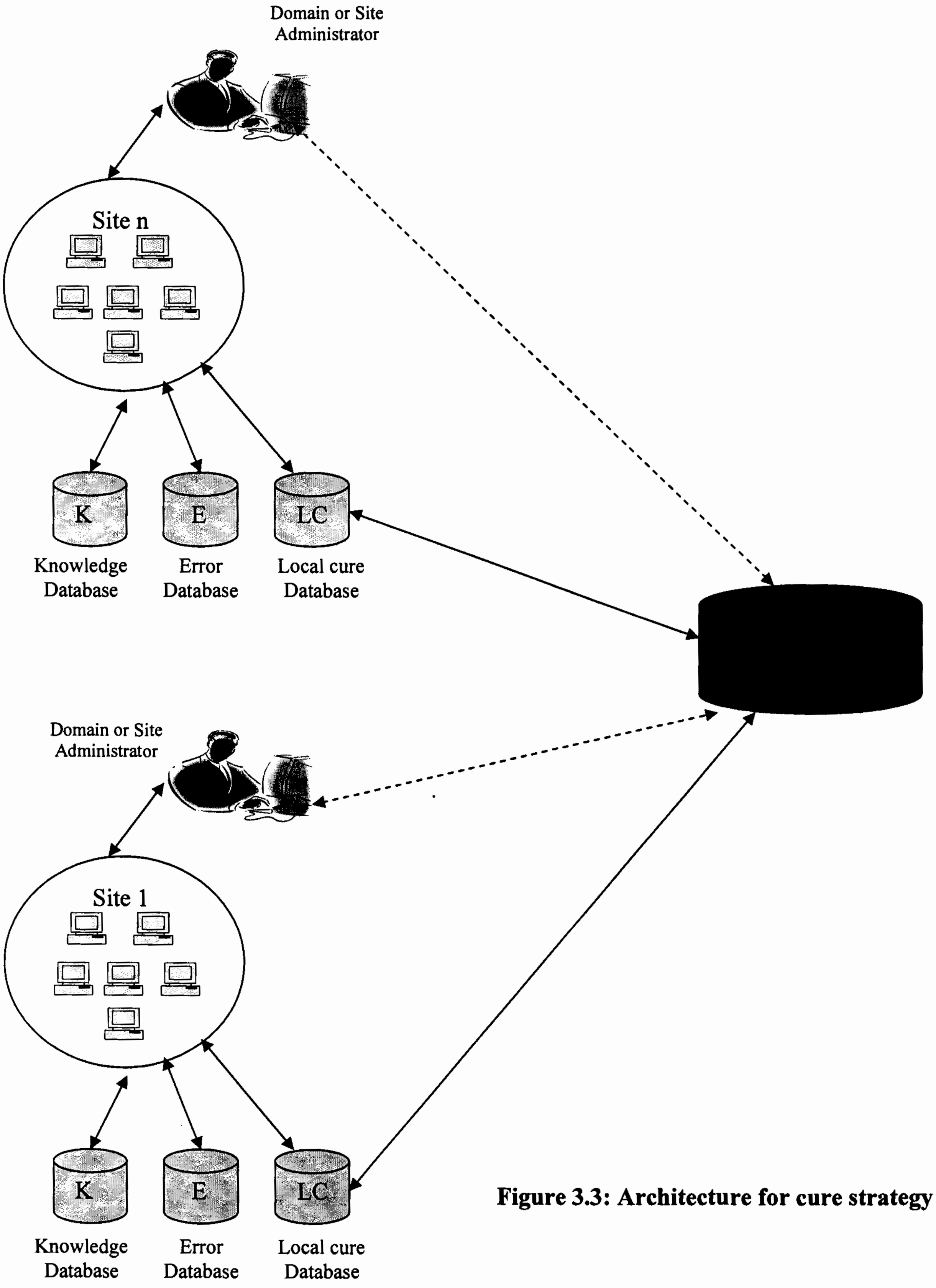


Figure 3.3: Architecture for cure strategy

Fig 3.3: In the Figure Database "LC" is local *cure database* in the centralized database server of a site (administrative domain). If some attack is detected then ID service will try to find the cure or action against attack from local cure database "LC". If the cure or action against intrusion or attack is not found locally, then ID service will consult with the Domain administrator. Domain administrator will consult with the Global cure database to find out the required actions against the intrusion/attacks.

When a GIED service at a node will detect an intrusion or an error(not intrusion) then it consults the local cure database in order to check out that which actions it should take against the intrusion or an error(not intrusion).

If GIED find the required action to be taken against the detected intrusion or error then after taking action GIED will report the domain administrator about the action taken by using report service otherwise If GIED service does not find the required action against intrusion or error then GIED will consult with domain administrator by using a report service. Domain administrator will consult with the Global cure database to find out the required action against the intrusion or error (detected by local node).

If the domain administrator finds the required action from the global cure database then he will store the information about cure in local cure database for making future references easy. Actually Global cure database is the database which is synchronized with all the local cure databases of different sites (domain administrations).

If domain administrator could not find the required cure from Global Cure databases then he will himself suggest the required action to be taken against the intrusion or error. Domain administrator will store that *newly suggested cure information* at the local cure database of his domain. When this local cure database will synchronize with the global cure database, then this newly suggested cure information will also be stored in global cure database. Now in this way this *newly suggested cure information* will be available for all other domain administrators for future references.

The beauty of this proposed architecture is that if an intrusion will not be detected by HID service then it will be detected by NID service and if it will not be detected by NID service then it will be detected by GIED service.

Claims:

Current IDS could only detect intrusions but they don't detect errors which could aid in downgrading the performance and resources availability just like intrusions. But our proposed architecture is flexible in a way that it could not only detect intrusion but it could also detect system errors which Current IDS do not detect. Moreover our claims are that, our proposed architecture is comprehensive in terms of providing cure against detected intrusions and system errors moreover with the passage of time our proposed model will evolve in terms of providing cures against detected intrusions and system errors.

Chapter # 4

Prototype & Results

Prototype & Results

4.1. Implementation details of HID and NID services

As we have mentioned that for HID and NID services we will use commercially available best software or hardware for detecting network and host specific intrusions. Therefore hardware which we will use for detecting host and network specific intrusion will be a Cisco firewall or any other device.

4.2. Implementation details of GIED service

4.2.1. Audit Data Collected for GIED service

In prototype Different logs are collected as audit data for GIED service.

i) Worker Node Logs.

ii) PBS server logs

iii) PBS client logs

iv) Var system log (a Log maintained at all the nodes)

4.2.2. Database Used

In prototype we have used a Microsoft SQL server light as a back end server for storing and accessing knowledge and error database and moreover local cure database. Global cure database has also been created by using Microsoft SQL server light.

4.2.3. Language Used

Java has been used for implementing GIED service. We have been chosen because the libraries in Java are more comprehensive in terms of providing functionalities as compared to libraries of other languages like C++, sea sharp etc.

4.3. UML Diagrams

UML diagrams for impimenting GIED service are shown below.

4.3.1. Class Diagram

Class diagram of GIED service is given below

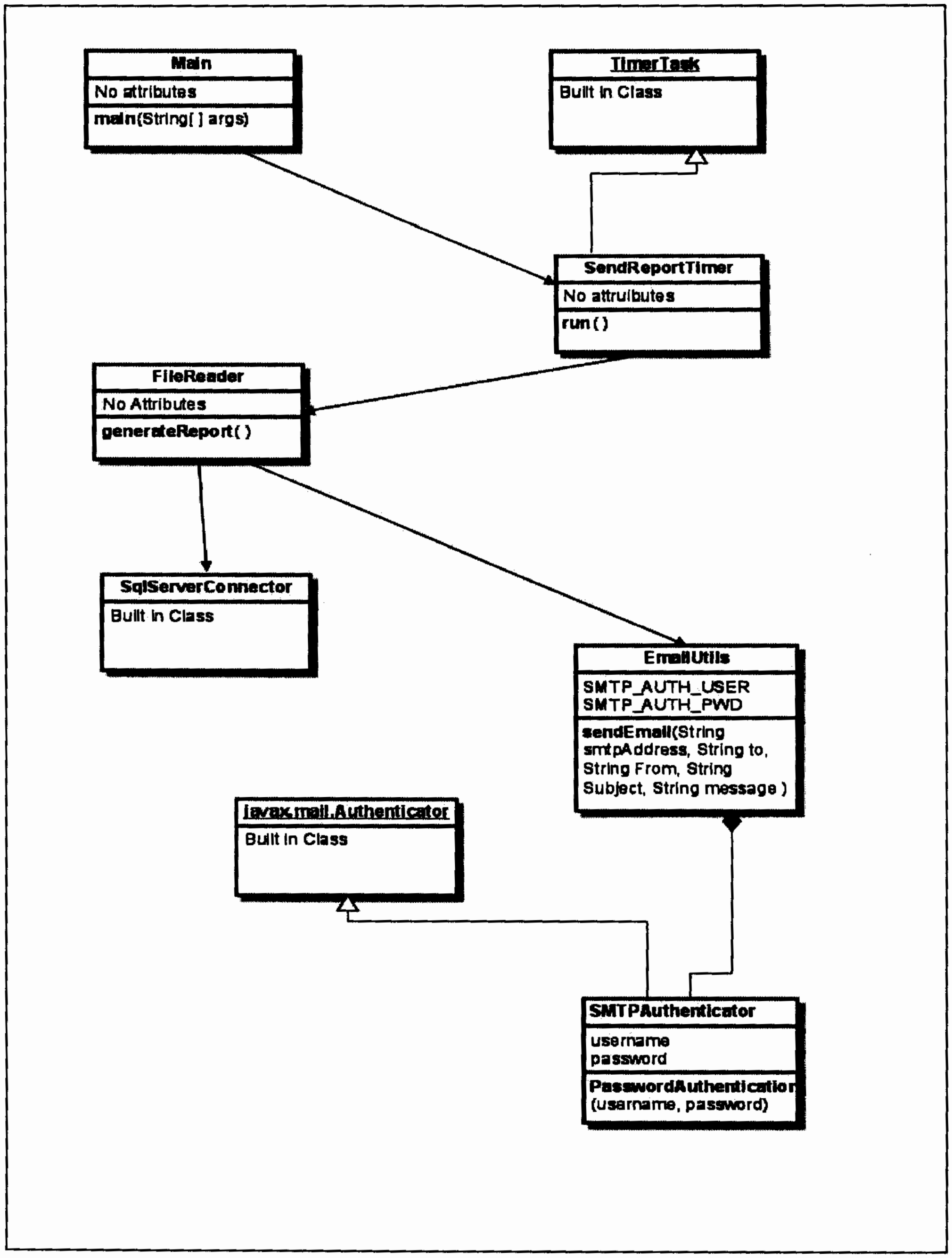


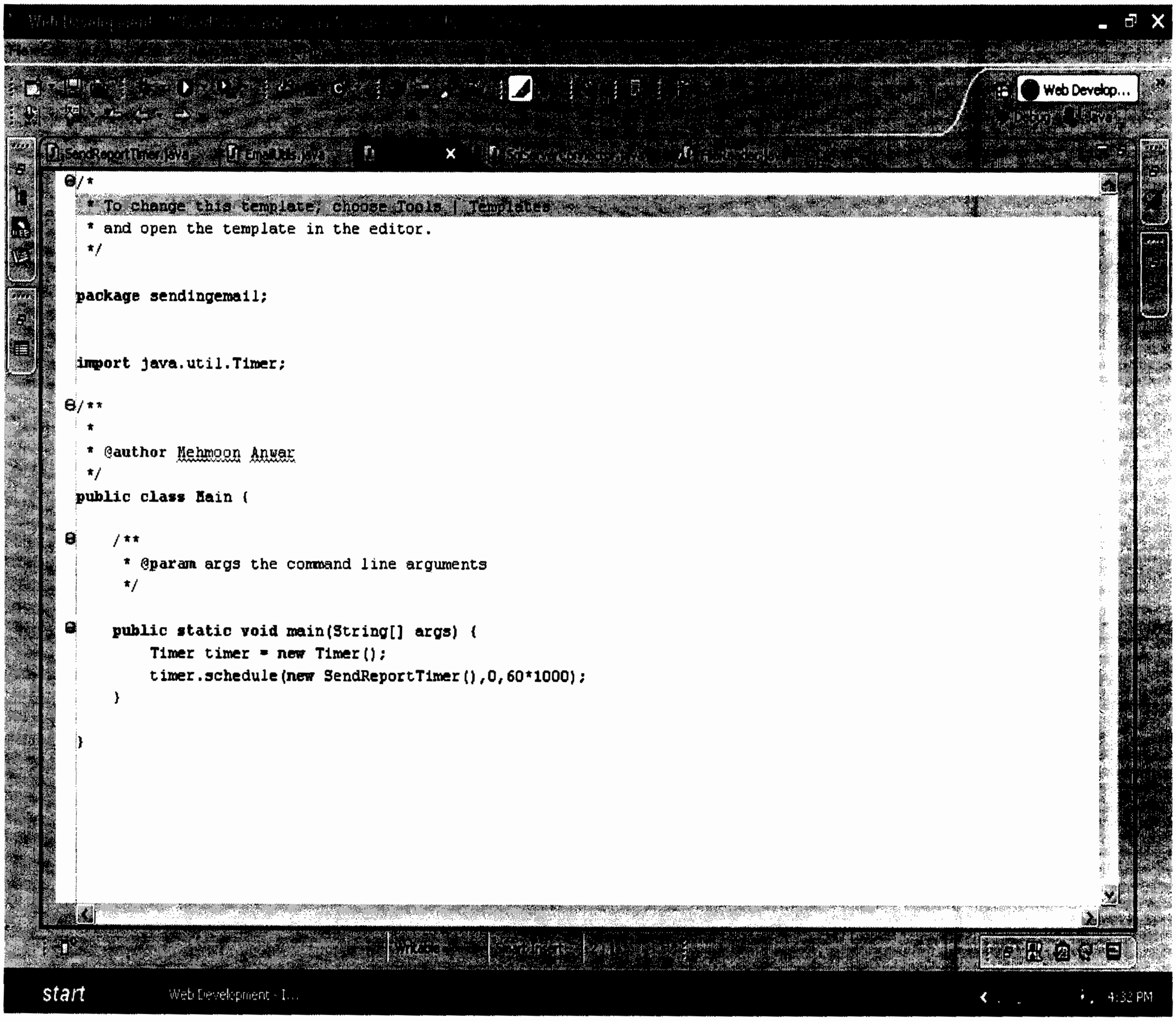
Figure 4.1: Class diagram

4.4. Code for implementing GIED service

Classes used for implementing GIED service are shown below.

4.4.1. 'Main' Class

Basically this is our boundary class. By using this our application will start functioning. Basically this class has a method called 'Main( )'. This method starts our timer and calls 'SendReportTimer' class after every 10 minutes. Screen shot of this Main.Java class is shown below.

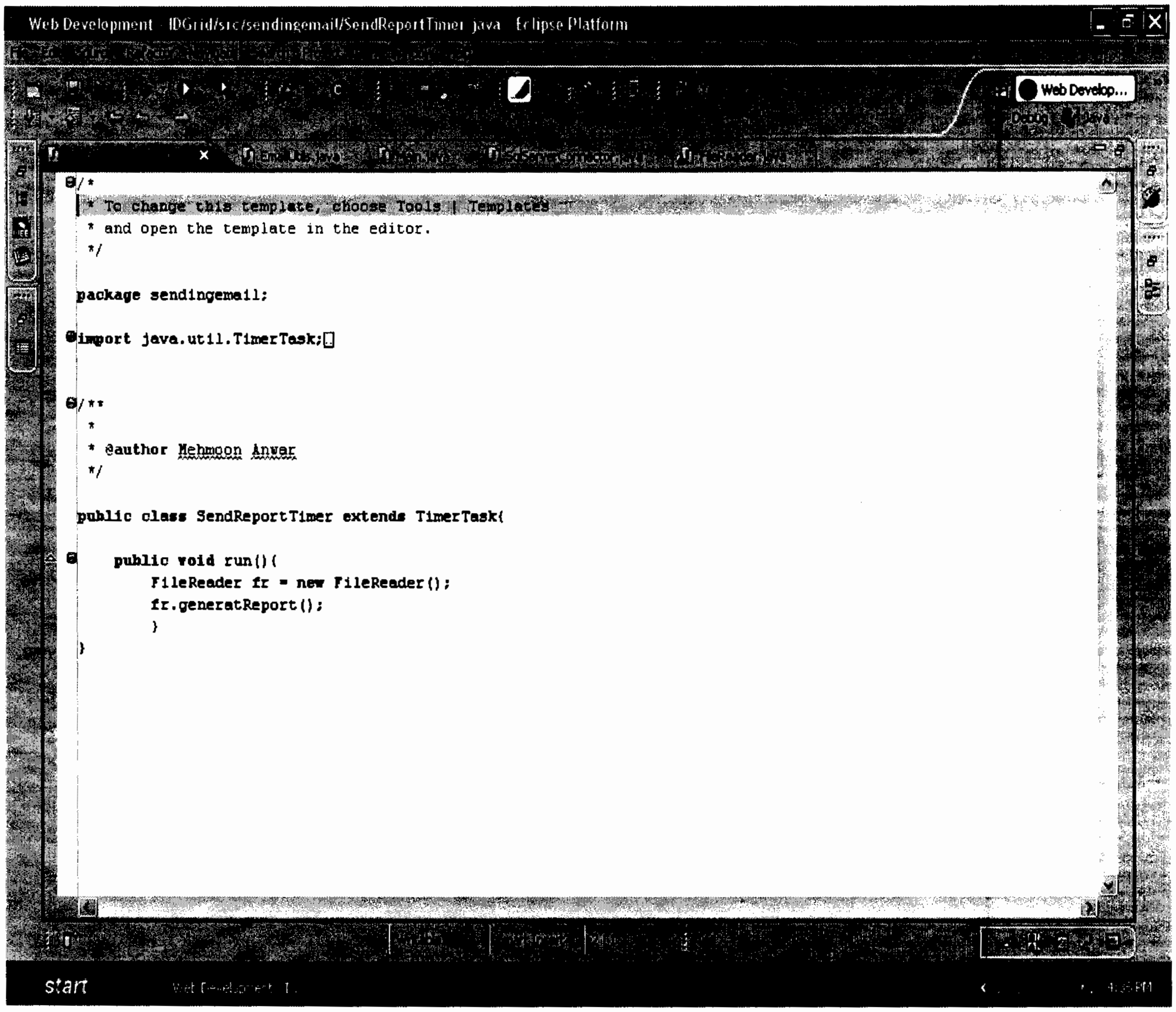


```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package sendingemail;


import java.util.Timer;

/**
 *
 * @author Mehmoon Anwar
 */
public class Main {

    /**
     * @param args the command line arguments
     */

    public static void main(String[] args) {
        Timer timer = new Timer();
        timer.schedule(new SendReportTimer(),0,60*1000);
    }

}
```

Figure 4.4: Main class

4.4.2. 'SendReportTimer' Class

This class will search a specific word within a log file and generate a report by calling a 'generateReport' method of 'fileReader' class. Screen shot of 'SendReportTimer' class is shown below.



Web Development IDGrid/src/sendingemail/SendReportTimer java Eclipse Platform
Web Develop...
start

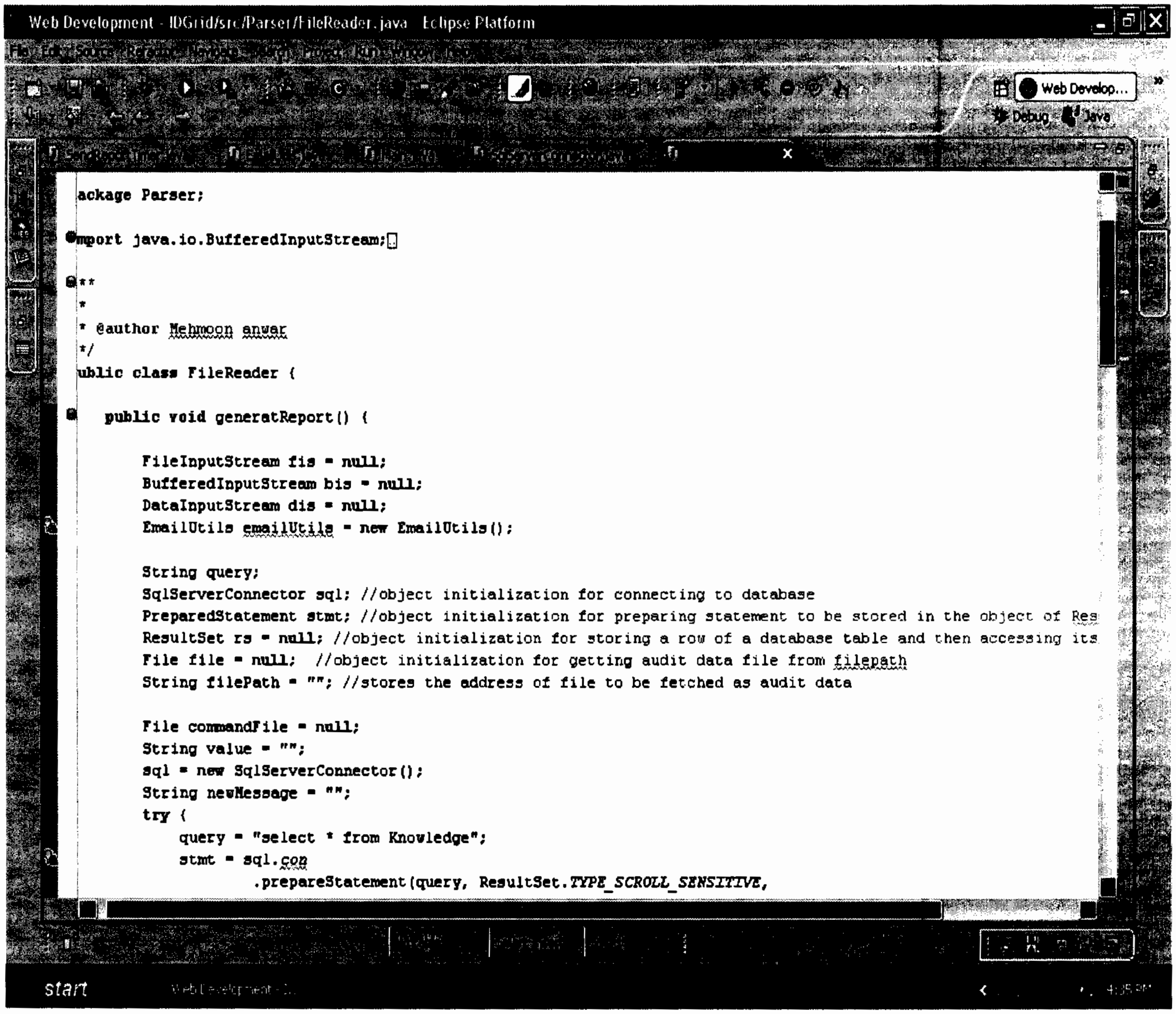
```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package sendingemail;

import java.util.TimerTask;

/**
 *
 * @author Mehmoon Anwar
 */

public class SendReportTimer extends TimerTask{

    public void run(){
        FileReader fr = new FileReader();
        fr.generatReport();
        }
}
```

Figure 4.5: SendReportTimer class

4.4.3 'FileReader' Class

First of all 'fileReader' class will make a connection with database by using 'SqlServerConnector' class. After making connection with the database 'fileReader' class takes data from 'knowledge' table in the databases and then it search for intrusion and error patterns in audit data log file. Then it makes a report and send it to a domain administrator by calling a method ‘sendEmail’ of class ‘EmailUtils’.

The code of this ‘fileReader’ class is shown below.

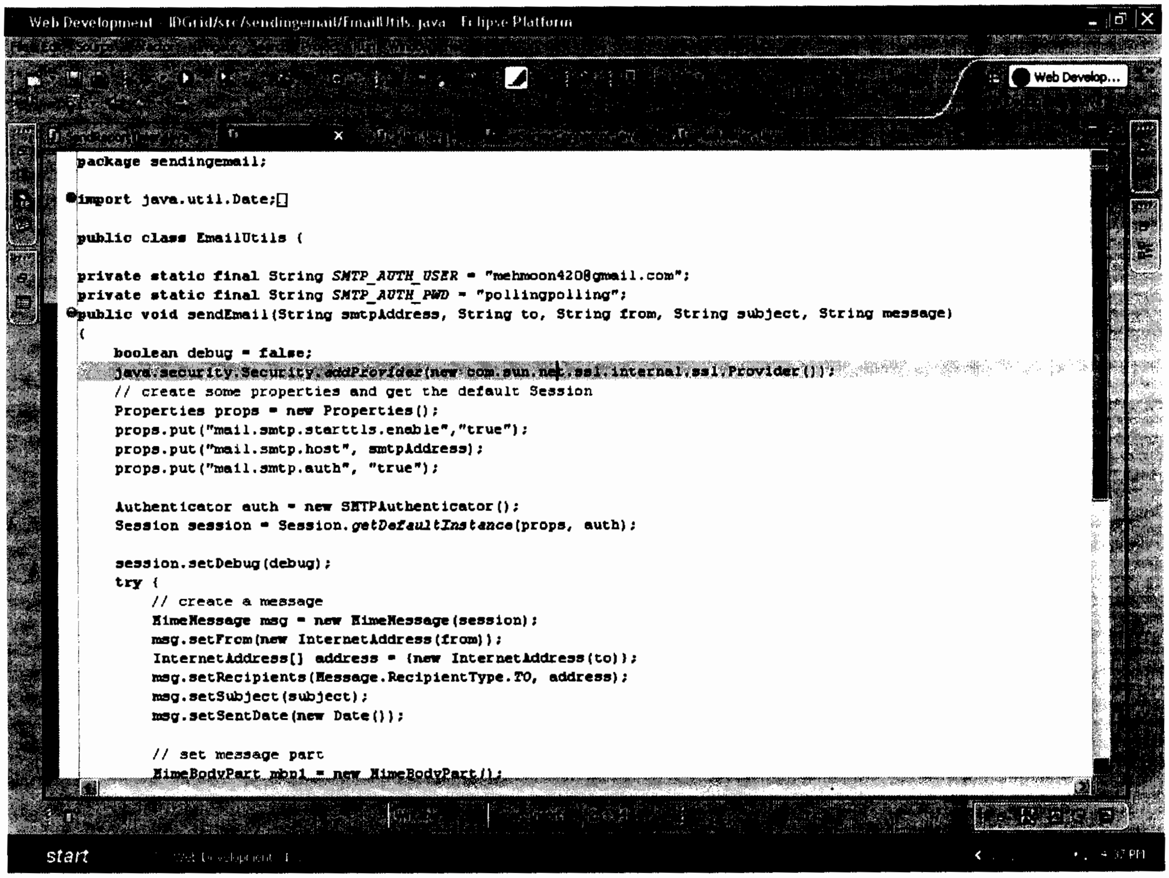

Web Development - IDGrid/src/Parser/FileReader.java - Eclipse Platform
Web Develop...
Debug
Java
start

```
ackage Parser;

mport java.io.BufferedInputStream;

/**
 *
 * @author Mehmoon anwar
 */
ublic class FileReader {

    public void generatReport() {

        FileInputStream fis = null;
        BufferedInputStream bis = null;
        DataInputStream dis = null;
        EmailUtils emailUtils = new EmailUtils();

        String query;
        SqlServerConnector sql; //object initialization for connecting to database
        PreparedStatement stmt; //object initialization for preparing statement to be stored in the object of Res
        ResultSet rs = null; //object initialization for storing a row of a database table and then accessing its
        File file = null;  //object initialization for getting audit data file from filepath
        String filePath = ""; //stores the address of file to be fetched as audit data

        File commandFile = null;
        String value = "";
        sql = new SqlServerConnector();
        String newMessage = "";
        try {
            query = "select * from Knowledge";
            stmt = sql.con
                    .prepareStatement(query, ResultSet.TYPE_SCROLL_SENSITIVE,
```

Figure 4.6: fileReader class

4.4.4 'EmailUtils' Class

Basically 'EmailUtils' has only one method 'sendEmail' and one composed class 'SMTPAuthenticator' which extends 'javax.mail.Authenticator' class. 'sendEmail' methods sends report to the domain administrator in the form of an email. 'EmailUtils' class's Screen shot is shown below.

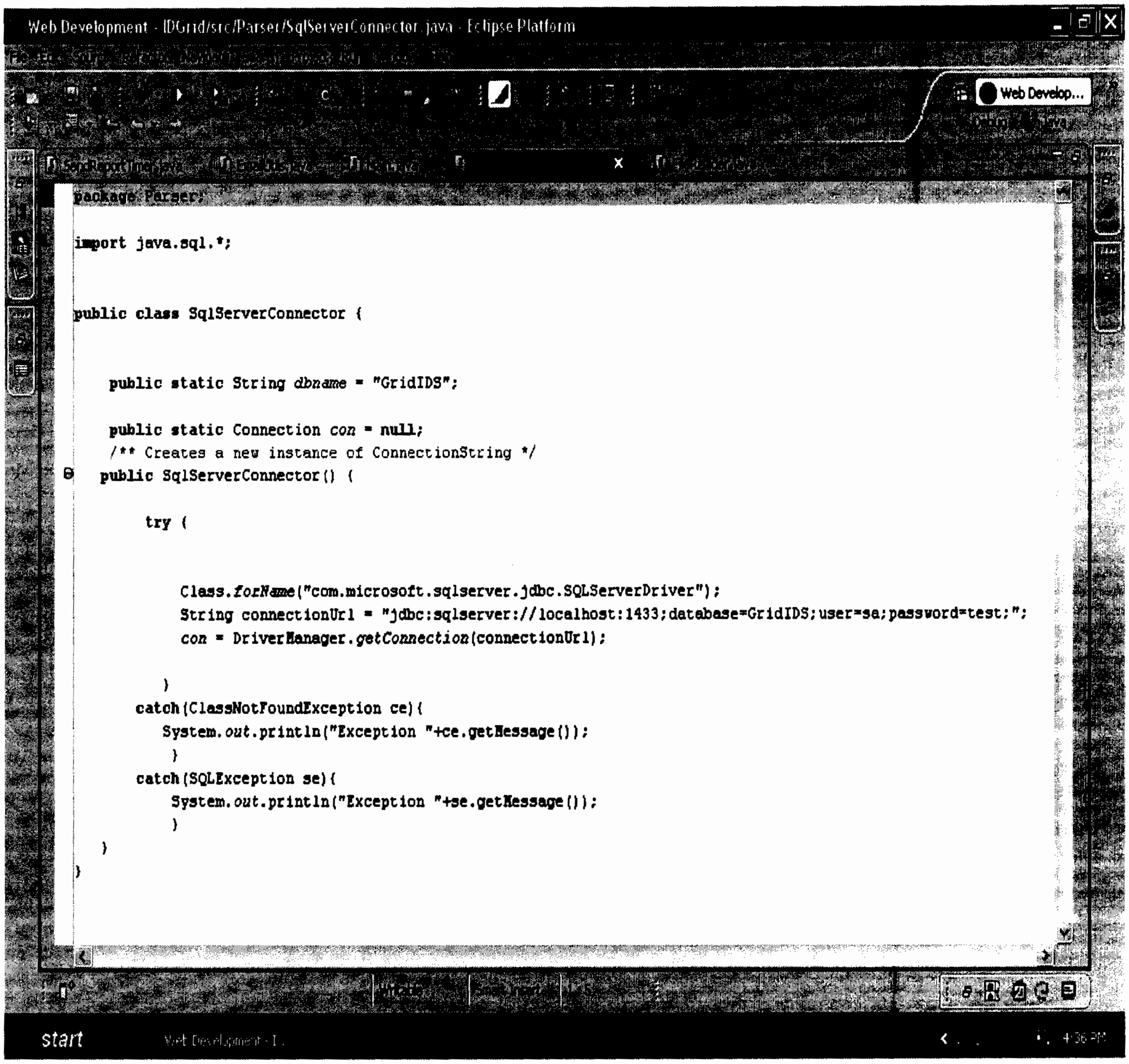

Web Development - IDGrid/src/sendingemail/EmailUtils.java - Eclipse Platform
Web Develop...
start

```
package sendingemail;

import java.util.Date;□

public class EmailUtils {

private static final String SMTP_AUTH_USER = "mehmoon420@gmail.com";
private static final String SMTP_AUTH_PWD = "pollingpolling";
public void sendEmail(String smtpAddress, String to, String from, String subject, String message)
{
    boolean debug = false;
    java.security.Security.addProvider(new com.sun.net.ssl.internal.ssl.Provider());
    // create some properties and get the default Session
    Properties props = new Properties();
    props.put("mail.smtp.starttls.enable","true");
    props.put("mail.smtp.host", smtpAddress);
    props.put("mail.smtp.auth", "true");

    Authenticator auth = new SMTPAuthenticator();
    Session session = Session.getDefaultInstance(props, auth);

    session.setDebug(debug);
    try {
        // create a message
        MimeMessage msg = new MimeMessage(session);
        msg.setFrom(new InternetAddress(from));
        InternetAddress[] address = {new InternetAddress(to)};
        msg.setRecipients(Message.RecipientType.TO, address);
        msg.setSubject(subject);
        msg.setSentDate(new Date());

        // set message part
        MimeBodyPart mbp1 = new MimeBodyPart();
```

Figure 4.7: EmailUtils class

4.4.5 'SqlServerConnector' Class

This class is used for connecting with the database. This class has a method 'SqlServerConnector( )' which is used by 'FileReader class for connecting with the databases. Screen shot of this class is shown on the next page.


Web Development - IDGrid/src/Parser/SqlServerConnector.java - Eclipse Platform
Web Develop...
start

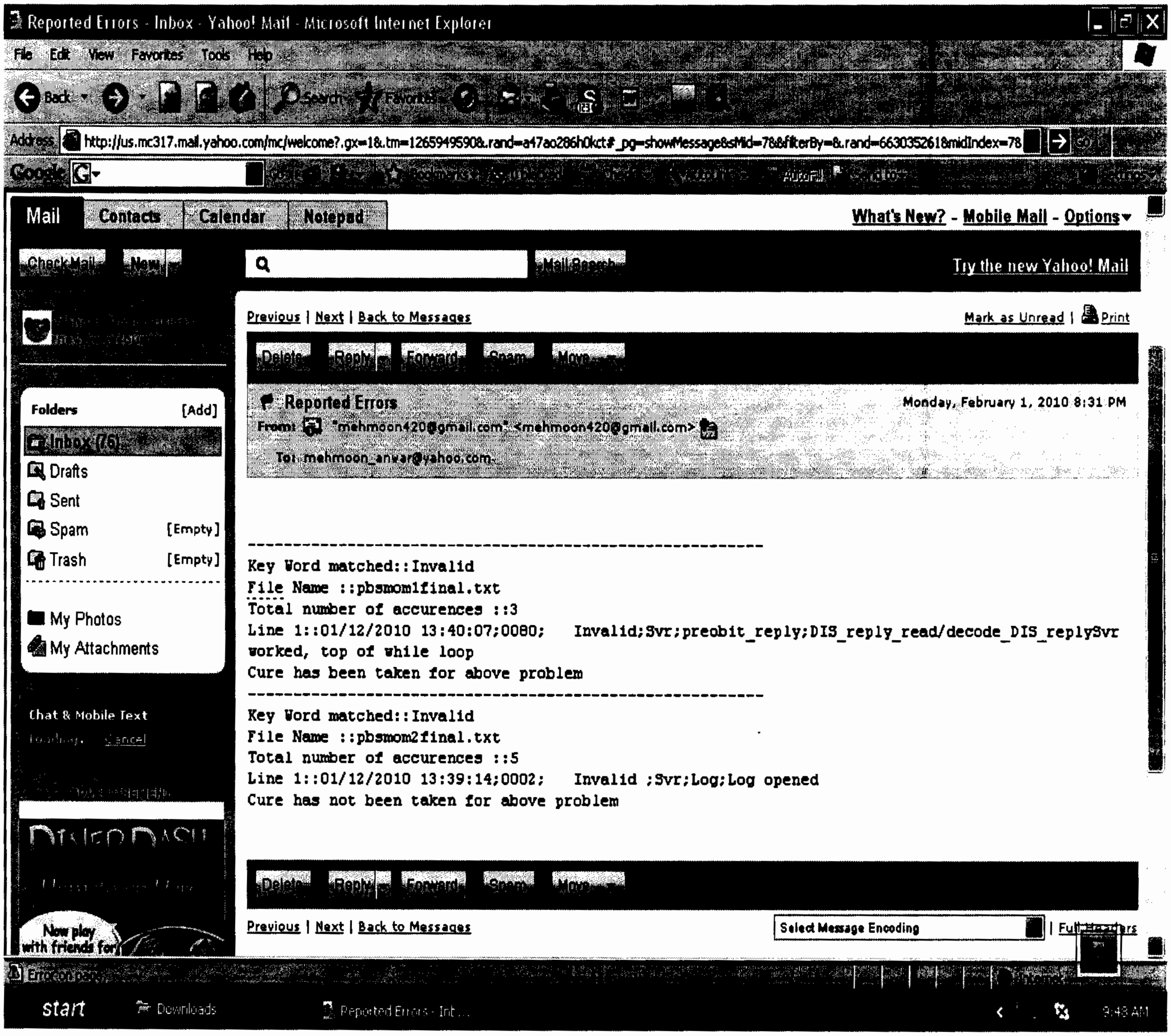
```
package Parser;

import java.sql.*;


public class SqlServerConnector {


    public static String dbname = "GridIDS";

    public static Connection con = null;
    /** Creates a new instance of ConnectionString */
   public SqlServerConnector() {

        try {


            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            String connectionUrl = "jdbc:sqlserver://localhost:1433;database=GridIDS;user=sa;password=test;";
            con = DriverManager.getConnection(connectionUrl);

          }
       catch(ClassNotFoundException ce){
          System.out.println("Exception "+ce.getMessage());
           }
       catch(SQLException se){
           System.out.println("Exception "+se.getMessage());
           }
   }
}
```

Figure 4.8: SqlServerConnector Class

4.5. Results:

4.5.1. Output of GIED Service:

Screen Shot of an output of our GIED service is shown below. This output is a report which is used to send to a domain administrator in the form of an e-mail. This report is used to send to a domain administrator only when Grid specific intrusions are detected.


Reported Errors - Inbox - Yahoo! Mail - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://us.mc317.mail.yahoo.com/mc/welcome?.gx=1&.tm=1265949590&.rand=a47ao286h0kct#_pg=showMessage&sMid=78&fiterBy=&.rand=663035261&midIndex=78
Google
Mail Contacts Calendar Notepad
What's New? - Mobile Mail - Options
Check Mail New
Mail Search
Try the new Yahoo! Mail
Previous | Next | Back to Messages
Mark as Unread | Print
Delete Reply Forward Spam Move...
Folders [Add]
Inbox (76)
Drafts
Sent
Spam [Empty]
Trash [Empty]
My Photos
My Attachments
Chat & Mobile Text
Cancel
Now play with friends for
Select Message Encoding
Full Headers
start
Downloads
Reported Errors - Int...

Reported Errors

Monday, February 1, 2010 8:31 PM

From: "mehmoon420@gmail.com" <mehmoon420@gmail.com>

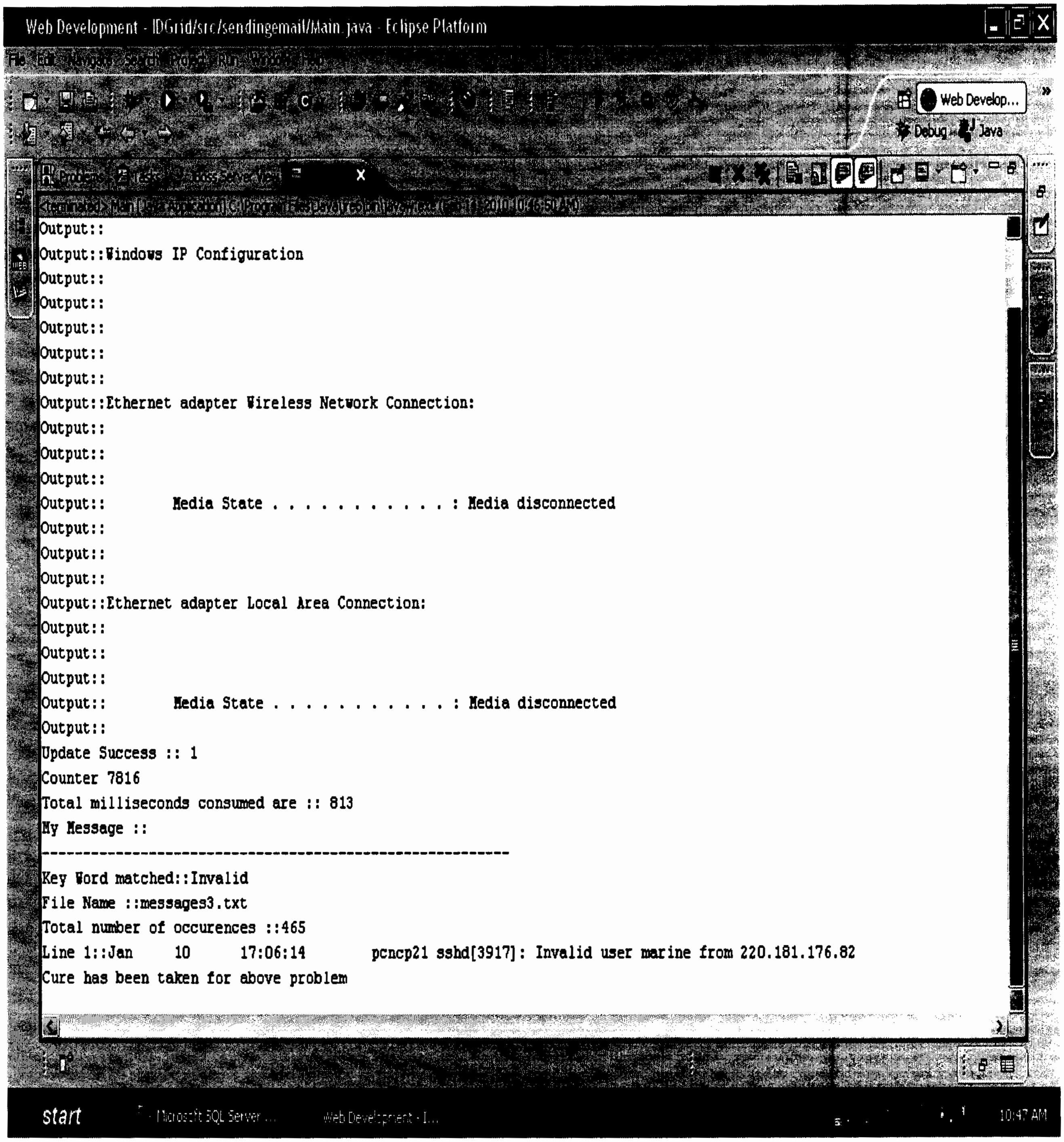
To: mehmoon_anwar@yahoo.com

```
-----------------------------------------------------------
Key Word matched::Invalid
File Name ::pbsmom1final.txt
Total number of accurences ::3
Line 1::01/12/2010 13:40:07;0080;   Invalid;Svr;preobit_reply;DIS_reply_read/decode_DIS_replySvr
worked, top of while loop
Cure has been taken for above problem
-----------------------------------------------------------
Key Word matched::Invalid
File Name ::pbsmom2final.txt
Total number of accurences ::5
Line 1::01/12/2010 13:39:14;0002;   Invalid ;Svr;Log;Log opened
Cure has not been taken for above problem
```

Figure 4.9: output of GIED service

4.5.2. Cure Taken by GIED Service:

Output of cure taken by GIED service is shown below in the figure.



```
Output::
Output::Windows IP Configuration
Output::
Output::
Output::
Output::
Output::
Output::Ethernet adapter Wireless Network Connection:
Output::
Output::
Output::
Output::        Media State . . . . . . . . . . . : Media disconnected
Output::
Output::
Output::
Output::Ethernet adapter Local Area Connection:
Output::
Output::
Output::
Output::        Media State . . . . . . . . . . . : Media disconnected
Output::
Update Success :: 1
Counter 7816
Total milliseconds consumed are :: 813
My Message ::
---------------------------------------------------------
Key Word matched::Invalid
File Name ::messages3.txt
Total number of occurences ::465
Line 1::Jan    10     17:06:14      pcncp21 sshd[3917]: Invalid user marine from 220.181.176.82
Cure has been taken for above problem
```

Figure 4.10: cure taken by GIED service

4.6. Performance Evaluation:

Performance of GIED service in terms of time (in milliseconds) for different kind of logs is shown below.

4.6.1. Maui Log:

Performance of GIED service in terms of time for Maui Log is shown below.

No. of Lines	Approximate Time in Milliseconds	Intrusion or error Detected and cure taken
20	375	No
60	390	No
140	406	No
240	457	No
360	604	Yes

Table 4.1: maui log

Graphical Representations of above readings are shown below.

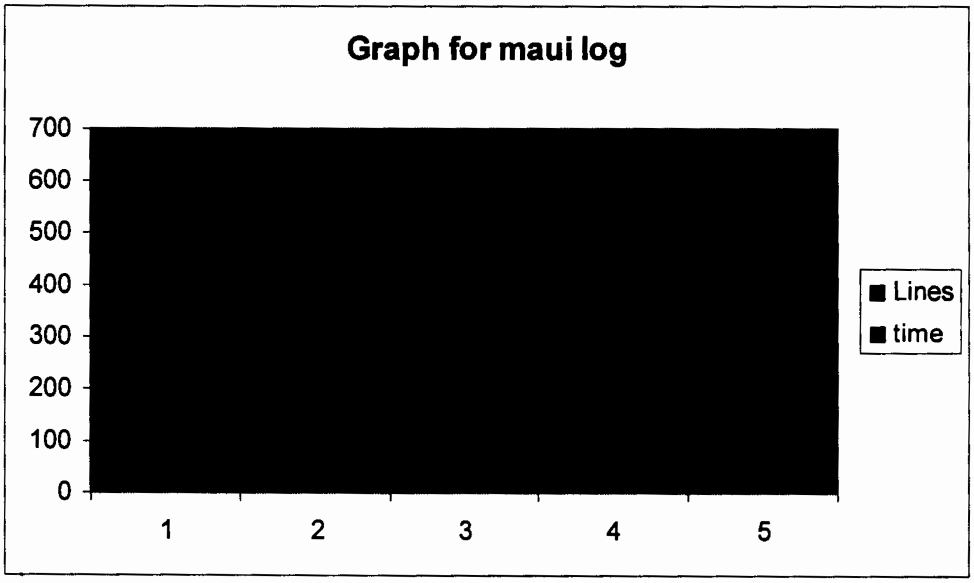


Figure 4.11: Graph for maui log

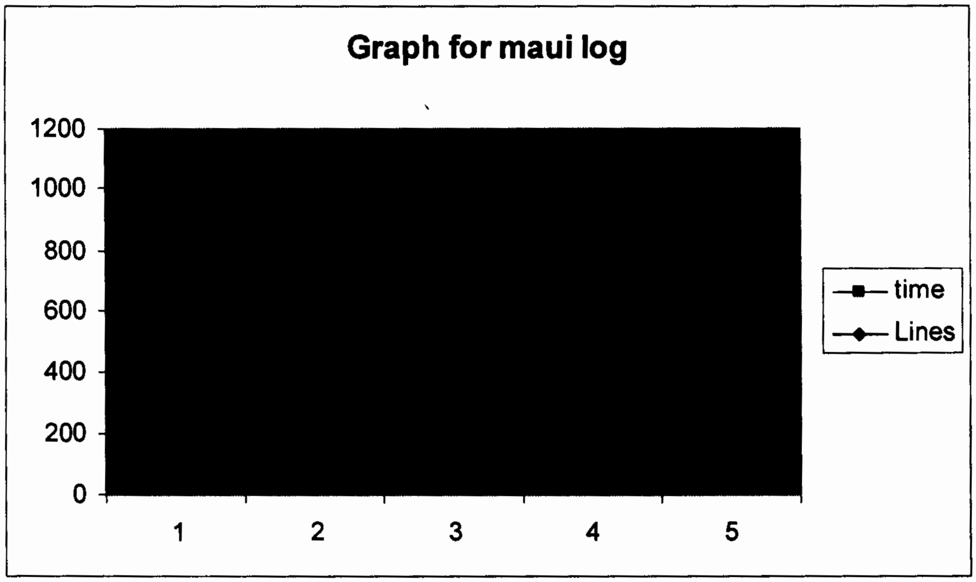


Figure 4.12: Graph for maui log

Above results show that large variation in time occurs when an intrusion or error is detected and cure is taken against it.

4.6.2. Pbs_mom Log:

Performance of GIED service in terms of time for Pbs_mom Log is shown below.

No. of lines	Approximate Time in Milliseconds	Intrusion or error Detected and cure taken
20	375	No
60	390	No
140	422	No
240	437	No
360	604	Yes

Table 4.2: pbs_mom log

Graphical Representations of above readings are shown on the next page.

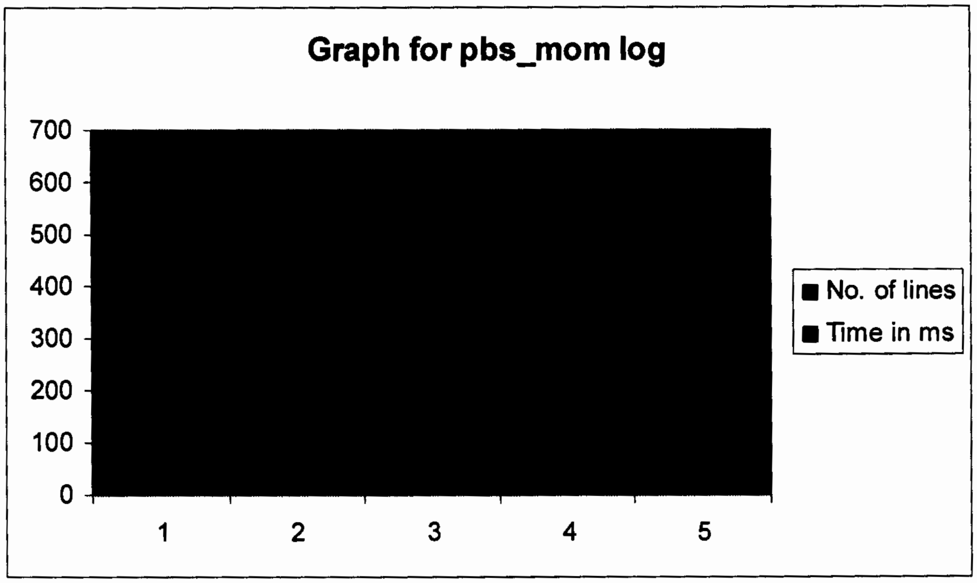


Figure 4.13: Graph for pbs_mom log

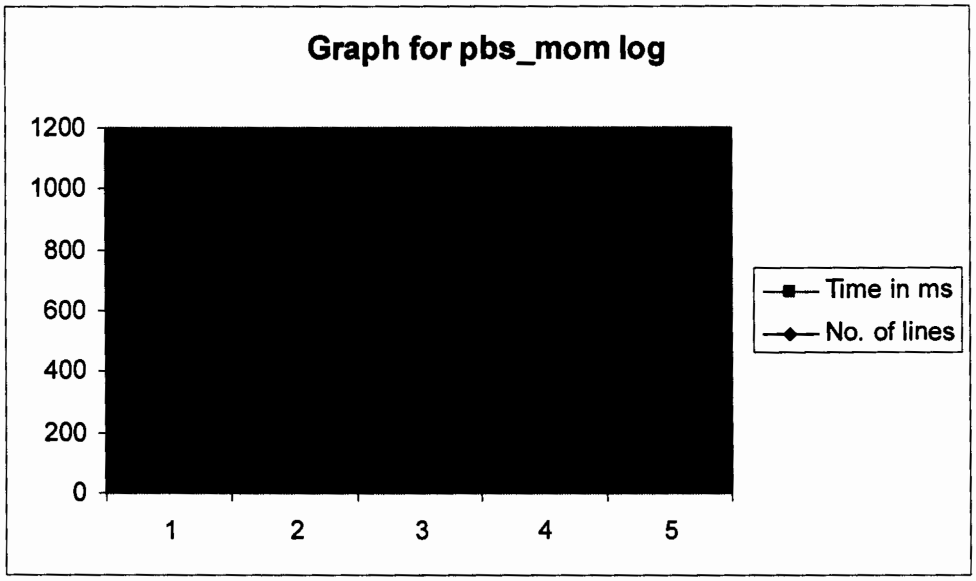


Figure 4.14: Graph for pbs_mom log

Above results show that when no intrusions or error is detected and no cure is taken against them then there will be no large variation in time as in case of maui log.

4.6.3. System_Var Log:

Performance of GIED service in terms of time for System_Var Log is shown below.

No. of lines	Approximate Time in Milliseconds	Intrusion or error Detected and cure taken
13	375	No
43	391	No
123	437	No
223	453	No
326	609	Yes

Table 4.3: system_var log

Graphical Representations of above readings are shown below.

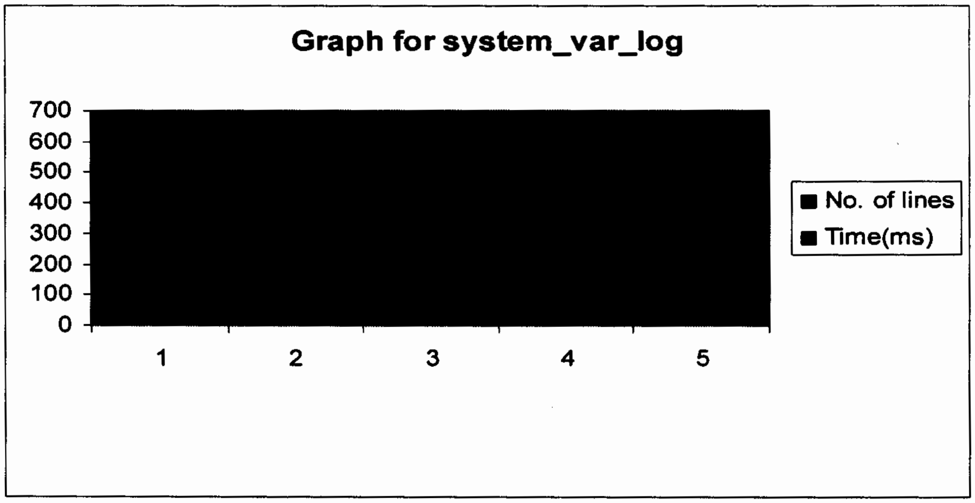


Figure 4.15: Graph for system_var log

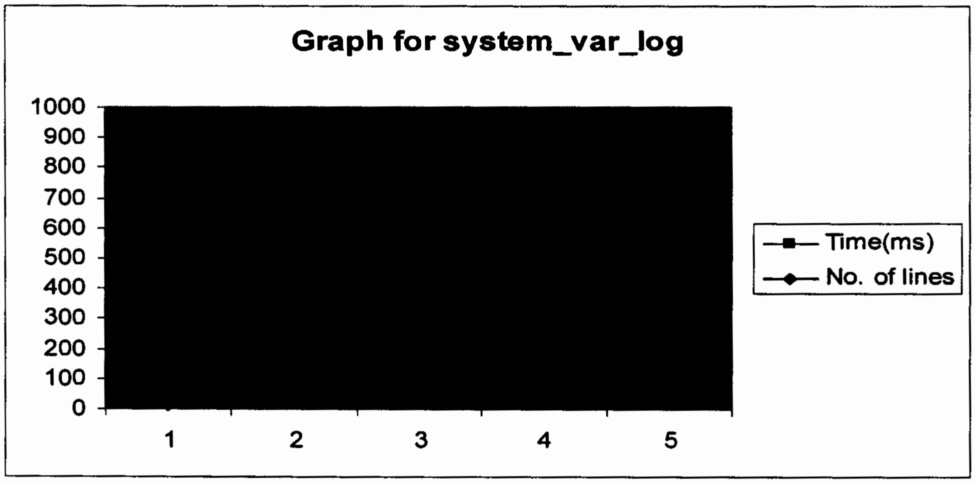


Figure 4.16: Graph for system_var log

Just like maui log the large variation in time in case of system var log is because of the reason that intrusion or error is detected and cure is taken against the intrusion.

4.6.4. Pbs_server Log:

Performance of GIED service in terms of time for Pbs_server Log is shown below.

No. of lines	Average or Approximate Time in Milliseconds	Intrusion or error Detected and cure taken
20	375	No
60	390	No
140	406	No
240	422	No
360	682	Yes

Table 4.4: pbs_server log

Graphical Representations of above readings are shown below.

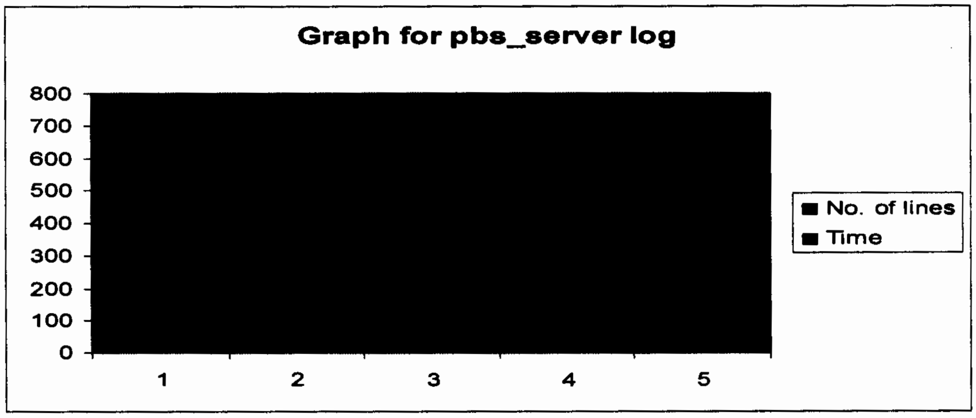


Figure 4.17: Graph for pbs_server log

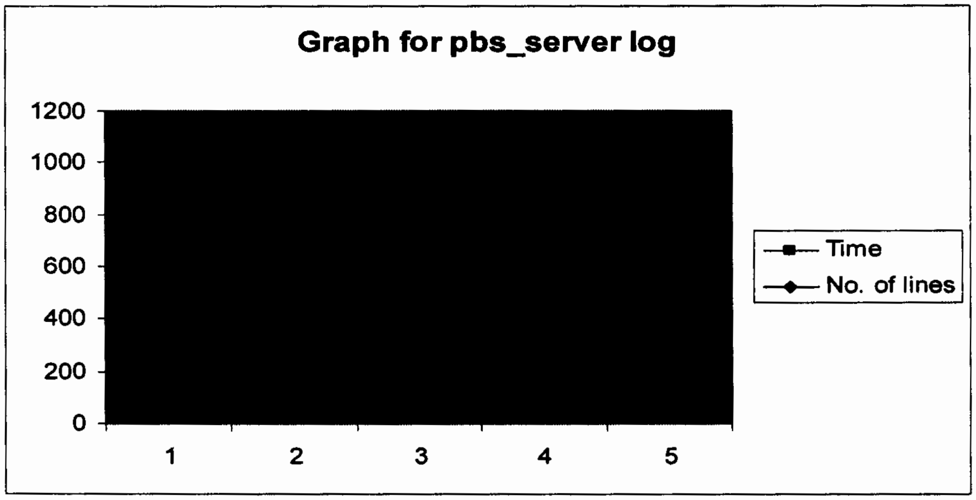


Figure 4.18: Graph for pbs_server log

Just like Pbs mom log the variation in time of Pbs server log is not so high because no intrusion or error is detected and no cure is taken.

Above results show that performance of our GIED service in terms of time is dependent on many factors which include.

1) Number of Characters in a line of a log.
2) Number of lines of a log
3) Processor Speed
4) Cure taken Against Intrusions.
5) Intrusions Detected.

4.6.5. False Negative

Name of log file	Attack or Error Pattern	Actual Number of occurrences of Pattern	Number of occurenes of Pattern Detected by GIED service
Maui	Cannot get node info	6	6
Pbs_mom	15001	4	4
Pbs_server	15056	34	34
System_var_log	Invalid	48	48

Table 4.5: False Negative

Above results show that False Negative rate of our GIED service is zero. It means that it never considers an intrusion or error as a normal activity.

4.6.6. False Positive

Name of log file	Attack or Error Pattern	Actual Number of occurrences of Pattern	Number of occurenes of Pattern Detected by GIED service
Maui	Cannot get node info	0	0
Pbs_mom	15001	0	0
Pbs_server	15056	0	0
System_var_log	Invalid	0	0

Table 4.6: False Positive

Similarly results in above tables show that False positive rate of our GIED service is zero i.e. our GIED service does not considers a normal activity as an intrusion or an error.

Chapter # 5

Conclusion

5.1. Conclusion

Limitations in current IDSs are that they could only detect Intrusions but they are not comprehensive in terms of providing cure against detected intrusions. Moreover current IDSs could only detect intrusions but they could not detect different 'System Errors' like failure of some service or application at a node etc. The impact of these 'System level Errors' on systems is just like intrusions. So there was a need to develop an IDS which could not only detect intrusions but can also detect system level errors. Moreover there was a need to develop an intrusion detection system which could not only detect but can also provide cure for detected intrusions and system level errors.

We have provided a model and prototype of IDS which is flexible in a way because it will detect both 'Intrusions' and 'System level Errors'. Moreover proposed IDS will also provide cure against detected 'intrusions' and 'System Level Errors'.

5.2. Future Work:

Future Includes

1) Making this IDS model more comprehensive by also introducing User Behavior Deviation Approach for Intrusion detection.

Appendix-A
References

References:

1) Syed Naqvi, Michel Riguidel "Evaluation of grid security solutions using common criteria," Computing in High Energy Physics and Nuclear Physics, Interlaken, Switzerland, pp.854, 27 Sep - 1 Oct 2004.
2) LHC Computing Grid http://lcg.web.cern.ch/
3) www.entropia.com
4) SETI (Search for Extraterrestrial Intelligence) http://setiathome.berkeley.edu/index.php
5) Yuri Demchenko "White collar Attacks on Web Services and Grid Security threats analysis and Grid Security Incident data model definition," Draft version 0.2, August 12, 2004
6) Sajjad Asghar "A survey of grid computing models for self healing," SZABIST, 9th National Research Conference, 2009.
7) Grid Today http://www.gridtoday.com/
8) M. Gastpar and M. Vetterli, "On the capacity of large Gaussian relay networks," IEEE Trans. Inf. Theory, vol. 51, no. 3, pp. 765–779, March 2005.
9) Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang "A Performance-Based Grid Intrusion Detection System," Computer Software and Applications Conference (COMPSAC), Vol. 2, Page(s): 525 – 530, July 2005.
10) Sanjeev Rana, Rajneesh Gujral and Manpreet Singh "Securing Grid Using Intrusion Detection System," Proceedings of National Conference on Challenges & Opportunities in Information Technology (COIT-2007) RIMT-IET, Mandi Gobindgarh. March 23, 2007.
11) Andrea Bosin, Nicoletta Dessì, Barbara Pes and Dipartimento di Matematica e Informatica "A Service Based Approach to a New Generation of Intrusion Detection Systems," Sixth European Conference on Web Services (ecows), pp.215-224, 2008

12) Fang-Yie Leu, Ming-Chang Li, Jia-Chun Lin "Intrusion Detection based on Grid," International Multi-Conference on Computing in the Global Information Technology (ICCGI), Volume, Page(s):62 – 62, Aug. 2006

13) Fang-Yie Leu, Jia-Chun Lin, Ming-Chang Li, Chao-Tung Yang and Po-Chi Shih "Integrating Grid with Intrusion Detection," 19th International Conference on Advanced Information Networking and Applications (AINA), Volume 1, Page(s): 304 – 309, March 2005.

14) Alexandre Schulter, Júlio Albuquerque Reis, Fernando Koch and Carlos Becker Westphall "A Grid-based Intrusion Detection System," Proceedings of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies (ICNICONSMCL'06), Volume, Page(s): 187 – 18, April 2006.

15) PEI-YOU ZHU, JI GAO1, BO-OU JIANG and HUI SONG "A NEW FLEXIBLE MULTI-AGENT APPROACH TO INTRUSION DETECTION FOR GRID," International Conference on Machine Learning and Cybernetics, page(s): 7-12, Aug 2006.

16) Kleber Vieira, Alexandre Schulter, Carlos Westphall and Carla Westphall "Intrusion Detection Techniques in Grid and Cloud Computing Environment," IEEE computer Society Digital Library IT Professional, 26 Aug. 2009. http://doi.ieeecomputersociety. org/10.1109/MITP.2009.89

17) Desheng Fu, Shu Zhou, Ping Guo "The Design and Implementation of a Distributed Network Intrusion Detection System Based on Data Mining," World Congress on Software Engineering (wcse), vol. 3, pp.446-450, 2009.

18) Yu-Fang Zhang, Zhong-Yang Xiong, Xiu-Qiong Wang "Distributed Intrusion Detection System Based on Clustering," Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005

19) Steven R. Snupp, James Brentano, Gihan V. Dias, Terrance L. Goan, Tim Grance, L. Todd Heberlein, Che-Lin Ho, Karl N. Levitt, Biswanath Mukherjee, Douglass L. Mansur, Kenneth L. Pon Stephen E. Smahd "A System for

Distributed Intrusion Detection," In COMPCOM Spring '91 Digest of Papers, pages 170-176, February/March 1991.

20) Sattarova Feruza Yusufovna "Integrating Intrusion Detection System and Data Mining," International Symposium on Ubiquitous Multimedia Computing (umc), pp.256-259, 2008

21) Yi Hu, Brajendra Panda "A Data Mining Approach for Database Intrusion Detection," Symposium on Applied Computing Proceedings of the ACM, Pages: 711 – 716, 2004