# A Study of a Network Intrusion Detection System Based on Java

**Ramesh Kumar[1] ,Savya Sachi[2]**

[1]Assistant Professor, Department of Computer Science & Engineering
B. P. Mandal College of Engineering, Madhepura, Bihar, India
Email id-ramscsc132@gmail.com
[2]Assistant Professor, Department of Information Technology
Muzaffarpur Institute of Technology, Muzaffarpur, Bihar, India

**Abstract**- As new technology is released, the number of hacking and infiltration occurrences rises rapidly every year. Sadly, there is nowhere to hide in the technologically connected world of today. There are numerous techniques to find you, including DNS, NSlookup, Newsgroups, website crawling, email properties, etc. As part of this research project, we created and constructed an Intrusion Detection System (IDS) that uses pre-established methods to recognize network threats. The system is developed in Java, and in order to grant access to the winpcap, JPCap is required. The network's packets are recorded online, or as soon as they arrive at the network interface. The IDS is intended to offer the fundamental detection methods in order in order to safeguard the systems found in networks that are either directly or indirectly linked to the Internet.

***Keywords***— *Intrusion Detection System (IDS), Jpcap Library, Network Security, DNS, Nslookup, Newsgroups, Website Crawling, Email Properties.*

## INTRODUCTION

The security of networked computers and the Internet has grown in importance due to their widespread use. The swift progress in network technologies has resulted in increased capacity and seamless connectivity for wireless and mobile devices. Anderson (1980) suggested using audit trails to keep an eye on potential dangers. At the time, the value of such data was not recognized, and all system security measures that were in place were intended to prevent unauthorized parties from accessing sensitive data. In the latter case, Dorothy put out the idea of intrusion detection as a fix for the issue of giving computer systems a feeling of security. The system, type of intrusion, and application environment are all unaffected by this intrusion detection model. An intrusion is somebody (“ hacker" or "cracker") attempting to break into or misuse your system. The word "misuse" is broad, and can reflect something severe as stealing confidential data to something minor such as misusing your email system for spam (though for many of us, that is a major issue!). With the emergence of Internet and the World Wide Web, the concept of Global village has taken its inception. There are facilities to virtually achieve any kind of information on the internet. All these advantages have been achieved because of networking computers and associated devices. There has been a rapid progress in this field. Along with this, there is the arms race between the intruders and people who provide security to the systems in networks. This project IDS (detection and protection) runs on the host machines and assists the network Administrators to detect several intrusion attacks and inform to the owner of the system and also provide security by blocking the malicious users based on their IP addresses. The number of hacking and intrusion incidents is increasing alarmingly each year as new technology rolls out. Unfortunately in today’ s digitally connected world, there is no place to hide. DNS, NSlookup, Newsgroups, web site trawling, e-mail properties etc. are just some of the many ways in which you can be found. In this research project, we designed and build an Intrusion Detection System (IDS) that implements pre-defined algorithms for identifying the attacks over a network. The Java programming language is used to develop the system, JPCap must be used to provide access to the winpcap. The packets in the network are captured online i.e., as they come on the interface of the network. The IDS is designed to provide the basic detection techniques so as to secure the systems present in the networks that are directly or indirectly connected to the Internet.

## SYSTEM STRUCTURE

The Architectural Design is depicted as a block diagram where each box in the diagram represents a sub-system. The arrows indicate data or control flow in the direction specified by them. The Architectural block diagram presents an overview of the system architecture (Figure 1).

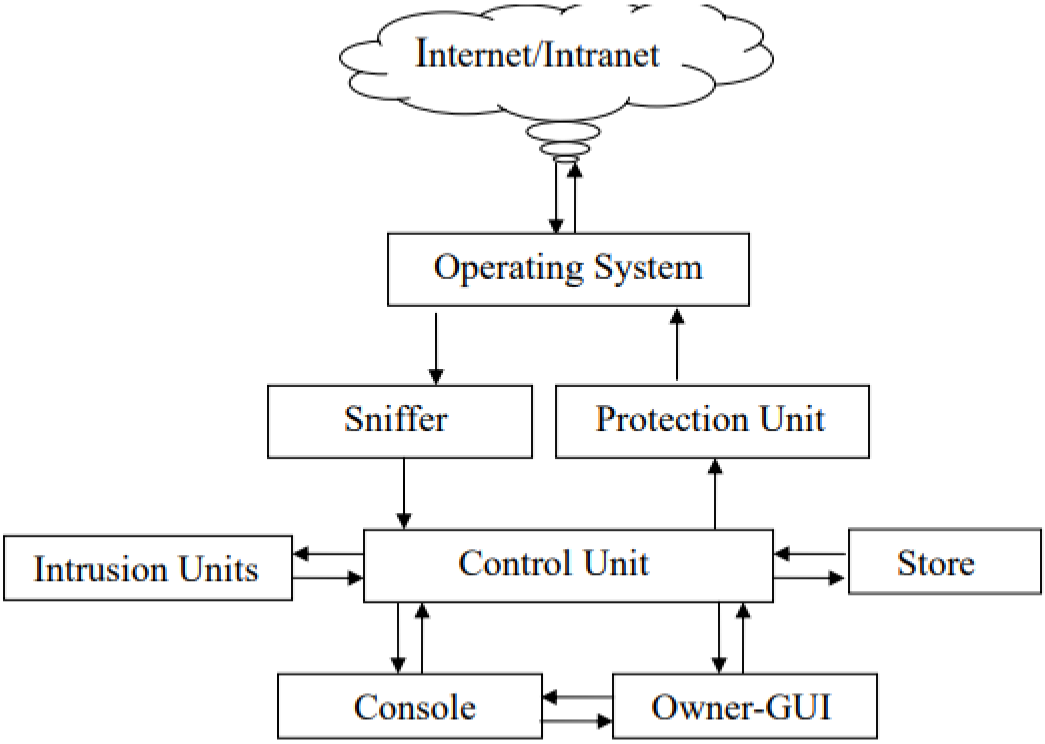

**Figure 1- Architectural diagram**

The above diagram is the structural model of architecture for the present system. In this system, the Sniffer sub-system captures the packets which flow into and out of the system. This sub-system then formats these packets in a format that is convenient for further processing. Jpcap is a part of this Sniffer sub-system .The packet capturing function is accomplished via Jpcap. It provides a Java API to the popular C packet capture library called pcap. While Jpcap is not a complete conversion of the popular C pcap library yet, it does provide the basic functionality we need. There are various Intrusion units each for a specific attack. So, there are individual intrusion units which detect Port Scanning, Smurf Attack, syn flood Attack, Efficient Mapping and Cerebral Mapping. All these intrusion units are independent of each other and interact only with the Control Unit. They run simultaneously continuously scanning for occurrence of specific attacks and report the attacks to the Control Unit when detected. The Store sub-system stores the various Rules defined and given to it by the Control Unit. It consists of various other sub-systems for data processing. The data in the form of XML files is stored after encryption using Simplified DES algorithm. The Owner-GUI sub-system displays to the user the defined Rules, the attack logs and the running status of the Intrusion units. It also provides facilities for starting and stopping intrusion units, clearing attack logs, adding new Rule to the store and deleting existing Rule from the store. The console sub-system performs functions similar to the Owner-GUI sub-unit, but displays at the command line. The Control Unit sub-system manages the sub-systems for detection of attacks by taking the packets from the sniffer, sending relevant packets to the Intrusion Units, gives Rules to the store and retrieves them and displays necessary messages to the user through the user interface. The Protection Unit sub-system takes the Rules from the Control unit and provides security by applying the rules on local Operating System as IPSec policies.

## NEED FOR AN INTRUSION DETECTION SYSTEM

Intrusion detection devices are an integral part of any network. The internet is constantly evolving, and new vulnerabilities and exploits are found regularly. They provide an additional level of protection to detect the presence of an intruder, and help to provide accountability for the attacker’ s action.

**(i) Denial of service**- Network-based denial-of-service attacks are one of the easiest types of attacks. It often requires little effort to fully consume resources on the target computer, to starve the target computer of resources, or to cause critical services to fail or malfunction. Internal corporate networks typically do not have internal filtering defenses against common denial-of-service attacks, such as flooding.

**(ii) Threat to Confidentiality**- Some viruses attach themselves to existing files on the system they infect and they send the infected files to others. This can result in confidential information being distributed without the author’ s permission.

**(iii) Modification of contents**- Intruders might be able to modify news sites, produce bogus press releases, and conduct other activities, all of which could have economic impact.

**(iv) Masquerade**- A masquerade takes place when one entity pretends to be a different entity. Authentication sequences can be captured and replayed after a valid authentication sequence has taken place, thus enabling an authorized entity with few privileges to obtain extra privileges by impersonating an entity that has those privileges.

## DIFFERENT APPROACHES TO INTRUSION DETECTION

Many classifications exist in literature about intrusion detection. The basic types of intrusion detection are host-based and network-based. Host-based systems were the first type of intrusion detection systems to be developed and implemented. These systems collect and analyze data that originate in a computer that hosts a service, such as a Web server. Once this data is aggregated for a given computer, it can either be analyzed locally or sent to a separate/central analysis machine. Instead of monitoring the activities that take place on a particular network, network-based intrusion detection analyzes data packets that travel over the actual network. These packets are examined and sometimes compared with empirical data to verify their nature: malicious or benign. Because they are responsible for monitoring a network, rather than a single host, network-based intrusion detection systems tend to be more distributed than host-based intrusion detection system. The two types of intrusion detection systems differ significantly from each other, but complement one another well. The network architecture of host-based is agent-based, which means that a software agent resides on each of the hosts that will be governed by the system. In addition, more efficient host-based intrusion detection systems are capable of monitoring and collecting system audit trails in real time as well as on a scheduled basis, thus distributing both CPU utilization and network overhead and providing for a flexible means of security administration. Two other approaches encountered in literature concerning intrusion detection systems for detecting intrusive behavior are misuse detection and anomaly detection.

**(i) Misuse Detection**- Misuse detection relies on matching known patterns of hostile activity against databases of past attacks. They are highly effective at identifying known attacks and vulnerabilities, but rather poor at identifying new security threats. Misuse-detection based intrusion detection systems can only detect known attacks.

**(ii) Anomaly Detection**- Anomaly detection will search for something rare or unusual by applying statistical measures or artificial intelligence to compare current activity against historical knowledge. Common problems with anomaly-based systems are that, they often require extensive training data for artificial learning algorithms, and they tend to be more computationally expensive, because several metrics are often maintained, and these need to be updated against every system’ s activities. Several approaches apply artificial neural networks in the intrusion detection system that has been proposed. Anomaly detection based intrusion detection systems can detect known attacks and new attacks by using heuristic method.

**(iii) Hybrid Intrusion Detection-** The hybrid intrusion detection system is obtained by combining packet header anomaly detection and network traffic anomaly detection which are anomaly-based intrusion detection systems with the misuse-based intrusion detection system. Snort is an example of an open-source paper for hybrid intrusion detection. The hybrid intrusion detection system is said to be more powerful than the signature-based on its own because it uses the advantages of anomaly-based approach for detecting unknown attacks.

## JAVA CLASS DESCRIPTIONS

There system contains a total of 31 classes where 7 of them are inner classes.

The following are the important classes

1. identified,
2. PacketCapture,
3. IntrusionAttacks,
4. IntrusionUnit,
5. Attack,
6. Rule,
7. ProtectionUnit,
8. SimpleDES,
9. XmlData,
10. DataProcessor,
11. ControlUnit,
12. Console,
13. Owner, and
14. IDSMain.

**(i) UML Diagram showing IntrusionAttacks-**

- ids.IntrusionAttacks
- +portScanner(java.io.ObjectInputStream ois,java.ioObjectOutputStream output):void
- +smurfAttack(java.io.ObjectInputStream ois,java.io.ObjectOutputStream output): void
- +synFloodAttack(java.io.ObjectInputStream ois,java.io.ObjectOutputStream output): void
- +efficientMapper(java.io.ObjectInputStream ois,java.io.ObjectOutputStream output): void
- +cerebralMapper(java.io.ObjectInputStream ois,java.io.ObjectOutputStream output): void

**(ii) XmlData showing various tags-**

- ids.XmlData
- +XmlData(java.lang.String xmlFile)
- +writeDecryptedFile(java.lang.String source, java.lang.String dest): void
- +writeEncryptedFile(java.lang.String source, java.lang.String dest): void
- +addSingleIP(java.lang.String ip, java.lang.String status): void
- +removeSingleIP(java.lang.String IP): void
- +addRangeIP(java.lang.String ipstart, java.lang.String ipend, java.lang.String status):void

- +removeRangeIP(java.lang.String ipstart, java.lang.String ipend): void
- +addSubnetIP(java.lang.String ip, java.lang.String mask, java.lang.String status): void
- +removeSubnetIP(java.lang.String ip, java.lang.String mask): void
- +getStatus(java.lang.String ip): java.lang.String
- +getAllRules(): ids.Rule[ ]
- +openXml(): org.w3c.dom.Element
- +writeXml(): void
- +newXml(): void
- +clearTempFile(): void

## CONCLUSION

In order to safeguard the systems found in networks that are either directly or indirectly connected to the Internet, the intrusion detection system (IDS) is made to offer the fundamental detection approaches. Whether one succeeds or fails in achieving the goal is always correlated with carrying out such a responsibility. It functions, at least. Ultimately, however, it is the Network Administrator's responsibility to ensure that his network is secure. While intrusion detection systems (IDS) cannot totally protect a network from outside threats, they can assist network administrators in locating online criminals who aim to compromise their system and expose it to intrusions. The actions that should be taken both while utilizing the program and after an attack has been identified by IDS are listed below, and this is just the beginning. The current system offers features for intrusion protection in contrast to previous traditional intrusion detection systems. This enables the application of pertinent rules on the Operating system to restrict or allow specific IPs, IP ranges, or subnet IPs. Because of its design, it is very easy to reuse the IDS system. A platform is set up in such a way that it is possible to identify some known assaults. It will be simple to add more attacks to the system because of the extreme flexibility and extensibility provided by the system design.

## REFERENCES

[1] J. P. Anderson, “ Computer security threat monitoring and surveillance,” Fort Washington, Pennsylvania, James P Anderson Co, Tech. Rep., 1980.

[2] D. Denning, “ An intrusion-detection model,” IEEE Transaction on Software Engineering, vol. 13, no. 2, pp. 222– 232, 1997. [3] R. G. Bace, Intrusion Detection. Technical Publising, 1995.

[4] B. Mukherjee et al., “ Network intrusion detection,” IEEE Network, vol. 8, no. 3, pp. 26– 41, 1994.

[5] K. Ramamohanarao et al., “ The curse of ease of access to the internet,” 3rd International Conference on Information Systems Security.

[6] N. Bashah et al., World Academy of Science, Engineering and Technol ogy. World Academy of Science, 2005.

[7] K. K. Gupta, “ Robust and efficient intrusion detection systems,” Ph.D. dissertation, The University of Melbourne, Department of Computer Science and Software Engineering, January 2009.

[8] N. J. Puketza et al., “ A methodology for testing intrusion detection systems,” IEEE Transaction on Software Engineering, vol. 22, no. 10, pp. 719– 729, 1996.

[9] M. A. Aydin et al., “ A hybrid intrusion detection system design for computer network security,” Computer and Electrical Engineering, vol. 35, pp. 517– 526, 2009.

[10] William Stallings, “ Cryptography and Network Security” , Principles and Practices, Third Edition.

[11] D. E. Denning, "An intrusion-detection model". IEEE Transactions on Software Engineering, Vol. SE-13(No. 2):222-232, Feb. 1987.

[12] Stephen Northcutt, Judy Novak, “ Network Intrusion Detection” , Third Edition, Pearson Education 2003.

[13] N. Debar et al., " A neural network component for an intrusion detection systems," in IEEE symposium on security and privacy, 1992, pp. 240– 250.
[14] L. M. Gassata, " The artificial immune model for network intrusion detection," in First international workshop on the recent advances in intrusion detection, 1998.
[15] J. Kim and P. Bentley, " The artificial immune model for network intru sion detection," in Seventh European congress on intelligent techniques and soft computing (EUFIT99), 1999.
[16] M. Sobirey. (2011, Jan.) Intrusion detection systems.
[17] M. Roesch, " Snort lightweight intrusion detection for networks."
[18] R. Russel, Snort intrusion detection 2.0. Rockland, MA: Syngress Publishing, Inc, 2003.
[19] D. Burgermeister and J. Krier. (2010, Dec.) Systeme de d ` etection d' intrusion.
[20] K. Fujii. (2007, Jan.) Jpcap tutorial.
[21] C. Thibaud, MySQL 5: installation, mise en oeuvre, administration et programmation. Edition Eyrolles, 2006.
[22] N. Cheswick and S. Bellovin, Firewalls and Internet Security: Repelling the Willy Hacker. Pearson Education Inc., 2003.
[23] P. G. Neumann and D. Parker, " A summary of computer misuse techniques," in 12th National Computer Security Conference, Baltimore, MD, 1989, pp. 396– 407.
[24] E. C. Ezin, " Implementation in java of a cryptosystem using a dynamic huffman coding and encryption methods," International Journal of Computer Science and Information Security, vol. 9, no. 3, pp. 154– 159, 2011.
[25] Intrusion detection system combining misuse detection and anomaly detection using Genetic Network Programming | IEEE Conference Publication | IEEE Xplore 2009s
[26] A survey on Intrusion Detection System (IDS) and Internal Intrusion Detection and protection system (IIDPS) | IEEE Conference Publication | IEEE Xplore 2017
[27] A Multi-Agent Model for Network Intrusion Detection | IEEE Conference Publication | IEEE Xplore 2019
[28] Studying the Fuzzy clustering algorithm for intrusion detection on the attacks to the Domain Name Sy stem | IEEE Conference Publication | IEEE Xplore 2021
[29] An intrusion detection system based on system call | IEEE Conference Publication | IEEE Xplore 2006
[30] MANET security: An intrusion detection system based on the combination of Negative Selection and danger theory concepts | IEEE Conference Publication | IEEE Xplore 2014
[31] Design of a New Intrusion Detection System Based on Database | IEEE Conference Publication | IEEE Xplore 2009
[32] Multi-layer Intrusion Detection and Defence Mechanisms Based on Immunity | IEEE Conference Publication | IEEE Xplore 2008