ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Development of an Enhanced Ripple Down System for Detecting Cybercrimes in a University Environment

Adeyemo Rukayat Ayokunmi[1], Ismaila Wasiu Oladimeji[2], Afolabi Adeolu Olabode[3], *Adeyemo Isiaka Akinkunmi[4] & Ismaila Folasade Muibat[5]

[1,2,3, 4]Ladoke Akintola University of Technology, Ogbomoso)

[5]Osun State University

DOI: https://doi.org/10.51583/IJLTEMAS.2026.150700170

Received: 10 August 2024; Accepted: 15 August 2024; Published: 26 August 2026

ABSTRACT

University environments have become increasingly vulnerable to cybercrimes, including identity theft, hacking, financial fraud, and data breaches. Existing detection systems often exhibit high false positive rates, causing alert fatigue and missed detections, alongside prolonged recognition times that delay response to critical incidents. These limitations undermine the effectiveness of cybersecurity measures in academic institutions. This research aims to develop and evaluate an Enhanced Ripple Down Rule (ERDR) system for detecting cybercrimes in a university environment, addressing the limitations of standard rule-based approaches.

A dataset comprising 3,800 cybercrime records, covering seven crime types (labelled Crime A through G) with four attributes each, was used. The dataset was partitioned into training (2,800 records) and testing (1,000 records) sets. The ERDR system, extending the standard Ripple Down Rule framework, employs a binary tree structure with enhanced rule processing capabilities. The system was implemented using MATLAB (R2023a) on a Windows 10 64-bit platform. Performance was evaluated using Sensitivity, Specificity, False Alarm Rate (FAR), Accuracy, and Computational Time (CT) across multiple threshold values.

At the optimum threshold of 0.80, the ERDR system achieved: Sensitivity of 97.86%, Specificity of 98.92%, FAR of 1.08%, Accuracy of 99.86%, and CT of 140.9 seconds. In comparison, the standard RDR achieved: Sensitivity of 96.24%, Specificity of 95.74%, FAR of 4.26%, Accuracy of 97.85%, and CT of 131.11 seconds.

The ERDR system demonstrated superior performance across all metrics, reducing false alarms by 74.6% while improving accuracy by 2.01%. The system provides a robust, accurate, and practical solution for cybercrime detection in university environments. Its superior performance, particularly in reducing false positives, makes it suitable for deployment across higher education institutions. This research contributes to the field of cybersecurity by extending the Ripple Down Rule methodology with enhancements that significantly improve detection accuracy and operational efficiency.

INTRODUCTION

Background to the Study

Information and Communication Technology (ICT) is an enabler of change, an essential element of new developments and a major contributor to successes in daily life. The impact of ICT is extensive, affecting individuals, businesses, industries, communities and governments. It changes the way daily activities are accomplished and also impacts social and economic development globally. The spread of ICT promotes the development of information society where economic success and social development became more dependent on

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

the availability and accessibility of ICT. The trend in ICT altered fundamental methodologies of doing business and gave rise to electronic services in communication, trade, employment, education, government, and health. Recently, there is increase in social media and mobile devices usage as they facilitate the interaction between individual and their participation in decision-making processes (Odumesi, 2014).

Today, our society is increasingly relying on ICT and the internet to conduct businesses, manage industrial activities and engage in personal communication among numerous benefits. These technologies allow enormous gain in efficiency, productivity and communication. They also create a vulnerability to those who choose to take an advantage of new situations or new development in the usage of ICT (Sumanjit and Tapaswini, 2013). The rapid growth of the internet and its global acceptance is producing increasing security threats. The cyber space creates unlimited opportunities for commercial, social and educational activities. Criminals perform operation in cyber space with the help of internet. Cybercrime is defined as a crime in which a computer is the object (hacking, phishing, spamming) or is used as a tool to commit an offense (child pornography, hate crimes). Cybercrime is increasing astronomically each day as the internet continues to reach every nook and crannies of our society and none can predict accurately the next dimension. It has caused a lot of harm to the society and to individual's lives. Nigeria's loss to cyber-crimes is now about N127 billion yearly and this has become a source of challenge to the Federal Government of Nigeria (Adeyemi and Nkechi, 2016).

More Nigerians embrace ICTs lately compared to other African countries (Stev et al., 2009). Some years ago, limited number of countries had local internet access but today the story has changed positively. Internet usage is on the increase in Nigeria with most of the citizens who cannot afford the private internet facilities, depends on the public and commercial internet access points such as cyber cafes for rudimentary internet access. All these cybercafe make use of this advantage to charge individual, group or companies exorbitant fees for their services. Despite the growth in ICT adoption, fundamental problem of erratic power supply also rears its ugly head. The inadequate telecommunication infrastructures also continue to hinder the nation from uninterrupted access to innovative information technology applications such as e-government, e-commerce, telemedicine and teleconferencing (Matanmi et al., 2013).

The present state of ICT involvement has brought renewed interest in the investment and innovative use of ICT for new development. According to the internet world Statistics, Nigeria as at November 2007 accounts for the highest number of internet users in Africa but as at June 2009 available statistics showed that Egypt is on the lead followed by Nigeria (Odumesi, 2014). However, as of December 2019, Nigeria had 99.05 million of internet users in Africa which upgrade them to first position in Africa in internet usage according to Joseph Johnson, (2021).

The rate of involvement of ICT in Nigeria has brought changes drastically in socio-economic growth and development in all facets of Nigerian economy such as stock market, banking, insurance, law and services. Cyber-criminals have seen ICT as a tool that could be exploited in performing different forms of cyber-crime. Cybercriminals perform these electronic crimes such as: email scam, fake lotteries, theft, child pornography, scam text messages as well as cyber piracy; these are few of the vices that have become recurring in the cyber space. Sending fraudulent mails and other sharp practices are referred to as scamming in Nigeria. This unpleasant trend of cybercrime affects the economic development in Nigeria and has compounded the challenges facing Nigeria in order to fight against corruption and other vices (Adelabu, 2015).

This research aims at developing effective technique for detecting and deterring cyber-criminals that specifically target web applications, as well as to investigate the socio-economic impact of cyber-crime in Nigeria. Web applications are programs that run over a network such as the internet or an intranet and enable websites to access dynamic contents by making connections to databases (Garfunkel and Spafford, 2011). Web applications have been designed to provide important functions such as online shopping, social networking, online banking, e-health, web searches, web logs, web mail, interactive information and other functions. All these aforementioned advantages of web application can be affected by cyber-criminals. Web application usage has

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

recently increased. At the same time, the vulnerability of web applications has also increased. Web vulnerability is a weakness in web application architecture that creates a potential threat. This could be from incorrect configuration, poor input validation design, or unsuitable and insecure coding methods. Even though the features of web applications make them convenient to support and maintain, many of these same features also make them vulnerable to attack (Blum et al., 2004).

As Internet usage is growing globally, it also gives rise to some people to take advantages of the internet usage negatively by causing different kind of havoc in the society. This problem is growing steadily and many people have become victims of hacking, theft, scamming and malicious software. One of the best ways to avoid being a victim and protecting one's sensitive information is by making use of impenetrable security that uses a unified system of software and hardware to authenticate any information that is sent or accessed over the Internet. Today, criminals that indulge in cybercrimes are not driven by expertise alone, instead, they want to use their knowledge to gain benefits quickly. They are using their expertise to steal, deceive and exploit people as they find it an easy way to earn money without doing any stressful work.

Cybercrimes have become a real threat today and are quite different from traditional crimes such as robbing, mugging or stealing. Unlike these crimes, cyber-crimes can be committed single handedly and does not require the physical presence of the criminals. The crimes can be committed from a remote location and the criminals need not worry about the law enforcement agencies in the country where they are committing crimes. The same systems that have made it easier for people to conduct e-commerce and online transactions are now being exploited by cyber criminals (Bursh and Cheswick, 2010). Nigerians and other perpetrators of cyber-crime must know that whatever affects the users of internet affect the economy of the country, due to the influence that ICT has on every aspect of the Nigerian economy. This research has developed a system for detecting fraudulent online activities using Enhanced ripple down rule and neural network and more so, the developed system has been able to handle multiple classification based on various attributes.

Statement of the Problem

Information and Communication Technology (ICT) is an essential element of new developments and a major contributor to successes in daily life. The impact of ICT is extensive, affecting individuals, businesses, industries, communities and governments. It changes the way daily activities are accomplished and also impacts social and economic development globally. The spread of ICT promotes the development of information society where economic success and social development became more dependent on the availability and accessibility of ICT. The trend in ICT altered fundamental methodologies of doing business and gave rise to electronic services in communication, trade, employment, education, government, and health. Recently, there is increase in social media and mobile devices usage as they facilitate the interaction between individual and their participation in decision-making processes (Odumesi, 2014)

Several researches have been conducted to investigate cyber related crimes. But it all depends on area of interest. This work aimed to investigate and develop intrusion detection and prevention system. There are some challenges that face the existing intrusion detection and prevention systems. It was observed from the reviewed literature that existing techniques can only detect and prevent individual attacks and not coordinated, distributed attacks. There is need to have a system that can both detect and prevent both coordinated and distributed attacks.

Most of the detection systems used to curb cybercrimes on web application generates a large number of false alarms. These false alarms are increased on network where there is large number of users. Another problem is that the existing system can detect identified attack, but unable to handle unknown attacks. Also, low detection rate of cyber-attack increases the rate at which cybercriminals operate. Hence, this research developed a technique which is characterized by low false alarm and solved the problem of known and unknown attack in web application.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Aim and Objectives

The aim of this research is to develop an Enhanced Ripple down rule system for detecting cybercrime in Nigeria university environment. The specific objectives are to:

i. formulate a cybercrime detector model for detecting fraudulent online activities using enhanced ripple down rule and RBFNN;
ii. implement the model in (i) using Python Programming Language
iii. compare the performance of the enhanced Ripple down rule and the existing RDR.
iv. evaluate the performance of the system in (iii) using Sensitivity, Specificity, False Alarm Rate (FAR), Accuracy and Computational Time(CT).

Significance of Study

The effect of cybercrime on Nigerian socio-economy has been an active area of research due to its adverse effect on Nigerian due to hacking, theft, cyber stalking, identity theft, malicious software and abuse. There is need to investigate the effect of cyber-crime on the society and design a system capable of detecting the action of cybercrime perpetrator and their location. This research reckons with the theoretical presentation (investigation), design of a system for detecting and preventing internet fraudulent activities using Ripple Down Rule system technique and Neural Networks.

Scope of Study

This work focused on designed and implementation of cybercrimes detection and prevention system that specifically targets web applications. Also, this work is limited to investigation of cybercrime act from year 2000 to year 2020 and development of intrusion detection system in web application by modifying Ripple Down Rule system technique. Training and evaluation data for the study were acquired from

LITERATURE REVIEW

Cybercriminal

Cybercriminal is an individual who commits cybercrimes, where he/she makes use of the computer either as a tool or as a target or as both. Cybercrime is any crime that involves a computer and a network to commit different crime activities. Cybercrime may be committed by individuals or small groups, as well as by criminal organizations that are often spread around the world and committing crimes on an unprecedented scale. Cybercrime has the unusual characteristic that the victim and the perpetrator may never come into direct contact; in many cases, perpetrators and victims are separated by thousands of miles. A Web application (Web app) is an application program that is stored on a remote server and delivered over the Internet through a browser interface. According to Adelabu (2015), Nigerian banks lost a total of N159 billion to electronic fraud between year 2000 and the first quarter of year 2013 while Africa retained its position as the region with the largest fraud cases.

Before the Internet, criminals had to dig through people's trash or intercept their mail to steal their personal information. Now that all of this information is available online, criminals also use the internet to steal people's identities, hack into their accounts, trick them into revealing the information, or infect their devices with malware. Most cybercrimes are committed by individuals or small groups. However, large organized crime groups also take advantage of the internet. These "professional" criminals find new ways to commit old crimes, treating cybercrime like a business and forming global criminal communities.

Criminal communities share strategies and tools and can combine forces to launch coordinated attacks. They even have an underground marketplace where cyber criminals can buy and sell stolen information and identities. It's

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

very difficult to crack down on cyber criminals because the Internet makes it easier for people to do things anonymously and from any location on the globe. Many computers used in cyber-attacks have actually been hacked and are being controlled by someone far away. Crime laws are different in every country too, which can make things really complicated when a criminal launches an attack in another country. Here are few types of attacks cyber criminals use to commit crimes: botnet, Fast Flux and Denial-of-Service. Some identity thieves target organizations that store people's personal information, like schools or credit card companies (Garcia et al., 2012). But most cyber criminals will target home computers rather than trying to break into a big institution's network because it's much easier. By taking measures to secure your own computer and protect your personal information, you are not only preventing cyber criminals from stealing your identity but also protecting others by preventing your computer from becoming part of a botnet.

Possible Attacks Caused by Cyber Criminals

The modern technologies and its deployment in computer and mobile devices have not only created new opportunities for better services but from other perspective, privacy of the users is highly questionable. The network-intruder and virus contagion extremely affect the computer system and its counterparts. They also alter the top confidential data. There are different types of cyber-crime that directly or indirectly has effect on our economy. According to malware research (1992), a malware is malicious software that is intended to damage or disable computer systems. Any unwanted software that is installed without adequate consent from the computer user is known as a malware. Viruses, worms, and Trojan horses are examples of malicious software that are often grouped together and referred to as malware

Malware is a major threat on the internet today based on their operational mode; they can be categorized into propagation, replication, corruption and self-execution. Malware is being designed by attackers, Malware writers/users. Some of the most popular names are black hats, hackers and crackers. The actual persons or organizations that take on the aforementioned names could be an external/internal threat, a foreign government or an industrial spy (Aru and Chiaghana, 2018).

There are essentially two phases in the lifecycle of software through which malware is inserted. These phases are referred to as the pre-release phase and the post release phase. An internal threat or insider is generally the only type of hacker capable of inserting malware into software before its being released to the end-users. An insider is a trusted developer, typically within an organization. All other persons or organizations that take on the hacker role insert malware during the post-release phase, which is when the software is available for its intended audience. In creating new malware, black hats generally employ one or both of the following techniques: obfuscation and behavior addition/modification in order to circumvent malware detectors. Obfuscation attempts to hide the true intentions of malicious code without extending the behaviors exhibited by the malware. Behavior addition/modification effectively creates new malware, although the essence of the malware may not have changed. The widespread use of the aforementioned techniques by malware coders along with those mentioned by researchers suggests that reused code is a major component in the development of new malware. This implication plays a critical role in some of the intrusion-based malware detection (Kolbitsh et al., 2013).

The following are agents of cybercrime: Viruses and worms, Spam emails, Trojan, Denial-of-service (DoS), Malware, Scareware, Phishing, Fiscal fraud, State sponsored cyber-attacks, Carders.

Viruses and worms

Viruses and worms are computer programs that affect the storage devices of a computer or network which then replicate information without the knowledge of the user.

Virus is a malicious program able to inject its code into other programs/applications or data files. After successful code replication the targeted areas become "infected". By definition virus installation is done without user's

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

consent and spreads in form of executable code transferred from one host to another. Purpose of viruses is very often of a harmful nature such as data deletion or corruption on the targeted host leading up to system in-operability in worst case scenario (Powers and He, 2008).

Viruses can spread pretty fast over network, shares or removable media. On many occasions the virus spread scenarios are connected with social engineering attacks, where end-users are tricked to execute malicious links or download malicious files, in some other cases malicious email attachments are being opened by end-users which ends in infection. Viruses as already mentioned have as well ability to inject the code in other legitimate executable files - when afterwards run by end-users - the virus code contained in the infected program is being executed simultaneously. Viruses can take advantages of known operating system security vulnerabilities that allow them to access the target host machines. Virus can be classified as follows: Resident virus, non-resident virus, boot sector virus and macro virus (Blum et al., 2004). A computer virus is a software usually hidden within another seemingly innocuous program that can produce copies of itself and insert them into other programs or files and that usually performs a harmful action (such as destroying data (Krugel et al., 2009)

Resident Virus: This is virus that embeds itself in the memory on a target host. In such way it becomes activated every time the OS starts or executes a specific action.

Non-resident Virus: This type of virus actively seeks targets for infections either on local, removable or network locations. Upon further infection it exits this way is not residing in the memory any more.

Boot sector Virus: This type of virus targets specifically a boot sector on the host's hard drive. This type of viruses is being loaded to memory every time when an attempt is being made to boot from the infected drive - this kind of viruses loads well before the OS loads. Boot sector viruses were quite common in the 90s where the infection was spread mostly through the infected floppy disks left in the bootable drives.

Macro Virus: Virus written in macro language, embedded in Word, Excel, Outlook etc. documents. This type of viruses is being executed as soon as the document that contain it is opened this corresponds to the macro execution within those documents that under normal circumstances is automatic. A well-known example of a macro virus is Melissa virus (1999), very widespread in that time. Virus causes a lot of damages to the communities.

File-infecting Virus (File-Infector): When the infected file is being executed the virus seeks out other files on the host and infects them with malicious code. The malicious code is being inserted either at the begging of the host file code (prepending virus); in the middle (mid-infector); or at the end (appending virus). A specific type of viruses called "cavity virus" can even inject the code in the gaps in the file structure itself. (Chao and Tan, 2009). The start point of the file executions is changed to the start of the virus code to ensure that it is run when the file is executed - afterwards the control may or may not be passed on to the original program in turn. Depending on the infections routing the host file may become otherwise corrupted and completely non-functional. More sophisticated viral forms allow though the host program execution while trying to hide their presence completely (Tong, wang and Yu, 2009).

Polymorphic Virus: This kind of viruses can change its own signature every time it replicates and infects a new file in order to stay undetected from antivirus programs. Every new variation of the virus is being achieved by using different encryption method each time the virus file is being copied. This type of viruses is especially difficult in detection by any detection programs due to the number of variants - sometimes going in hundreds or even thousands.

Metamorphic Virus: This kind of virus is capable of changing its own code with each infection. The rewriting process may cause the infection to appear different each time but the functionality of the code remains the same. The metamorphic nature of this virus type makes it possible to infect executable from two or more different

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

operating systems or even different computer architectures as well. The metamorphic viruses are ones of the most complex in build and very difficult to detect.

Stealth Virus: This is memory resident virus that utilizes various mechanisms to avoid detection. This avoidance can be achieved for example by removing itself from the infected files and placing a copy of itself in a different location. The virus can also maintain a clean copy of the infected files in order to provide it to the antivirus engine for scan, while the infected version still remains undetected. Furthermore the stealth viruses are actively working to conceal any traces of their activities and changes made to files (Tong et al., 2009).

Worm: This category of worm exploiting operating system vulnerabilities to spread itself. In its design worm is quite similar to a virus considered even its sub-class. Unlike the viruses though worms can reproduce/duplicate and spread by itself - during this process worm does not require attaching itself to any existing program or executable. In other words it does not require any interaction for reproduction process - this capability makes worm especially dangerous as they can spread and travel across network having a devastating effect on both the host machines, servers as well consuming network bandwidth. More invasive worms target to tunnel into the host system and from within to allow code execution or remote control from the attacker. Some worms can as well include a viral component that infects executable files. Worms are aptly named for their ability to "crawl" through networks. Worms replicate themselves but do not embed themselves in other programs as a virus tends to do. Worms move along a network connection seeking vulnerable machines to infect (Aru and Chiaghana, 2018).

The most common categorization of worms relies on the method and how they are spread:

i. Email worms: spread through email massages especially through those with attachments

ii. Internet worms: spread directly over the internet by exploiting access to open ports or system vulnerabilities

iii. Network worms: spread over open, unprotected network shares

iv. Multi-vector worms: having two or more various spread capabilities (Aru and Chiaghana, 2018).

Spam emails

Spam emails are unsolicited emails or junk newsgroup postings. Spam emails are sent without the consent of the receiver, potentially creating a wide range of problems if they are not filtered appropriately. Spam is the electronic equivalent of the ‘junk mail’ that arrives on your doormat or in your postbox. However, spam is more than just annoying. It can be dangerous especially if it’s part of a phishing scam (Vijesh and Santhadevi 2014). Spam emails are sent out in mass quantities by spammers. Spammers are not generally trying to get sensitive information from you, although they may try to elicit personal information to add to their database for future spam attempts and cybercriminals that are looking to do one or more of the following:

i. Make money from the small percentage of recipients that actually respond to the message
ii. Run phishing scams in order to obtain passwords, credit card numbers, bank account details and more
iii. Spread malicious code onto recipients’ computers

Trojan

A Trojan is a program that appears legitimate. However, once run, it moves on to locate password information or makes the system more vulnerable to future entry. Or a Trojan may simply destroy programs or data on the hard disk. A Trojan horse is a destructive program that masquerades as a benign application. Unlike viruses, Trojan

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

horses do not replicate themselves but they can be just as destructive. One of the most insidious types of Trojan horse is a program that claims to rid your computer of viruses but instead introduces viruses into your system.

Roesch (2009) described Trojan as the term that comes from the Greek myth of the Trojan War in which the Greeks give a giant wooden horse to their foes, the Trojans, ostensibly as a peace offering. But after the Trojans drag the horse inside their city walls, Greek soldiers sneak out of the horse's hollow belly and open the city gates, allowing their compatriots to pour in and capture Troy. Trojans appear to be benign programs to the user but will actually have some malicious purpose. Trojans usually carry some payload such as remote access methods, viruses and data destruction. Trojans provide a back door for the malicious attacker and gives them the following abilities: Session logging, Keystroke logging, File transfer, Program installation, Remote rebooting, Registry editing, and Process management. This Trojan horse is designed to destroy and delete files, and it's more like a virus than any other Trojan. It can often go undetected by anti-virus software.

Trojan horses are broken down into categories based on how they breach systems and the damage they cause (Sinclair et al., 2009). The following are the main types of Trojan horses: remote access trojans, data sending trojans, proxy trojans, file transfer protocol trojans, security software disabler trojans.

Remote Access Trojans (RATs)

RATs, a Remote Access Trojan is designed to provide the attacker with complete control of the victim's system. Attackers usually hide these Trojan horses in games and other small programs that unsuspecting users then execute on their PCs. Remote Access (William et al., 2003).

Trojans (RATs) provide cybercriminals with unlimited access to infected endpoints. Using the victim's access privileges, they can access and steal sensitive business and personal data including intellectual property, personally identifiable information and patient health information. While automated cyber-attacks such as Man-in-the-Browser allow cybercriminals to attack browser-based access to sensitive applications, RATs are used to steal information through manual operation of the endpoint on behalf of the victim. Most Advanced Persistent Threat (APT) attacks take advantage of RAT technology for reconnaissance, bypassing strong authentication, spreading the infection, and accessing sensitive applications to exfiltrate data. RATs are commercially available in Poison Ivy, Dark Comet and can be maliciously installed on endpoints using drive-by-download and spear-phishing tactics. Organization should specifically address RATs in their enterprise defense strategy at the endpoint layer. The risk is especially high when RAT infection occurs, as the detection of RATs in run-time is extremely difficult to do (William et al., 2003).

RATs enable cybercriminals to perform key-logging and session logging of the user activity as a way to capture credentials, sensitive data, and gather intelligence on internal application flows and structures. By preventing these run-time activities when sensitive applications like Virtual Private Network (VPN) and Virtual desktop infrastructure (VDI) clients (or the browser) are used, the ability of the attackers to leverage the RAT to execute an attack is dramatically reduced. Controls should be implemented to prevent RAT malware from infecting managed and unmanaged devices. If a device is infected, the controls need to quickly detect and remove RATs from end users' machines. Future infections must be stopped by blocking malware installation processes and spear-phishing attacks. Special focus should be given to resource consumption and management overhead when balancing strength of the protection and risk reduction with end user and IT security impact. Anti-malware solutions must cover the vast majority of managed and unmanaged device platforms, including PCs, Macs and Mobile (iOS and Android devices). The solution must be readily available to end users to instantly secure their devices. An on demand deployment option is required when enterprise resources are accessed from home computers or on the road. Organizations must have the ability to mandate that all VPN access be performed from secured endpoints (Dharmapurikar et al., 2003)

Data sending trojans

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

This type of Trojan horses is designed to provide the attacker with sensitive data such as passwords, credit card information, log files, e-mail address and IM contact lists. These Trojans can look for specific pre-defined data such as credit card information and passwords, or they install a key logger and send all recorded keystrokes back to the attacker. A data-sending Trojan is a kind of Trojan virus that relays sensitive information back to its owner. This type of Trojan can be used to retrieve sensitive data, including credit card information, email addresses, passwords, instant messaging contact lists, log files and so on (Dharmapurikar et al., 2003).

Some data-sending Trojans are not used for malicious acts, but they still relay information. Usually, this is to serve ads to the user. Hackers use data-sending Trojans to gain data about user activity on the Web, such as the sites visited and the ads or other icons clicked. Using this information the Trojans serve the relevant ads to the user. Sometimes, data-sending Trojans will install key loggers on a computer so that all key strokes are recorded and sent back to the hacker. Then another program parses the data and separates it into readable words. Data-sending Trojans are quite annoying, even though some are only used to present ads to a user. However, these ads may arrive irregularly and can be cunning if they advertise something like virus removal when in fact the Trojan pushing the ad for a product to remove itself from the user's computer.

Proxy trojans

This kind of Trojan horses is designed to use the victim's computer as a proxy server. This lets the attacker do anything from your computer, including credit card fraud and other illegal activities and even use your system to launch malicious attacks against other networks

File transfer protocol trojans

This Trojan horse opens port 21 (the port for FTP transfer) and lets the attacker connect to your computer using File Transfer Protocol (FTP)

Security software disabler trojans

This nasty Trojan horse stops or kills computer security software such as antivirus programs or firewalls without the user knowing. It’s usually combined with another type of Trojan as a “payload” (Manjeri and Shelke, 2014).

Denial-of-service

Denial-of-service (DoS) occurs when criminals attempt to bring down or cripple individual websites, computers or networks, often by flooding them with messages (Yoohwan, et al., 2004). A DoS is a type of attack that brings a network to its knees by flooding it with useless traffic. Many DoS attacks, such as the Ping of Death and Teardrop attacks, exploit limitations in the TCP/IP protocols. For all known DoS attacks, there are software fixes that system administrators can install to limit the damage caused by the attacks. But, like viruses, new DoS attacks are constantly dreamed up by hackers (Yoohwan et al., 2004)

Malware

Malware is software that takes control of any individual’s computer to spread a bug to other people’s devices or social networking profiles. Such software can also be used to create a ‘botnet’ a network of computers controlled remotely by hackers, known as ‘herders,’ to spread spam or viruses. Malware is shorthand for malicious software. It is software developed by cyber attackers with the intention of gaining access or causing damage to a computer or network, often while the victim remains oblivious to the fact there's been a compromise. A common alternative description of malware is computer virus although are big differences between these types malicious programs (Anderson et al., 2012). Malware is typically used to steal information that can be readily monetized, such as login credentials, credit card and bank account numbers, and intellectual property such as computer software,

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

financial algorithms, and trade secrets. Although many cybercriminal groups are trafficking in commodities shared by multiple industry sectors such as credit card numbers, there are some situations wherein a single company is obviously the target of a single adversary, whether it be an organized crime syndicate, nation-state or a single operative (Zou et al., 2015).

Scareware

Using fear tactics, some cyber criminals compel users to download certain software. While such software is usually presented as antivirus software, after some time these programs start attacking the user's system. The user then has to pay the criminals to remove such viruses (Neelabh, 2012).

Phishing

Phishing is when scammers use email messages with phony email addresses, websites, or pop-up windows to gather personal information. Phishing attacks are designed to steal a person's login and password. For instance, the phisher can access the victims' bank accounts or assume control of their social network. According to Roger (2008) phishing is simply a high-tech identity theft that does not only steal personal information and identity from unsuspecting consumers but also an act of fraud against the legitimate businesses and financial institutions that are victimized by phishing.

Phishing is usually a social engineering crime pervasive in attacking organizations' or individuals' (customers') information systems (IS) in order to gather private information to be used against organizations to extract some benefit for the perpetrator through the anonymity of identity theft or identity deception acts (Roger, 2008).

Study from the Anti-Phishing working group 2008 showed that phishing scams remain a relatively small percentage of spam sent worldwide today. Phishing attempts to pose significant dangers for unsuspecting victims. It has become one of the fastest-growing worldwide threats on the Internet. This rapid growth in destruction has prompt an effective technique that will curb the act of phishing crime.

Fiscal fraud

By targeting official online payment channels, cyber attackers can hamper processes such as tax collection or make fraudulent claims for benefits.

State cyber-attacks

Experts believe that some government agencies may also be using cyber-attacks as a new means of warfare. One such attack occurred in 2010, when a computer virus called Stuxnet which was used to carry out an invisible attack on Iran's secret nuclear program. The virus was aimed at disabling Iran's uranium enrichment centrifuges (Garcia et al., 2009).

Carders

Stealing bank or credit card details is another major cyber-crime. Duplicate cards are then used to withdraw cash at ATMs or in shops. Carding is a form of credit card fraud in which a stolen credit card is used to charge prepaid cards. Carding typically involves the holder of the stolen card purchasing store-branded gift cards, which can then be sold to others or used to purchase other goods that can be sold for cash. Credit card thieves who are involved in this type of fraud are called "carders."

How Carding Works

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Carding typically starts with a hacker gaining access to a store's or website's credit card processing system, with the hacker obtaining a list of credit or debit cards that were recently used to make a purchase. Hackers might exploit weaknesses in the security software and technology intended to protect credit card accounts. They might also procure credit card information by using scanners to copy the coding from the magnetic strips. Credit card information might also be compromised by accessing the account holder's other personal information, such as bank accounts the hacker has already gained entry to, targeting the information at its source. The hacker then sells the list of credit or debit card numbers to a third party a carder who uses the stolen information to purchase a gift card.

Most credit card companies offer cardholders protection from charges made if a credit or debit card is reported stolen, but by the time the cards are canceled, the carder has often made a purchase. The gift cards are used to purchase high-value goods, such as cell phones, televisions, and computers, as those goods do not require registration and can be resold later. If the carder purchases a gift card for an electronic retailer, such as Amazon, he or she may use a third party to receive the goods and then ship them to other locations. This limits the carder's risk of drawing attention. The carder may also sell the goods on websites offering a degree of anonymity.Because credit cards are often canceled quickly after being lost, a major part of carding involves testing the stolen card information to see if it still works. This may involve submitting purchase requests on the Internet.

Buffer overflows

Buffer overflows finds means of attacking a computer or network. They are rarely launched on their own and are usually part of a blended attack. Buffer overflows are used to exploit flawed programming in which buffers are allowed to be overfilled. If a buffer is filled beyond its capacity the data filling it can then overflow into the adjacent memory and then can either corrupt data or be used to change the execution of the program. A buffer is a temporary area for data storage. When more data (than was originally allocated to be stored) gets placed by a program or system process, the extra data overflows. It causes some of that data to leak out into other buffers, which can corrupt or overwrite whatever data they were holding. In a buffer-overflow attack, the extra data sometimes holds specific instructions for actions intended by a hacker or malicious user; for example, the data could trigger a response that damages files, changes data or unveils private information (Akash, 2015).

Attacker would use a buffer-overflow exploit to take advantage of a program that is waiting on a user's input. There are two types of buffer overflows: stack-based and heap-based. Heap-based, which are difficult to execute and the least common of the two, attack an application by flooding the memory space reserved for a program. Stack-based buffer overflows, which are more common among attackers, exploit applications and programs by using what is known as a stack: memory space used to store user input. The two main types of buffer overflows are described as follows:

Stack buffer overflow

A stack is an area of memory that a process uses to store data such as local variables, method parameters and return addresses. Often buffers are declared at the start of a program and so are stored in the stack. Each process has its own stack, and its own heap. Overflowing a stack buffer was one of the first types of buffer overflows and is one that is commonly used to gain control of a process. In this type of buffer overflow, a buffer is declared with a certain size. If the process controlling the buffer does not make adequate checks, an attacker can attempt to put in data that is larger than the size of the buffer. An attacker may place malicious code in the buffer. Part of the adjacent memory will often contain the pointer to the next line of code to execute (Chiueh and Hsu, 2001). Thus, the buffer overflow can overwrite the pointer to point to the beginning of the buffer, and hence the beginning of the malicious code. Thus, the stack buffer overflow can give control of a process to an attacker.

Heap overflows

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Heap overflows are similar to stack overflows but are generally more difficult to create. The heap is similar to the stack but stores dynamically allocated data. The heap does not usually contain return addresses like the stack, so it is harder to gain control over a process than if the stack is used. However, the heap contains pointers to data and to functions. A successful buffer overflow will allow the attacker to manipulate the processes execution. An example would be to overflow a string buffer containing a filename, so that the filename is now an important system file (Chiueh and Hsu, 2001). The attacker could then use the process to overwrite the system file (if the process has the correct privileges)

Web Application Attack

Web application is an application program that is stored on a remote server and delivered over the Internet through a browser interface. Web applications are designed to assist the end user to connect to the Database through a web browser (Connoly,Bgg and Stranchan, 2006). Unfortunately, most web applications are not designed with security in mind. Thus, they have a lot of exploitable vulnerabilities (Robertson, 2010). In computer security, vulnerability is a weakness which allows an attacker to reduce a system's information assurance. Vulnerability is the quality of being easily hurt or attacked. These weaknesses have encouraged malicious software writers to take advantage of such vulnerabilities to cause attack mostly for financial gains. For example, if a user's input is not handled correctly, the web application can be left vulnerable to a cyber-crime. For this reason, attackers have developed crime tool kits to look for vulnerabilities to launch attacks against web applications. The availability of such toolkits, combined with the motivation to launch attacks, makes web applications a critical area to study (Anderson and Moore, 2007). Table 2.1 shows the overview of most application readability

security risk as reported by the Open Web Application Security Project (OWASP) in 2010. The Table also shows the impact of each risk. According to the report of Common Vulnerabilities and Exposures (CVE), the number of web-related crime has increased steadily from 2005 –2010 (Corporation, 2012).

Cybercrime Detection Approaches

Cybercrimes are criminal offenses committed via the Internet or otherwise aided by various forms of computer technology, such as the use of online social networks to bully others or sending sexually explicit digital photos using smart gadget (Alazab et al., 2013). Cybercrime can be defined as a series of actions aimed at compromising the security of a computer network system. It attempts to bypass security mechanisms of computer systems that threaten the availability, integrity, or confidentiality of computer resources. Cyber-criminal used different ways to gain information from the computer network. Intrusions can take two primary forms: external attacks, insider attacks (Sheikhan et al., 2010).

Cybercrime detection is said to be a method of identifying user activities that may possibly project computer systems from a secured state to an unsecure state. However, the cyber-crime detection systems could be hardware or software systems that observe activities on computer.

systems or computer networks in order to find abnormal activities (Stakhanova et al., 2007). The ability of cyber-crime detection and prevention to prevent up-to-date attacks is inadequate (Garcia et al., 2009).

Based on the input data sources of IDSs examined, there are two main types of IDSs: network-based and host-based. Host-based IDSs (HIDS) examine data that originate from the host system, audit sources, such as application system audits, operating system, server, or

Table 2.1: The web application security risks (Alazab et al., 2013)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

S/N	Web application Security attacks	Description of the crime	Impact of the attack
1	Injection	The hacker sends a malicious stringas a query statement to the database.	Data loss, data stolen, Enhanced , deleted, or denial of access.
2	Cross-Site Scripting (XSS)	Attacker executed malicious code in the user's browser	Redirect users to malicious
3	Broken Authentication	When a session ID is visible due to leaks in the authentication process	Attacker could do anything with the victim's privileged accounts.
4	Insecure Direct Object References	The attacker change the URL Parameter	Compromises all the data that can be referenced by the parameter
5	Cross-Site Request Forgery (CSRF)	Attacker sends fake HTTP requests to trick a victim into submitting them via image tags.	Loss or Enhanced data.
6	Security Misconfiguration	Attackers try to find unused pages, unprotected files and unprotected directories, etc. to reach for sensitive data.	The web application could be completely compromised without knowledge.
7	Insecure Cryptographic Storage	Here attackers don't break the cryptographic, but disclose something else to get the original.	Lost data such as health records, credentials, personal data, and credit cards.
8	Failure to restrict URL access	Simply changes the URL to a privileged page	Attacker can do anything the victim can do
9	Insufficient Transport Layer Protection.	The attacker simply monitors network traffic, and observes an authenticated victim's session cookie.	Phishing, resulting in a stolen account.
10	Invalidated Redirects and Forwards	Attacker redirects links to invalidate and trick victims into clicking it. Victims will probably click on it, with the security mechanism bypassed.	Redirections such as these may attempt to install Malware or trick victims into disclosing passwords or other sensitive information.

database logs. Host- Based Intrusion Detection System detector plays a significant role for detection of inside attacks that do not involve network traffic

Network-based IDSs (NIDS) examine network packets that are taken from a network. Network-based IDS can be implemented to protect several hosts that are connected to a network. NIDS can report an attack that could be launched from the external attacker at an earlier stage, before the attacks actually reach the host. However, NIDSs have a difficulty to examine all packets in a high-speed network due to rate and scale of packets (Rodger, 2008).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Based on investigation, there are two approaches to analyze events using intrusion detection techniques. These can be categorized into two classes: signature-based detection technique and anomaly-based detection technique (Vigna and Kemmerer, 2009). Overview of the two techniques was discussed in the next session

Signature based Intrusion Detection System

Signature based Intrusion Detection System (SIDS) is based on pattern matching techniques to find a known attack. In other words, when a known intrusion collides with a malicious string database, an alert is raised. SIDS commonly provides good detection results for specified, well-known attacks. However, SIDS cannot detect unknown attacks because the signature does not exist in the database until the signature database is updated (Shimirit, 2011). The main advantage of a SIDS is that they are very efficient in detecting known attacks without raising false positive alarms and can quickly detect an attack. However, SIDS has the disadvantage that a signature must be created for every cybercrime, therefore SIDS is time consuming. A SIDS is also prone to false negative, since they are commonly based on regular expressions and string matching. While SIDS works well against cybercrime with a fixed behavioral pattern, they do not work well against the multitude of crime patterns created by a human or self-modifying behavioural characteristic. The SIDS has been implemented in many popular tools, such as Snort (Roesch, 2009) and NetSTAT (Vigna and Kemmerer, 2009).

Anomaly based Intrusion Detection System (AIDS)

Anomaly based intrusion detection system has attracted the interest of many researchers to overcome the disadvantage of SIDS. Anomaly detectors identify unusual user activity on a computer system. The assumption for this approach is that behaviour of an attacker deviates from normal user behaviour. AIDS constructs a behavior profile of users by using features that are accepted as normal behaviour. It then monitors the activities of users and compares the new data with the obtained normal behavior profile and tries to detect deviations. Those activities of new users which are different from normal behaviour are considered as attacks (Cova et al., 2008). To achieve this, AIDS involves two stages: the training stage; and a testing stage. In the training stage, the normal traffic profile is labeled by using data that is accepted as normal behaviour; in the latter, the testing stage is applied to new data set. The result of this is that the AIDS have the ability to monitor the activities of new users and compare the new data with the obtained profile in an attempt to detect deviations. Those different from normal behavior are considered as attacks (Garcia et al., 2009).

Anomaly based intrusion detection system may be categorized into different sub-kinds based on the methods employed to do learning in the literature such as statistical techniques, data mining, Artificial Neural Networks (ANN) and Genetic Algorithms (GA). The main benefit of an anomaly-based technique is the ability to detect zero days malware because the unusual user profile is not dependent on a signature database. AIDS raises an alarm because it differs from the usual activity. Moreover, AIDS has many advantages: First, they have the ability to find insider attacks. If an attacker using a stolen account starts making actions that are unknown in the normal user-profile, it generates an alarm. Second, because the system is constructed from customized profiles, it is extremely hard for an attacker to identify what is normal user activity without generating an alarm (Patcher and Park, 2007). The other added disadvantage of AIDS is the possibility of detecting a known attack, that is, mistake to determine non-attack. However, anomaly detection has an advantage over signature-based engines in detecting new attack.

Two major kinds of intrusion detection systems that used SIDS or AIDS are Host- based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS) (Patcher and Park, 2007).

Host Based IDS and Network Based IDS

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Based on the location in a network, IDS can be categorized into two groups such as host-based IDS and the other is network based IDS. Both have its own advantages and disadvantages. Before you decide which IDS suits your network environment the best you need is to have a clear concept of both types of IDS.

Host Based IDS: Host- based Intrusion Detection Systems (HIDS) rely on the data that is derived from a single host, whereas Network- Based Intrusion Detection Systems (NIDS) derived information by monitoring the traffic movement in the network to which the hosts are connected. A host-based intrusion detection system (HIDS) is a system that monitors a computer system on which it is installed to detect an intrusion or misuse and responds by logging the activity and notifying the designated authority. A HIDS can be thought of as an agent that monitors and analyzes whether anything or anyone whether internal or external has affected the system's security policy. HIDS has the ability of determining if an attempted attack was indeed successful. Software set up directly on the host protects against not just the attack, but against the consequences of an attack. It is independent of network architecture; and can detect local attacks, privilege growth attacks and attacks which are encrypted (Longe and Chiemeke, 2008).

However, such system be problematic to deploy and run, particularly with the amount of hosts requiring such protection in web application. Additionally, these systems are not equipped to identify attacks against multiple hosts in a network. A HIDS analyzes the traffic to and from the specific computer on which the intrusion detection software is installed. A host-based system also has the ability to monitor key system files (Shafi and Abbass, 2009).

Network Based IDS: Network Intrusion Detection System (NIDS) is a system that attempts to detect hacking activities, denial of service attacks or port scans on a computer network or a computer itself. The NIDS can monitor incoming, outgoing, and local traffic. NIDS is able to monitor a large number of hosts with reasonably small deployment costs, and are efficient to detect attacks over multiple hosts. They can detect network Denial of Services (DoS), bandwidth-oriented attacks, SYN flood (SYN flood is a form of denial-of-service attack in which an attacker sends a succession of SYN requests to a target's system in an attempt to consume enough server resources to make the system unresponsive to legitimate traffic) and others; and can provide a larger observation of attack types such as scans, probes and attacks against non-host system based. A NIDS is often a standalone hardware appliance that includes network detection capabilities. It will usually consist of hardware sensors located at various points along the network. It may also consist of software that is installed on various computers connected along the network. The NIDS analyzes data packets both inbound and outbound and offer real-time detection. NIDS relies on statistical anomaly detection techniques. It has a profile of normal user activity behavior which is provided by the statistical techniques (Amrit and Manik, 2004). This helps the IDS to compare the existing user action with the normal behaviour of the variables that are stored in the system and then raise an alarm if the result of the comparison is considerably far from normal behavior.

This type of IDS is strategically positioned in a network to detect any attack on the hosts of that network. To capture all the data passing through the network, you need to position your IDS at the entry and exit point of data from your network to the outside world. You can also position some IDS near the strategic positions of your internal network, depending on the level of security needed in your network. Since a network based IDS need to monitor all the data passing through the network, it needs to be very fast to analyze the traffic and should drop as little traffic as possible.

Depending on how they function, network based IDS can be divided into two types:

i. Statistical anomaly IDS
ii. Pattern matching IDS

In statistical based IDS model, the IDS try to find out users' or system's behavior that seem abnormal. Actually, IDS make a profile of every user and system during the normal operation time. When the deviation of this normal

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

behavior is detected the IDS trigger its alarm for intrusion. One of the main advantages of this type of IDS is that they can detect the type of intrusion that has no records of its previous occurrence. In that sense, statistical anomaly can detect new type of attack patterns. A large number of false alarms are the main problem with this type of detection system (Jamil, 2003).

Techniques for anomaly detection

Many research works carried out by different researcher make use of different techniques to detect intrusion in anomaly intrusion detection system. The main methods are statistical techniques, knowledge-based techniques and machine learning techniques

Statistical Techniques

Statistical techniques take a different approach to intrusion detection. The following are statistical technique to be discussed: Principal Component Analysis (PCA) and Chi-square distribution. The concept of the Statistical techniques is simple: it determines "normal" network activity and then all traffic that falls outside the scope of normal is flagged as anomalous (not normal). Statistical methods of anomaly detection attempt to learn network traffic patterns on a particular network. This process of traffic analysis continues as long as the Statistical techniques is active, so, assuming network traffic patterns remain constant, the longer the system is on the network, the more accurate it becomes. By analyzing network traffic and processing the information with complex statistical algorithms, statistical techniques look for anomalies in the established normal network traffic patterns.

All packets are given an anomaly score (indicating the degree of irregularity for the specific event) and if the anomaly score is higher than a certain threshold, the IDS generated an alert. The key to any statistical techniques is its ability to learn and distinguish normal from anomalous network activity (Jamil, 2003).

The statistical techniques are designed to learn from the previous event. The system learns what is normal by observing an activity over a period of time. When an unknown or rare (anomalous) event occurs, the statistical techniques observe it and subsequently generate an alert. The difference between an alert and a non-alert is based on the anomaly threshold. If the anomaly threshold is high, minor anomalies are not noteworthy. If the threshold is low, most of the anomalies are a cause for investigation (Jamil, 2003).

In statistical approaches for AIDS, the system is looking for some traffic parameters, which can be used to define the network traffic and create profiles to be normal activities (Stibor et al., 2005).

The principal component analysis

The Principal Component Analysis (PCA) is a common technique to find the patterns in a high dimension data by reducing its dimensions without losing the information contained in it. It produces a set of principal components which are orthonormal eigenvalue/eigenvector pairs, i.e., it projects a new set of axes that best suit the data. In our proposed scheme, these set of axes represent the normal feature data. Outlier detection occurs by mapping the data on to these 'normal' axes and calculating the distances from these axes. If the distance is greater than a certain threshold, we may confirm there is an attack i.e. outlier detection. Principal components are particular linear combinations of m random variables: X1, X2, ... ,Xm that have two important properties:

a. X_1 , X_2 , ... , X_m are uncorrelated and sorted in descending order.

b. Total variance, X, of X_1 , X_2 , ... , X_m is given by

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

$$X = \sum_{i=1}^{m} X_i \qquad (2.1)$$

These variables are found from eigenanalysis of the correlation matrix of the original variables X_1, X_2, ... ,Xm. The values of correlation matrix and covariance are not same (Yeh, Lee and Wang, 2013). Let the original data, X, be an nxm data matrix of n observations with each observation consisting of m fields (dimensions) X1, X2,..., Xm and R be a m x m correlation matrix of X1, X2,...,Xm. The eigenvalues of R are the roots of the following polynomial equation:

$$| R - \lambda I | = 0 \qquad (2.2)$$

Each eigenvalue of R has a corresponding non-zero vector e, called an eigenvector,

$$Re = \lambda e \qquad (2.3)$$

If (λ_1 , e_1), (λ_2 , e_2), ..., (λ_m, e_m) are the m eigenvalue-eigenvector pairs of the correlation matrix R, the i^{th} principal component is given by

$$y_i = e_i^T(x - \bar{x})$$
$$= e_{i1}(x_1 - \overline{x_1}) + e_{i2}(x_2 - \overline{x_2}) + \ldots + e_{im}(x_m - \overline{x_m}),\ i = 1, 2, \ldots, m.$$

where

eigen values are ordered as $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda m \geq 0$,

e_i^T= (ei1 , ei2 ,..., eim) is ith eigenvector,

x=(x_1, x_2, … , xm)T is transpose of observation data

x =(x_1 , x_2 , ..., xm)T is transpose of sample mean vector of the observation data.

For any attribute xi, let the observed data be a1, a2, … , an, then we have

$$\overline{x_i} = 1/n \sum_{i=1}^{n} a_i \qquad (2.4)$$

The principal components derived from the covariance matrix are usually different from the principal components generated from the correlation matrix (Yeh et al., 2013). When some values are much larger than others, their corresponding eigenvalues have larger weights. One of these metrics is known as the Mahalanobis distance

$$d^2(x, y) = (x - y)^T R^{-1} (x - y) \qquad (2.5)$$

Where R^{-1} is the sample correlation matrix, x and y are vectors and here X^T represents the transpose of X. Using the correlation matrix, the relationships between the data fields is represented more effectively. There are two main issues in applying PCA, interpretation of the set of principal components and calculation of distance.

Each eigenvalue of a principal component corresponds to the amount of variation it encompasses. The larger eigenvalues are more significant and correspond to their projected eigenvectors. The principal components are sorted in descending order. Eigenvectors of the principal components represent axes which best suit a data sample.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Points which lie at a far distance from these axes are assumed to exhibit abnormal behavior and they can be easily identified. Using a threshold value, the data generated by normal system with Mahalanobis distance greater than the threshold is considered an outlier and, here, it is an intrusion. But, sometimes it alerts the user as intrusion if the data is on the threshold boundary.

Consider the sample principal components, y_1 , y_2 , ... ,y_m of an observation X where

$$y_i = e_i^T (X - \bar{x}), \qquad i = 1, 2, ..., m \tag{2.6}$$

The sum of squares of the partial principal component scores is equal to the principal component score:

$$\sum^{m} y_i^2/\lambda_i = y_1^2/\lambda_1 + y_2^2/\lambda_2 + ... + y_m^2/\lambda_m \tag{2.7}$$

equates to the Mahalanob is distance of the observation X from the mean of the normal sample dataset(Yee et al., 2013) Major principal component score is used to detect extreme deviations with large values on the original features. The minor principal component score is used to detect some attacks which may not follow the same correlation model. As a result, two thresholds are needed to detect attacks. q is a subset of major principal components and r is a subset of minor principal components. The major principal component score threshold is denoted T_q while the minor principal component score threshold is referred to as T_r. q is most significant principal components and r is least significant principal components. An attack occurs for any observation X if :

$$\sum_{i=1}^{q} y_i^2/\lambda_i > T_q \quad \text{or} \quad \sum_{i=m-r+1}^{m} y_i^2/\lambda_i > T_r \tag{2.8}$$

The chi-square distribution

The Chi-square distribution is an asymmetric distribution that has a minimum value of 0, but no maximum value. The curve reaches a peak to the right of 0, and then gradually declines. The mean of the Chi square distribution is the degree of freedom and the standard deviation is twice the degrees of freedom. For each degree of freedom, there is a different X2 distribution. This implies that the Chi square distribution is more spread out with a peak farther to the right, for larger than for smaller degrees of freedom. As a result, for any given level of significance, the critical region begins at a larger Chi-square value, the larger the degree of freedom (Danforth, 2009). The size of multivariate data are quantified by the covariance matrix. A well-known distance measure which takes into account the covariance matrix is the Mahalanobis distance. For a p-dimensional multivariate sample xi (i = 1, …, n), the Mahalanobis distance is defined as MDi = ((xi - t)T C-1 (xi - t))1/2 , i = 1, … , n

Where t is the estimated multivariate location and C the estimated covariance matrix. Usually, t is the multivariate arithmetic mean, and C the sample covariance matrix. For multivariate normally distributed data, the values are approximately Chi-square distributed with p degrees of freedom (Xp^2). The multivariate outliers can now simply be defined as observations having a large (squared) Mahalanobis distance. For this purpose, a quantile of the Chi-squared distribution such as 99.5% quantile could be considered (Danforth, 2009). However, this approach has several limitations. The Mahalanobis distances need to be estimated by a robust procedure in order to provide reliable measures for the recognition of outliers.

Single extreme observations, or groups of observations, departing from the main data structure can have a severe influence to this distance measure because both location and covariance are usually estimated in a non-robust manner. The minimum covariance determinant (MCD) estimator is probably most commonly used in

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

practice. Using robust estimators of location and scatter in the above equation leads to robust distances (RD). If the squared RD for an observation is larger than, say, $X^2p;0.995$, it can be declared as an outlier. The Chi-square plot are obtained by the squared Mahalanobis distances (which have to be computed ont the basis of robust estimations of location and scatter) against the quantiles of Xp^2, the most extreme points are deleted until the remaining points follow a straight line. The deleted points are identified as outliers. This method needs user interaction and experience on the part of the analyst.

Moreover, especially for large data sets, it can be time consuming, and also to some extent it is subjective.

Advantages of statistical techniques

Statistical techniques have a number of advantages. The behavior-based analysis of traffic patterns and subsequent notification of anomalous activity can be critical in today's world of constant cyber threats. Because no prior knowledge of an attack is required, as is generally the case with rule-based IDS, statistical techniques can detect "0 day" (extremely new) attacks. If a network is attacked by a previously unseen worm, virus, or Denial of Service (DOS) attack, statistical methods is likely to alert based on the presence of the unusual activity. Statistical techniques are also capable of detecting "low and slow" attacks. These attacks (or portscans), usually performed by skilled intruders, are characterized by their lengthy duration (possibly months at a time), precision, and methodical execution. Usually these attacks are intended to enumerate the network or gather information about a specific system. It doesn't matter how "low and slow" that attack is though, in most cases, even if the attacker sends a mere packet a month (or less), the statistical techniques will note that it is anomalous traffic and alert on the event (Seog and Narasimha, 2005).

Another major benefit of statistical techniques is that it is potentially easier to maintain than an rule-base intrusion detection system. There is no need to update signatures because the system doesn't rely on specific attacks or conditions. You may say: "A self-updating, "0-day" attack catching, port-scan notifying.

Disadvantages of statistical techniques

Statistical techniques are wonderful for what they can potentially accomplish but there are a few significant shortfalls in the technology. A statistical technique is useful by relies completely upon its ability to learn the regular patterns of network traffic for a given network segment. Theoretically, this principle is sound, but not necessarily in practice.

The truth is that most networks are extremely diverse in terms of protocols, services and usage times. In many cases, the only thing "normal" about a network is the fact that it is constantly changing. Statistical techniques also suffer from the ability to be "taught" by intruders (Iwasokun, Alese, Thompson, 2012). For example, an attacker could use a program like Nmap and send numerous SYN-scans at the network. Over time that activity would be deemed normal by the Statistical Based Intrusion Detection (SBID) system. The SBID system learns what is "normal" for a given network so if Web traffic, for instance, is the only activity on a network, anything that falls outside of that scope will be alerted on. On the flip side, if flood pings are commonplace on a given network, then a SBID would allow the activity because it cannot determine "good" traffic from "bad", only normal from anomalous (This is analogous to a situation where the "check engine" light in your car turns on. Initially it is a concern but if it continues to stay lit with no noticeable problems to the car, the light becomes more and more ignored. Eventually, the illuminated warning light becomes "normal" to you).

Another issue with statistical-based intrusion detection is the implementation of such a system. Unlike a Rule Based Intrusion Detection (RBID) system, Statistical Based Intrusion Detection system will not have an immediate effect when it is connected to the network it cannot hit the ground running. Even if the SBID system is on a relatively consistent network, the learning process takes days or weeks to become accurate and effective. SBID systems require that operators be highly skilled in the art of packet analysis (Sumanjit et al.,2013). Another

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

concern is that SBID systems do not generate alerts that clearly state the issue at hand like a RBID system does. The data analyst will not receive alerts with a header "EXPLOIT dtspcd exploit attempt" for example. The alert will simply be a packet trace and it will be up to the operator to determine the nature of the alert. Furthermore, SBID system operators must establish an appropriate, effective threshold. Similar to determining the number of rules to implement in a RBID system, the threshold will either make or break the effectiveness of the SBID system. Set the threshold too high and the system will not alert on the necessary traffic, too low and the system will produce an overabundance of false positives.

Machine Learning Techniques

Different researchers have used machine learning techniques for anomaly detection techniques. Both supervised and unsupervised machine learning techniques can be used to solve several pattern recognition problems (Amusan and Arulogun, 2015). Supervised learning is based on using the training data to create a model, in which each of the training data has a class label. The training stage is to build a classifier model. Once the model is built, it can categorize new instances based on a learned class label. A number of machine learning-based structures have been used. Some of the machine learning techniques are enumerated below and their main capabilities and problems are explained for the purpose of this research work. Some of the most machine learning techniques are Bayesian Networks, Genetic Algorithms (GA), Artificial Neural Network (ANN).

Bayesian Networks

A Bayesian network is applied to model a domain having uncertainty. They have been widely used in AIDSs in different approaches. For example, Kruegel et al., (2008) developed an anomaly detection system that employed Naive Bayes. This model returns an anomaly score or a probability value that represents the 'normality' of this event as stated by their current profiles. It is a simple multiple-class classification algorithm with an assumption that a value of every feature is independent of the other regardless of any correlation between different features (Hesham and Saeed, 2011). Despite being simple, Naive-Bayes algorithm has been quite successful in many physical world problems. It is based on Bayes' theorem and it first calculates the conditional probability of each feature for a given label and then applies Bayes theorem to get the probability. Bayes theorem provides a method to find the posterior probability P(c/x), given P(c),P(x), and P(x/c) by following rule:

$$P(c|x) = \frac{P(x|c) * P(c)}{P(x)} \qquad (2.9)$$

Where:

P(c/x) it gives posterior probability for a class for given features x.

P(c) is the prior probability of class.

P(x/c) is the prior probability of features, given class.

P(x) is prior probability of features. But in our case

P(x) is not known, so there is another way round to solve this equation:

$P(c|x) = P(x_1/c) * P(x_1|c) * ....P(x_n/c) * P(c)$ (2.10)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Here, frequency table of each feature is drawn against the target and multiplied with the probability of class. Once, we get the posterior probability of each class; we compare them, and the class with higher probability gets labeled for particular data (Dargorn, 2008).

Genetic Algorithms (GA)

Genetic Algorithms (GA) have been employed in diverse ways in IDSs. Genetic Algorithm GA is a search method that finds an approximate solution to an optimization task inspired from biological, evolution process and natural genetics as proposed by Holland (2005). GA uses hill climbing method from an arbitrarily selected number of genes. GA has four operators: initialization, selection, crossover and mutation. GA has been used in different ways in IDS. There are many researchers that used evolutionary algorithms and especially GAs in IDS to detect malicious intrusion from normal use Genetic algorithm based intrusion detection system is used to detect intrusion based on past behavior. A profile is created for the normal behavior based on that genetic algorithm learns and takes the decision for the unseen patterns. Genetic algorithms are also used to develop rules for network intrusion detection. A chromosome in an individual contains genes corresponding to attributes such as the service, flags, logged in or not and uper-user attempts. These attacks that are common can be detected more accurately compared to uncommon attributes (Holland, 2015).

Genetic algorithms are capable of deriving classification rules and selecting optimal parameters for detection process. The application of Genetic Algorithm to the network data consist primarily of the following steps: The Intrusion Detection System collects the

information about the traffic passing through a particular network. The Intrusion Detection System then applies Genetic Algorithms which is trained with the classification rules learned from the information collected from the network analysis done by the Intrusion Detection System. The Intrusion Detection System then uses the set of rules to classify the incoming traffic as anomalous or normal based on their pattern.

Generic Algorithm as evolutionary algorithms was successfully used in different types of IDS. Using GA returned impressive results, the best fitness value was very closely to the ideal fitness value. GA is a randomization search method often used for optimization problem. GA was successfully able to generate a model with the desired characteristics of high correct detection rate and low false positive rate for IDS. And it is used successfully in IDS to distinguish the normal action and the intruded actions. So, Mather et al., (2010) observed that clustering GAs is a promising method for the detection of malicious intrusions into computer systems.

Artificial neural network

Artificial Neural Networks (ANN) is a purely inspired method of distributed computation. It is comprised of component processing units and the connections between them. ANN have been utilized to resolve a variety of problems that are very hard to resolve by ordinary programming which is based upon certain rules. The possibility of adaptive learning in neural networks has attracted the attention of researchers. Many functions like classification pattern, anomaly detection, functions approximation, sequence recognition, clustering, filtering, blind source separation, and controls all have certain security requirements (Wang et al., 2010).

Artificial Neural Networks are form of machine learning algorithm inspired by the behaviour of biological neurons located in the brain and central nervous system. Inputs to the ANN are typically fed to the artificial neurons in one or more hidden layers, where they are weighted and processed to decide the output to the next layer. ANNs make use of a “learning rule” (often gradient descent based back-propagation of errors) that allows the set of weights and biases for the hidden layer and output layer neurons to be adaptively tuned. This self-adaptive nature means that ANNs are capable of capturing highly complex and non-linear relationships between both dependent and independent variables without prior knowledge (zhang, 2000).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

ANNs have been used in a wide variety of classification tasks across many application domains. In contrast to traditional classification methods, such as logistic regression and discriminant analysis, which require a good understanding of the underlying assumptions of the probability model of the system that produced the data, ANNs are a "black-box" technique capable of adapting to the underlying system model. This makes them particularly useful in fields such as decision support for concealed weapons detection (Rostami 2015), prediction and classification of Internet traffic and signature verification where their ability to adapt to the data, especially in high dimensional datasets, overcomes many of the difficulties in model building associated with conventional classification techniques such as decision trees and k-nearest neighbour algorithms (Hauang, 2007).

ANNs have also been used in several computer security domains, including the analysis of software design flaws (Adebiyi and Arreymbi, 2013) and computer virus detection. ANN approaches to detection of multiple types of network attacks have also been shown to be effective, though their application to the detection of shell code was not considered (wu et al., 2015).

Neural networks are a uniquely powerful tool in multiple class classification (Theodorios 2009) especially when used in applications where formal analysis would be very difficult or even impossible, such as pattern recognition, nonlinear system identification and control. Provided the neural network has been given sufficient time to train the property of generalization ensures that the network will be able to classify patterns that have never been seen before. However, the accuracy of classification problems depends on a variety of parameters, ranging from the architecture of the actual neural network to the training algorithm of choice.

Radial Basis Function Neural Network

Radial basis function (RBF) networks are a commonly used type of artificial neural network for function approximation problems. Radial basis function networks are distinguished from other neural networks due to their universal approximation and faster learning speed. An RBF network is a type of feed forward neural network composed of three layers, namely the input layer, the hidden layer and the output layer. Each of these layers has different tasks (Figueiredo., 2000). A brief description of the RBF network is presented as follows. The training of the RBF model is terminated once the calculated error reached the desired values (i.e. 0.01) or number of training iterations (i.e. 500) already was completed. An RBF network with a specific number of nodes (i.e. 10) in its hidden layer is chosen. A Gaussian function is used as the transfer function in computational units. Depending on the case, it is typically observed that the RBF network required less time to reach the end of training.

RBF neural network is a novel and effective feed-forward neural network, which has good performance of best approximation and global optimum. It has been broadly used in considerable applications, such as function approximate, classification, regression problems, prediction, signal processing, and other problems. The RBF neural network architecture has three layers composed of input layer, hidden layer, and output layer (Huang, Saratchandran and Sundararajan, 2005). The input layer is composed of input vectors. Before the input vectors are input to the network, data processing should be done, such as normalization processing. This processing also can be done in the input layer.

The hidden layer is composed of hidden neurons, the number of which is determined by the issues described. The RBF networks are different from other types of neural networks mainly in the hidden neurons. Each hidden neuron has a radial basis function which is a center symmetric nonlinear function with local distribution. The radial basis function consists of a center position and a width parameter. Once the center and width are determined, the input vectors are mapped to the hidden space by the mapping F: $R^i \rightarrow R^j$. Suppose that the input layer has I input units; the hidden layer has J radial basis functions. The output of the J_{th} hidden neuron is expressed as

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

$$h_j(X) = \varphi\left(-\frac{\|X - c_j\|}{d_j}\right) \quad j = 1, 2, \ldots, J, \tag{2.11}$$

Where X is the overall input sample as $X = (X_1, X_2, \ldots, X_N)$, and each X_N is the I -dimension input vector expressed as $X_n = (X_{n1}, X_{n2}, \ldots, X_{nI})$, C_J. and d_j are, respectively, the center and the width of the J_{th} hidden neuron, and C_J is an I -dimension vector as $C_J = (C_{J1}, C_{J2}, \ldots, C_{JI})$.‖ . ‖ . is Euclidean norm usually taking 2-norm. ϕ(.) is the radial basis function. It can take a variable of formula expression such as B-Spline RBF, thin-plate Spline RBF, Cauchy RBF, and Gaussian RBF (Zhang, Chen and Karimi, 2013). Among them the Gaussian function is the most used, as in

$$\varphi(r) = \exp\left(-\frac{r^2}{2}\right), \tag{2.12}$$

where r is the variable of radial basis function ϕ(.).

The output layer implements the mapping F: $R^J \rightarrow R^K$. K is the number of output neurons. The output function is a linear combination of the outputs of the radial basis functions through connection weights which connect the hidden layer and the output layer, which is shown as

$$y_k = \sum_{j=1}^{J} \omega_{jk} h_j(X) \quad k = 1, 2, \ldots, K, \tag{2.13}$$

Where W_{Jk} is the connection weight between the Jth hidden layer and the Kth output of the network. RBF neural networks have three layers as highlighted below:

i. Input layer: There is one neuron in the input layer for each predictor variable. In the case of categorical variables, N-1 neurons are used where N is the number of categories. The input neurons (or processing before the input layer) standardizes the range of the values by subtracting the median and dividing by the interquartile range. The input neurons then feed the values to each of the neurons in the hidden layer.

ii. Hidden layer: This layer has a variable number of neurons (the optimal number is determined by the training process). Each neuron consists of a radial basis function centered on a point with as many dimensions as there are predictor variables. The spread (radius) of the RBF function may be different for each dimension. The centers and spreads are determined by the training process. When presented with the x vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function to this distance using the spread values. The resulting value is passed to the the summation layer.

iii. Summation layer: The value coming out of a neuron in the hidden layer is multiplied by a weight associated with the neuron (W1, W2, ...,Wn) and passed to the summation which adds up the weighted values and presents this sum as the output of the network. Figure 2.1 displayed Radia Basis Function Neural Network Architecture.

Training RBF Networks

The following parameters are determined by the training process:

1. The number of neurons in the hidden layer.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

2. The coordinates of the center of each hidden-layer RBF function.

3. The radius (spread) of each RBF function in each dimension.

4. The weights applied to the RBF function outputs as they are passed to the summation layer.

Advantages of using RBF Neural Network

1. Training in RBNN is faster.
2. We can easily interpret what is the meaning / function of the each node in hidden layer of the RBNN.
3. Radial basis function (RBF) networks have good generalization
4. Radial basis function (RBF) networks have strong tolerance to input noise
5. RBF networks have online learning ability. The properties of RBF networks make it very suitable to design flexible control systems. Classification will take more time in RBNN than MLP.

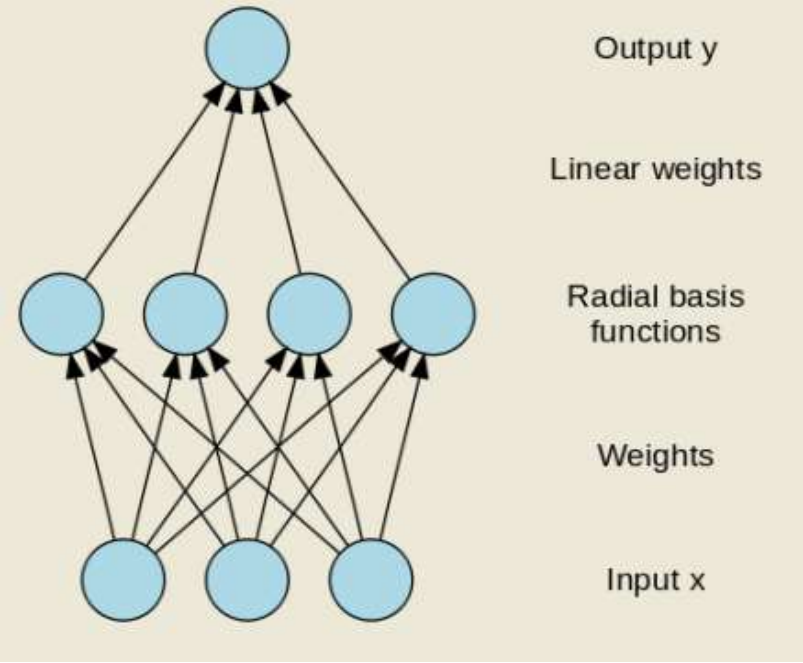


Figure 2.1: Radia Basis Function Neural Network (Zhang, Chen and Karimi, 2013).

Decision Tree

Decision tree induction is one of the classification algorithms in data mining. The Classification algorithm is inductively learned to construct a model from the pre-classified dataset. Each data item is defined by values of the attributes. Classification may be viewed as mapping from a set of attributes to a particular class. The Decision tree classifies the given data item using the values of its attributes. The decision tree is initially constructed from a set of pre-classified data. The main approach is to select the attributes, which best divides the data items into their classes. According to the values of these attributes the data items are partitioned. This process is recursively applied to each partitioned subset of the data items. The process terminates when all the data items in current subset belongs to the same class. A node of a decision tree specifies an attribute by which the data is to be partitioned. Each node has a number of edges, which are labeled according to a possible value of the attribute in the parent node. An edge connects either two nodes or a node and a leaf. Leaves are labeled with a decision value for categorization of the data (Ameer and Chafika, 2015).

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Induction of the decision tree uses the training data, which is described in terms of the attributes. The main problem here is deciding the attribute, which will best partition the data into various classes. The ID3 algorithm (Quinlan 2006) uses the information theoretic approach to solve this problem. Information theory uses the concept of entropy, which measures the impurity of a data items. The value of entropy is small when the class distribution is uneven, that is when all the data items belong to one class. The entropy value is higher when the class distribution is more even, that is when the data items have more classes. Information gain is a measure on the utility of each attribute in classifying the data items. It is measured using the entropy value. Information gain measures the decrease of the weighted average impurity (entropy) of the attributes compared with the impurity of the complete set of data items. Therefore, the attributes with the largest information gain are considered as the most useful for classifying the data items.

Decision trees as intrusion detection model

Intrusion detection can be considered as classification problem where each connection or user is identified either as one of the attack types or normal based on some existing data. Decision trees can solve this classification problem of intrusion detection as they learn the model from the data set and can classify the new data item into one of the classes specified in the data set. Decision trees can be used as a misuse intrusion detection as they can learn a model based on the training data and can predict the future data as one of the attack types or normal based on the learned model. Decision trees work well with large data sets. This is important as large amounts of data flow across computer networks. The high performance of Decision trees makes them useful in real-time intrusion detection. Decision trees construct easily interpretable models, which is useful for a security officer to inspect and edit. These models can also be used in the rule-based models with minimum processing. Generalization accuracy of decision trees is another useful property for intrusion detection model. There will always be some new attacks on the system which are small variations of known attacks after the intrusion detection models are built. The ability to detect these new intrusions is possible due to the generalization accuracy of decision trees.

Ripple down Rule Technique

Ripple down Rule is a knowledge acquisition and representation methodology proposed by Compton and Jansen (Gaikward and Thool, 2014). The knowledge is rapidly acquired and maintained by the domain expert using Ripple Down rule. The knowledge is learned based on the current context and is added incrementally. It is an interesting representation scheme for knowledge acquisition. It consists of a data structure and knowledge acquisition scenarios. The data structure is used to store human expert's knowledge in coded form which is called as rule. A ripple down rule is a list of rules, each of which may be connected to another ripple down rule, specifying exceptions. An expert can create a new rule and insert in list, which only have effects in the given context of the parent rule. The ripple down rule technique creates a two-way dependency relation between rules such that rule activation is investigated only in the context of other rule activation. Thus, ripple down rules form a binary decision tree that differs from standard decision trees in which all decisions are made at root nodes. Firstly, the default rules are generated and then the exceptions are generated by the default rule with the least error rate. It does an expansion of exception like a tree (Gaikward and Thool, 2014).

A set of rules that predict classes other than the default is called as exception rules. A set of rules that forecast classes other than the default is called as exception rules. An incremental reduced error pruning (IREP) rule is used to find exceptions with the smallest error rate, finding the best exceptions for each exception and iteration. The most excellent exceptions are created by each exception produces the tree-like expansion of exceptions. In this rule learning, rules are never adapted or removed but they are locally repaired. The approach is based on the combined use of cases to motivate knowledge acquisition and validation. The knowledge is acquired as making rules and structured into decision lists of exceptions. The rules are easily managed in the database because they are never deleted or adapted but are locally patched (Juvonen and

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Sipola, 2013). New rules are exceptions to previous rules and the new rule is endorsed in the context of previously seen cases (Priyanka, 2009). Ordered list of rules and their exceptions are formed and the first condition that is satisfied fires the corresponding rule. A main difficulty in revising a rule set is to confirm that any changes leave the rule set inconsistent. Normally, we don't want changes to break previously working rules.

The general format of rule of Ripple Down Rule learner is shown:

If condition then conclusion except

If.....

Else if ….

For instance: If ‘a’ and ‘b’ then’ c’ except if ‘d’ then ‘e’ else if ‘f’ and ‘g’ then ‘h’. This is interpreted as "if ‘a’ and ‘b’ are true then we conclude ‘c’ unless ‘d’ is true. In that case, we conclude e.

Following are some outputs of Ripple down Rule under different conditions:

If a and b then

If d then e else c

Else if f and g then h (Juvonen and Sipola, 2013).

Crime Toolkit

Cybercrime is a crime in which a computer is the object of the crime or is used as a tool to commit an offense (child pornography) most especially with the use of web application. Cybercrime continue to grow based on the ready availability of crime-ware toolkits which present challenges in the detection and prevention of malware. These crime-ware toolkits, which required in-depth knowledge in both computer system and hardware have common interest in targeting victims including individual’s business and financial institutions. Cybercriminals are taking advantages of the users of computer by compromising web applications and the user's browser to gain advantages such as using the computer's resources for launching further attacks or stealing data such as identifying information. Crimeware toolkits are being developed to attack an increasing number of applications and can now be deployed by attackers with little technical knowledge (Ammar, 2013). Methods that are used in crime toolkits are presented in this section. Development and current trend of crime toolkits are studied and reveal the methods that have been used to commit cybercrime successfully.

Types of Financial Cybercrime

The following are different types of financial cybercrime in our society:

1. Credit-card fraud
2. Bank fraud
3. Insurance fraud
4. Tax fraud

Credit Card Fraud

Rule based system for credit card fraud:

1. If number of transactions increased rapidly within short time then the transaction is Suspicious

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

2. If current transaction amount is very much greater than average transaction amount and income range is medium then transaction is Suspicious
3. If purchase of same product of luxury category within short time then transaction is Suspicious

Detection Technique: When the trend of purchases tends to change the present record will be out of character compared to previous consumer transactions. Frequency Monetary and Recency (FMR) techniques can be examined and employed. Time sequence accumulated-risk scores may be used as an input to aggregated risk exposure. A change in location may indicate a ring operation, suppose the user performs the transaction in Ogbomoso at 9am by providing his details and same users details are used in Lagos by 9:30am, such transaction may raise a suspicious alarm. There are a number of leads that relate specifically to credit card and debit card fraud. They are common points-of purchase (CPP) detection, particularly with regard to new merchant agents. The main method of detection is to look for outliers and changes in the normal patterns of usage.

Bank Fraud

Banks remain the favorite target of skilled cybercriminals. This has been true for more than a decade. Cybercrime imposes a heavy cost on financial institutions as they struggle to combat fraud and outright theft (James, 2018)

Detection Technique: The method of detection relies on out-of-pattern transactions or anomalous account use. As with other financial crimes, detection must occur before any loss is sustained. There are lead indicators like the "manipulation of credit" described above and in the lack of references, high associations of matching attributes, and dubious acceptance criteria.

The critical factors for detecting all of these financial fraud crimes is knowing the behavior of credit, bank, and loan accounts and developing an understanding of the categories of customers. Data mining can be used to spot outliers or account usages that are normal and out of character. Sometimes the account seems "too good to be true," and it often is. The absence of telephone numbers or other contact information may indicate a "ring." These rings enable fraudulent activities to be distanced from their sources and add complexity to criminal detection. Another clue is the multiple use of the same address or phone number for different accounts.

Insurance Fraud

Insurance fraud costs companies billions of dollars per year across the globe, making it imperative that insurers take a proactive stance against fraud. Insurance companies should establish a technology framework, tap into advanced automation and analytics, and take steps to prevent it. Because you know that knee injury can be given more risk points than hand injury. When an individual tries to make an insurance claim for an incident that never occurred this is considered a fraud crime. Likewise, grossly misstating the amount of damage in order to receive more money also constitutes fraud. Insurance fraud can happen with nearly any type of policy, including workers compensation to automobile coverage (Kou et al., 2004).

In the insurance industry, there are various methods by which carriers attempt to review for fraud while processing policy claims. The following are some important data attributes for detecting potential fraud claims:

i. Duration of illness
ii. Net amount cost
iii. Illness (disease)
iv. Claimant sex
v. Claimant age
vi. Claim cost
vii. Hospital

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Rule based system for insurance fraud

1. If the attribute listed above are not correlating with the supplied information then the claim is suspicious
2. If the net amount of the claim is too large compared to the average amount of similar claims then the claim is suspicious

The following are some examples of insurance fraud

(1) False claim
(2) Illegal billing

False claim

Detection Technique: Depending on the insurance carrier, various methods can be used in other to identify false claims, the method include: red-flag reviews by fraud specialists, both on-line and offline (physical display). Insurance carrier may also use an expert system which is a rule-based program that codifies the rules of a human reviewer. Data mining tools such as neural networks may be used for training and detection if samples of fraud cases exist. When the net amount of the claim may be too large compared to the average amount of similar claims.

Illegal billing

Detection Technique: The methods used for false claims can also be adopted. In addition, insurance carrier may use models and rules developed coupled with those from data mining analyses, such as decision trees or rule generators to detect these type of insurance frauds (Kou et al.,2004)

Tax fraud

Tax fraud is the intentional act of lying on a tax return form with intent to lower one's tax liability. The system can demonstrate the ability of the model to identify under-reporting taxpayers on real tax payment declarations, reducing the number of potential fraudulent tax payers to audit.

Effect of Cybercrime in Nigeria

Individuals and businesses can suffer significant financial loss because of cybercrime with the most obvious impact being theft. Loss of business can also be significant in the instance of a denial of service attacks for large corporations. The role of the internet in the development of a nation‟s economy has well been established. Accordingly, Ehimen and Bola (2009) argued that the internet has created a geometric growth and accelerated windows of opportunities for businesses and the removal of economic barriers hitherto faced by nations of the world.

Considering these limitless advantages of the internet, one can easily subscribe to the fact that it is an important tool for national development in a developing country like Nigeria. However, the internet based cybercrime has become a huge menace threatening the socio-economic and technological advancement of Nigeria. Shehu (2014) maintains that socially, cybercrime activities such as cyber stalking, harassment, and blackmail and cyber terrorism are a menace to individual's right to privacy and fundamental freedom. Similarly, cybercrimes like pornography, child predation, online gambling, online prostitution and so on undermines morality in society and puts the society at risks of breakdown of social norms and values. Cybercrime has thrown up emergency millionaires or billionaires in Nigeria which is injurious to the socioeconomic development of the country as most of such funds acquired illegally are not been used productively to promote the economy of the nation. The crime has diverted the attention of so many Nigerian youth from undertaking productive activities such as

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

manufacturing, construction as well as large scale farming that would have grown the economy to such criminal activities because of the flamboyant life chances and life styles that it presents (Folashade and Abimbola, 2013).

The prevalence of cybercrime has also created a bad image for Nigeria amongst the committee of nations as one of the most corrupt nations in the world. This tarnished national image affects the way Nigerians are treated abroad with suspicion and extreme caution as Nigerians are stereotyped to be scammer and hence not to be trusted (Folashade and Abimbola, 2013). Private companies around the world are beginning to take steps geared towards blocking e-mail originating from the country and financial instrument are accepted with extreme caution. Foreign investors are scared of the country, considering it as risky and unattractive business zone (Cohen and Felson, 2009).

Cybercrime also has an implication in the socioeconomic advancement of the country as information flowing from the country is been characterized as questionable because of the criminal element that make it unreliable, inaccurate and untrustworthy, (Iwarimie, 2010). Indeed one cannot overemphasize the economic sabotage resulting from cybercrime in Nigeria. According to Saulawa and Abubakar, (2014), in 2012, an estimated $1 trillion was lost to cyber-related frauds globally although only $390 billion was reported for obvious reasons and only recently a report by the South African based Institute of Digital Communication indicates that Nigeria is losing about $ 80 million dollars yearly to software piracy alone.

Similarly, Sesan and Oluwafemi (2012) reported that in 2012 alone, an estimated customer loss of N2,146,666,345,014.75($13,547,910,034.80) was incurred to cybercrime in Nigeria. In highlighting the consequences of cybercrime in Nigeria, Folashade and Abimbola, (2013) posited that cybercrime hinders the socio-economic development of the country as it engenders lack of trust and confidence in profitable transactions, promotes denial of innocent Nigerians opportunities abroad and causes loss of employment and revenue loss.

Also, the findings of the study by Maitanmi et al (2013) showed that cybercrime impedes socio-economic development in Nigeria as it scares away foreign investors due to the low level of confidence it has created for the Nigerian economy. The findings also showed that cybercrime has aided other illicit activities in Nigeria such as intellectual plagiarism, disruption of public services, drug trafficking, and terrorism. From the foregoing, the effect of cybercrime continues to be evident with many potential negative impacts on the socio-economic development of Nigeria.

Summarily, the socio-economic impacts of cybercrime include the following:

i. Widespread cybercrime has tarnished the image of Nigeria in the international community thereby making the country unsafe for foreign investors
ii. Cybercrime has negatively impacted on confidence Nigerians have on the digital economy thus inhibiting economic growth
iii. Cyber-attacks against businesses and organizations has the ability to damage organizational reputation and result in a loss of customers and revenue;
iv. The need to develop measures to combat and respond to cyber-attacks imposes significant costs on businesses and organizations;
v. Financial losses accrued by consumers and businesses resulting from the theft of information and money, or extortion impedes economic growth
vi. Nigeria critical infrastructure may be targeted by cyber-attacks and this can lead to immediate and long term economic losses.
vii. Cybercrime has the potentials to fuel other criminal activities and increases cost in time and resources for law enforcement agencies.
viii. It can lead to loss of business assets and cost of government agencies and business in re-establishing credits histories, accounts and identities

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

ix. Loss of personal financial resources and the subsequent emotional damage

Consequences of Cybercrime on Nigeria Economy

Criminals take advantage of technology in many different ways. The internet, in particular is a great tool for scammers and other miscreants, since it allows them to ply their trade while hiding behind a shield of digital anonymity. Cybercrime affects society in a number of different ways both online and in the offline world.

Life is about a mix of good and evil. So is the "Internet". Despite its advantages the internet has its dark ides too. Cybercrime is not without costs and these costs increase daily. These losses manifest themselves in various ways such as loss of life, loss of dignity and loss of employment. The impact cybercrime is not only limited to the victims, it spreads its impact to the society as a whole. The following are some of the consequences cybercrime has on the economy. These are: loss of revenue, waste time, Damaged Reputations, Reduced Productivity,

Loss of Revenue

One of the main effects of cyber-crime on an economy is the loss of revenue. For example, in financial institution or multinational companies' loss of revenue can be caused by an outside party who obtains sensitive financial information and using it to withdraw funds. It can also occur when a business's e-commerce site becomes compromised while inoperable, valuable income is lost when consumers are unable to use the site. It is also applicable to the individual.

Wasted time

Another major effect or consequence of cyber-crime is the time that is wasted when IT personnel must devote great portions of their day handling such incidences. Rather than working on productive measures for an organization, many IT staff members spend a large percentage of their time handling security breaches and other problems associated with cyber-crime. Waste of time could also be viewed when international banks often delay Nigerian financial transactions, pending proper verification.

Piracy

The cybercrime of piracy has had major effects on the entertainment, music and software industries. Claims of damages are hard to estimate and even harder to verify, with estimates ranging widely from hundreds of millions to hundreds of billions of dollars per year. In response, copyright holders have lobbied for stricter laws against intellectual property theft, resulting in laws like the Digital Millennium Copyright Act. These laws allow copyright holders to target file sharers and sue them for large sums of money to counteract the financial damage of their activities online.

Damaged reputations

In cases where customer records are compromised by a security breach associated with cyber-crime, a company's reputation can take a major hit. Customers whose credit cards or other financial data become intercepted by hackers or other infiltrators lose confidence in an organization and often begin taking their business elsewhere. Due to these, foreign investors often consider Nigeria as an unattractive market and it has detrimental impact on Nigerian citizenry when they travel outside to do business.

Reduced productivity

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Due to the measures that many companies must implement to counteract cyber-crime, there is often a negative effect on employees' productivity. This is because, due to security measures, employees must enter more passwords and perform other time-consuming acts in order to do their jobs. Every second wasted performing this task is a second not spent working in a productive manner.

The impacts of cybercrime already have detrimental influence on Nigerian economy. Global anticorruption bodies, such as transparency international have listed Nigeria as one of the most corrupt countries in the world (Oyesanya, 2004). Private companies around the world are beginning to take steps geared at blocking email traffic originating from Nigeria. Nigerian financial instruments are now accepted with extreme due diligence around the world and some international banks have completely denied access to their web sites if the traffic originates from Nigeria. Cybercrime might also hinder technological development within Nigeria.

Identity theft

Becoming the victim of cybercrime can have long-lasting effects on your life. One common technique scammers employ is phishing, sending false emails purporting to come from a bank or other financial institution requesting personal information. If you hand over this information, it can allow the criminal to access your bank and credit accounts, as well as open new accounts and destroy your credit rating. This type of damage can take months or even years to fix, so protecting your personal information online is an important skill to learn.

Sales fraud and forgery

In our society today, fraudulent sales of products that do not exist or that are replicas are increasingly common. The purchase of an item before actually seeing it has created ways for fraudsters to make money via the sale of unoriginal products or in some cases, the total absence of the product. Many persons have fallen victim of these particular crimes on popular e-commerce websites.

Related Works

Lebbe et al., (2007) analyzed various Artificial Immune System (AIS) models used in Intrusion detection system (IDSs) and introduced Danger Theory in AIS as a method for danger response in wireless mesh networks. For classification of network dangers, they used Self-organizing Maps (SOMs) as classifiers. The experiments validated their proposal of applying Danger Theory to security of wireless mesh networks. It is deduced that the model works only on the test or fraud attack.

Garcial-teodoro et al. (2008) studied how internet and computer networks are exposed to an increasing number of security threats. With new types of attacks appearing continually, developing flexible and adaptive security-oriented approaches is a severe challenge. In this context, anomaly-based network intrusion detection techniques are a valuable technology to protect target systems and networks against malicious activities. However, despite the variety of such methods described in the literature in recent years, security tools incorporating anomaly detection functionalities are just starting to appear, and several important problems remain to be solved. The researcher began with a review of the most well-known anomaly-based intrusion detection techniques. Then, available platforms, systems under development and research projects in the area were presented. Finally, the main challenges were outlined to be dealt with for the wide scale deployment of anomaly-based intrusion detectors with special emphasis on assessment issues. The solution here result to low detection response time because it is not scalable.

Hausken (2008) also used a strategic reliability approach to anomaly detection. His work recommends different techniques, depending on the type of the network. He uses Markov analysis to repeated games and studies the strategic defense of a system, which has been targeted by multiple attackers. Hausken took into account the essential dissimilarities of the network elements and considered several parallels and complicated series of

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

defensive techniques. Ghose and Hausken (2007) described the optimality process of a possible interaction between the hackers, depending on the information structure and availability. They consider a process of building a shared information platform among the hackers so that the hackers can be aware of the level of vulnerabilities among the different protected systems. That implies that the weakest systems were targeted and there are better chances of a successful attack on the protected system.

Gianini et al. (2009) proposed an extension of Anomaly detection model for computer system security to ambient intelligence domain. The extended model can provide perceptual functions and detection capabilities with device intelligence such as multimedia sensor system interpretation also studied the benefits of AI in general for improving Intrusion detection system by investigating different IDS designs based on Artificial intelligent. The results showed how Artificial intelligent approach to IDS design can be fruitful for future applications. The work is limited to the used of sensing devices to secure the system from intrusion.

Zhang et al. (2011) proposed SGDIDS a new hierarchical Distributed Intrusion Detection System for improving cyber security of the Smart Grid. The system consists of an intelligent module (among other modules) which uses AIS to detect and classify malicious data and possible cyber-attacks. Simulation results showed that the system is applicable to identification of malicious network traffic and improving system security also enhanced the anomaly detection based on AIS and showed how the model improves AIS performance in applications such as anomaly detection, ensuring security, detecting errors and performing data mining in mobile ad hoc networks. The limitation of the proposed SGDIDS is that it identifies malicious network traffic rather than detecting the anomaly behaviours posed.

Asmaa and sharad (2011) discussed differences between Intrusion Detection System and Intrusion Prevention System (IDS/IPS) technology in computer networks. IDS and IPS systems stability, performance and accuracy were used as metrics in comparing the IDS and IPS. Intrusion types of systems are put in place to serve a business needs for meeting an objective of network security. The IDS and IPS are to provide a foundation of technology meets to tracking, identifying network attacks to which detect through logs of IDS systems and prevent an action through IPS systems. If the host with critical systems, confidential data and strict compliance regulations, then it is great to use of IDS, IPS or both in network environments. IPS and IDS both examine traffic looking for attacks but they are critically different. The differences between deployment of these system in networks in which IDS are out of band in system means it cannot sit within the network path but IPS are in-line in the system, means it can pass through in between the devices. IDS generates only alerts if anomaly traffic passes in network traffic, it would be false positive or false negative, means IDS detects only malicious activities but no action taken on those activities but IPS has feature of detection and prevention with auto or manual action taken on those detected malicious activities like drop or block or terminate the connections also evaluate some techniques such as genetic algorithm and scatter search to score each transaction and based on these scores the transaction can be classified as fraudulent or genuine transaction and these approaches are based on the classification problem. The system proposed by Asmaa and Sharad generate a large number of false alarm rate.

Garcia et al. (2012) proposed an off-line solution to mine client log files to identify the source of a security breach. Given a security incident has already been detected, and a set of client log files, their system will attempt to locate the exploit in one of the log files. Postmortem intrusion detection is primarily used to discover how an intruder gained access to a system, what subsystems were accessed, and what information was compromised. The solution assumes that a security breach has already occurred and bypassed the Intrusion Detection System or any other security controls in place. This solution used a combination of anomaly detection and a classification technique called KHMM which utilizes a Hidden Markov Model (HMM) and k-means clustering.

The main idea around the work is that an attack would result in a sequence of system calls being logged that would not normally appear in normal activity. Normal log data is used to create a normal behavior profile. First,

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

the log files are shrunk by replacing repetitive sequences with a meta-symbol. The log files are then pre-processed using a sliding window containing one hundred elements, stepping through the log file one hundred elements at a time. The last step builds the normal activity model from vector sequences in each window. The resulting model is used for detection. The KHMM process is composed of three steps. First, the preprocessed input is clustered using K-means. Then the sliding window approach is used to create an HMM for each window. The last step uses an anomaly detection to compare each window with the average HMM from the previous step. If two or more consecutive abnormal windows are detected, they are marked for verification by a security analyst. A major shortcoming of the solution proposed by Garcia et al. (2012) is that it does not detect intrusions; instead, it attempted to locate abnormal activity in a collection of client log files after a security breach has already been deemed to have occurred. Secondly, the solution can be only implemented in an off-line manner because it is not linearly scalable.

Maitanmi (2013) investigated the level of awareness of individuals on the existing phenomenon in Nigeria and the impacts of cybercrime on Nigerian economy. Questionnaire was used as an instrument in gathering the information. A lot of people in the world, mostly Nigerian have a limited knowledge of the crime occurring in cyberspace, known as cybercrime. Cybercrime happens in the world of computer and the Internet. This kind of crime has a severe impact on our economy, lives and society, because our society is becoming an information society, full of information exchange that is happening in cyberspace. The work is aimed at knowing the level of awareness of individuals on the existing phenomenon in Nigeria, and their impacts on Nigerian economy. A survey was carried out with the aims of getting these results using questionnaire as an instrument, the responses were quantitatively analyzed using some statistical techniques. The results show that cracking, software piracy, and pornography among others are prevalent crimes in Nigeria. While the impacts of these crimes on Nigerian economy cannot be over emphasized. Recommendations were proposed on how these crimes can be minimized if not totally eradicated.

Kim (2013) proposed two methods in detecting fraudulent acts based on the finite fixture model to detect fraudulent items automatically without the requirement of labeled items and modeled the dependency between the description and the price of an item by considering the possible combinations of the item description and price clusters according to item clusters and a real-world dataset to evaluate the effectiveness of the proposed models and compared them to existing outlier detection methods so the proposed model significantly identified the fraudulent transactions. The model proposed by Kim is limited to the numbers of trained dataset.

Lee (2013) adopted an algorithm called online oversampling principal component analysis algorithm (OSPCA) to solve real world applications problem such as intrusion detection or credit card fraud detection and the aim of the author was to detect the presence of outliers from a large amount of data via an online updating technique.

Okoye and Gbegi (2013) introduced an evaluation technique for the effect of fraud and related crimes in Nigerian economy. The work is to determine the impact of fraud and related financial crimes on the growth and development of Nigerian economy. Data for the study were collected from secondary sources only. The research analyzed the data generated using regression analysis. The research findings revealed that, fraud and related financial crime has significant effect on the Nigerian economy while fraud and financial crime have no significant effect on inflation. The research therefore recommends that Auditors and Accountants in organizations and financial institutions should be trained on how to carry out forensic investigation since the fraudsters are now sophisticated in their act. Also internal control systems should be strengthened to block opportunities that attract fraud perpetrators and oversight function of the National Assembly be strengthened to make public office holders accountable.

Razzaq and Abdul (2014) proposed a solution for detecting web application attacks by analyzing HTTP requests. The proposed solution was deployed as a web proxy that evaluates all network traffic before it is delivered to the web server. Even though the solution only analyzes the HTTP protocol and claimed it could be expanded to other

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

protocols. Additionally, the solution only examined portions of the headers and payload of user requests. They developed an ontology model (OWL) to build rules to analyze the user request to detect web application attacks, such as SQL Injection, DNS Cache poisoning attack and HTTP response splitting attacks. These rules are applied to all user requests by analyzing portions of the HTTP traffic before being processed by the web server. Test attack vectors consisted of SQL Injection Cross Site Scripting (XSS) attacks using an open source tool called Web Goat to simulate the attack vectors. The technique used detected web application attacks with an average detection rate of 86%. The detection rate is calculated using the total number of attack records and the number of false negatives (FN). A false negative is an attack vector that is classified as normal.

Ng and Banik (2015) proposed an offline solution to detect Denial of Service (DoS) and brute force password attacks. The technique implements both anomaly detection and signature recognition methods. Attack signature database was maintained as well as a normal signature database. A Clustering algorithm was used on pre-processed log data to identify multiple occurrences of similar log messages. The tool used searches the signature databases by using log patterns detected while processing the log data. When the clustering algorithm detects an unusual number of event occurrences the signature is compared to the normal log database and is ignored if found. If the signature is found in the existing attack signature database then an alarm is generated. However, if the signature is not found in either signature database then it is presented to the user for manual classification. The initial log data was obtained from one host running the Ubuntu operating system. Attack log data was obtained by performing ICMP flood and brute force attacks against the host. The limitation of the developed system is that, it generate a large number of false alarm rate.

A set of normal and attack patterns obtained from the initial data collection is stored in the signature database. A real-time intrusion detection system was identifies as potential future work. The primary shortcoming of the solution developed by Ng et al. (2015) was that it depends on a single client log file source from one platform (Ubuntu). Additionally, it does not differentiate between events that have occurred recently or far in the past. Since the solution maintained a database of all normal activity patterns; it can only be implemented as an off-line solution. As such, it is not linearly scalable, and cannot detect suspicious user activity in real-time at an enterprise scale.

Parvinder and Mandeep (2015) designed a fraud detection system that monitored customers behavior and activities using unsupervised learning approach; the author aimed to study the customer log and perform analysis based on some assumption and laid down rules.

Jyothsna et al. (2016) proposed a meta-heuristic statistical method to detect intrusion in network. Phenomenal growth in the volume of computer network users leads to the drastic divergence of anomaly activities. Henceforth, it is quite obvious to consider the associability between network transactions and the feature involved to form those transactions. In this regard, the majority of current research is involved to devise signature-based intrusion detection using soft computing techniques. Most of these soft computing approaches are delivering the computational complexity, which is due to magnification of number evolutions. The work presented a meta-heuristic statistical scaling process derived to estimate if a network transaction is safe, suspicious or intrusion. The proposed model employed duplex graph strategy to estimate the strong associability of the features towards network transactions. The results explored from the empirical study are successfully delivering the accuracy towards identifying the intrusive state of a network transaction. The proposed strategy is able to stabilize the computational complexity to O(n*log(n)) towards assessing the network intrusion using the scale derived.

Reazul et al. (2017) developed a network intrusion detection frame work based on Bayesian network using wrapper approach. Increasing internet usage and connectivity demands a network intrusion detection system combating cynical network attacks. Data mining technique was used by intrusion detection system to prevent the network attacks and classify the network events as either normal or attack. The study presented a wrapper approach for intrusion detection. Framework Feature selection technique eliminate the irrelevant features to

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

reduce the time complexity and build a better model to predict the result with a greater accuracy and Bayesian network works as a base classifier to predict the types of attack. The experiment shows that the proposed framework exhibits a superior performance in terms of accuracy but has high false alarm rate.

Ismaila et al. (2018) introduced soft-computing approach to anomaly detection in call profiles. The work focus on telecommunication fraud because telecommunication fraud have been a problem affecting all operators and customers and is an important factor in their annual revenue losses detection and prevention of telecommunication frauds were focused by the researcher. Here, Ismaila et al 2018 employed Counter Propagation Neural Network (CPNN) classifier to detect anomaly in subscribers' call records in telecommunication industry where eighty user's datasets were gotten from a renowned telecommunication industry. The dataset was re-processed to adapt to the neural network processing terrain, and then classified the data into normal and abnormal in an offline processing. The researcher concluded that the theft of telecommunication services has been one of the most enduring types of telecommunications crimes which has been evident since the beginning of telephone systems.

Each technological development designed to thwart criminal endeavours has been quickly followed by the creation of a new form of crime designed to exploit new security. In particular, fraud detection is important to the telecommunications industry because subscribers, companies and suppliers of telecommunications services lose a significant proportion of their subscriptions or revenues as a result. Moreover, the modelling and characterization of users' behavior in telecommunications can be used to improve network security, improve services, provide personalized applications, and optimize the operation of electronic equipment and/or communication protocols. Thus, neural network variants have become an increasing presence in major aspects of telecommunication networks improving efficiency, adapting to changing calling patterns, and providing better information about the use of networks. The limitation of the developed model is that the time taken in detection is very high.

Ismaila et al. (2019) investigated two-step approach to identify theft detection in electronic payments. The work introduced identity theft and identity fraud as terms used to represent all types of crime in which someone wrongfully obtains and uses another person's personal data in some way that involves fraud, typically for economic gain. In e-commerce, the growing number credit card transactions provides more opportunity for thieves to steel credit card numbers and subsequently commit fraud. The research used Communal detection (CD) and Counter Propagation Neural Network (CPNN) for fraud detection in identity theft in credit cards online transactions. Here three thousand credit cards transaction were simulated as dataset. The selected simulated applicants' attributes were worked on by Communal detection to produce whitelists and blacklists. The simulation of the developed system reported 88.7% for prediction accuracy and 64.8% for false positive rate. It is deduced that the system generates large number of false alarm rate.

Various techniques used in detecting online financial crime and their classifications based on several approaches has been reviewed. As mentioned in this chapter, the central idea of anomaly detection techniques is to build models of legitimate activities based on the normal data and then any unusual activities from the normal model will be considered as an attack or anomaly. A broad study of several anomalies-based methods used in detecting online financial crimes were reviewed. However, it was observed that most of the investigated online financial crime detection systems has low anomaly detection rates and high false alarms. Most of the existing systems also, can only detect and prevent individual attack and not coordinated, distributed attack. Whereas online financial crime detection system should be able to detect and prevent both coordinated and distributed attack consistently with minimal false alarms and high detection rate.

METHODOLOGY

Research Approach

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

In this work, detection and prevention of cybercrime technique was developed using Enhanced Ripple Down Rule (ERDR) technique and Neural Networks. In developed the detection and prevention system, a database of cybercrimes was created; which was used to trained the developed system.

Cybercrime detection and prevention stages

The stages involved in this developed technique are in four stages. The stages include the following:

1. Data Acquisition: datasets were gotten from LAUTECH students
2. Data pre-processing such as merging and integrating, dimensionality, reduction, treatment of missing values, insertion and data slitting will be carried out on dataset.
3. Enhanced the Ripple Down Rule system using python libraries and Radia basis function (RBF) and Neural Network.
4. Machine Learning
5. Classification stage

Overview of the developed process is shown in Figure 3.1 and described in this section.

Data acquisition is the first stage in developing the system. This stage deals with acquiring crimes from student which are gotten from an online store. In developing the technique, the data were collected from an online shopping store. Even though the store provided real credit card data for this research, it is required that the store name be kept confidential. Though real credit card transactional data were obtained, real credit card number and customer personal information were not given due to confidentiality and fraudulent transactional records available. Synthetic database was also explored in order to test the developed technique speed on large scale data.

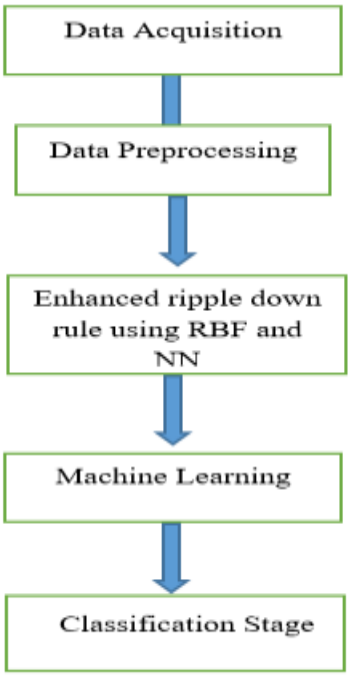


Figure 3.1: Cyber-crime detection and prevention procedure.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Enhanced Ripple Down Rule system contains the dynamic rule which is generally based on human knowledge such as human insight. If the students passes through this phase then it goes to the next stage. Machine learning technique is a key stage in the developed system; it generates dynamic rules based on past attacks. Here learning is totally dynamic; so if the pattern of cyber-attacks changes then the technique learns itself from transactions and generates dynamic rules for prediction of cybercrime. Radial Basis Function (RBF) Neural Network and Ripple Down Rule method were used. Radial Basis Function (RBF) was used to check the profiling of the student or the person that wishes to perform the cybercrimes. ERDR was used as rule which operates in real time and is very precise.

The ERDR would deny a cybercrime request only if such a request clearly represents fraudulent attempt. ERDR are IF –THEN (ELSE) rules that associate the class fraud or genuine to a transaction when specific conditions are met. An example is " IF a cybercrime is done in Ogbomoso and another transaction is performed in Italy within 5 minutes from the same student or same person THEN deny the student and raise alarm", because the individual cannot physically be in Ogbomoso and Italy over a short period of time. The last stage in the developed system is the classification of normal cybercrimes and cyberattacks.

Cyber-crime Data Acquisition

The dataset used for this research contains seven (7) cards which were labeled Card A, B, C, D, E, F and G. The card comprised four attributes and consisted of Three thousand Eight Hundreds (3800) where Two thousand eight hundred were used for training and one thousand were used for the testing. The numbers of transactions for each card were 250, 350, 450, 550, 650, 750 and 800 transactions respectively, each with fraudulent type and non-fraudulent type. Using cross-validation method, 60% transactions of each card were used to train the system and 40% transactions were used to test the system. These were subjected to Enhanced Ripple Down Rule (MRDR) and Ripple Down Rule (RDR). In this work, real time data and synthetic data were used. The real time data used was collected from an online shopping store. Gaussian distribution was used to generate synthetic data. The input to the developed system was the data collected from online shopping store and the synthetic data generated by Gaussian distribution.

The developed strategy was implemented using MATLAB R2023a software on Windows 10 64-bit operating system. The results gotten were examined at different threshold values using some selected performance metrics which are Sensitivity, Specificity, False Alarm Rate (FAR), Accuracy and Computational Time(CT).

As discussed from the literature there are many types of attack that are used by cyber criminals to affect online transactions but for the purpose of this work, network-based attacks were addressed. Network-based attacks described several kinds of attack that operate on networks and the protocols that run the networks. Network spoofing is the process in which an attacker passes themselves off as someone else. There are several ways of spoofing in the standard TCP/IP network protocol stack, including: MAC address spoofing at the data-link layer and IP spoofing at the network layer. By spoofing who they are, an attacker can pretend to be a legitimate user or can manipulate existing communications from the victim's host.

Generation of Sets of Rules using Ripple Down Rule

Ripple Down Rule (RDR) System is simple to implement and it was used to generate a set of rules from the selected attributes obtained from the dataset. The RDR consist of little or more than a set of if-then statements, but provide the basis for so-called "expert systems" which are widely used in many fields. The Ripple Down Rule System itself uses a simple technique: It starts with a rule-base which contains all of the appropriate knowledge encoded into **If-Then rules** and a working memory which may or may not initially contain any data, assertions or initially known information.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Operational rule of the designed system

This section explained how ripple down rule will be used in the designed system of cybercrime technique. RDR has a binary tree structure in which each node corresponds to a rule. Here, the root node always true by default. The default node was connected to a network of nodes, also connected to their parent nodes through either a false or true branch. Every parent node have two possible branches; the true and false branches. The parent node was connected to a child node through the branch that represents the evaluation of the parent's rule. For example, if the parent node's rule evaluates to true for a given case then the child connected to the parent through the true branch were evaluated next.

The same case was applied for a child node on the false branch if the rule evaluation returns false. Ultimately, the terminating node was reached after all its parents' nodes have been evaluated. The conclusion from the last successful node was returned as the ultimate conclusion for the case. If the last firing rule was false, then the last true ancestor's conclusion was returned as the effective conclusion. RDR's root node has a default conclusion such that if no other rules evaluate to true, the default conclusion was returned. All these explanations showed the process to follow in design cybercrime techniques with ripple down rules. This technique guarantees that a conclusion is returned every time. Figure 3.2 shows how ripple down rule was used in the design of cybercrime technique.

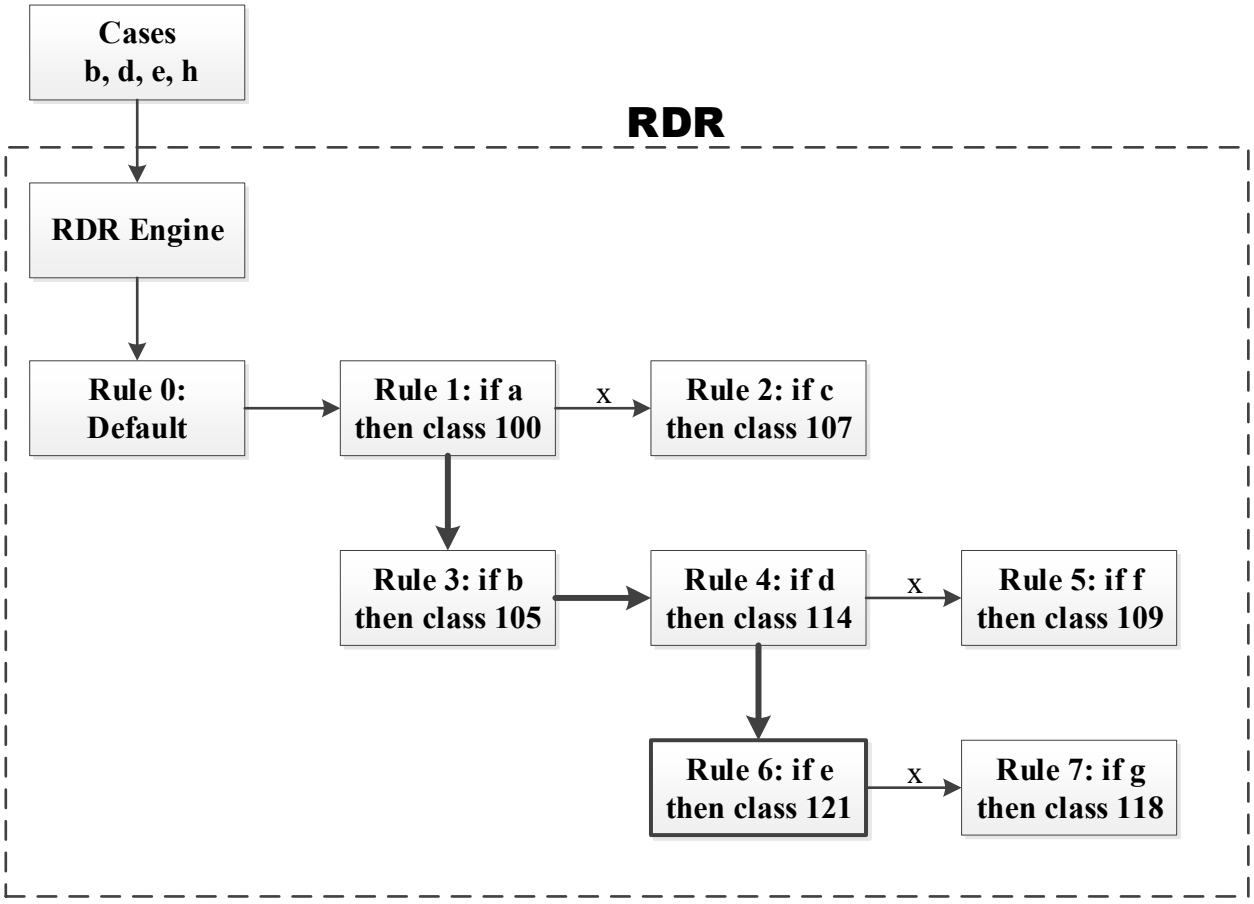


Figure 3.2: Ripple down Rules operation

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

There must be an assumption or a case for the rule to hold:

i. the spending behaviour
ii. the location of the person
iii. the time variance
iv. the devices used/ IP address

From Figure 3.2, Rule 0 is the default rule so, always return true. The first rule tested after the default node is Rule 1, this node evaluates to false since the condition does not satisfy any of the case attributes. The false branch of Rule 1 leads to Rule 3. Rule 3 evaluates to true for the current case so Rule 4 is evaluated next. Rule 4 evaluates to false and leads to Rule 6, which returns true and leads to Rule 7. Rule 7 is false so the effective conclusion is 121 from Rule 6 since this is the last tested rule that returned true.

The Modification of RDR

The operation of Ripple Down Rules (RDR) is that, after a case is processed a corresponding conclusion or result was returned. In the modification of RDR after a case is processed, a model was returned instead of a conclusion or a class. The model was called Situated Profile (SP). This SP contains profiles describing each of the attributes matching the current case. Figure 3.3(a) and Figure 3.3(b) depict the basic RDR operation and the Enhanced RDR operation respectively.

The modification of RDR (ERDR) was an improved RDR with the added capability to handle multiple classifications. The inability of Rule Based System to provide multiple classifications was seen as a limitation to its applicability in many domains. ERDR was developed to be applicable for domains where a single case may lead to multiple conclusions while still retaining the advantages of RDR. In fact, ERDR should be shown to cover a domain

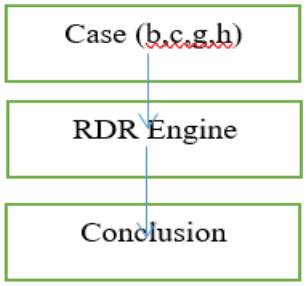


Figure 3.3(a): Basic Ripple Down Rules operation (Gaikwad and Thool, 2014)

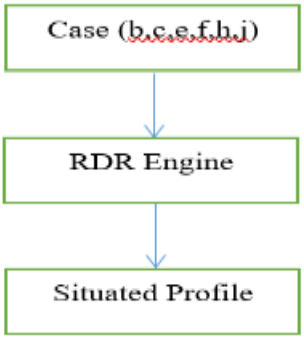


Figure 3.3(b): Enhanced Ripple Down Rules operation

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

quicker than its single class counterpart and it should be able to produce a more compact Knowledge Base with fewer redundancies than single class RDR.

One of the main differences between RDR and MRDR is that the modifying RDR structure was an K-ary tree compared to RDR's binary tree structure. A k-ary tree is a rooted tree in which each node has no more than k children. It is also sometimes known as a k-way tree, an N-ary tree, or an M-ary tree. Like RDR, MRDR has a default root node which always returns a true to guarantee a conclusion with every input. However, unlike RDR's true and false branch at each parent node, MRDR have exclusively exception (or true) branches. An MRDR parent node can have any number of exception branches and each branch will be followed when the parent node condition is true for a given case. Inferencing in MRDR is generally similar to RDR except that if a condition is false at a parent node, then no child nodes should be evaluated. Another variation between the two methods is that unlike in RDR, a case in MRDR can return multiple effective conclusions depending on how many of the terminating nodes evaluate to true for the case.

Inferencing in MRDR is such that the root node was evaluated first, and then all nodes connected to the root are tested next. The nodes that evaluate to true have their child nodes tested and the ripple continues until a terminating node is reached or until all children are false. The effective conclusions for the case were a collation of the firing terminal rules in each branch. Figure 3.4 illustrates the inferencing process in an MRDR structure. In the diagram, it is assumed the structure has the following inputs:

i. the spending behaviour
ii. the location of the person
iii. the time variance
iv. the devices used/ IP address

For the example in Figure 3.4, the root node is evaluated first, returning true by default. Rules 1, 2, and 3 are evaluated next. Rule 1 return false so its child is not checked. Rule 2 returns true so all its branches (Rules 5, 6 and 7) are checked. Rule 5 returns true and Rule 10 is checked which returned true and is a terminating rule. Rule 6 also returns true so Rule 11 is checked, which also returns true and terminates. Rule 7 returns false so its leaf node is not checked. Since Rule 3 returns true, its branch (Rule 8) is evaluated and returns true. Rule 8's child (Rule 12) is then evaluated and returns false. After all inferencing is done, the effective conclusions are "109,110 and 107". These are the last true firing nodes in each independent branch.

Representation of the parameters:

a: The spending behavoiur

b: The location of the person

c: The time in performing the crime

d: Buying costly product

e: The person was performed in Ogbomoso

f: The crime was performed in Ibadan

g: The crime was performed in London

h: The time the transaction was performed was 9am

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

i: Buying wrist watch

j: The crime was performed at Califonia in Ogbomoso

k: The crime was performed at Bodija inIbadan

l: The time the crime was performed was at the mid night

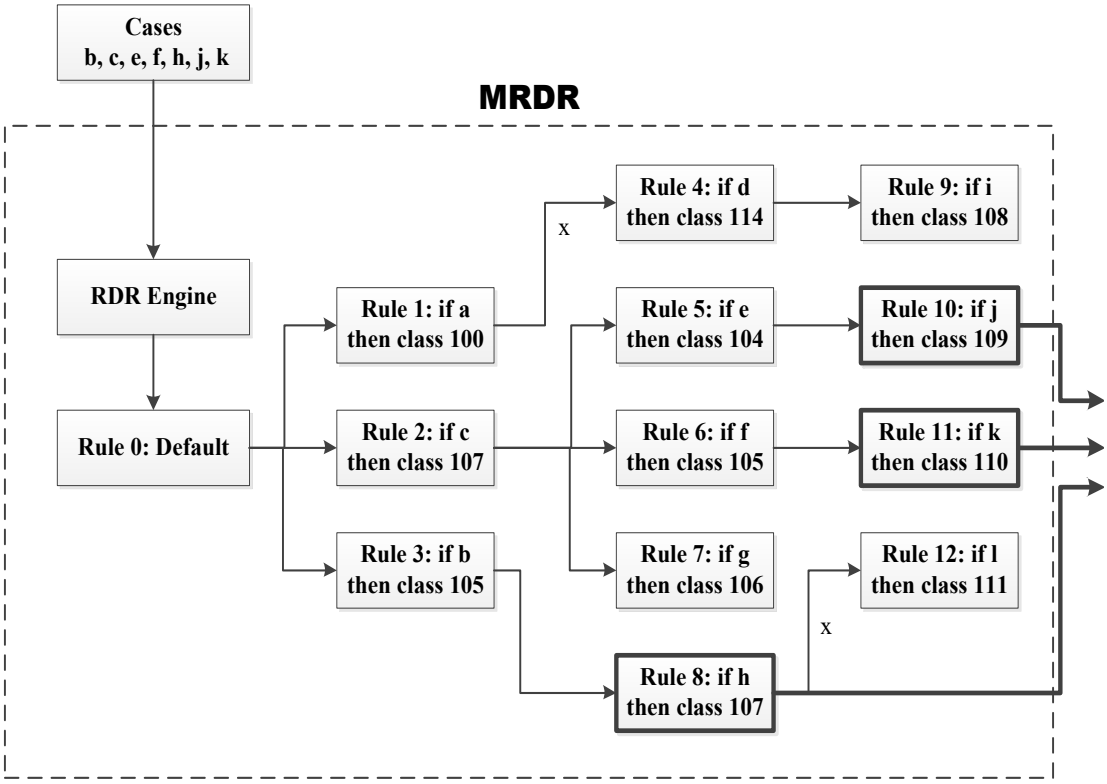


Figure 3.4: Enhanced Ripple Down Rule Structure

The assumption/rules made

The assumptions or rules made in the designed technique are as follows:

a. If the transaction is performed during the late night and no past transaction exist in late night, the transaction is considered as suspicious transaction.
b. If the user is active and performs the transaction continuously but suddenly stops performing the transaction and after some years becomes again active to perform online transaction. The transaction is considered suspicious
c. Suppose the users normally take 60 seconds to perform the transaction and suddenly the user is taking 10 minutes to perform the transaction then the transaction is also considered suspicious
d. In Nigeria not everybody demands for costly and luxury product but suddenly customer begins to purchase a costly and luxury product, the transaction is considered suspicious
e. Suppose the user performs the transaction in Nigeria at 10:00am by providing his details and same users details are used in Germany at 11:00am then there may be possibilities that the transaction is fraudulent and it has to be suspicious.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Parameter to Checkmate the Suspicious Online Transaction

The parameters to checkmate suspicious online transaction are as follows:

i. Check the spending behaviour
ii. Check the location of the person
iii. Check the time variance
iv. Check the devices used/ IP address

Classifying Suspicious/ Fraudulent Online Transaction

Radial basis function (RBF) networks was used in this research as classifier. RBR neural networks is easy to design, it has good generalization, strong tolerance to input noise, and have online learning ability. The properties of RBF networks make it very suitable to design flexible control systems. Neural Networks has been utilized to resolve a variety of problems that are very hard to resolve by ordinary programming which is based upon certain rules. The possibility of adaptive learning in neural networks has attracted the attention of the researchers. Many functions like classification pattern, anomaly detection, functions approximation, sequence recognition, clustering, filtering, blind source separation and controls all have certain security requirements. RBFNN was used in the place of data classification in this work. Radial basis function Neural Networks are a biologically inspired form of distributed computing usually comprising a set of nodes (including input, hidden and output) and weighted connections between them (Singh et al., 2017). RBFNNs can also be defined as a topology formed by organizing nodes into layers and linking the layers of neurons.

The nodes are interconnected by weighted connections, and the weights are adjusted when data is presented to the network during a training process (Wu, et al., 2015). NNs provide a mapping from the input space to the output space so can learn from the given cases and generalize the internal patterns of a given data set (Alex, David and Aladdin, 2018). They adapt the connection weights between neurons and approximate a mapping function that models the provided training data (Simone, 2019).

In a typical feed-forward process, the sum of the products of the input nodes and their weights are passed through a threshold and the result at each hidden node is multiplied by the corresponding weights (hidden-to-output node connections). The sum of each connection and hidden node product is collected at each output node, passed through a continuously differentiable non-liner function and the NN output is determined. Usually, the sigmoid function is used at the hidden and output nodes. Figure 3.5 give a detail of three-layered RBNN. The sigmoid function provides enough information about the output to earlier nodes (hidden and input) so that the weights can be adjusted accordingly to reduce the difference between the NN's calculated output and the desired target output (Alex et al., 2018). Equation 3.1 represents sigmoid function.

$$f(\text{net}) = \frac{1}{(1+e^{-k\text{net}})} \qquad (3.1)$$

From this equation K is the positive constant and it control the breadth of the function

Many of neural networks variation with Enhanced algorithms are in use today in different applications, including fraud detection. The major forms of fraud including: telecommunications fraud, financial fraud and computer intrusion fraud among others. For the purpose of this work, RBFNN was used as a secondary classifier where MRDR outputs was indexed and passed to the Radial Basis Function (RBF) NN and the output was used to substantiate the MRDR result. Figure 3.6 depicts the detection process of the online transaction, starting from where the users supply some information and to the stage of allowing the users to perform any transaction on the platform. Figure 3.7 represent the block diagram of the developed system where acquired data is inputed, the inputed data is store in a data storage where the features of the supplied data was extracted based on the expected

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

attributes. MRDR checked the features extracted if it agreed with the laid down rules. Radia basis function neural network was used to classifies the extracted features whether it is anomaly transaction or genue transaction.

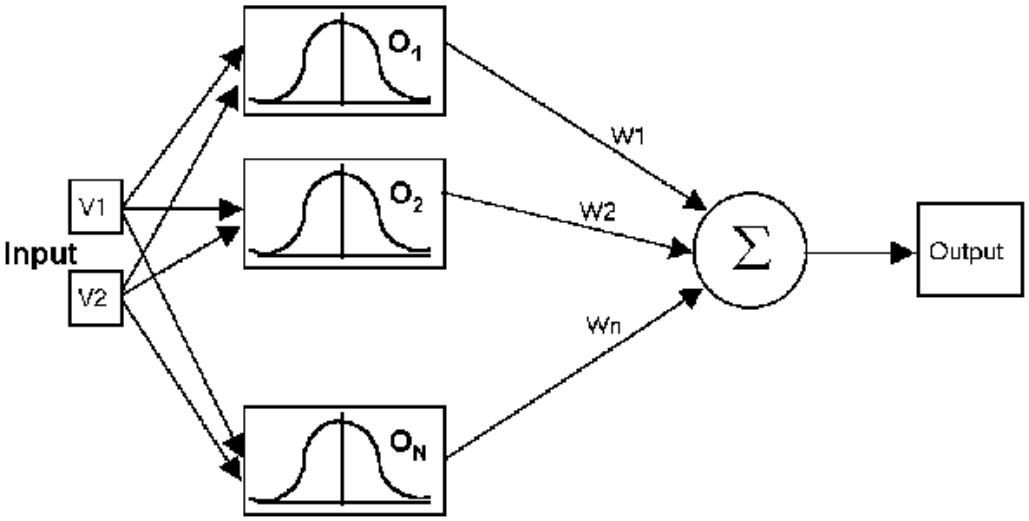


Figure 3.5: Radia Basis Function Neural Network Architecture

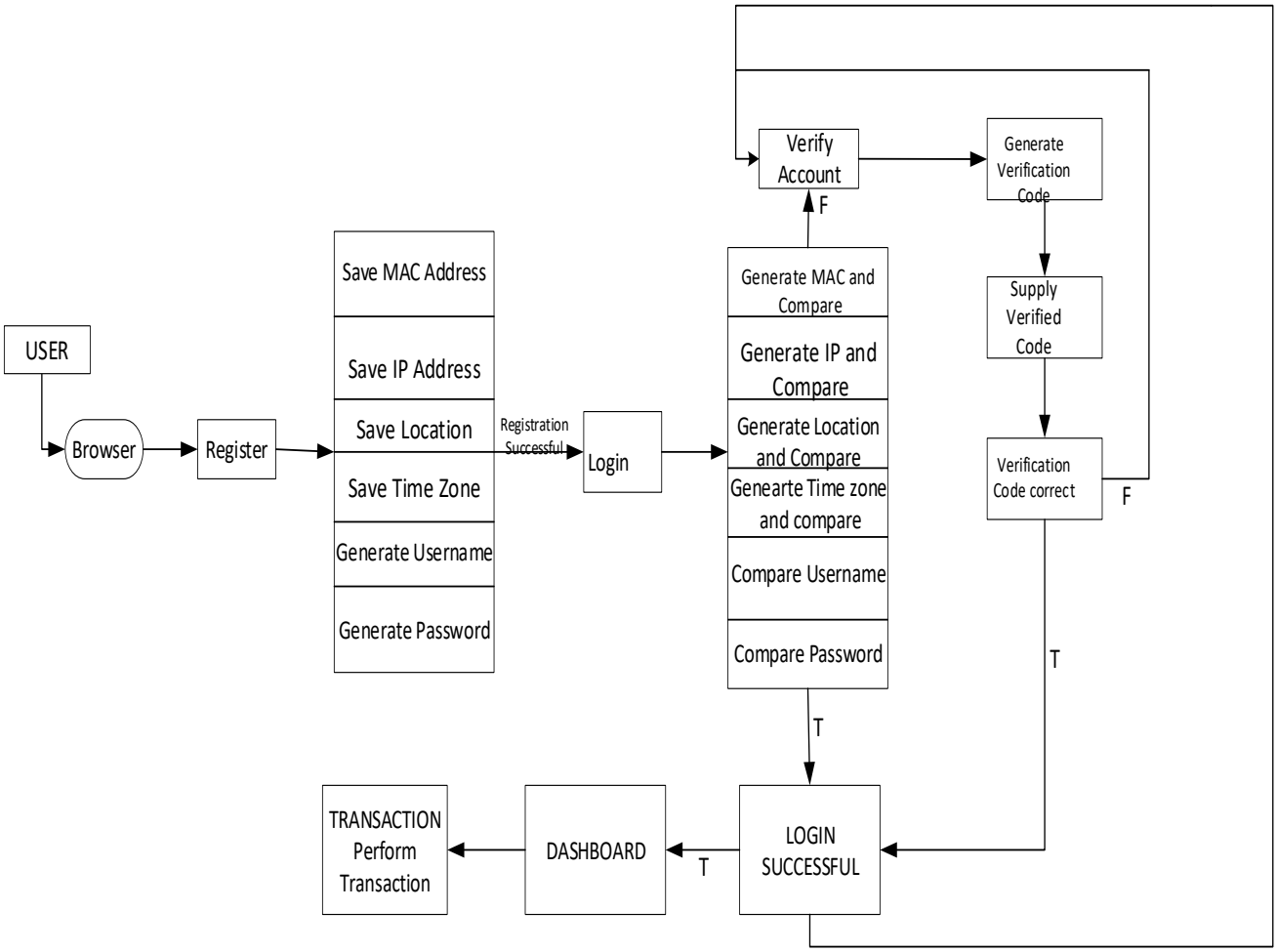


Figure 3.6: Detection processes

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

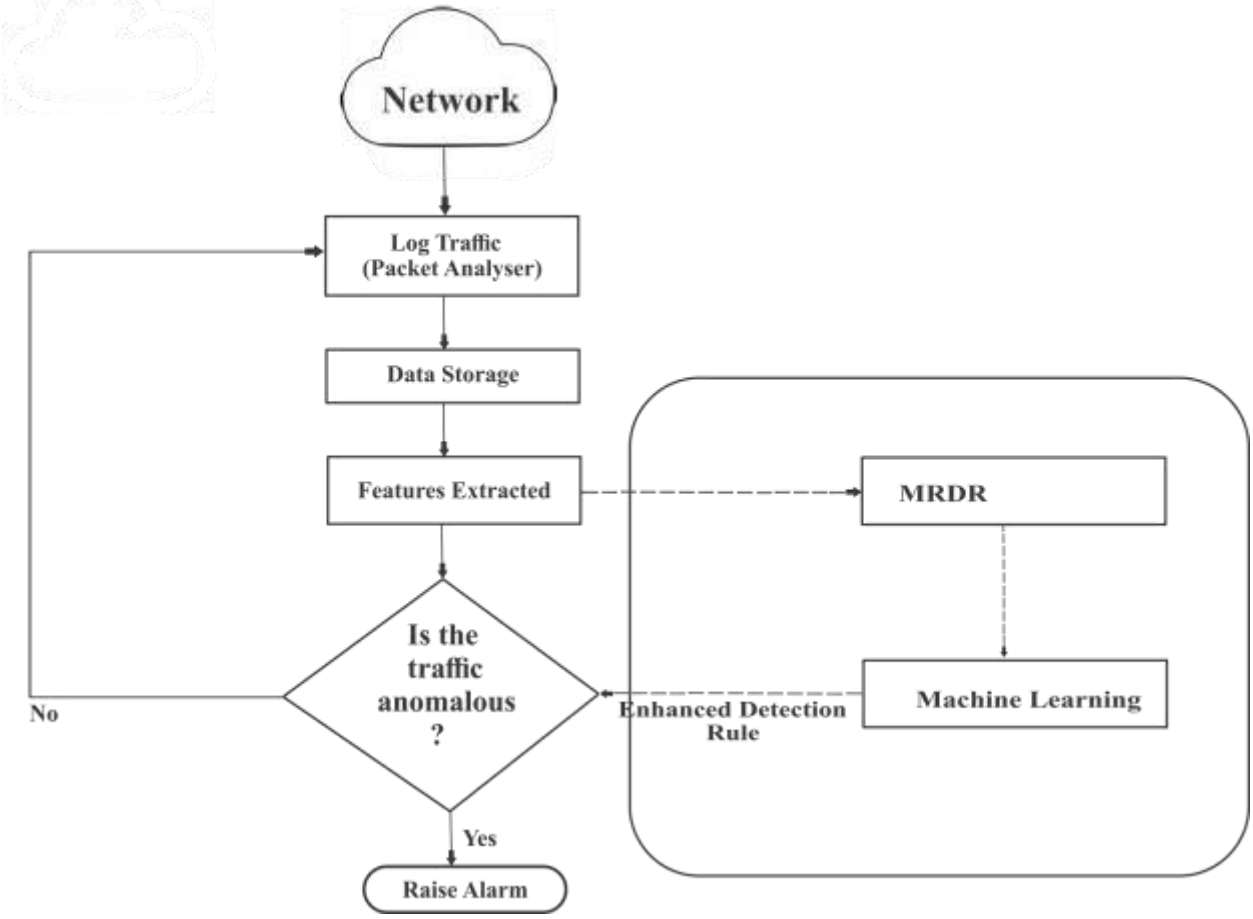


Figure 3.7: Block diagram of the Developed system

Enhanced Ripple Down Framework

Enhanced Ripple Down Framework **(**ERDF) was developed for network intrusion detection system as shown in Figure 3.8. The system depends on Enhanced ripple down rules which is an idea or conclusion that's drawn from evidence and reasoning in which new partitions are created as new rules are added. Exploiting this particular feature of MRDR and some others, MRDF was designed to be an online intrusion detection system which would detect outliers as the system learnt the network behaviour. The system assumed that the partitions resulting from RDR are homogenous and that the data within each partition is uniformly distributed. The framework of the developed system gives rise to the system Model. The Ripple Down Framework was divided into three main components; RDR modeling, Outlier estimation and Final RDR classification.

The Enhanced ripple down model engine processes a case and passes a Situated Profile to the Outlier Detector which passes Outlier Indexes to the Enhanced Ripple Down Rule engine for final classifications. A case was introduced to the system through the MRDR modeling component. In this phase, the RDR engine retrieved an appropriate Situated Profile (SP) for the case. As in RDR, there is a default situated profile in case none of the available SPs match the current case. The situated profile is then passed to the outlier estimation components where each attribute value is searched for possible outliers. The outlier detection results are passed to the RDR decision base where another RDR process would take place to classify the case as either an anomaly or not. The RDR classification is first confirmed with an expert and is stored in the RDR decision base if correct. Otherwise, a new situated profile and a new rule are created for the case.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

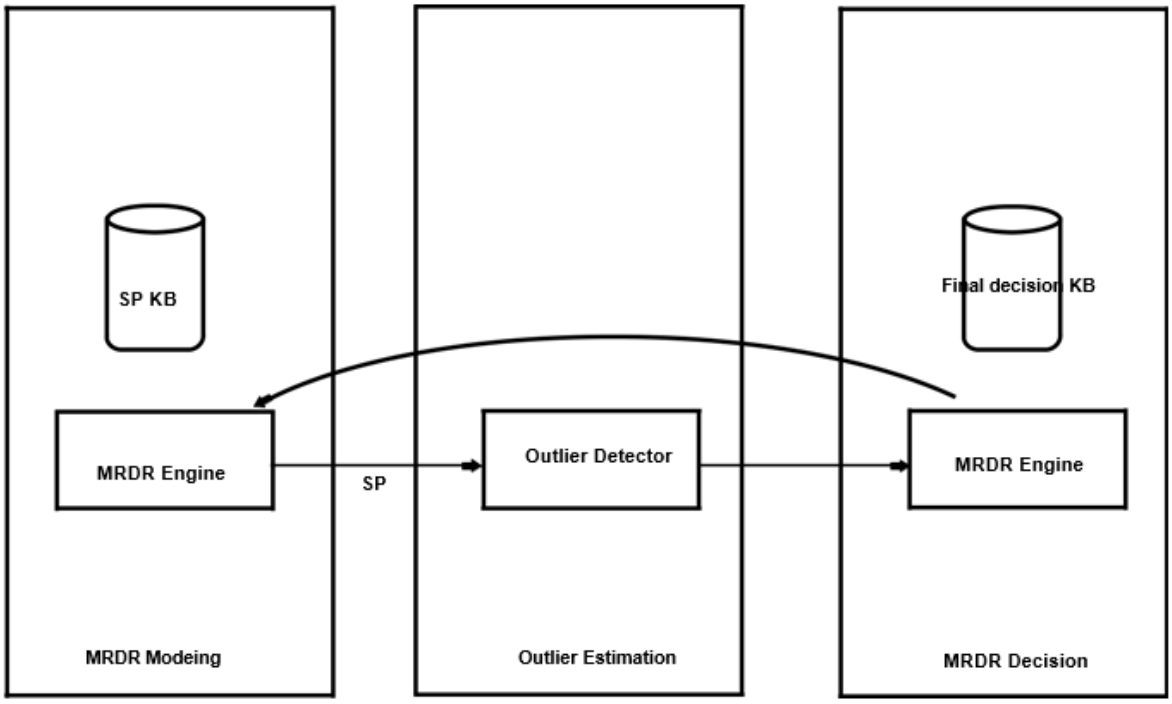


Figure 3.8: Enhanced Ripple Down Framework

Figure 3.9 represents the flow process of the developed system. The flow diagram in Figure 3.9 starts by inputting the acquired data from financial institute then locate the machine address if the transaction is done via mobile phone or laptop then the address is the IP address which is unique to each of the devices used in transaction. The system prompted to ascertain if there is any machine address then traces the IP address and if the system cannot ascertain the IP address returned back to the acquired data. Another measure is to compare the current geographical location and the spending behavior of the present transaction with the previous transaction and decide if the transaction is suspicious or genue.

Implementation of the Designed System

The developed cybercrime detection system was implemented using MATLAB R2023a software on Windows 10 64-bit operating system, Intel (R) Corer ^{TM} i7 Central Processing Unit with a spood of 2.7GHZ, 12GB Random Acess Memory and 500GB hard disk drive. The results gotten were examined at different threshold values using some selected performance metrics which are Sensitivity, Specificity, False Alarm Rate (FAR), Accuracy and Computational Time(CT). The dataset used for this work contains Six cards which were labeled Card A, B, C, D, E and F. The card comprised four attributes and consisted of Two thousand Seven Hundreds (2700) where One thousand six hundred and twenty were used for training and one thousand and eighty were used for the testing.

The numbers of transactions for each card were 250, 350, 450, 550, 650, 750 and 800 transactions respectively, each with fraudulent type and non-fraudulent type. Using cross-validation method, 60% transactions of each card were used to train the system and 40% transactions were used to test the system. These were subjected to Enhanced Ripple Down Rule

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

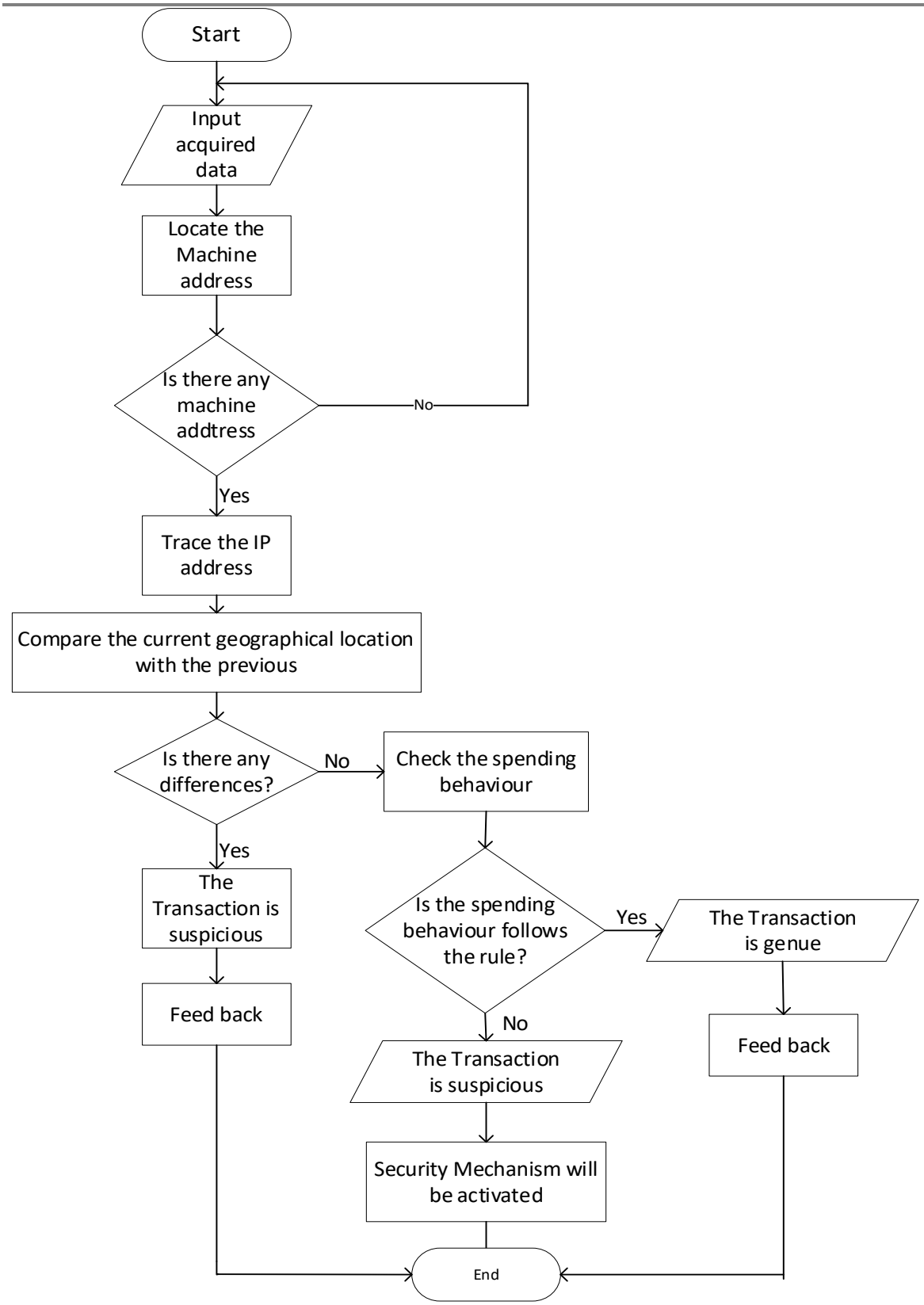


Figure 3.9: Flow chart of the Developed system

(MRDR) and Ripple Down Rule (RDR). The system compared the results generated by MRDR and RDR at different transaction

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

At the point of execution, the Graphical User Interface (GUI) window appeared as the file name and executed, the needed class of attributes to be pre-processed, extracted and detected is selected. The GUI is made up of a button which helps to navigate from one function to another as shown in appendix B .The button performed the loading process of number of attributes, pre-processing operation, detection process; detected the status of the transaction whether it is suspicious , fraudulent or not fraudulent act as analyzed in appendix D.

Performance evaluation of the system

The performance of the developed system was evaluated using Sensitivity, Specificity, False Alarm Rate (FAR), classification accuracy and Computational Time by considering True positive (TP) , False Positive (FP), True Negative(TN) and False Negative(FN).

Mathematically, Sensitivity, Specificity, False Alarm Rate(FAR), classification accuracy were calculated using equation 3.2, 3.3, 3.4 and 3.5 respectively:

$$\text{Sensitivity} = \frac{TP}{TP+FN} X\ 100 \quad (3.2)$$

$$\text{Specificity} = \frac{FN}{TP+FN} X\ 100 \quad (3.3)$$

$$\text{FAR} = \frac{FP}{TN+FP} X\ 100 \quad (3.4)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} X\ 100 \quad (3.5)$$

Where TP, FP, TN and FN are defined correctly:

i. True positive (TP): assigned to a case if a warning was produced correctly.

ii. False Positive (FP): assigned to a case if the system produced a warning incorrectly.

iii. True Negative (TN): assigned to a case if the system did not produce a warning when it was not supposed to.

iv. False Negative (FN): assigned if a warning was required but the system failed to do so.

RESULTS AND DISCUSSION

Evaluation Result of the developed system for Detecting Cybercrimes

Examination results data contained seven (7) crimes which were labeled Card A, B, C, D, E, F and G. The card comprised four attributes and consisted of Three thousand Eight Hundreds (3800) where Two thousand eight hundred were used for training and one thousand were used for the testing. The numbers of transactions for each card were 250, 350, 450, 550, 650, 750 and 800 transactions respectively, each with crime type and non-crime type. Using cross-validation method, 60% crime of each card were used to train the system and 40% crime were used to test the system. These was subjected to Enhanced Ripple Down Rule (ERDR) and Ripple Down Rule

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

(RDR). The strategy was implemented using MATLAB (R2023a) software on Windows 10 64-bit operating system and the results was also examined at different threshold values using Sensitivity, Specificity, False Alarm Rate (FAR), Accuracy and Computational Time(CT).

Tables 4.1(a) and 4.1(b) show results generated by ERDR and RDR, respectively at 250 transactions based on prediction accuracy, sensitivity, specificity, false alarm rate, and Computational time as analyzed under the threshold 0.2, 0.4, 0.6 and 0.8. Likewise, Tables 4.2(a) and 4.2(b) show results generated by ERDR and RDR, respectively at 350 transactions. Tables 4.3(a) and 4.3(b) show results generated by ERDR and RDR respectively at 450 transactions. Tables 4.4(a) and 4.4(b) show results generated by ERDR and RDR, respectively at 550 transactions. Tables 4.5(a) and 4.5(b) show results generated by ERDR and RDR, respectively at 650 transactions. Tables 4.6(a) and 4.6(b) show results generated by ERDR and RDR, respectively at 750 transactions, Tables 4.7(a) and 4.6(b) show results generated by ERDR and RDR, respectively at 800 transactions.

Table 4.1(a): Result of Card A using ERDR at 250 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate (%)	Accuracy (%)	CT (sec)
0.2	98.11	66.67	33.33	87.50	36.71
0.4	96.23	77.78	22.22	90.00	37.63
0.6	94.34	88.89	11.11	92.50	36.66
0.8	92.45	96.30	3.70	93.75	36.68

Table 4.1(b): Result of Card A using RDR at 250 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate (%)	Accuracy (%)	CT (sec)
0.2	92.45	59.26	40.74	81.25	27.81
0.4	90.57	66.67	33.33	82.50	27.69
0.6	88.68	74.07	25.93	83.75	27.14
0.8	86.79	85.19	14.81	86.25	27.68

Table 4.2(a): Result of Card B using ERDR at 350 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	98.75	77.50	22.50	91.67	56.75
0.4	97.50	85.00	15.00	93.33	57.80
0.6	96.25	92.50	7.50	95.00	57.62
0.8	95.00	97.50	2.50	95.83	58.05

Table 4.2(b): Result of Card B using RDR at 350 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	95.00	72.50	27.50	87.50	48.94

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

0.4	93.75	77.50	22.50	88.33	48.73
0.6	92.50	82.50	17.50	89.17	41.24
0.8	91.25	90.00	10.00	90.83	46.63

Table 4.3(a): Result of Card C using ERDR at 450 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	99.07	83.02	16.98	93.75	103.48
0.4	98.13	88.68	11.32	95.00	93.00
0.6	97.20	94.34	5.66	96.25	77.89
0.8	96.26	98.11	1.89	96.88	79.03

Table 4.3(b): Result of Card C using RDR at 450 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT(sec)
0.2	96.23	79.63	20.37	90.63	67.61
0.4	95.28	83.33	16.67	91.25	68.03
0.6	94.34	87.04	12.96	91.88	67.74
0.8	93.40	92.59	7.41	93.13	67.87

Table 4.4(a): Result of Card D using ERDR at 550 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate (%)	Accuracy (%)	CT (sec)
0.2	99.25	86.57	13.43	95.00	97.00
0.4	98.50	91.04	8.96	96.00	97.75
0.6	97.74	95.52	4.48	97.00	97.83
0.8	96.99	98.51	1.49	97.50	97.80

Table 4.4(b): Result of Card D using RDR at 550 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate (%)	Accuracy (%)	CT (sec)
0.2	96.99	83.58	16.42	92.50	97.77
0.4	96.24	86.57	13.43	93.00	101.43
0.6	95.49	89.55	10.45	93.50	98.40
0.8	94.74	94.03	5.97	94.50	97.69

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Table 4.5(a): Result of Card E using ERDR at 650 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	99.38	88.75	11.25	95.83	119.58
0.4	98.75	92.50	7.50	96.67	118.95
0.6	98.13	96.25	3.75	97.50	123.75
0.8	97.50	98.75	1.25	97.92	120.47

Table 4.5(b): Result of Card E using RDR at 650 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	97.50	86.25	13.75	93.75	109.70
0.4	96.88	88.75	11.25	94.17	108.53
0.6	96.25	91.25	8.75	94.58	113.10
0.8	95.63	95.00	5.00	95.42	111.91

Table 4.6(a): Result of Card F using ERDR at 750 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT(sec)
0.2	97.30	89.16	10.84	91.67	206.06
0.4	98.93	93.55	6.45	97.14	138.28
0.6	98.40	96.77	3.23	97.86	149.36
0.8	97.86	98.92	1.08	98..24	140.91

Table 4.6(b): Result of Card F using RDR at 750 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	97.85	88.30	11.70	94.64	130.57
0.4	97.31	90.43	9.57	95.00	131.40
0.6	96.77	92.55	7.45	95.36	130.08
0.8	96.24	95.74	4.26	96.07	131.11

Table 4.6(b): Result of Card G using RDR at 800 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	97.85	88.30	11.70	94.64	130.57
0.4	97.31	90.43	9.57	95.00	131.40
0.6	96.77	92.55	7.45	95.36	130.08
0.8	96.24	95.74	4.26	99.86	131.11

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Table 4.6(b): Result of Card G using RDR at 800 Crimes

Threshold	Sensitivity (%)	Specificity (%)	False Alarm Rate(%)	Accuracy (%)	CT (sec)
0.2	97.85	88.30	11.70	94.64	130.57
0.4	97.31	90.43	9.57	95.00	131.40
0.6	96.77	92.55	7.45	95.36	130.08
0.8	96.24	95.74	4.26	97.85	131.11

Out of all the experimental datasets for training and testing as obtained in Table 4.1, Table 4.2, Table 4.3, Table 4.4 ,Table 4.5, Table 4.6, and Table 4.7, it was observed that ERDR has high sensitivity and specificity at all threshold values. More so, ERDR has low False Alarm Rates at all threshold values compared to RDR.

DISCUSSION OF RESULTS

Table 4.1 to Table 4.6 show results generated for cybercrimes detection system highlighting Sensitivity, Specificity, False Alarm Rate(FAR), Accuracy and Computational Time(CT) for six different cards(Card A,B,C,D,E, F & G) at each threshold value 0.20,0.40.0.60 and 0.80. The results were generated for both ERDR and RDR for result validation and comparison.

Result of card A using ERDR for two hundreds (250) with 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.11%, 66.67%, 33.33%, 87.50% and 36.71sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.23%, 77.78%, 22.22%, 90% and 37.63sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 94.34%, 88.89%, 11.11%, 92.50% and 36.66sec, respectively. However, at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 92.45%, 96.30%, 3.70%, 93.75% and 36.68sec, respectively. Looking at the result shown in Table 4.1(a), it would be deduced that ERDR with 250 transactions performs best at 0.80 threshold value.

Table 4.1(b) shows the result of card A using RDR for 250 crimes. At 0.20 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 92.45%, 59.26%, 40.74% , 81.25% and 27.81sec respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 90.57%, 66.67%, 33.33%, 82.50%% and 27.69sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 88.68%, 74.07%, 25.93%, 83.75% and 27.14sec, respectively. However, at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 86.79%, 85.19%, 14.81%, 86.25% and 27.68sec, respectively for RDR. When compare the two results of card A using ERDR and RDR for two hundred transactions it is recorded that at each of the threshold value, it is observed that MRDR performs better in terms of sensitivity, specificity, FAR and Accuracy with high computational time. MRDR is highly sensitive to correctly identify person with genue transaction and also has the ability to correctly identified person without genue transactions. The ERDR has low false alarm because it has the ability to classify each transaction correctly.

Card B with 350 crimes for ERDR as displayed in Table 4.2(a) using 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.75%, 77.50%, 22.50%, 91.67% and 56.75sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.50%, 85%, 15%, 93% and 57.80sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.25%, 92.50%, 7.50%, 95% and 57.62sec, respectively. However, at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 95%, 97.50%, 2.50%, 95.83% and 58.05sec, respectively.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Table 4.2(b) represents card B with 300 crimes for RDR. At 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 95%, 72.50%, 27.50%, 87.50% and 48.94sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 93.75%, 77.50%, 22.50%, 88.33%% and 48.73sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 92.50%, 82.50%, 17.50%, 89.17% and 41.24sec, respectively. However, at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 91.25%, 90%, 10%, 90.83% and 46.63sec, respectively for RDR.

Card C with 450 transactions for ERDR as displayed in Table 4.3(a) using 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 99.07%, 83.02%, 16.98%, 93.75% and 103.48sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.13%, 88.68%, 11.32%, 95% and 93sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.20%, 94.34%, 5.66%, 96.25% and 77.89sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 96.26%, 98.11%, 1.89%, 96.88% and 79.03, respectively.

Table 4.3(b) represent card C with 450 transactions for RDR. At 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.23%, 79.63%, 20.37%, 90.63% and 67.61sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 95.28%, 83.33%, 16.67%, 91.25%% and 68.03sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 94.34%, 87.04%, 12.96%, 91.88% and 67.74sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 93.40%, 92.59%, 7.41%, 93.13% and 67.87sec, respectively for RDR.

Result of card D using ERDR for five hundreds(550) with 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 99.25%, 86.5%, 13.43%, 95.00% and 97sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.50%, 91.04%, 8.96%, 96% and 97.75sec respectively. At 0.60 thresholds, the Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.74%, 95.52%, 4.48%, 97% and 97.83sec, respectively. However, at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 96.99%, 98.51%,1.49%, 97.50% and 97.80sec, respectively. Looking at the result shown in Table 4.4(a), it would be deduced that MRDR with 500 transactions performs best at 0.80 threshold value.

Table 4.4(b) shows the result of card D using RDR for 550 transactions. At 0.20 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.99%, 83.58%, 16.42%, 92.50% and 97.7sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.24%, 86.57%, 13.43%, 93% and 101.43sec, respectively. At 0.60 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 95.49%, 89.55%, 10.45%, 96% and 98.40sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 94.74%, 94.03%, 5.97%, 94.50% and 97.69sec, respectively for RDR. When compare the two results of card D using MRDR and RDR for five hundreds transactions it is recorded that at each of the threshold value, it is observed that MRDR performs better in terms of sensitivity, specificity, FAR and Accuracy with high computational time.

Card E with 650 crimes for ERDR as displayed in Table 4.5(a) using 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 99.38%, 88.75%, 11.25% , 95.83% and 119.58sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.75%, 92.50%, 7.50%, 96.67% and 118.95sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.13%, 96.25%, 3.75%, 97.50% and 123.75,

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 97.50%, 98.75%, 1.25%, 97.92% and 120.47sec, respectively.

Table 4.5(b) represent card E with 650 transactions for RDR. At 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.50%, 86.25%, 13.75% , 93.75% and 108.53sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.88%, 88.75%, 11.25%, 94.17%% and 108.53sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.25%, 91.25%, 8.75%, 94.58% and 113.10sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 95.63%, 95%, 5%, 95.42% and 111.91sec, respectively for RDR.

Card F with 750 crimes for ERDR as displayed in Table 4.6(a) using 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.30%, 89.16%, 10.84% , 91.67% and 206.06sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.93%, 93.55%, 6.45%, 97.14% and 138.28sec, respectively. At 0.60 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.40%, 96.77%, 3.23%, 97.86% and 149.36sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 97.86%, 98.92%, 1.08%, 98.21% and 140.91sec, respectively

Table 4.6(b) displayed card F with 750 transactions for RDR. At 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.85%, 88.30%, 11.70% , 94.64% and 130.57sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.31%, 90.43%, 9.57%, 95% and 131.40sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.77%, 92.55%, 7.45%, 95.36% and 130.08sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 98.24%, 95.74%, 4.26%, 96.07% and 131.11sec, respectively for RDR.

Card G with 800 crimes for ERDR as displayed in Table 4.6(a) using 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.30%, 89.16%, 10.84% , 91.67% and 206.06sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.93%, 93.55%, 6.45%, 97.14% and 138.28sec, respectively. At 0.60 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 98.40%, 96.77%, 3.23%, 97.86% and 149.36sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 98.86%, 98.92%, 1.08%, 97.85% and 140.91sec, respectively

Table 4.6(b) displayed card G with 800 transactions for RDR. At 0.20 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.85%, 88.30%, 11.70% , 94.64% and 130.57sec, respectively. At 0.40 threshold, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 97.31%, 90.43%, 9.57%, 95% and 131.40sec, respectively. At 0.60 thresholds, Sensitivity, Specificity, FAR, Accuracy and Computational Time value recorded were 96.77%, 92.55%, 7.45%, 95.36% and 130.08sec, respectively. However at 0.80 thresholds, the system recorded Sensitivity, Specificity, FAR, Accuracy and Computational Time value of 99.24%, 95.74%, 4.26%, 97.07% and 131.11sec, respectively for RDR.

Summarily, computational time, accuracy, false alarm rate, specificity and sensitivity were the evaluation metrics adopted for the evaluation of the application ERDR and RDR on a dataset containing 3800 number of crimes for seven crimes respectively, comprising fraudulent type and non-fraudulent type. After the experimentation of the performance of the developed system at thresholds 0.2, 0.4, 0.6 and 0.8, respectively, the threshold at 0.8 produced the best performance and consequently was used to determine the best of the two systems. It is also deducted that false alarm rates decreases as the number of transaction increases. ERDR is highly sensitive to correctly identify person with genue transaction and also has the ability to correctly identified person without genue transactions.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Appendices A, B and C described the graphical User Interface (GUI) of the cybercrimes detection system at design level, training level and testing level. Appendix D defined the crimes analysis of card A for 250 crimes and Appendix E showed the forty percent of 250 crimes data of card A. Figure 4.1 to Figure 4.5 depicted the Graphical representation of the comparisons of the two techniques with respect to the performance metric used in evaluating the system. The results generated with ERDR produced most efficient results in terms of prediction accuracy compared with RDR. In addition, sensitivity and specificity with ERDR was higher than RDR. It shows that MRDR generated the most efficient values than RDR. Evaluating performance of ERDR and RDR based on metric such as FAR revealed that ERDR had a lesser False Alarm Rate compared to RDR as described in Figure 4.3 which is one of our statements of problem. ERDR level of accuracy was higher than that of RDR as displayed in Figure 4.4.

The graphical representation showing sensitivity with ERDR and RDR was depicted in Figure 4.1. The Figure 4.1 revealed that ERDR had higher sensitivity than RDR reason being that ERDR system contains a dynamic rule which is generally based on human knowledge such as human insight. Figure 4.2 showed the plot showing specificity with ERDR and RDR. The points on the graphical representation shown in Figure 4.2 revealed that the specificity for ERDR was higher. Figure 4.5 showed that RDR used less time to complete execution compared to ERDR.

All these positive results of ERDR were achieved because ERDR has an added capability to handle multiple classifications. The inability of Rule Based System to provide multiple classifications was seen from the results generated. It was noticed that, ERDR covered a domain accurately and more sensitively than its single class counterpart and produced a more compact Knowledge Base with fewer redundancies than single class RDR. ERDR was able to detect outliers as the system learnt the network behavior with the least false alarm rate.

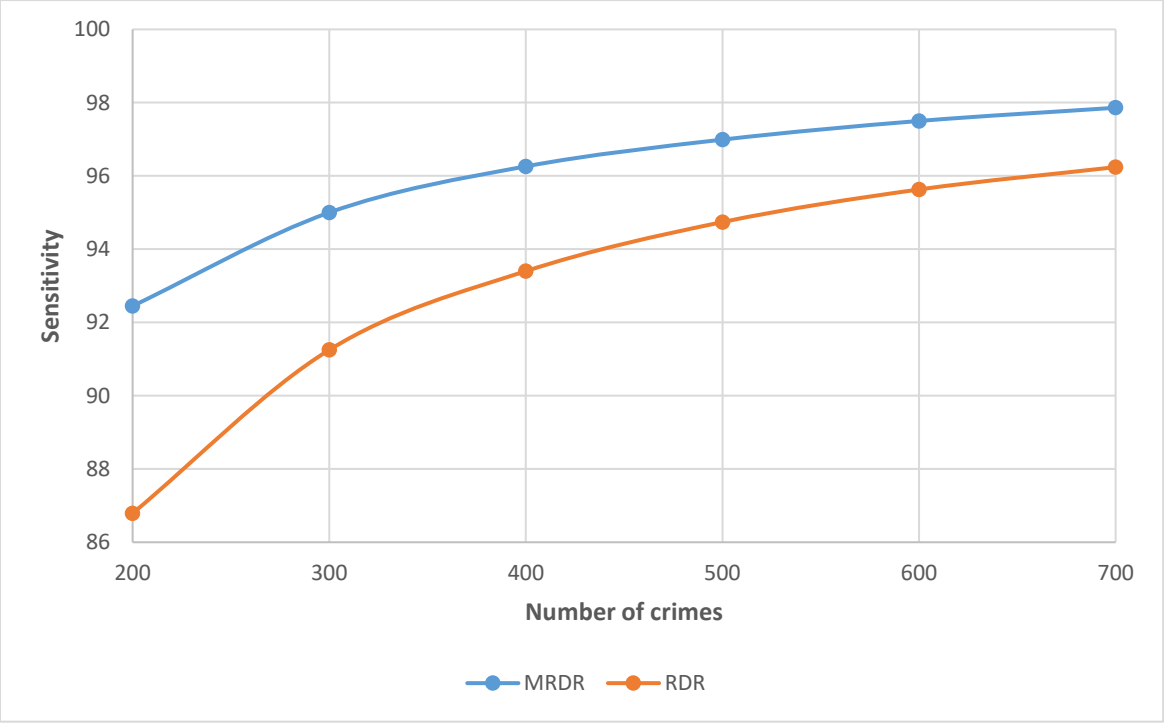


Figure 4.1: Graphical representation showing sensitivity against number of crimes

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

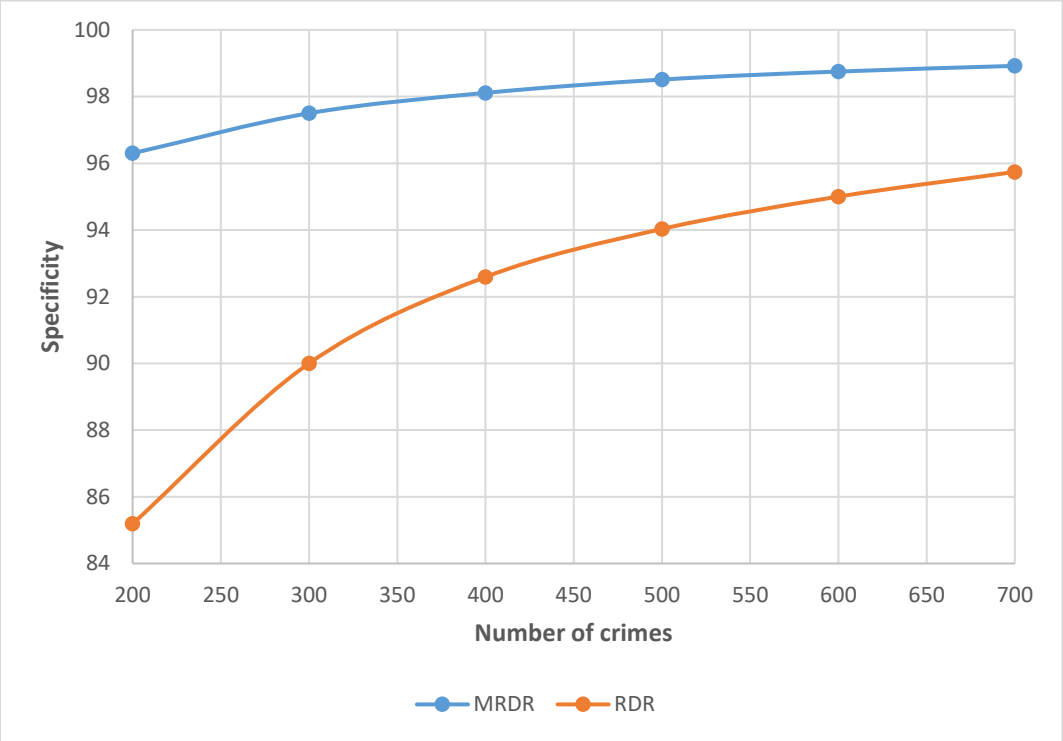


Figure 4.2: Graphical representation showing specificity against number of crimes

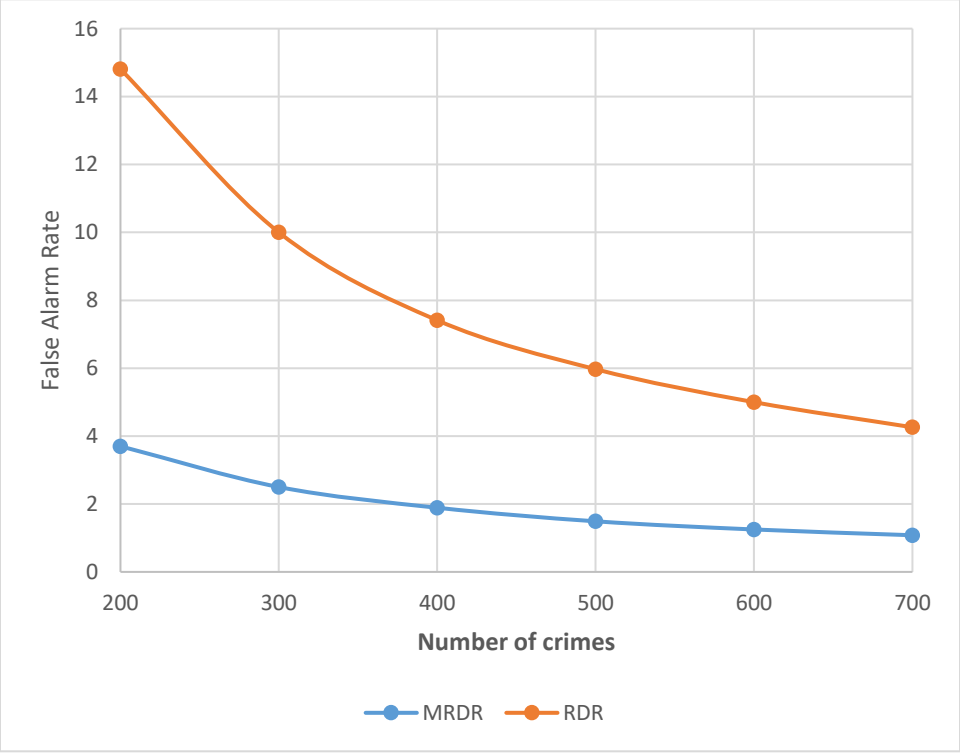


Figure 4.3: Graphical representation showing false alarm rate against number of crimes

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

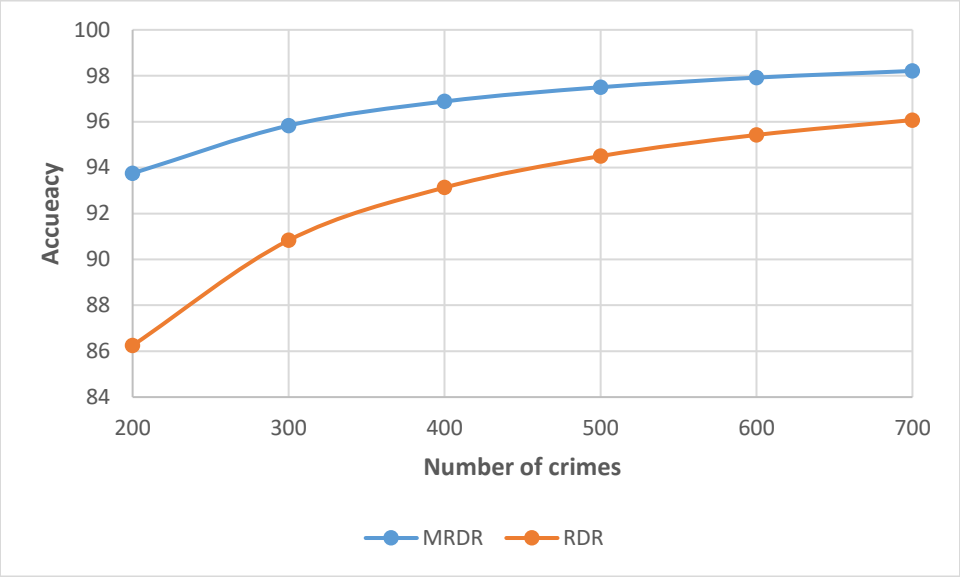


Figure 4.4: Graphical representation showing accuracy against number of crimes

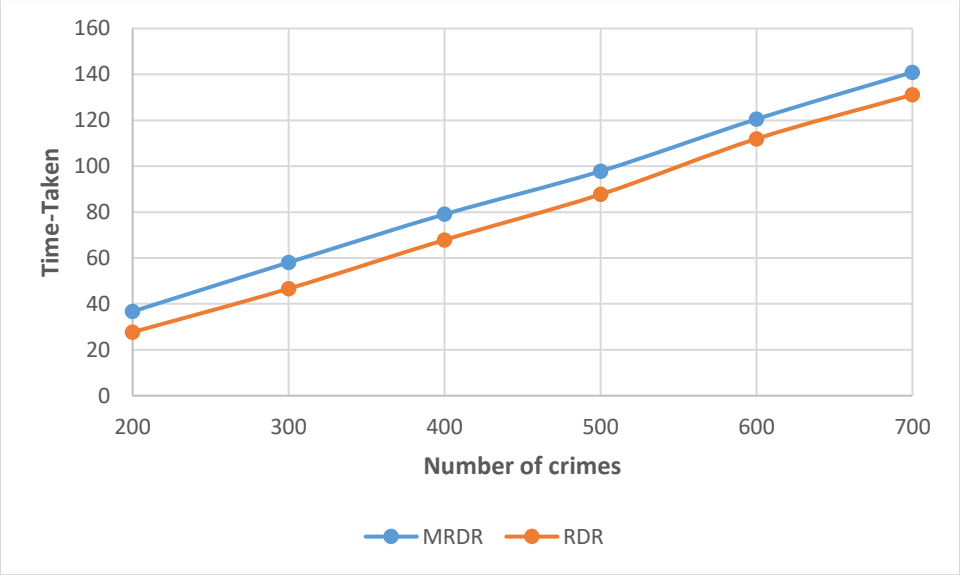


Figure 4.5: Graphical representation showing time taken against number of crimes

Performance evaluation of the result

Performance evaluation of cybercrime detection system is a challenge on its own, since there is no actual or common reference point for benchmarking. Many developed systems from literature claimed an accuracy of less than 90.00% with high false alarm rates. However the truth is that all related existing studies worked only on restricted dataset. It is very important that while evaluating system performance, all our practical test cases should be worked on.

In fulfilment of the last objective of this research work, the performances of the developed system were computed from the values of TP, FP, TN, FN obtained and summarized in Table 4.7. The performance metrics used were

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Sensitivity, Specificity, False Alarm Rate (FAR) and classification accuracy which is calculated averagely to be 96%, 98%, 1.04% and 97% respectively.

Table 4.7(a): Table showing summary of cybercrime detection system with MRDR at 0.80 threshold value

Category	no of transaction	sensitivity (%)	specificity (%)	FAR (%)	Accuracy (%)	CT (Sec)
CRIME A	250	92.45	96.30	3.70	93.75	36.68
CRIME B	350	95.00	97.50	2.50	95.83	58.05
CRIME C	450	96.26	98.11	1.89	96.88	79.03
CRIME D	550	96.99	98.51	1.04	97.50	97.80
CRIME E	650	97.50	98.75	1.28	97.92	120.47
CRIME F	750	97.86	98.92	1.08	98.21	140.91
CRIME G	800	97.86	98.92	1.08	99.86	140.91

Table 4.7(b): Table showing summary of cybercrime detection system with RDR at 0.80 threshold value

Category	**no of transaction**	**Sensitivity**	**Specificity**	**FAR**	**Accuracy**	**CT**
		(%)	**(%)**	**(%)**	**(%)**	**(sec)**
CRIME A	250	86.79	85.19	14.81	86.25	27.68
CRIME B	350	91.25	90.00	10.00	90.83	46.63
CRIME C	450	93.40	92.59	7.41	93.13	67.87
CRIME D	550	94.74	94.03	5.97	94.50	97.69
CRIME E	650	95.63	95.00	5.00	95.42	111.91
CRIME F	750	96.24	95.74	4.26	96.07	111.91
CRIME G	800	96.24	95.74	4.26	97.85	131.11

CONCLUSION AND RECOMMENDATION

Conclusion

In this research, a system has been developed to detect cybercrimes in university and LAUTECH as a case study to reduce the rate of fraudulent act with low false alarm rate. Four important attributes are involved in cybercrime detection system. The cybercrime detection system is critically important for security and economic operation in banking system.

This research developed an Enhanced or Enhanced Ripple Down Rule that was used to detect and prevent cybercrimes. The combination of Radial Basis Function (RBF) and RDR established powerful problem-solving ability called ERDR. It contains the dynamic rule which is generally based on human knowledge. The MRDR technique reported a reasonably and effective results in terms of accuracy, sensitivity, specificity and false alarm rate in comparison with RDR.

ERDR provided results that enhanced accuracy in detecting cybercrimes with lowest false alarm rates. The overall decrease in false alarm rate reflects that the ERDR based approach was superior as compared to RDR which means that ERDR approach has a significant improvement in learning and classification than RDR.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Recommendations

It is recommended that:

i. Future work can be carried out by comparing the effect of other artificial neural network algorithms with RDR.
ii. Nature inspired optimization techniques like particle swarm optimization, firefly algorithm and evolutionary algorithms should be applied to optimize Enhanced Ripple Down Rule (ERDR).

Contributions to Knowledge

This research contributed to the body of knowledge by handling different cybercrimes in our university based on various attributes and yielded low false alarm rates. More so, this research has successfully demonstrated the effectiveness of enhanced RDR for anomaly detection in cybercrimes which has considerably improved the performance of the system in terms of accuracy, sensitivity and specificity.

REFFERNCES

1. Alazab, A., Abawajy, J. H., & Hobbs, M. (2013). Web Malware that Targets Web Applications. Social Network Engineering for Secure Web Data and Services, 12, 248-264. doi:10.4018/978-1-4666-3926-3.
2. Alex S., David D., Aladdin A. (2018). Intelligent intrusion detection systems using artificial neural networks. The Korean Institute of Communication and Information Science, ICT Express 4, 95–99.
3. Ameer A. and Chafika B., (2015). Cybercrime Profiling: Decision-Tree Induction, Examining Perceptions of Internet Risk and Cybercrime Victimization. IEEE Trustcom.
4. Amrit, P. S., and Manik, D.S. (2014). Analysis of Host-Based and Network-Based Intrusion Detection System, International Journal of Computer Network and Information Security, 8, PP.41-47.
5. Aru, O. E and Chiaghana, C. E. (2018). Malware Analysis and Mitigation in Information Preservation, Computer Engineering (IOSR-JCE), e-ISSN: 2278-0661, 20(4): 53-62.
6. Arulogun, O.T., Amusan D. G. (2015). Vehicle license plate recognition using edge detection and neural network. International Journal of Science and Research (IJSR), ISSN (Online): 2319-7064.
7. Anderson, R., and Moore, T. (2007). Information security economics and beyond. Advances in Cryptology, 68-91.
8. Anderson, B., Storlie, C., and Lane, T. (2012). Improving malware classification: bridging the static/dynamic gap. In Proceedings of the 5th ACM Workshop on Security and Artificial Intelligence, 3–14.
9. Asmaa, S. and Sharad, G. (2011). Intrusion detection system and intrusion prevention system. International Journal of Scientific and Engineering Research, 2(7): 64-71.
10. Blum, A, Song, D, and Venkataraman, S. (2004). Detection of interactive stepping stones. Algorithms and confidence bounds, in Proceedings of the 7th International Symposium on Recent Advances in Intrusion Detection (RAID), 27, 1134-1142.
11. Burch, H. and Cheswick, B. (2000). Tracing anonymous packets to their approximate source, in Proceedings of USENIX LISA, New Orleans, USA, 319-327.
12. Chao, R., Tan, Y. (2009). A Virus Detection System Based on Artificial Immune System, International Conference on Computational Intelligence and Security, 1, 6–10.
13. Cker Chiueh, T., and Hsu, F., H. (2001). A Compile-Time Solution to Buffer Overflow Attacks. Proc. 21st Int'l Conf. Distributed Computing Systems (ICDCS).
14. Cohen, LE., Felson M. (1979). Social Change and Crime Rate Trends. A Routine Activity Approach. Am. Sociol. 44(2): 588-605.
15. Connolly, T. M., Begg, C. E., and Strachan. A. D. (2006). Database systems: Addison-Wesley. Journal of Image and Vision Computing, 19, 142-160.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

16. Corporation. M., (2012). Common Vulnerabilities and Exposures Available: http://cve.mitre.org/.
17. Cova, M., Balzarott D., Felmetsger V., and Vigna, G. (2008). Swaddler: An approach for an anomaly-based detection of state violations in web applications, 63-86.
18. Dagorn, N. (2008). A Cooperative Bayesian Anomaly-Based Intrusion Detection System for Web Applications (Extended Abstract), Web IDS: 392-393.
19. Danforth, M. (2009). Towards a Classifying Artificial Immune System for Web Server Attacks. International Conference on Machine Learning and Applications, 523–527.
20. Dharmapurikar, S., Krishnamurthy. P, Sproull. T, and Lockwood J. W. (2004). Deep Packet inspection using parallel bloom filters. IEEE. 52-61.
21. Duman, E. and Ozcelik. H. M. (2011). Detecting credit card fraud by genetic algorithm and scatter search. Science Direct, Expert System with Applications 38, 13057-13063.
22. EshghiShargh, A. (2009). Using Artificial Immune System on Implementation of Intrusion Detection Systems. Third UK Sim European Symposium on Computer Modeling and Simulation, 164-168.
23. Figueiredo, M. A. (2000). On gaussian radial basis function approximations: interpretation, extensions, and learning strategies. In Proceedings of the IEEE 15th International Conference on Pattern Recognition 2, 618–621.
24. Folashade, B. O and Abimbola K.A. (2013). The Nature, Causes and Consequences of Cyber Crime in Tertiary Institutions in Zaria-Kaduna State, Nigeria. American International Journal of Contemporary Research. 3(9): 21-32.
25. Garcia K. A. (2012). Analyzing Log Files for Postmortem Intrusion Detection. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(6): 1690-704.
26. Garcia-Teodoro, P., Diaz-Verdejo, J., Macia-Fernandez. G., and Vazquez. E. (2009). Anomaly-based network intrusion detection. Techniques, systems and challenges, Computers & Security, 28, 18-28.
27. Gianini, G., Anisetti, M., Azzini, V., Bellandi, V., Damiani, E., Marrara, S. (2009). An Artificial Immune System approach to Anomaly Detection in Multimedia Ambient Intelligence. 3rd IEEE International Conference on Digital Ecosystems and Technologies. 502–506.
28. Hesham, A. and Saeed A. (2011). Bayesian based intrusion detection system. Journal of King Saud University Computer and Information Sciences. 2-7.
29. Huang, G., Saratchandran, P., and Sundararajan, N. (2005). A generalized growing and pruning RBF (GGAP-RBF) neural network for function approximation. IEEE Transactions on Neural Networks. 16(1), 57–67.
30. Ismaila, W. O, Ismaila, F. M, Falohun, A. S. Oladoye S. F. (2018). Soft Computing Approach to Anomaly Detection in Subscribers Call Profiles. The International Journal of Engineering and Science (IJES). 7(12), 23-30.
31. Ismaila, W. O., Falohun, A.S., Ismaila F.M., Ogunjinmi T.O., Babalola O.R. (2019). Soft Computing: Investigation of two-step approach to identify theft detection in electronic payments. The proceedings for Academic Conference of the sub-sahara African Academic Research publications on unleashing sub-sahara African Resources held at Multipurpose Hall, Gonbe State University, Gombe State, Nigeria. 18(2), 13–23.
32. Iwarimie, J., D. (2010). Criminology, Crime and Delinquency in Nigeria. Port Harcourt, Pearl Publishers.
33. Iwasokun, G. B., Alese, B. K., T A., and Aranuwa, F. O. (2012). Statistical evaluation of the impact of ICT on Nigerian universities. International Journal of Education and Development using Information and Communication Technology (IJEDICT), 8(1), 104-120.
34. James l. (2018). Economic Impact of Cybercrime—No Slowing Down. McAfee, one of the world's leading independent cybersecurity companies. 1-28.
35. Jamil F. (2003). Statistical based Intrusion Detection. Symantec. Security Focus.
36. Juvonen and Sipola T. (2013). Combining conjunctive rule extraction with diffusion maps for network intrusion detection. 20-13.
37. Kim, K, Choi, Y and Park, J. (2013). Pricing fraud detection in online shopping malls using a finite mixture model. Science Direct Electronic Commerce Research and Applications, 195-207.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

38. Kolbitsch. (2013). Discriminant Malware Distance Learning on Structural Information for Automated Malware Classification. Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems. 347-348.
39. Kruegel, C. Comparetti, P. M., Hlauschek, C. and Bayer, U. (2009). Scalable, Behavior-Based Malware Clustering. IEEE.
40. Kruegel, C., Mutz, D., Robertson, W., and Valeur F. (2003). Bayesian event classification for intrusion detection. In Computer Security Applications Conference, 2003. Proceedings. 19th Annual. 14-23.
41. Lebbe, M. A., Agbinya, J. I., Chaczko, Z., Chiang F. (2007). Self-Organized Classification of Dangers for Secure Wireless Mesh Networks. Australasian Telecommunication Networks and Applications Conference. 322–327.
42. Longe, B., Chiemeke, C. (2008). Cybercrime and criminality in Nigeria. Journal of Social Science. 6(4), 133-139.
43. Maher, A., Hossain, M.A., Keshav, D., Fadi, T. (2010). Predicting Phishing Websites using Classification Mining Techniques with Experimental Case Studies. Seventh International Conference on Information Technology.
44. Maitanmi, O, Ogunlere S, Ayinde S, Adekunle Y. (2013). Impact of Cyber Crimes on Nigerian Economy. The International Journal Of Engineering And Science (IJES), 2(4), 45-51.
45. Manjeri N. K., and Shelke C. J. (2014). Remote Administrative Trojan/Tool (RAT). JCSMC. 3(3), 482–487.
46. Moore. T., and Anderson. R. (2007). Information security economics–And beyond. Advances in Cryptology-CRYPTO, 68-91.
47. Neelabh (2012). Tracking Digital Footprints of Scareware to Thwart Cyber Hypnotism Through Cyber Vigilantism in Cyberspace. BVICAM's International Journal of Information Technology (BIJIT). 4(2), 0973–5658.
48. Ng, J. D. Joshi, and Banik. S. M. (2015). Applying Data Mining Techniques to Intrusion Detection. Information Technology: New Generations (ITNG) 2015 Proceedings of the 12th International Conference on Information Technology, Las Vegas, NV, USA, 800-810.
49. Odumesi, J. O. (2014). A socio-technological analysis of cybercrime and cybersecurity in Nigeria. International Journal of Sociology and Anthropology, 6(3), 116-125.
50. Okoye R and Gbegi N. (2013). An evaluation of the effect of fraud and related financial crimes in Nigerian economy. Kuwait chapter of Arabian Journal of Business and Management Review 2(7).
51. Open Web Application Security Project. (2010). The Top 10 Most Critical Web Application Security Risks.
52. Patcha. A and Park J. M. (2007). An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer Networks, 51, 3448-3470.
53. Parvinder, S., and Mandeep, S. (2015). Fraud Detection by Monitoring Customer Behavior and Activities. International Journal of Computer Applications. 111(11), 0975–8887.
54. Powers, S. T., and He, J. (2008). A hybrid artificial immune system and Self Organizing Map for network intrusion detection. Information Sciences, 178, 3024-3042.
55. Priyanka, S. (2009). Ripple-Down Rules for Knowledge Acquisition in Intelligent System (JTES) Delving. Journal of Technology and Engineering Sciences. 1(1).
56. Quinlan, J. R. (2006). Induction of Decision Trees. Machine Learning, 1, 81-106.
57. Razzaq and Abdul. (2014). Semantic Security Against Web Application Attacks: 254 Vol. Elsevier Inc, 2014.
58. Reazul K. M., Abdur Rahman O., Tanvir S. (2017). A Network Intrusion Detection Framework based on Bayesian Network using Wrapper Approach. International Journal of Computer Applications. 166(4): 0975–8887.
59. Robertson, W. K. (2010). Detecting and preventing attacks against web applications. Information of Technology, University of California, Santa California.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

60. Roesch, M. (1999). Snort-lightweight intrusion detection for networks. In Proceedings of the 13th USENIX Conference on System Administration. 1.9, 229-238.
61. Roger, E.S. (2008). Rogers Communications Inc, 2008 Annual Report.
62. Samanvay, G. (2012). Buffer Overflow Attacka. IOSR Journal of Computer Engineering (IOSRJCE), ISSN: 2278-0661, 1(1): 10-23.
63. Seong, S.K., and Narasimha R. (2005). A study of analyzing network traffic as images in real-time. In IEEE Infocom. 2056–2067.
64. Sesan, G., Soremi, B., and Oluwafemi B. (2013). Economic Cost of Cybercrime in Nigeria. Cyber Steward Network Project of the Citizen Lab. University of Toronto.
65. Shafi, K., and Abbass, H. A. (2009). An adaptive genetic-based signature learning system for intrusion detection. Expert Systems with Applications. 36, 12036-12043.
66. Sheikhan, M., Jadidi, Z., and Beheshti, M. (2010). Effects of feature reduction on the performance of attack recognition by static and dynamic neural networks. World Applied Sciences Journal, 8, 302-308.
67. Shimrit Tzur-David (2011). Network Intrusion Prevention System. Signature-based and Anomaly Detection. Ph.D. Thesis, School Computer Science. The Hebrew University of Jerusalem.
68. Simone, A. L. (2019). Applying a Neural Network Ensemble to Intrusion Detection. JAISCR, 9(6).
69. Sinclair, C., Pierce, L., and Matzner, S. (2009). An application of machine learning to network intrusion detection. In Computer Security Applications Conference, 1999 (ACSAC'99) Proceedings. 15th Annual. 371-377.
70. Singh R., Kumar H., Singla R.K., Ketti, R.R. (2017). Internet attacks and intrusion detection System. A review of the literature, Online Inform. 41(2): 171–184.
71. Stakhanova, N., Basu, S., and Wong, J. (2007). A taxonomy of intrusion response systems. International Journal of Information and Computer Security, 1, 169-184.
72. Steve U. B, Diepreye O, Uduak D. A. (2009). Information Communication Technologies in the Management of Education for Sustainable Development in Africa. An International Multi-Disciplinary Journal, Ethiopia, 3(3): 414-428.
73. Stibor, T., Timmis, J., and Eckert, C. (2005). A comparative study of real-valued negative selection to statistical anomaly detection techniques. In Artificial Immune Systems, Springer, 262-275.
74. Sumanjit, D. and Tapaswini N. (2013). Impact of Cyber Crime: Issues and Challenges. International Journal of Engineering Sciences & Emerging Technologies, 6(2): 142-153.
75. Tong, X, Wang, Z, and Yu, H. (2009). Hybrid RBF/Elman neural networks for intrusion detection system secure model. Computer Physics Communications, 180, 1795-1801.
76. Vigna, G., and Kemmerer, R. A. (2009). A network-based intrusion detection system. Journal of Computer Security. 7, 37-72.
77. Vijesh K., Santhadevi P. (2014). Internet spam threats and email exploitation – A scuffle with inbox attack. Int. Journal of Applied Sciences and Engineering Research, 3(4): 46-55.
78. Wang, G, Hao, J, Ma, J, and Huang, L. (2010). A new approach to intrusion detection using Artificial Neural Networks and fuzzy clustering. Expert Systems with Applications, 37, 6225-6232.
79. William R. Cheswick., Steven M. Bellovin., Aviel D. R. (2003). Firewalls and Internet Security. Repelling the Wily Hacker, 2nd Edition, Addison Wesley Publication.
80. Wu J., Peng D., Li Z., Zhao L., Ling H. (2015). Network intrusion detection based on a general regression neural network optimized by an improved artificial immune algorithm. PLOS ONE, 10(3): 1–13.
81. Yeh, Y.R., Lee, Y.J., and Wang, Y.C.F. (2013). Anomaly Detection via Online Oversampling Principal Component Analysis. IEEE Transactions on Knowledge and Data Engineering, 25(7): 1460-1470.
82. Yoohwan, K., Wing, C., Lau, M., Choo, C., and Jonathan, C. (2004). Statistics-based overload control against distributed denial-of-service attacks. In IEEE Infocom, 2594–2604.
83. Zhang, A., Chen, C., and Karimi, H. (2013). A new adaptive LSSVR with on-line multikernel RBF tuning to evaluate analog circuit performance. Abstract and Applied Analysis. 20, 1-7. Article ID 231735.
84. Zou, C. Towsley, Gong, W. & Cai. Routing S. (2015). Worm: A fast, selective attack worm based on IP address information. In Proc. IEEE Work. Princ. Adv. and Dist. Simul. (PADS), 199–206.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

APPENDIX A

CYBERCRIMES DETECTION SYSTEM

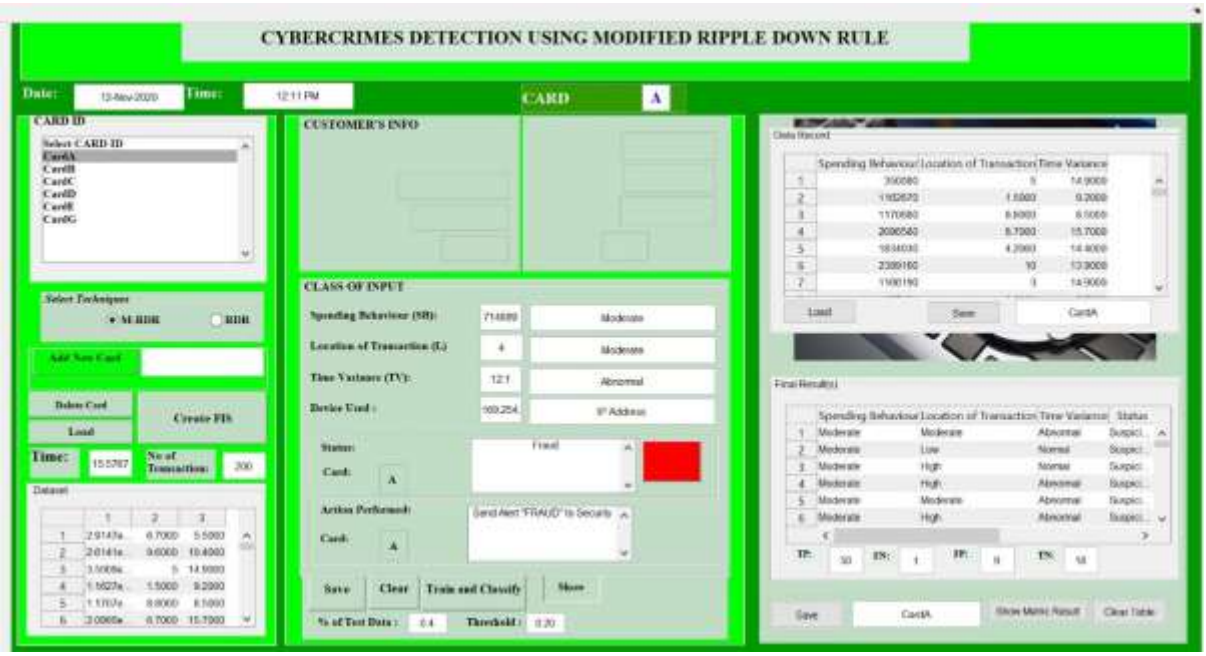
APPENDIX B

TRAINING THE CYBERCRIMES DETECTION SYSTEM

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

APPENDIX C

CLASSIFYING THE CYBERCRIMES DETECTION SYSTEM



APPENDIX D

TRANSACTION ANALYSIS OF CARD A FOR 200 TRANSACTIONS

Spending Behaviour	Location of Transaction	Time Variance	Status	Actions Performed	Card ID	Time	Confusion Matrix
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	0.8862	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	0.17151	TN
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	0.32545	TN
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	0.50693	TN
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	0.68309	TN
Moderate	Low	Normal	Suspicious	Send Alert "CHECK" to Security	A	0.82842	TN
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	1.0271	TN

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	1.1978	TN
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	1.4101	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	1.5912	TN
Moderate	High	Normal	No Fraud	None	A	1.8071	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	1.9888	TN
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	2.2123	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	2.427	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	2.6695	TP
Moderate	Low	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	2.8953	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	3.1413	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	3.389	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	3.6694	TP
Moderate	High	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	3.946	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	4.2081	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	4.4953	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	4.7849	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	5.0931	TP

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	5.3869	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	5.7126	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	6.0311	TP
Moderate	High	Normal	No Fraud	None	A	6.3779	TP
Moderate	Low	Normal	Suspicious	Send Alert "CHECK" to Security	A	6.7239	TP
Moderate	Low	Normal	Suspicious	Send Alert "CHECK" to Security	A	7.1268	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	7.5221	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	7.9157	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	8.278	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	8.6921	TP
Moderate	Moderate	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	9.0845	TP
Moderate	High	Normal	No Fraud	None	A	9.4757	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	9.8922	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	10.2709	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	10.7285	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	11.1457	TP
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	11.6657	TP

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	12.173	TP
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	12.6076	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	13.0782	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	13.5137	TP
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	14.0023	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	14.4689	TP
Moderate	Moderate	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	14.9026	TP
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	15.2829	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	15.6934	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	16.0933	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	16.5147	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	16.9274	TP
Moderate	Low	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	17.363	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	17.7764	TP
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	18.2116	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	18.6592	TP

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Moderate	High	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	19.097	TP
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	19.5512	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	20.0069	TP
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	20.4838	TP
Moderate	High	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	20.941	FN
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	21.4243	FN
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	21.8998	FN
Moderate	High	Normal	Suspicious	Send Alert "CHECK" to Security	A	22.3973	FN
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	22.8829	FP
Moderate	Low	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	23.3851	TN
Moderate	Low	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	23.872	TN
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	24.3628	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	24.8518	TN
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	25.3465	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	25.8526	TN
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	26.3493	TN

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	26.8489	TN
Moderate	Moderate	Abnormal	Fraud	Send Alert "FRAUD" to Security	A	27.3399	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	27.8342	TN
Moderate	High	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	28.325	TN
Moderate	Moderate	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	28.8239	TN
Moderate	Moderate	Normal	Suspicious	Send Alert "CHECK" to Security	A	29.3196	TN
Moderate	Low	Abnormal	Suspicious	Send Alert "CHECK" to Security	A	29.8206	TN

APPENDIX E

FORTY PERCENT OF 200 TRANSACTION DATA OF CARD A

Spending Behaviour	Location of Transaction	Time Variance
1409220	5.2	11.6
1488340	5.8	6.3
2236380	9.9	5.4
1799770	2	17.1
2207420	6.7	15.1
1237760	2.2	9.2
2669860	9.1	7.9
2884780	9.3	10.9
1776250	6	14.5
1041740	3.9	8.4
222511	9.2	5.6
1250020	4.9	6.3
2217520	8.1	9.9
1860220	3.1	14.1
2185070	3.2	11.4
827887	1.6	18.5
254680	5.8	12.9
2301260	5.4	17.5
2379200	8.2	12.8

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

345437	7.1	10.9
2321840	7.3	8.1
2361570	4.2	15.8
2395930	6	10.5
2461990	5.8	13.3
1816970	4.5	14.6
1548590	7.8	16.3
1782880	7.1	11.8
461506	6.3	5.3
2393840	1.6	7.9
875886	2.4	8.9
2363050	3.9	12.7
2613190	8.9	8.7
2442740	8.1	14.6
1089750	7.1	15.4
728462	4.6	16.4
408294	9.6	8.9
2429960	8.7	7.9
1585410	4.5	17.6
1397310	7.1	8.2
1795740	5.2	15.8
1006250	2.4	14
691901	3.2	7.3
1949460	3	13.6
1507920	6.2	11
397709	3.9	10.1
283781	1.2	13.2
2194090	6.4	17.8
802867	5.9	11.8
2135570	6.5	10.3
1686260	5.4	11.2
2330840	10	9.8
2214440	3.8	18.6
1726260	3.4	19.2
879373	2.8	15.6
1104370	8.7	6.4
2315750	2.2	17.1
1725170	8.4	9.1
433587	8.7	15.3
1824660	5.5	18.7
1307890	8.9	6.4
1342420	7.4	9.5
461800	6.7	16.6

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

1087450	6.7	12.8
1153700	9.3	14.3
1851230	9.9	6.8
2054400	7	13.4
723777	2.3	19.3
537040	1.3	15.9
457206	2.9	18.8
846561	4	5.8
200956	6	15.9
2565530	4.6	6.2
2209810	1.4	15.9
1297500	5.5	5.7
523571	5	13.8
1688280	3.8	9.5
2441540	8.7	11.2
1153600	5.9	19.4
1472560	4.2	8
1401390	1.8	12.4

APPENDIX F

COMBINED RESULTS OF SENSITIVITY, SPECIFICITY, FAR, ACCURACY AND TIME-TAKEN AT 0.8 THRESHOLD

(a) Sensitivity at 0.8 threshold

Number of transactions	MRDR	RDR
200	92.45	86.79
300	95.00	91.25
400	96.26	93.40
500	96.99	94.74
600	97.50	95.63
700	97.86	96.24

(b) Specificity at 0.8 threshold

Number of transactions	MRDR	RDR
200	96.30	85.19
300	97.50	90.00
400	98.11	92.59
500	98.51	94.03
600	98.75	95.00
700	98.92	95.74

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue VII, July 2026

(c) False Alarm Rate at 0.8 threshold

Number of transactions	MRDR	RDR
200	3.70	14.81
300	2.50	10.00
400	1.89	7.41
500	1.49	5.97
600	1.25	5.00
700	1.08	4.26

(d) Accuracy at 0.8 threshold

Number of transactions	MRDR	RDR
200	93.75	86.25
300	95.83	90.83
400	96.88	93.13
500	97.50	94.50
600	97.92	95.42
700	98.21	96.07

(e) Time-taken at 0.8 threshold

Number of transactions	MRDR	RDR
200	36.68	27.68
300	58.05	46.63
400	79.03	67.87
500	97.80	87.69
600	120.47	111.91
700	140.91	131.11