ISSN: 2959-6386 (Online), Volume 2, Issue 1

Journal of Knowledge Learning and Science Technology

Journal homepage: https://jklst.org/index.php/home

Understanding Ransomware Attacks: Trends and Prevention Strategies

FNU Jimmy

Senior Cloud consultant, Deloitte, USA

Abstract

Ransomware attacks pose significant security risks to both personal and corporate data, leading to profound privacy breaches and financial losses. Detecting ransomware accurately and promptly is crucial to mitigating its impact. This paper explores current trends and future prospects in automated ransomware detection, covering a background on ransomware, its historical evolution, and various detection, prevention, mitigation, and recovery approaches. Notably, it presents a comprehensive timeline of ransomware attacks from 1989 to 2021, offering insights into the state-of-the-art techniques, particularly those published between 2015 and 2022. By highlighting recent advancements in machine learning, deep learning, and neural network technologies, this paper identifies research gaps and challenges in combating ransomware, providing valuable direction for future studies.

Keywords: machine learning, deep learning, neural network, security, ransomware attack, ransomware detection

Article Information:

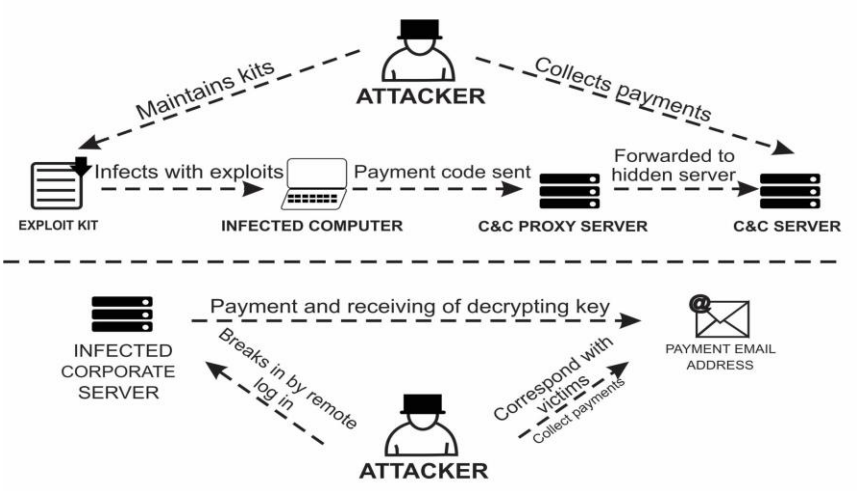
Article history: Received: 01/04/2023 Accepted:15/04/2023 Online: 30/04/2023 Published: 30/04/2023

DOI: https://doi.org/10.60087/jklst.vol2.n1.p214

[i] ***Correspondence author: FNU Jimmy***

Introduction

Ransomware, a form of malicious software, seizes control of data or systems, effectively denying rightful access to their owners. This nefarious software employs intricate processes, tools, and techniques, often making decryption or unlocking a formidable challenge even for skilled computer experts. In addition to encryption or system locking, ransomware may also pilfer sensitive data from victimized computers and networks. Its targets range from personal computers to business systems, including industrial control systems and internet of things (IoT) devices (Celdrán et al., 2022). Typically, a ransomware attack employs private key encryption, coercing legitimate users to pay a ransom, frequently in bitcoin, to regain access to their systems or data (Richardson & North, 2017). Moreover, ransomware tactics may involve data exfiltration, where attackers threaten to expose compromised files unless the ransom is paid. This insidious malware propagates through various vectors, such as email attachments, malicious ads, or links to infected websites. Once infiltrated, it scours drives on the victim's system or network, encrypting files and obstructing owners' access (Morhurle & Patil, 2017). Subsequently, the attacker furnishes instructions, often in the form of files, detailing the ransom payment process. Upon receipt of payment confirmation, the attacker provides the decryption key, granting the victim access to their data. Infected or encrypted files often bear distinctive extensions like .aaa, .micro, or .locked, indicative of the ransomware strain. Notable examples include Reveton, CryptoLocker, and WannaCry (Andronio et al., 2017). Ransomware can be categorized into crypto ransomware, locker ransomware, and scareware, each with distinct characteristics and operational modes (Andronio et al., 2017). Figure 1 elucidates the modus operandi of locker and crypto ransomware (F-Secure Labs, 2013).



Crypto ransomware stands as the predominant threat among ransomware variants, targeting computer systems and networks. Employing symmetric and/or asymmetric cryptographic algorithms, this category encrypts files and data, rendering them inaccessible even if the malware is removed or compromised storage media is transferred to another device (Savage et al., 2015). In contrast, locker ransomware locks devices, preventing their use, but typically does not affect stored data. Removal of the malicious software often allows for data recovery by inserting the infected storage device into another system, making locker ransomware less lucrative for extorting money (Savage et al., 2015). Scareware, on the other hand, deceives victims by displaying fake infection warnings, enticing them to purchase and install bogus antivirus software (Brewer, 2016). Other ransomware categories include human-operated ransomware and fileless ransomware. Human-operated ransomware targets entire organizations, exploiting security vulnerabilities to penetrate networks and launch attacks against critical data (Microsoft Ignite, 2022). Fileless ransomware employs native system tools for attacks, making detection challenging as no external code installation is required (Crowdstrike, 2022b).

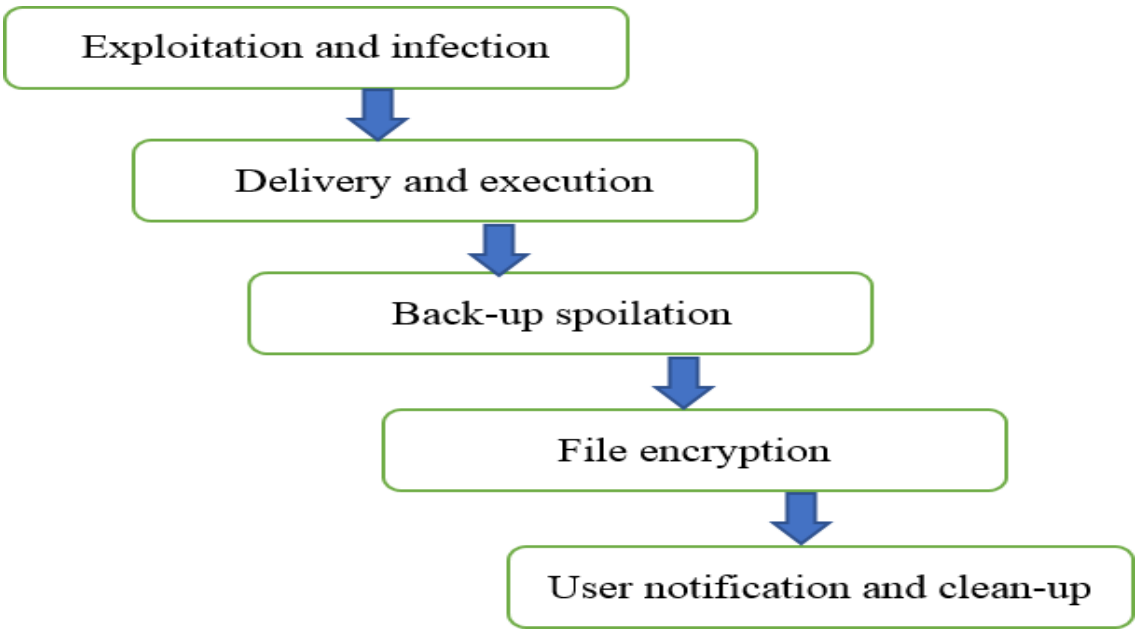
The proliferation of ransomware poses significant threats to both business and individual data and devices, with victims typically coerced into paying ransom, often in bitcoin, to regain access. However, payment may not always result in decryption, and decryption attempts using provided keys can further damage files (Zetter, 2015). Technological advancements such as ransomware development kits and bitcoins facilitate the continual rise in ransomware attacks, causing annual losses in the hundreds of millions of dollars (Fitzpatrick et al., 2016). The lucrative

nature of ransomware fuels the development of new variants each year since 2013, evolving to evade conventional antivirus and intrusion detection systems. This underscores the urgent need for innovative, effective techniques for detection, prevention, and mitigation of ransomware attacks.

This paper contributes to the field through its comprehensive historical overview of ransomware, spanning from its inception in 1989 to recent attacks in 2021, surpassing existing studies in depth and scope (Vehabovic et al., 2022). Additionally, it provides a detailed review of detection, prevention, mitigation, and recovery methods, surveying a significant number of 2022 of papers for up-to-date insights, unlike previous works (McIntosh et al., 2021; Oz et al., 2021). Furthermore, it distinguishes itself from prior research by covering a broader spectrum of ransomware aspects, from history to recovery, rather than focusing solely on mitigation or defense (McIntosh et al., 2021; Oz et al., 2021). The methodology, historical background, state-of-the-art detection methods, and future research suggestions are outlined in subsequent sections, culminating in a comprehensive understanding of ransomware and its countermeasures.

The execution of a ransomware attack unfolds through several distinct stages.

Figure 2 delineates the sequential flow of activities essential for orchestrating such an assault.



The ransomware attack unfolds through distinct phases, beginning with exploitation and infection, where the attacker identifies vulnerabilities within the victim's computer system. This may involve leveraging malicious email attachments or exploit kits, such as the Angler exploit kit utilized by cryptolocker ransomware to exploit common vulnerabilities in Adobe Flash and Internet Explorer. Once vulnerabilities are identified, the delivery and execution stage commence, involving the installation and execution of the ransomware code on the victim's system. Upon execution, the ransomware establishes a connection with the attacker via a command-and-control mechanism to perpetrate further damage.

Following execution, backup spoliation ensues, where the ransomware identifies and removes backup files and folders, thwarting attempts to restore compromised files. For instance, CryptoLocker and Locky employ tools like vssadmin to delete volume shadow copies from Windows systems. Subsequently, file encryption takes place, facilitated by a secure key exchange with the command-and-control server, utilizing robust encryption algorithms like AES 256 or RSA 1024 to lock files on the local system. Some ransomware variants, like SamSam, perform encryption locally without accessing a command-and-control server.

Finally, the hacker notifies the victim of the attack and provides instructions for ransom payment. This typically occurs after backup files are removed and main files are encrypted. Victims are often given a short timeframe to comply, with failure resulting in escalated ransom demands. Payment instructions are usually stored on the hard drive or within infected file folders, sometimes in specific locations. To evade detection, the malicious executable file self-deletes from the infected system, eradicating forensic evidence that could aid in reconstructing the attack or defending against future malware threats.

Methodology

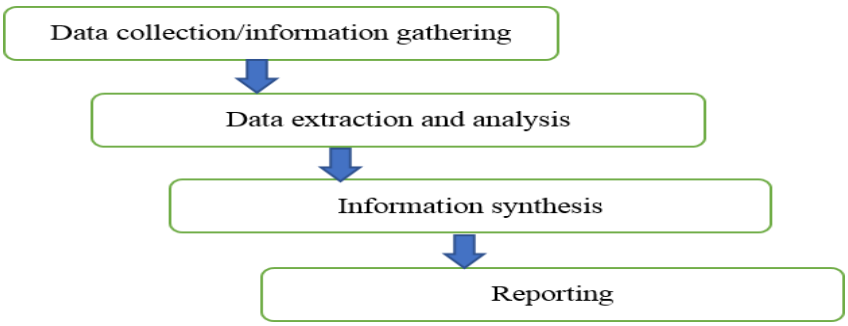
The methodology employed to achieve the paper's overarching objectives encompassed several phases: data collection/information gathering, data extraction/analysis, information synthesis, and reporting. Figure 3 delineates the research process flow, illustrating the interplay between these phases.

Data collection commenced by selecting pertinent and contemporary journal and conference papers from reputable databases such as IEEE, Springer, MDPI, Elsevier, IET, and Archive.org. Additionally, university-based journals,

theses/dissertations, and blogs from esteemed organizations like Microsoft, Crowdstrike, Symantec, and Techspot were considered. These materials were categorized into two main groups: non-technical sources, providing general information on ransomware for sections like introduction and ransomware history/chronology of attacks, and technical sources, focusing on proposed solutions categorized into detection, prevention, mitigation, and recovery. Detection papers were further categorized into artificial intelligence (AI)-based and non-AI-based approaches, with AI-based approaches subdivided into machine learning, deep learning, and artificial neural networks methods, while non-AI approaches were grouped into packet and traffic analysis categories.

Data extraction entailed a meticulous analysis and summary of each technical paper, identifying the addressed problem, objectives, methodology, achieved results, and limitations. Information synthesis identified similarities or correlations among papers within each group, highlighting improvements or resolutions to previous limitations.

In the reporting phase, papers addressing similar issues or employing similar techniques were grouped together, and their reviews were presented within the same paragraph. This organization facilitated coherent communication and enhanced the paper's readability, ensuring readers gained a clear understanding of the discussed concepts.



An Overview of Historical Development and Chronological Incidents

Ransomware has inflicted havoc on over 2 million victims spanning various sectors including health, business, education, and government. Notorious among these is WannaCry, which encrypts user data, leaving behind only two files: the encrypted file itself and a document containing ransom payment instructions. This document also threatens deletion of hijacked data if the ransom is not paid. WannaCry's modus operandi involves opening an original file,

reading its contents, creating an encrypted version, and subsequently closing the file (Scaife et al., 2016). India bore the brunt of the WannaCry onslaught, with states like Madhya Pradesh, Maharashtra, and Delhi enduring 32.63%, 18.84%, and 8.76% of the total attacks on the country respectively (eScan, 2017). High-profile corporations including FedEx, Nissan, and railway companies in Germany and Russia, along with NHS organizations in the United Kingdom, were among the targets. Moreover, universities and students in China, as well as prominent internet service providers like RailTel and Vodafone, suffered significant damages (Mohurle & Patil, 2017).

Table 1 presents a chronological account of major ransomware attacks, offering crucial insights into ransomware evolution, including the emergence year, name, mode of attack, propagation method, encryption strategy, and payment methods employed by victims.

Year	**Ransomware Name**	**Attack mode**	**Mode of spread**	**encryption strategy**	**Ransom payment method**
1989	AIDS Trojan	Encryption of file names	Infected floppy disk	Symmetric encryption	$189 postal order
2005	Trojan PGPcoder	File encryption	Spam email attachment	Asymmetric RSA-1024 encryption	N/A
2006	Trojan Cryzip	Creates password-protected archives of infected files	Spam email attachment	Password locking	No payment; malware code includes password
	Archievus	Encryption of **My Documents** folder	Phishing emails	Asymmetric RSA-1024 encryption	Purchase of 30-digit recovery password
2007	Locker	Display of pornographic image on the machine	Phishing attack	AES and RSA	SMS text message or calling a premium-rate phone number
2008	GPcode.AK	File encryption of subdirectory	Email phishing	Asymmetric RSA-1024 encryption	$100 to $200 in e-gold or Liberty Reserve
2011	60,000 new samples	Varying attack modes	Different modes of spread	Varying encryption and locking methods	Anonymous payment services
2012	Reveton	Password stealing	Clicking malicious link	Malicious JavaScript files	Around $300

	Trojan.Randsom.C	Device locking	N/A	N/A	calling a premium-rate phone number to reactivate Windows license
2013	CyptoLocker	File encryption	Gameover ZeuS banking Trojan botnet;	public and private cryptographic keys	Two Bitcoins (or $100), CashU, Ukash,
			malicious email		Paysafecard, and MoneyPak $150 via Perfect Money or QIWI Visa Virtual Card number earned $34,000 in its first month more than $1,000,000 $500 Unspecified amount in bitcoin $300 1 bitcoin $100 $300 in bitcoin $280 bitcoin $500-$600 0.5 bitcoin
	Locker	File encryption	Spam campaigns	AES	
			Spear phishing		

2014	CryptoDefens	File encryption		RSA-2048	
	e		email		
	CryptoWall	File encryption	Infected USB drive,	RSA-2048	
			email, malicious		
			executables,		
			malicious websites		
2015	LockerPin	Device locking	Adult entertainment	AES	
			app		
	Linux.Encoder	Encryption of	Exploits the flaw in	AES and RSA	
	.1	data and web	Magento shopping		
		applications	cart software		
		files			
2016	Petya	File overwriting	MEDoc tax and	Master boot record	
		and full hard	accounting software	(MBR) and file	
		disk encryption		encryption	
	KeRanger	File encryption	Infected web link	RSA	
	Xbot	File encryption	SMS messages	N/A	
		and stealing			
		online banking			
		details			
2017	WannaCry	File and device	Unknown	Hybrid (AES and	
		encryption		RSA)	
	Bad Rabbit	Device locking	Drive-by-download	Locks users'	
			on infected	devices when they	
			websites	click on alicious	
				Adobe Flash	
				installer	
2018	GandCrab	File encryption	Infected phishing	Installs on a device	
			email, Microsoft	and encrypts user	
			Office macros,	files when they	
			VBScript and	access infected	
			ransomware-as-a-	email	
			service		
	Katyusha	File encryption	Malware trojan	Infects networks	
			encrypts and adds	using EternaBlue	
			'Katyusha'	and DoublePulsar	
			extension to	exploits	
			infected files		
	Ryuk	File encryption	Massive spam	Symmetric AES-	15-50 bitcoins
			attacks and exploit	256 and asymmetric	
			kits	RSA-2048	
				encryption	
2019	Prolock/	File	Qakbot Trojan	Asymmetric RSA-	Bitcoin
	PwndLocker	lock/encryption		2048 encryption	
	LockerGoga	File encryption	Logs users out of	Cryptographic	N/A

		and file wiping	systems, encrypts	encryption and	
			files and deactivate	deletion of infected	
			devices	files	
	PewCrypt	File encryption	Spam email	Symmetric 256-bit	Free
			messages	Advanced	
				Encryption Scheme	
				(AES-256)	
	Dharma v2019	File encryption	Malicious email	Symmetric AES-	N/A
				256 algorithm	
2020	Nefilim	File encryption	Remote desktop	AES-256	Via email
			protocol (RDP)	encryption for	communication
			attack	victim's files; RSA-	
				2048 algorithm to	
				encrypt the AES-	
				256 keys	
	Ransomware	Attack mode	Mode of spread	encryption strategy	Ransom payment
	Name				method
	Paradise	File encryption	Spam message	RSA-1024 and	No ransom. Tools
	v2020		containing internet	RSA-2048	are available to
			query attachments	algorithms	retrieve encrypted
					files
	Maze	File encryption	Exploit kits such as	RSA and ChaCha20	$6m - $15m
			Fallout and Spelva	stream cipher	
	REvil	File	Phishing email and	AES or Salsa20	$70m in bitcoin
		encryption/file	malicious		
		blocking	attachment		
	Tycoon	Password	Insecure connection	RSA	N/A
		exploitation of	to an RDP server		
		file servers and	and a malicious		
		domain	(trojanized) Java		
		controllers	Runtime		
			Environment		
	NetWalker	Full Windows	Network-wide	Salsa20	More than $30m
		device	executable files and		total ransom since
		encryption	VBS script		March 2021
			attachments in		
			Corona virus		
			phishing emails.		
2021	Dark side	File encryption	VPN password	Lightweight	75 bitcoin or
		and data		Salsa20 with RSA-	$4.4m
		exfiltration		1024	
	ReVil	File encryption/file blocking	Vulnerability in Microsoft Exchange servers	AES or Salsa20	$50m in Monero cryptocurrency demanded
	Phoenix locker	File encryption on desktop and network shares	Spam emails	RSA-2048 algorithm	$40m

	ContiLocker	File encryption and data exfiltration	Via unprotected remote desktop protocol (RDP) port	RSA-4096 and AES-256-CBC	$2.6m
	Avaddon	File encryption, data exfiltration and DDoS	Malicious JavaScript files	AES-256	$40,000 or its equivalent in bitcoin

The data in the table illustrates a significant uptrend in both the development and deployment of ransomware attacks since the emergence of the first known instance in 1989. Typically, ransomware attacks involve the encryption of files and sub-directories, rendering them inaccessible to legitimate users while allowing the infected devices to continue functioning. Less frequently, ransomware may block users from accessing their devices altogether, even if the files remain accessible. Since 2013, there has been a consistent emergence of new malware variants, attributed to the accessibility of sophisticated tools enabling attackers to craft ransomware scripts easily, coupled with the substantial profits hackers accrue from ransom payments.

Currently, notable ransomware attacks such as Maze, REvil, Ryuk, Tycoon, and NetWalker pose significant threats (Ransomware Attacks, 2021). Several factors contribute to the proliferation of ransomware and the continual increase in ransomware attacks. These include the ready availability of powerful encryption algorithms (both symmetric and asymmetric), facilitating the creation of tailor-made ransomware for specific attacks or environments. Effective infection vectors like spam emails and malvertising further exacerbate the rapid spread of ransomware among users (Adamov & Carlson, 2017).

Moreover, the widespread accessibility of cryptocurrency enables victims to make ransom payments easily, with attackers able to convert cryptocurrency to cash without traceability. Additionally, the emergence of Ransomware as a Service (RaaS) allows less skilled attackers to obtain customized ransomware and track victims through user-friendly interfaces (Gellegos-Segovia et al., 2017). The creators of RaaS typically earn a percentage of profits from ransomware attacks launched via their platforms, further fueling the growth of ransomware.

Ransomware Detection

Research indicates a surge in ransomware attacks, which doubled in the first quarter of 2020, attributed to the widespread adoption of remote working necessitated by the COVID-19 pandemic. Many remote workers, operating outside the conventional office environment, often lack robust cybersecurity measures typically enforced in corporate

settings. Additionally, the use of personal devices, inadequately equipped with security tools such as antivirus software, firewalls, intrusion detection/prevention systems, password management utilities, and encryption software, further compounds the vulnerability landscape. Exploiting newly discovered vulnerabilities in systems and networks, ransomware attacks indiscriminately target small, medium, and large enterprises embracing remote work practices.

Beyond encrypting files and locking devices, ransomware employs sophisticated techniques for data exfiltration, exposing sensitive information and posing grave security and privacy risks. Instances like the ransomware attack on Dusseldorf University Hospital underscore the life-threatening consequences, where emergency services were disrupted, leading to the tragic death of a patient due to delayed treatment (Fingers, 2020). Moreover, ransom payments constitute a lucrative revenue stream for attackers, extorting millions of dollars from victims annually (Symantec Corporation, 2016). In 2020, ransomware attacks accounted for over 41% of cyber insurance claims, with projected total losses for organizations reaching $20 billion by the year's end (Potoroaca, 2020). These financial losses impede business growth, diverting resources that could otherwise be invested in productive ventures.

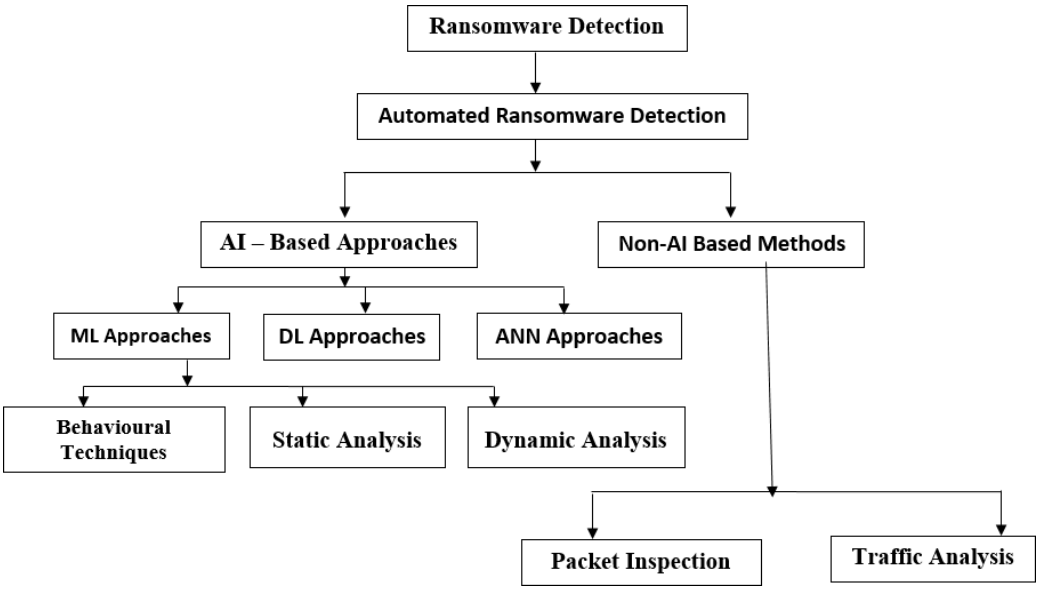
Given these pressing concerns, there is a critical need for effective methods for ransomware detection, prevention, mitigation, and recovery. Ransomware detection methods generally fall into two categories: automated and manual. Automated approaches leverage specialized tools to detect and report ransomware attacks, often equipped with functionalities to block such attacks. Conversely, manual detection methods entail regular inspection of files and devices for telltale signs of ransomware, such as changes in file extensions and unauthorized access attempts. This includes verifying file integrity and ensuring authorized users can access their devices and files. The flow of presentation in this section is illustrated in Figure 4.

Automated Ransomware Detection

Current strategies for detecting ransomware primarily center on system-level monitoring, often focusing on tracking file system characteristics. Automated ransomware detection methods can be broadly categorized into two main groups: artificial intelligence (AI)-based methods and non-artificial intelligence (non-AI)-based methods. AI-based approaches commonly utilize techniques such as machine learning (ML), deep learning (DL), and artificial neural networks (ANN) for identifying ransomware. Some tools employ variations or hybrid combinations of these

techniques to combat ransomware threats effectively. Non-AI methods, on the other hand, employ approaches like packet inspection and traffic analysis for ransomware detection.

Automated detection methods offer significant advantages, including the capability to detect, block, and recover from ransomware attacks autonomously, without requiring human intervention. These tools are known for their high accuracy and reliability in detecting, preventing, and recovering from ransomware incidents.



Artificial Intelligence-Based Methods

Artificial intelligence-based methods harness various techniques such as machine learning (including behavioral techniques, static analysis, and dynamic analysis), deep learning, and artificial neural networks to automate the detection of ransomware attacks.

Machine Learning Approaches

Machine learning (ML) represents a subset of artificial intelligence that empowers systems to learn from data and discern patterns, subsequently making decisions with minimal human intervention (Dontov, 2019). It facilitates

automated model building by enabling computers to make predictions based on patterns discerned from extensive datasets. ML algorithms possess the capability to adapt and refine predictions as dataset sizes increase. ML's capacity to predict based on file behavior, including both known and unknown datasets, renders it a valuable tool for detecting previously unidentified ransomware variants. However, to ensure reliable predictions, ML techniques typically necessitate a minimum dataset size ranging from 50 to 1,000 data points, as insufficient samples may lead to overfitting and biased predictions. Moreover, training ML algorithms demands considerable time investment.

File behavior detection constitutes a primary application of machine learning in ransomware detection. ML algorithms leverage specialized analyses, such as interactive debugging or post-mortem code execution, to extract significant and distinctive information, enabling the learning of legitimate or normal application behavior. ML-based ransomware detection tools conduct meticulous analyses of legitimate code execution, proficiently identifying malicious applications and prompting specific actions by leveraging their ability to discern between normal and abnormal program execution. The machine learning approaches examined in this study include behavioral techniques, static analysis, and dynamic analysis.

Behavioral Techniques

Behavioral techniques entail measuring normal application behavior from both user and resource perspectives, establishing a baseline representing routine operations within a computer system or network. This baseline encompasses activities like logins, file access, user and file behaviors, resource utilization, and other indicators of standard system activity (Acronis International, 2021). The duration of the learning process depends on the quantity of data required to establish a representative baseline. These tools identify and scrutinize behavioral anomalies deviating from the established baseline.

Juan et al. (2017) proposed a ransomware detection and prevention model utilizing machine learning techniques to identify abnormal behavioral patterns associated with Microsoft Windows-based ransomware, utilizing unstructured datasets from Ecuadorian control and regulatory institution (EcuCERT) logs. Feature selection techniques were employed to extract the most pertinent and discriminating information from the log data, forming the input feature set

for automatic learning algorithms. These algorithms model abnormal behavioral patterns, facilitating timely and reliable ransomware detection.

To address the limitations of signature-based methods in detecting rapidly evolving ransomware attacks, Shaukat & Ribeiro (2018) introduced RansomWall, a layered and hybrid approach leveraging static and dynamic analyses to generate a new feature set modeling ransomware behavior. This approach incorporates a robust trap layer for early ransomware detection, proving effective against zero-day attacks. Evaluation of RansomWall and Gradient Tree Boosting Algorithm on a dataset comprising 574 samples of 12 Microsoft Windows operating system-based cryptographic ransomware yielded a 98.25% detection rate and minimal false positives, with the capability to detect 30 zero-day attack samples.

Another notable tool, CryptoDrop, was developed for early ransomware detection based on suspicious file activity (Scaife et al., 2016). CryptoDrop employs a set of behavioral features to terminate processes altering large amounts of user data, integrating common ransomware features to achieve rapid detection with minimal false positives. Experimental analysis demonstrated CryptoDrop's efficacy in ransomware detection and prevention, preventing execution of ransomware files with minimal data loss.

However, CryptoDrop's limitation lies in its inability to discern the intent behind changes in file behavior, such as distinguishing between user-initiated encryption and ransomware activity. It flags legitimate activities like file compression, potentially leading to false positives. Future iterations of CryptoDrop should aim to differentiate between legitimate bulk transformation activities and malicious attacks.

Table 2 presents a summary of previous studies on behavioural techniques for ransomware detection.

Author	Problem addressed	Method used	Result	Limitation
Shaukat & Ribeiro (2018)	Ransomware detection	Layered and hybrid mechanism (RansomWall)	Suitable for detectingzero-day attacks	N/A

Scaife et al. (2016)	Ransomware detection	Evaluation of RansomWall and Gradient Tree Boosting Algorithm (CryptoDrop)	Median loss of only 10 files out of almost 5,100 tested files	Inability to determine the intent of attack indicated by changes in file behavior
Makinde et al. (2019)	To detect the susceptibility of a real network system to ransomware attack	Machine Learning	Correlation above 0.8	It simulated the behaviour of few users
Ahmad et al. (2019)	To distinguish members of the Locky ransomware	Behavioural ransomware detection approach (parallel classifiers)	Highly accurate detection with low false positive rate	N/A
Zahra & Sha (2019)	Detecting Cryptowall ransomware attack	Command and control (C&C) server black listing	Extracts TCP/IP header from web proxy server which serves as the gateway to TCP/IP traffic.	The model was not implemented to demonstrate its accuracy and effectiveness in detecting ransomware and their modes of attack against different operating system environments
Singh et al., (2022)	Detection of previously unknown ransomware families and classification of new	Examines access privileges in process memory to achieve easy and accurate detection of ransomware	accuracy ranges between 81.38% and 96.28%.	N/A

	ransomware attacks			

A variant of behavioral detection approaches employed a machine learning baseline model to simulate and predict individual network user behavior patterns at a micro level, aiming to detect potential vulnerabilities or actual ransomware attacks (Makinde et al., 2019). This method aimed to gauge the susceptibility of real network systems to ransomware attacks. Comparative analysis between results obtained from the simulated network and log data from the server in the real-life network system indicated a realistic model with a correlation above 0.8. However, a limitation of this approach is its simulation of the behavior of only a few users. Future endeavors should focus on employing tools for big data analytics to simulate the behavior of a larger number of users.

A recent behavioral ransomware detection approach utilized two parallel classifiers to categorize members of the Locky ransomware family based on their types (Ahmad et al., 2019). This method concentrated on early detection through behavioral analysis of ransomware network traffic to preempt ransomware from connecting to command-and-control servers and executing harmful payloads. The study employed a dedicated network to gather network information and extract relevant features of network traffic. Two independent classifiers processed the extracted features of the Locky ransomware family, operating on data at packet and datagram levels. Experimental outcomes demonstrated the method's efficacy in extracting valid features and offering highly accurate detection with a low false positive rate.

Zahra and Sha (2019) proposed a domain-specific framework for detecting Cryptowall ransomware attacks, centered on communication and behavioral analysis of the ransomware in an IoT environment. This framework employed command and control (C&C) server blacklisting to identify ransomware attacks. The method extracted TCP/IP headers from a web proxy server, serving as the gateway to TCP/IP traffic, and compared source and destination IP addresses with blacklisted IP addresses of Command-and-Control servers to detect ransomware attacks for IoT

devices. However, the model was not implemented to demonstrate its accuracy and effectiveness in detecting ransomware and their modes of attack against different operating system environments.

A very recent behavioral-based detection approach leveraged access privileges in process memory to achieve straightforward and accurate ransomware detection (Singh et al., 2022). This method also detected previously unknown ransomware families and classified new ransomware attacks based on the access privileges a file or application possessed and the area of memory it intended to access. Experimental results based on multiple algorithms yielded good detection accuracy ranging between 81.38% and 96.28%.

Static and Dynamic Analysis

A novel detection technique based on static analysis extracted features directly from raw ransomware binaries using frequent pattern mining (Khammas, 2020). It utilized the Gain Ratio technique to select 1000 features for optimal ransomware detection. The approach demonstrated a detection rate of 97.74%. Direct extraction of raw ransomware binaries significantly accelerated the detection speed.

An enhanced approach integrated dynamic analysis with machine learning, forming a hybrid ransomware detection model based on Markov model and Random Forest model (Hwang et al., 2020). This method utilized Windows API call sequence patterns to build a Markov model, extracting unique ransomware features. Subsequently, Random Forest was employed to minimize error rates. The two-stage mixed detection technique achieved good detection rates with an overall accuracy of 97.3%, a false positive rate (FPR) of 4.8%, and a false negative rate (FNR) of 1.5%.

A similar approach, known as EldeRan, utilized dynamic analysis to detect ransomware at runtime (Sgandurra et al., 2016). This technique leveraged runtime features exhibited by ransomware samples, which are consistent across ransomware families, to perform dynamic analysis and detection. Experimental results indicated an area under the ROC curve of 0.995, showcasing the approach's effectiveness in dynamic analysis and ransomware detection, even with incomplete datasets of ransomware families.

An improved technique for ransomware detection employed an integrated approach combining static and dynamic analysis (Bazrafshan et al., 2013). This analysis framework, based on support vector machines, utilized "run-time" and "static code" features for early detection of known and previously unknown ransomware variants. Experimentation across various ransomware types highlighted the integrated approach's superior ransomware detection compared to using static or dynamic analysis individually.

The integration of static and dynamic analysis has also been applied to analyze ransomware threats against mobile devices and perform mobile ransomware detection (Yang et al., 2015). This two-phase approach integrated data states and software execution on the critical test path of the Android API, combining static analysis to detect attack likelihood and runtime dynamic analysis to identify attack nature and potential data confidentiality violations.

A related study detected unknown ransomware by utilizing the most discriminating API calls to train a classifier (Sheen & Yadav, 2018). This approach demonstrated effectiveness with random forest, producing a detection rate of over 98%. However, a limitation lies in the class imbalance in the dataset due to differences in sample numbers between ransomware and benign data.

An enhanced approach integrated feature generation engines and machine learning to analyze malware samples, identifying malicious code intentions (Poudyal et al., 2018). Supervised ML techniques applied to features extracted from ransomware and benign binaries achieved detection accuracy ranging from 76% to 97%, with seven out of eight classifiers achieving a detection rate of at least 90%. The integration of static level analysis with data obtained from ASM-level and DLL-level features led to better ransomware detection rates.

Similarly, Dehghantanha et al. (2018) proposed a Decision Tree (J48) classifier known as NetConverse for high-speed and reliable Windows ransomware detection. Experimental results based on conversation-based network traffic features dataset showed a true positive detection rate of 97.1% using the Decision Tree (J48) classifier.

Static and Dynamic Analysis

Static and dynamic techniques can also be employed for real-time detection and prevention of ransomware attacks (Lalson et al., 2019). This technique offers robust and effective protection against a variety of ransomware, halting attacks before significant system or network damage occurs. However, it does not include file recovery capabilities, and ransomware may encrypt some files before detection or blocking.

Lee et al. (2022) addressed the ineffectiveness of static analysis against obfuscating ransomware and the low-speed detection of dynamic analysis by proposing statistical analysis using heuristics to distinguish between normal files and those attacked by ransomware. This approach provides real-time detection of known crypto-ransomware variants with minimal overhead during the detection process.

Recent ML approaches, such as the one proposed by Rani and Dhavale (2022), utilized multiple machine learning models to build an effective proof of concept for a product-specific ransomware, achieving an accuracy of 98.21%. Similarly, three different machine learning algorithms, namely decision tree (J48), random forest, and radial basis function (RBF), applied on 1000 dominant features obtained from raw, byte-level ransomware data using the gain ratio feature selection method, yielded a detection accuracy of approximately 98% (Khammas, 2022).

An enhanced approach integrated ensemble learning with a voting-based method, monitored memory usage, system call logs, CPU usage, and performed static and dynamic analysis of text, permissions, and network-based features (Ahmed et al., 2022). Experimental results demonstrated the technique's effectiveness in detecting unknown ransomware attacks based on malicious application behavior, even against adversarial evasion attacks.

Prevention, Mitigation, and Recovery Strategies

Detecting ransomware attacks after significant damage to data and systems is crucial, but it's equally important to implement strategies for preventing attacks and mitigating potential damages while ensuring recovery without paying

ransom. One such prevention method focuses on protecting computer systems by fooling attackers into encrypting large dummy files over an extended period, rendering the remaining contents of the file system inaccessible to ransomware (Patel & Tailor, 2020). Evaluation in real-time environments showed its effectiveness against ransomware attacks.

Similarly, a study proposed strategies to minimize susceptibility to Petya ransomware attacks, including blocking server message block (SMB) ports or disabling SMBv1, and preventing the execution of perfc.dat and psexec.dat files (Aidan et al., 2018). Additional measures involved using Software Restriction Policies (SRP) to disable certain binaries' execution paths and deploying email and web filtering on the network to restrict malicious files. However, file- and behavior-based detection methods struggle with unknown ransomware variants and cloud-based data storage attacks. Lee et al. (2019) addressed this by proposing a machine learning technique called file entropy analysis, capable of retrieving infected files synchronized to backup servers, irrespective of the host system's infection status.

Du et al. (2022) presented defensive strategies, including a hybrid machine learning solution integrating KNN and density-based algorithms, achieving a high ransomware attack prediction accuracy of 98%. Another method utilized random forest, recording a 99% accuracy rate. Recent advancements include system-architecture-based risk transference, relocating sensitive data to highly protected storage, minimizing susceptibility to ransomware attacks (Sreejith Gopinath & Aspen Olmstead, 2022).

Gómez-Hernández et al. (2022) introduced an enhanced tool deploying "honey files" near sensitive system files for early ransomware detection. This tool was adapted to Windows platforms, with improvements in system-wide "honey file" management and dynamic white-/black-lists, reducing the need for human intervention during attacks. WmRmR (weighted minimum Redundancy maximum Relevance) is a mitigation strategy estimating the importance of dominant features in ransomware attack data (Ahmed et al., 2022). It employs a hybrid solution integrating enhanced minimum redundancy maximum relevance (EmRmR) and Term Frequency-Inverse Document Frequency (TF-IDF), achieving early detection with low false positives.

Simple techniques for recovery from ransomware attacks have also been proposed. Wecksten (2016) suggested a method for easy recovery by renaming the system tool handling shadow copies of files, regardless of the attacker's tools on the victim's system. Kim et al. (2022) enabled partial recovery of master keys used by attackers to launch Hive ransomware, without requiring the attacker's RSA private key or paying ransom. Al-Dwairi et al. (2022) proposed a framework for efficient recovery of compromised XML documents using distributed storage and access control mechanisms, demonstrating efficient processing time, CPU utilization, and memory usage.

These strategies and techniques collectively offer a comprehensive approach to preventing, mitigating, and recovering from ransomware attacks, ensuring minimal disruption and data loss.

Future Research Directions

The path enumeration technique proposed by Lalson et al. (2019) offers effective decoy file creation but requires optimization for very large file systems. Balancing file size and computation time is crucial, with adjustable thresholds catering to system peculiarities. Future research should explore enhancing techniques for detecting multi-stage crypto ransomware attacks, prioritizing security for production network devices via cascaded network segmentation (Zimba et al., 2018). Moreover, detecting network-level ransomware attacks, particularly those using encrypted channels like HTTPS, demands attention.

Makinde et al.'s (2019) simulation limitations due to few users highlight the need for big data analytics tools to simulate a larger user base. Sheen and Yadav's (2018) imbalance issue in ransomware datasets warrants exploration on balanced datasets using the same classifiers for improved outcomes.

Aragom et al.'s (2016) Deep Packet Inspection, though accurate, currently supports only static analysis and could benefit from dynamic analysis implementation in a software-defined network for real-time detection. Vinayakumar et al.'s (2017) simple MLP network results prompt future studies to employ more complex architectures in distributed environments. Additionally, Zahra and Sha's (2017) IoT ransomware detection technique requires prototyping and deployment in real-world environments for validation and refinement.

Alzahrani et al.'s (2018) Randroid accuracy could be bolstered by adding more malicious image and string samples to the databases. Similarly, text recognition techniques may enhance dynamic payload detection. Chen et al.'s (2018) constraints on mobile ransomware research signal a need for comprehensive datasets and deeper insights into real-time attack scenarios.

Future research on Rani and Dhavale's (2022) work could integrate the model with ELK for practical ransomware detection tools. Ahmed et al.'s (2022) study could explore distinct features of known ransomware, qualitative strategies for feature extraction, and performance metrics for defence mechanisms. Kim et al.'s (2022) byte-frequency-based indicators should address detection limitations for smaller files and ransomware variants like DMA Locker2.

Nazarovs et al.'s (2022) Bayesian Neural Network, requiring human intervention, could evolve into an automated solution for system isolation during ransomware attacks. Abbasi et al.'s (2022) particle swarm optimization could benefit from additional feature sets and dataset variations to capture diverse ransomware traits. Exploring system call sequences as classification features may further improve ransomware family identification.

Conclusion

Ransomware attacks continue to inflict significant damage on computers, data, and information systems, leading to unauthorized access, disclosure, and destruction of critical resources. Both individuals and organizations have suffered severe financial losses and reputational harm due to these attacks. In response, numerous methods have been proposed for accurate, timely, and reliable ransomware detection.

This study has provided readers with a comprehensive introduction to ransomware detection by discussing the background of ransomware and presenting a chronological overview of ransomware attacks. The critical review of recent papers offers readers an up-to-date understanding of the current trends in automated ransomware detection. This knowledge equips readers with insights into the state-of-the-art approaches for ransomware detection, prevention, mitigation, and recovery.

Furthermore, this study has highlighted future research directions, identifying open issues and potential research problems in the detection, prevention, mitigation, and recovery from ransomware attacks. By addressing these challenges, future research endeavors can contribute to enhancing the effectiveness and resilience of ransomware defense mechanisms.

References:

[1]. Maharjan, R., Chy, M. S. H., Arju, M. A., & Cerny, T. (2023, June). Benchmarking Message Queues. In *Telecom* (Vol. 4, No. 2, pp. 298-312). MDPI. https://doi.org/10.3390/telecom4020018

[2]. Chy, M. S. H., Arju, M. A. R., Tella, S. M., & Cerny, T. (2023). Comparative Evaluation of Java Virtual Machine-Based Message Queue Services: A Study on Kafka, Artemis, Pulsar, and RocketMQ. *Electronics*, *12*(23), 4792. https://doi.org/10.3390/electronics12234792

[3]. Rahman, M., Chy, M. S. H., & Saha, S. (2023, August). A Systematic Review on Software Design Patterns in Today's Perspective. In *2023 IEEE 11th International Conference on Serious Games and Applications for Health (SeGAH)* (pp. 1-8). IEEE. https://doi.org/10.1109/SeGAH57547.2023.10253758

[4]. Shivakumar, S. K., & Sethii, S. (2019). *Building Digital Experience Platforms: A Guide to Developing Next-Generation Enterprise Applications*. Apress.

[6]. Sethi, P. Karmuru, & Tayal.(2023). Analyzing and Designing a Full-Text Enterprise Search Engine for Data-Intensive Applications. *International Journal of Science, Engineering and Technology*, *11*. https://www.ijset.in/wp-content/uploads/IJSET_V11_issue6_628.pdf

[7]. Sethi, S., Panda, S., & Kamuru, R. (2023). Comparative study of middle tier caching solution. *International Journal of Development Research*, *13*(11), 64225-64229.

[8]. Gitte, M., Bawaskar, H., Sethi, S., & Shinde, A. (2014). Content based video retrieval system. *International Journal of Research in Engineering and Technology*, *3*(06), 123-129.

[9]. Gitte, M., Bawaskar, H., Sethi, S., & Shinde, A. (2014). Content based video retrieval system. *International Journal of Research in Engineering and Technology*, *3*(06), 123-129.

[10]. Sethi, S., & Shivakumar, S. K. (2023). DXPs Digital Experience Platforms Transforming Fintech Applications: Revolutionizing Customer Engagement and Financial Services. *International Journal of Advance Research, Ideas and Innovations in Technology*, *9*, 419-423.

[11]. Jhurani, J. REVOLUTIONIZING ENTERPRISE RESOURCE PLANNING: THE IMPACT OF ARTIFICIAL INTELLIGENCE ON EFFICIENCY AND DECISION-MAKING FOR CORPORATE STRATEGIES.

[12]. Jhurani, J. Enhancing Customer Relationship Management in ERP Systems through AI: Personalized Interactions, Predictive Modeling, and Service Automation.

[13]. Jhurani, J. DRIVING ECONOMIC EFFICIENCY AND INNOVATION: THE IMPACT OF WORKDAY FINANCIALS IN CLOUD-BASED ERP ADOPTION.

[14]. Smith, J. D. Influence of Self-Efficacy, Stress, and Culture on the Productivity of Industrial Sales Executives in Latin American Sales Networks.

[15]. Miah, S., Rahaman, M. H., Saha, S., Khan, M. A. T., Islam, M. A., Islam, M. N., ... & Ahsan, M. H. (2013). Study of the internal structure of electronic components RAM DDR-2 and motherboard of nokia-3120 by using neutron radiography technique. *International Journal of Modern Engineering Research (IJMER)*, *3*(60), 3429-3432

[16]. Rahaman, M. H., Faruque, S. B., Khan, M. A. T., Miah, S., & Islam, M. A. (2013). Comparison of General Relativity and Brans-Dicke Theory using Gravitomagnetic clock effect. *International Journal of Modern Engineering Research*, *3*, 3517-3520.

[17]. Miah, M. H., & Miah, S. (2015). The Investigation of the Effects of Blackberry Dye as a Sensitizer in TiO2 Nano Particle Based Dye Sensitized Solar Cell. *Asian Journal of Applied Sciences*, *3*(4).

[18]. Miah, S., Miah, M. H., Hossain, M. S., & Ahsan, M. H. (2018). Study of the Homogeneity of Glass Fiber Reinforced Polymer Composite by using Neutron Radiography. *Am. J. Constr. Build. Mater*, *2*, 22-28.

[19]. Miah, S., Islam, G. J., Das, S. K., Islam, S., Islam, M., & Islam, K. K. (2019). Internet of Things (IoT) based automatic electrical energy meter billing system. *IOSR Journal of Electronics and Communication Engineering*, *14*(4 (I)), 39-50.

[20]. Nadia, A., Hossain, M. S., Hasan, M. M., Islam, K. Z., & Miah, S. (2021). Quantifying TRM by modified DCQ load flow method. *European Journal of Electrical Engineering*, *23*(2), 157-163.

[21]. Miah, S., Raihan, S. R., Sagor, M. M. H., Hasan, M. M., Talukdar, D., Sajib, S., ... & Suaiba, U. (2022). Rooftop Garden and Lighting Automation by the Internet of Things (IoT). *European Journal of Engineering and Technology Research*, *7*(1), 37-43.
DOI: https://doi.org/10.24018/ejeng.2022.7.1.2700

[22]. Prasad, A. B., Singh, S., Miah, S., Singh, A., & Gonzales-Yanac, T. A Comparative Study on Effects of Work Culture on employee satisfaction in Public & Private Sector Bank with special reference to SBI and ICICI Bank.

[23]. Ravichandra, T. (2022). A Study On Women Empowerment Of Self-Help Group With Reference To Indian Context.https://www.webology.org/data-cms/articles/20220203075142pmwebology%2019%20(1)%20-%2053.pdf

[24]. Kumar, H., Aoudni, Y., Ortiz, G. G. R., Jindal, L., Miah, S., & Tripathi, R. (2022). Light weighted CNN model to detect DDoS attack over distributed scenario. *Security and Communication Networks*, *2022*. https://doi.org/10.1155/2022/7585457

[25]. Ma, R., Kareem, S. W., Kalra, A., Doewes, R. I., Kumar, P., & Miah, S. (2022). Optimization of electric automation control model based on artificial intelligence algorithm. *Wireless Communications and Mobile Computing*, *2022*. https://doi.org/10.1155/2022/7762493

[26]. Devi, O. R., Webber, J., Mehbodniya, A., Chaitanya, M., Jawarkar, P. S., Soni, M., & Miah, S. (2022). The Future Development Direction of Cloud-Associated Edge-Computing Security in the Era of 5G as Edge Intelligence. *Scientific Programming*, *2022*. https://doi.org/10.1155/2022/1473901

[27]. Al Noman, M. A., Zhai, L., Almukhtar, F. H., Rahaman, M. F., Omarov, B., Ray, S., ... & Wang, C. (2023). A computer vision-based lane detection technique using gradient threshold and hue-lightness-saturation value for an autonomous vehicle. *International Journal of Electrical and Computer Engineering*, *13*(1), 347.

[28]. Patidar, M., Shrivastava, A., Miah, S., Kumar, Y., & Sivaraman, A. K. (2022). An energy efficient high-speed quantum-dot based full adder design and parity gate for nano application. *Materials Today: Proceedings*, *62*, 4880-4890.https://doi.org/10.1016/j.matpr.2022.03.532

[29]. Pillai, A. S. (2023). Advancements in Natural Language Processing for Automotive Virtual Assistants Enhancing User Experience and Safety. *Journal of Computational Intelligence and Robotics*, *3*(1), 27-36.

[30]. Rahman, S., Mursal, S. N. F., Latif, M. A., Mushtaq, Z., Irfan, M., & Waqar, A. (2023, November). Enhancing Network Intrusion Detection Using Effective Stacking of Ensemble Classifiers With Multi-Pronged Feature Selection Technique. In *2023 2nd International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE)* (pp. 1-6). IEEE. https://doi.org/10.1109/ETECTE59617.2023.10396717

[31]. Latif, M. A., Afshan, N., Mushtaq, Z., Khan, N. A., Irfan, M., Nowakowski, G., ... & Telenyk, S. (2023). Enhanced classification of coffee leaf biotic stress by synergizing feature concatenation and dimensionality reduction. *IEEE Access*.

https://doi.org/10.1109/ACCESS.2023.3314590

[32]. Irfan, M., Mushtaq, Z., Khan, N. A., Mursal, S. N. F., Rahman, S., Magzoub, M. A., ... & Abbas, G. (2023). A Scalo gram-based CNN ensemble method with density-aware smote oversampling for improving bearing fault diagnosis. *IEEE Access*, *11*, 127783-127799.

https://doi.org/10.1109/ACCESS.2023.3332243

[33]. Irfan, M., Mushtaq, Z., Khan, N. A., Althobiani, F., Mursal, S. N. F., Rahman, S., ... & Khan, I. (2023). Improving Bearing Fault Identification by Using Novel Hybrid Involution-Convolution Feature Extraction with Adversarial Noise Injection in Conditional GANs. *IEEE Access*. https://doi.org/10.1109/ACCESS.2023.3326367

[34]. Latif, M. A., Mushtaq, Z., Arif, S., Rehman, S., Qureshi, M. F., Samee, N. A., ... & Al-masni, M. A. Improving Thyroid Disorder Diagnosis via Ensemble Stacking and Bidirectional Feature Selection. https://www.techscience.com/cmc/v78n3/55928/html

[35]. Gunasekaran, K. P., Babrich, B. C., Shirodkar, S., & Hwang, H. (2023, August). Text2Time: Transformer-based Article Time Period Prediction. In *2023 IEEE 6th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)* (pp. 449-455). IEEE. https://doi.org/10.1109/PRAI59366.2023.10331985

[36]. Gunasekaran, K., & Jaiman, N. (2023, August). Now you see me: Robust approach to partial occlusions. In *2023 IEEE 4th International Conference on Pattern Recognition and Machine Learning (PRML)* (pp. 168-175). IEEE. https://doi.org/10.1109/PRML59573.2023.10348337

[37]. Kommaraju, V., Gunasekaran, K., Li, K., Bansal, T., McCallum, A., Williams, I., & Istrate, A. M. (2020). Unsupervised pre-training for biomedical question answering. *arXiv preprint arXiv:2009.12952*. https://doi.org/10.48550/arXiv.2009.12952

[38]. Bansal, T., Gunasekaran, K., Wang, T., Munkhdalai, T., & McCallum, A. (2021). Diverse distributions of self-supervised tasks for meta-learning in NLP. *arXiv preprint arXiv:2111.01322*. https://doi.org/10.48550/arXiv.2111.01322

[39]. Mahalingam, H., Velupillai Meikandan, P., Thenmozhi, K., Moria, K. M., Lakshmi, C., Chidambaram, N., & Amirtharajan, R. (2023). Neural attractor-based adaptive key generator

with DNA-coded security and privacy framework for multimedia data in cloud environments. *Mathematics*, *11*(8), 1769. https://doi.org/10.3390/math11081769

[40]. Padmapriya, V. M. (2018). Image transmission in 4g lte using dwt based sc-fdma system. *Biomedical & Pharmacology Journal*, *11*(3), 1633. https://dx.doi.org/10.13005/bpj/1531

[41]. Padmapriya, V. M., Thenmozhi, K., Praveenkumar, P., & Amirtharajan, R. (2020). ECC joins first time with SC-FDMA for Mission "security". *Multimedia Tools and Applications*, *79*(25), 17945-17967. https://doi.org/10.1007/s11042-020-08610-5

[42]. Padmapriya, V. M., Sowmya, B., Sumanjali, M., & Jayapalan, A. (2019, March). Chaotic Encryption based secure Transmission. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)* (pp. 1-5). IEEE. https://doi.org/10.1109/ViTECoN.2019.8899588

[43]. Padmapriya, V. M., Thenmozhi, K., Praveenkumar, P., & Amirtharajan, R. (2022). Misconstrued voice on SC-FDMA for secured comprehension-a cooperative influence of DWT and ECC. *Multimedia Tools and Applications*, *81*(5), 7201-7217. https://doi.org/10.1007/s11042-022-11996-z

[44]. Padmapriya, V. M., Thenmozhi, K., Avila, J., Amirtharajan, R., & Praveenkumar, P. (2020). Real Time Authenticated Spectrum Access and Encrypted Image Transmission via Cloud Enabled Fusion centre. *Wireless Personal Communications*, *115*, 2127-2148. https://doi.org/10.1007/s11277-020-07674-8

[45]. Padmapriya, V. M., Priyanka, M., Shruthy, K. S., Shanmukh, S., Thenmozhi, K., & Amirtharajan, R. (2019, March). Chaos aided audio secure communication over SC-FDMA system. In *2019 International Conference on Vision Towards Emerging Trends in*

Communication and Networking (ViTECoN) (pp. 1-5). IEEE.

https://doi.org/10.1109/ViTECoN.2019.8899413