International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. , No. (Mon-20..)

http://dx.doi.org/10.12785/ijcds/XXXXXX

Analyzing Machine Learning Techniques in Detecting and Preventing Ransomware

Alexander Veach[1] **and Munther Abualkibash**[1]

[1]*School of Information Security and Applied Computing, Eastern Michigan University, 211 Sill Hall, Michigan, Ypsilanti 48197, United States of America*

Received Mon. 20, Revised Mon. 20, Accepted Mon. 20, Published Mon. 20

Abstract: Ransomware is one of the biggest threats to organizations in the current cybersecurity landscape with severe attacks causing millions of United States Dollars in damages. Many have looked to newer technology, such as machine learning and artificial intelligence, to identify and prevent these costly attacks. This review gathers and analyzes one hundred and five research papers to understand what is being done in the field and the results of the reported experiments. The papers were then separated into groups depending on the contents of the research. The suggested frameworks and reviews are judged qualitatively, and the experiment groups were judged quantitatively by using simple statistics generated by the average reported accuracy of each machine learning classifier is calculated to give a simple overview of popular classifiers and their performance. This data was then analyzed further by generating median, mode, and standard deviation to better understand the reported performance of each classifier that appeared enough to make reasonable inferences. Furthermore, this paper gives a generalized overview of commonly suggested implementations, and analyzes current commercial solutions to show how these techniques have been adopted by major security providers such as Microsoft and CrowdStrike. This paper concludes with suggestions of commonly successful classifiers in traditional testing, alongside suggestions for future research.

Keywords: Ransomware, Machine Learning, Artificial Intelligence, Cybersecurity

1. INTRODUCTION

Ransomware is one of the most costly attacks in the modern business landscape. When done correctly a business must take months, if not years, to fully recover from the damage done by a severe ransomware attack. According to Cloudwards, a cybersecurity firm, “Ransomware cost the world $20 billion in 2021” and “that number is expected to rise to $265 billion by 2031” [1]. All it takes for hackers to gain access to a business’ system is a vulnerability in their network. Once access has been gained, it becomes a race against time to detect the intrusion and isolate the malicious software before the damage spreads. This threat is one of the most prominent in the current cybersecurity landscape, with many researchers and business leaders questioning what the best course of action is to prevent these attacks.

When it comes to defending against these attacks, there are two parts of the common defensive strategy: preventative measures such as frameworks focusing on limiting access to systems that can be affected by an attack and reactive measures where analysis is done on system logs for potential signs of an attack. If done correctly, the attack is prevented or detected and stopped before serious damage can be done. In research circles, and select commercial offerings, machine learning and artificial intelligence has been experimented with and used to detect signs of ransomware and theoretically react faster than a person could. This has become increasingly necessary as many businesses struggle to deal with the massive amounts of data they generate and process across multiple locations. This paper analyzes the current research using quantitative and qualitative methods to draw conclusions from a bevy of published works on the topic and make suggestions for future research based upon their findings and statistical inference.

2. A BRIEF PRIMER ON IMPORTANT CONCEPTS

This topic intersects two massive fields of study, malware analysis with a focus on ransomware and the field of machine learning and artificial intelligence. To ensure clarity of the work provided, this section will explain the essentials of these critical concepts.

A. A Primer on Ransomware

Ransomware is a type of malicious software that uses various methods to extort capital from affected people and organizations. There are many types of ransomware, with many different methods of attack. These methods can involve gathering information for blackmail, changing login credentials, or more commonly the encryption of critical data in a way that massively damages a company. Often

E-mail address: aveach1@emich.edu, mabualki@emich.edu

these attacks include a message declaring their data as forfeit unless a ransom is paid, or specific actions are taken. Encryption based ransomware, better known as cryptographic ransomware, is more common with several stories of critical infrastructure in the United States being affected by it such as the Colonial pipeline in 2021 [2], or more recently the ransomware attack on London hospitals[3].

The way blackmail-based ransomware works is by malicious software implemented inside a critical system, which then either exfiltrates as much key data initially or gathers the data over a long period of time to avoid detection. Once enough data is gathered to coerce the target, the attacker will send a letter of ransom outlining an ultimatum of paying the ransom or have your information published/sold online. Cryptographic ransomware, commonly referred to as crypto ransomware, works similarly to the blackmail-based ransomware in the initial stages. However, once the malware infiltrates the system it attempts to gain elevated permissions to use encryption functions on as much data as it can access. During this the ransomware can either exfiltrate the data or focus on encrypting as much data as possible. To prevent easy decryption of the data external encrypting algorithms are often used via API calls, with the decryption key being held by the attackers if any such decryption key exists.

One of the major issues when it comes to stopping ransomware is that the methods used to attack and encrypt are rapidly changing, common detection methods can be thwarted by the evolution of these attacks. Alongside this, many ransomware attacks on major organizations are specifically targeted and often done by hacker groups. This makes it difficult to stop if they have accurately mapped the network and permissions of a business's systems and can lead to massive breaches and disruptions to critical organizational services.Currently the best methods are not reactive but rather preventative. By following a zero-trust model or similar framework, which assumes there will be a breach, a company can design their networks to limit the potential damage caused by a ransomware attack. The reason this is the preferred method currently is because detecting an active ransomware attack and reacting to it before critical damage has been done is extremely difficult. This can be detected by noticing a spike in encryption traffic; however, by the time it is detected several systems could have been compromised. It is also difficult because encryption is a commonly used function in daily operating procedures, meaning that ransomware can potentially be overlooked as normal activity. Thus, AI technologies have been seen as a potential solution to increase the efficacy of reactive defense solutions as these technologies allow for an additional wall of defense, offering a solution that can look for key signs of ransomware attacks and report them immediately.

B. A Primer on Machine Learning and Artificial Intelligence

Machine learning (ML) and Artificial Intelligence (AI) are commonly used terms with a broad set of meanings. ML refers to the training of statistical decision models by analyzing recorded data sets with the goal of accurate predictions. The threshold for an accurate prediction depends on the subject and use of the model, but an accuracy of 95 percent or higher is generally preferred. AI refers to the trained model itself and its applied use to predict the outcome based on the information given, while ML is focused on the training and development of these models. These models then can predict outcomes based on the data given. In the case of this paper, one of the common frameworks used had a trained AI model receive a device's activity logs and looked for signs of ransomware activity. If the model detected these signs and returned a high enough confidence value, it would return a value stating that there is ransomware activity within the logs ingested. From this point either the software will either automatically quarantine the system affected, or inform a person to verify that the system has been infected immediately.

Most of the work that goes into creating these models is split up into two sections, the development of the training dataset and the testing of classification methods. Datasets are collections of information pertinent to the prediction of the outcome. In the case of ransomware some common predictors are the amount of encryption calls, API calls, and privilege escalation. These predictors are called features and make up a major portion of work that goes into developing these models. Thus, the features used should be able to detect the difference between infected and uninfected operating systems from the provided data. Another important part of predicting these outcomes is the weight assigned to each feature. Feature weight refers to the amount of variance the feature accounts for, or more simply how much effect that feature has on the result. Features with high weight have a large amount of effect on the predicted result, while those with lower weight scores have less effect on the result. The proper weighting of features has a major impact on the predictive capabilities of a model and is very important to analyze to ensure accuracy.

Beyond this, the last part of machine learning process is the classifier used. A classifier is a statistical equation used to analyze the features and determine the result of the data ingested. There is a multitude of classifiers that can be broken down into multiple groups. There are linear classifiers that take the features given and calculate the result based on a single equation. There are neural network classifiers that attempt to emulate human neural patterns to calculate the result which can quantify complex equations more accurately at the cost of speed and additional time to train. Another common classifier family is that of the ensemble methods, which as the name implies works by combining multiple predictive equations to better quantify the data processed. Depending on what is used in the ensemble method it can return high levels of accuracy with less computing than other high power processing methods such as neural networks.

The best classifier to be used will change based on a multitude of factors: the features analyzed, the complexity of the prediction, the speed needed, etc. This means that what works perfectly for one occasion might perform worse for another. Even predicting the same outcome, the classifier that works best might change depending on the features used and the desired outcome. Something that wants rapid training and decision making will prefer faster classifiers, while tasks more focused on accuracy over speed will want more process intensive tasks that have higher rates of accuracy and lower chances of false positives.

There is also another dimension of classifiers, those that use manual feature weights and those that calculate feature weights based on the training set. The latter is often faster and lends itself to data sets that are constantly being adjusted with the features changing due to shifts in the methods of detection. While static feature weights are often tuned for a specific task to find the best fit for prediction, and are more prone to error from variance over time. This works better for predictions that are more static, such as physics predictions and other calculations with fixed unchanging formulas.

3. RELATED WORK

The studies in the review section were analyzed and categorized to see how they were done, and their major conclusions were added below. A common trend in review literature was to focus more on the general concepts rather than the reported results, the reason often being cited is due to the variance in methods used and the values being reported. This review contains that information to showcase the reported accuracy of current studies, and aggregated this information to show what common trends have been established in research on the topic. However, these results are not definitive due to the methods used in collection and the differences in each reported results as cited by similar reviews, thus the information presented is better suited for assisting in the selection of classifiers for initial testing rather than a definitive statement of which classifiers should be used.

Alzahrani and Alghazzawi [4] in 2020 searched for Android malware detection using machine learning techniques. Specifically looking for research using deep learning methods. What they found was that in the Android space most of the research was focused on simpler methods such as random forest rather than deep learning, with only a handful of papers attempting it. They cite that deep learning has several weaknesses due to the amount of data required for accurate detection and the necessity of continuous updates to the dataset to get the most out of deep learning methods.

Bertia et al. [5] in 2022 researched common methods used to detect ransomware using machine learning. The study was more generic in scope and looked at recent research. They listed classifiers used in the studies they had found, then explained the results and methods used. Alongside this, they explain the ever-changing nature of malicious software and the issues that occur due to it and gave a general primer on the topic of ransomware. Sneha, Arya and Agarwal[6] similarly researched the topic in a general sense and gave similar explanations.

Thamer and Alubady [7] in 2021 specifically looked at ransomware attacks on healthcare systems and what can be done to mitigate the threat. This study explained the common vectors of attacks and suggested fixes and frameworks that many hospitals utilize to give a fuller understanding of the issues that healthcare services face when it comes to ransomware.

Moussaileb et al. [8] , Oz et al. [9], and Razaulla et al. [10] all broke down the evolution of ransomware, how ransomware works and is designed, and the potential defenses against it. In the research they thoroughly outline key detectors that can be used in machine learning to train a model to detect ransomware activity. Moussaileb also analyzed research on ransomware that targeted mobile systems. All studies offered suggestions of other preventative measures that can assist in a system's layers of defense.

Mcintosh et al [11] in 2021 did a thorough survey of ransomware studies at the time and outlined key information they noticed. Specifically, they noted that much of the research used generalized terms that sometimes overlapped with other terms and lacked decisive terminology. They also noted several different methods of defense from organizational configurations to machine learning techniques.

Ortloff, Vossen and Tiefenau [12] in 2021 readministered a survey done in the United States of America in Germany to see the potential cultural differences between the populations when it came to dealing with ransomware. They found that the US on average had more experience with ransomware attacks, and that German participants were more likely to restore the computer from a backup or use a tool to remove the malicious software than those surveyed in the US.

Davies, Macfarlane, and Buchanan [13] in 2021 analyzed anti-ransomware implementations outlined in research and their methods. They analyzed the performance of each and noted that many implementations struggled to accurately detect the difference between high entropy file types and encrypted files.

Bansal et al. [14] in 2020 used web search logs from Bing to try and detect users that had been affected by ransomware via their search queries. Using this method, they did a case study on the spread of Nemty, a ransomware that started spreading around August of 2019 and found that they could see the increase in searches per global region coinciding with the spread of the ransomware.

4. METHODOLOGY AND SAMPLE SELECTION

To analyze this topic, several research papers were gathered from scholarly sources totaling 128. These papers were then analyzed and papers that were not related or tenuously related were removed from the sample bringing the number to 123. A final round of selections culled papers from before 2019 to keep focus on recent research. The number of papers remaining after the final selections was 105. These papers were then classified into broad categories depending on the type of research. These categories were as follows: Experiments, Frameworks, and Reviews. Experiments were research papers focused on the applied use of machine learning to predict and stop ransomware attacks, frameworks were theoretical designs and standards focused on stopping ransomware through defensive strategies, and reviews were research focused on the analysis of other studies on the topic or closely related topics. The data reported by the experiment group was then aggregated to find the popular machine learning classifiers used, and the reported accuracy of commonly recurring classifiers to showcase what methods are traditionally successful.

Another category that stood out was a section specifically focused on the detection of ransomware on Android systems. A sizable group of twelve experiment papers were focused specifically on this topic, with an additional four review/framework papers covering this topic with sizable interest. These studies often cited the rising number of mobile endpoints using Android and the potential threats it faced[15]. Android is a major part of the internet-of-things so it is unsurprising that research would be done into securing these devices against ransomware which is a major pressing issue. The papers in this group often applied the techniques highlighted in non-mobile endpoint detection methods, such as analyzing rapid API calls, and the information contained within newly downloaded files. A benefit that may have assisted in the popularity of the android group in experiments was the publicly available dataset provided by the Canadian Institute of Cybersecurity for both 2017[16] and 2020[17] which was used in five of the experiments. By allowing use of this dataset others were able to experiment and modify their experiment to further the collective knowledge on the topic.

To generate statistical results from the classifiers used in the experiment group the reported accuracy was averaged in groups based on the classifier or classifier group depending on the results. Alongside this, groups that had less than ten results reported in the research surveyed were excluded from the descriptive statistics due to a high influence from outlier data. Alongside this the median and mode of the accuracy data is included to show a fuller breakdown of the trends in each group alongside the standard deviation to show the average variance. This information showcases the expected general accuracy of classifiers used to detect malicious ransomware activity; however, performance may vary due to a multitude of factors. Alongside this, machine learning has other critical factors such as recall and F1 scores which were not consistently reported in the surveyed works thus their exclusion in calculations. Thus, these numbers should not be taken as a concrete classification of performance but rather as a classification of general trends in performance via the common methods of detection against ransomware.

A. *Breakdown of Research Surveyed*

In the framework group there were 20 articles of research offering different theoretical implementations of defensive measures against ransomware. These frameworks were often focused on user controls and company policies rather than active defenses, and some works cited a potential for the use of machine learning to train defensive models that could analyze computer logs and potentially alert the company of incoming attacks. There were also 16 review papers which focused on this topic using different methods to classify and analyze future research. Most of these reviews covered recent research on machine learning for either general operating systems or focused on a singular system such as Android or Windows. In the experiment group there were 69 research papers, focused specially on using machine learning techniques to predict or respond to ransomware attacks. There were notably two sub-groups: those that provided their dataset publicly or used a publicly available dataset and those that outlined what was contained in their dataset but did not provide it. This paper, built off the information gathered, opts to analyze the currently suggested classifiers by reported performance to showcase the reported results of the assorted classifier against popular ransomware.

5. ANALYSIS OF RESULTS

A. *Preface*

Of the 69 pieces of research in the experiment group with reported results, 26 provided the dataset used with most being a combination of publicly available datasets and newer data points often used in a training dataset. The remaining 43 papers detailed how they generated the dataset but did not provide the specific dataset used to get their results. Using the outlined methods should theoretically return similar results as those reported, however an issue occurs when replicating results obtained from years prior as their is some variability when it comes to ransomware samples. Each ransomware sample is often modified or changed in slight ways which can alter the results in either a small way, or a larger way depending on the amount of variance. Alongside this, each major family of ransomware has slight differences which can alter the results if the same exact variants are not used. As mentioned prior[5], ransomware rapidly evolves to evade detection and become more optimal. This means that the ransomware of 2012 and the ransomware of 2020 may achieve a similar result but have enough differences that can make it difficult for a model trained on old data to predict against new methods. By providing the dataset it becomes easier to validate the work, and the publicly available data can assist in the building of testing sets for specific years or the creation of congregate datasets for future research.

In the surveyed papers, there were commonly used classifiers and classification families due to a mixture of factors. For posterity's sake, below is a breakdown of the commonly appearing families and individual classifiers, alongside a generalized overview of how they work.

Random forest, commonly abbreviated as RF, is an ensemble classifier that creates multiple decision trees that influence the final result. This method is commonly used across multiple fields and has a reputation for performing well on a wide-berth of tasks.

Decision Tree, commonly abbreviated as DT, is a linear classifier that creates a single flowchart structure based on the features provided and determines the result from that process. Decision Tree is commonly used in many ensemble classifiers, such as random forest, and is also used on its own for simple estimations.

K-Nearest Neighbor, or KNN, is a classifier that works with non-parametric data and determines classification based on distance from the nearest grouping of data. This method works well for the classification of data that is more prone to outliers.

Meta Algorithms is a label used to classify ensemble methods that alters traditional classifiers by using higher level alterations such as feature weight, feature selection, and other similar higher level methods. In the scope of this review, all algorithms under this classification use a statistical method to automatically assign feature weight and other meta characteristics based on classifier performance on the training data, which can lead to optimal feature weighting. However, these classifiers can over-fit feature weights based on the dataset, which can result in a reduction of accuracy. An upside is that these methods allow for the rapid retraining of a model with variant weights, and can increase the accuracy of traditional classifiers when used properly.

Support Vector Machine, commonly referred to as SVM, works by finding clusters and dividing them into groups using planes. SVM also has methods to reclassify outliers that may cross between groups, so results are consistent. SVM is commonly used in meta methods and is often used in conjunction with other classification methods.

Neural Networks, often abbreviated as NN, are classifiers that work by emulating the neural layout of a brain and creating links between features that then cascade across the linked nodes. By doing this it emulates a more complex decision-making process and when given enough data it can reach a high level of accuracy. There are a multitude of methods in this family of classifiers, with some methods being bi-directional and others only progressing in a single direction. These methods are commonly used in research and are seen as one of the biggest futures when it comes to artificial intelligence. However, these methods are often more complex requiring more compute to achieve a similar result to their less intensive counterparts on simple tasks.

Logistic Regression, or LR, is a linear statistical classifier used to predict the result of an event based upon the data provided. LR can be customized to classify on a binary scale or have multiple classification results. Due to the nature of linear regression, it is usually limited to simple decision making.

Bayesian is a family of classifiers based upon the Bayes' theorem, which assumes that all features are independent. Alongside this Bayesian classifiers often work best when all features are assumed to have a similar effect. This makes it a simple decision-making classifier; however, it lacks the complexity that other classifiers have and makes it less suited for multiple feature datasets.

To analyze the reported results, each paper was sorted by classifier, or type of classifier, used and can be seen in a visual format in Fig 1. The first most popular was Random Forest (RF) which had 39 appearances across the surveyed literature. The second most popular was the family of Meta algorithms with 29 appearances. These two classifiers were the most represented by a large margin, likely due to RF being an easy to use and high performing classifier, and Meta algorithms being similarly popular and representing a wider family of classifiers. The third most commonly used classifier was Decision Tree, which had twenty two appearances in the surveyed experiments. Then the Neural Network family, K-Nearest Neighbor, and the Bayesian family of classifiers were also popular with each having twenty test results across the surveyed works. Beyond this are Support Vector Machine with 18, Logistic Regression which had thirteen reported results, and finally 15 miscellaneous methods including novel approaches. To quantify the reported results the values were calculated together in table 1, which breaks down the descriptive statistics of the reported performances separated by classifier/classifier family.

B. *Results and Data Found*

As shown in table 1, the most tested method and one of the best performer by accuracy overall was Random Forest. Random Forest had an average accuracy of 94.52% and had a low standard deviation of ±4.42 while being the most represented classifier in the research surveyed. This means that Random Forest performed consistently using multiple different feature sets across multiple works. Following this, the next most popular classifier was a group of similar ensemble methods using meta algorithms. This group had the second most appearances with 29, and had an average accuracy of 93.63 percent but a wider standard deviation of ±7.30. This is likely due to the broader nature of this category and the amount of variance caused by the different methods being used for classification. However, the median and mode being 95.11% and 96.80% respectively shows that

TABLE I. Statistical Breakdown of Reported Accuracy from Experiments per Classifier

Classifier	No. Appearances	Mean	Median	Mode	Standard Deviation
Random Forest	39	94.52%	95.11%	96.90%	±4.42%
Meta Ensemble	29	93.27%	96.71%	96.80%	±7.14%
Decision Tree	22	93.68%	96.15%	98.00%	±6.72%
Neural Network	20	90.17%	90.90%	86.00%	±7.14%
Bayesian	20	88.63%	89.86%	85.11%	±7.78%
K-Nearest Neighbor	20	91.81%	94.39%	94.40%	±7.48%
Support Vector Machine	18	90.00%	90.00%	85.83%	±7.85%
Logistic Regression	12	88.78%	92.92%	96.00%	±10.44%

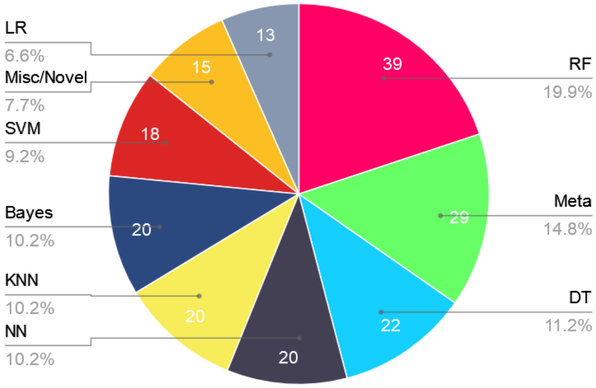

Figure 1. Number of Classifier Appearances Across All Works

with the proper feature sets this method is a great candidate for detecting ransomware.

The next most tested classifier, was Decision Tree which has a mean accuracy of 93.68% with a standard deviation of 6.72%. This means that DT averaged slightly better than the Meta classification methods with less variance, however it also had seven less appearances which could skew the data in its favor. DT still shows great potential for future experiments with a similar level of performance to meta methods. The next classification family represented in the surveyed experiments were Neural Networks, which had a mean accuracy of 90.17% and a standard deviation of 7.14%. This means that it performed on average worse than the three prior classifiers and was more prone to variance from feature selection and other variables. This is not unsurprising as NN classifiers often work better with large amounts of data and are more complex than traditional ensemble methods, which can increase the likely hood of subpar results depending on how the design of the model.

Bayesian, KNN, and SVM however had similar amounts of appearances (20/20/18 respectively), mean accuracy within a five percent deviation (88.63%, 91.81%, and 90% respectively), and a standard deviation between seven and eight percent(7.78%, 7.48%, and 7.85% respectively). This group had a wide level of variance in their accuracy between the reported experiments, and often performed worse than other more common classification methods. That is not to say that these methods cannot be used to predict ransomware attacks, only that they are more likely to be effected by the features selected. Thus, models using these classifiers must design experiments with this information in mind.

LR was the classifier that appeared the least outside of novel/miscellaneous classification methods, with only thirteen appearances in the experiment group. It also performed the worst with an average accuracy of 88.78% and a large standard deviation of 10.44%. This means that most LR results are expected to be within the range of 78.78-98.78%, which means that features and other variables have a major influence on its performance. However, this could instead be attributed to the smaller amount of occurrences in the dataset causing outliers to have more effect on the statistical results. This classifier, if used in experiments, should use carefully selected features and methods to ensure the model can reliably classify ransomware signs.

Following this, the next aspect of importance is the common features being used. Unlike accuracy which was reliably reported in most of the works surveyed, the features used were sometimes obfuscated behind generalized breakdowns of what was being targeted. Some papers would list features used in groups due to the massive amount of features used in their decision model. For example, some papers would report that they were focused on API calls and had 23 features dedicated to its detection, or the features chosen had a focus on the entropy of encrypted files and were classified into 18 features. This means that statistical reporting of these common features is less clear compared to reported classification accuracy.

The most common features used depended on how the study was attempting to detect the ransomware. For systems targeting newly downloaded files, there was a focus on detecting files containing ransomware as they were downloaded. Thus, the features were based upon the contents of the file such as encryption statements in the code, references to external API, the file metadata, etc. For reactive systems that worked on detecting ransomware activity in live system logs, the common features were focused on rapid API calls, rapid encryption calls, high levels of entropy in encrypted files, and rapid delivery of data to an external source. These features were often the most weighted in the models used, however many smaller features were noted that assisted the decision process such as the type of encryption used, the source of the file, if the file contained account logon code, etc. Other features that were commonly used among both were the levels of permissions required/asked of by the files, the operating codes used by the systems in question, VPN activity, and keywords contained in the files.

The ransomware commonly used in the surveyed datasets were often the most common families used at the time of the study. As these studies were conducted after or in 2019, the commonly used ransomware families were often WannaCry, TeslaCrypt, Locky, and REvil. The samples were often gathered from online malicious file repositories and tested in virtualized systems, often using tools such as Cuckoo Sandbox to obtain the data. For those using publicly available datasets the most common were those provided by government organizations, and research universities. These datasets often contained system logs and other runtime data gathered from a multitude of devices.

Altogether, the data gathered shows promising results. However, there are issues due to the nature of ransomware and how malicious software often evolves. In the research surveyed multiple studies[8],[10][6] note that linear ensemble classifiers had a stronger initial result compared to more complex classifiers such as neural networks. Studies also reported that using meta techniques such as genetic programming [18], gradient tree boosting [19], and particle swarm optimization[15], [20] increased the performance of the trained models in their tests. However, other studies such as Mcintosh et al.[11] also cited that the general high-performance models popular in many studies would lose accuracy with time as the features that had were used for detection during testing may become irrelevant over time as the ransomware methods used change. Alongside this, the features needed to detect each family of ransomware change depending on the method of attack which could further decrease the efficacy of general ransomware detection methods. Some ransomware exfiltrated the data then encrypted it, which makes models that look for high levels of network traffic and rapid encryption more effective. While other variants of ransomware may focus on quietly exfiltraing data to an off-network location which could lead to false negatives. Thus, the features used must either be chosen to efficiently target a single family or style of ransomware, or focus on including as many features as possible to cover all potential families at the cost of accuracy. This issue of targeting and optimization makes it so multiple models for different families of ransomware may be more efficient in detection.

The way these models are often designed around implementation on a singular client, or as part of a network security center servicing multiple clients. Each version offers another layer of defense in the scenarios they are designed for. For singular clients, one of the more popular designs focused on analyzing the activity of the device it is on to detect ransomware activity.

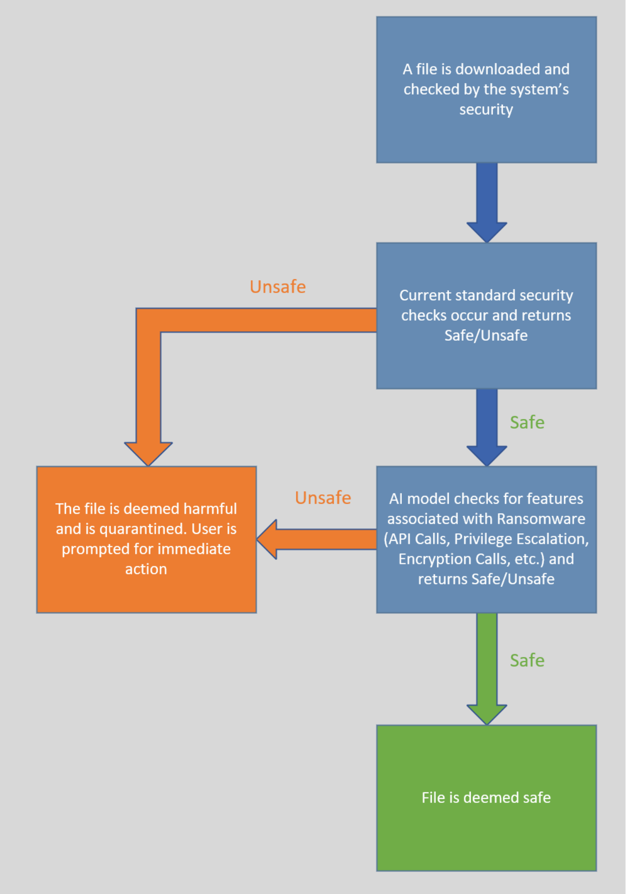
A file is downloaded and checked by the system's security
Current standard security checks occur and returns Safe/Unsafe
Unsafe
Safe
The file is deemed harmful and is quarantined. User is prompted for immediate action
Unsafe
AI model checks for features associated with Ransomware (API Calls, Privilege Escalation, Encryption Calls, etc.) and returns Safe/Unsafe
Safe
File is deemed safe

Figure 2. Example of client activity anti-ransomware

This method relies on the computer's processing capabilities to rapidly detect suspicious activity and take immediate action. However, being a reactive measure there is a drawback in that malicious activity detected could be reported too late and the device would already be compromised. The general design of this style is shown in Fig. 2.

Another singular device design focuses the AI decision model on the analysis of newly acquired files. Alongside the normal security checks done when a file is downloaded, a trained model will analyze the contents and return its own value of safe/unsafe in another layer of defense. This method focuses on being proactive, and could head off attacks before they start. However, like the prior example incorrect classification can lead to the device being compromised.

Finally, in network-based infrastructure the models are deployed as part of the networks security center, with

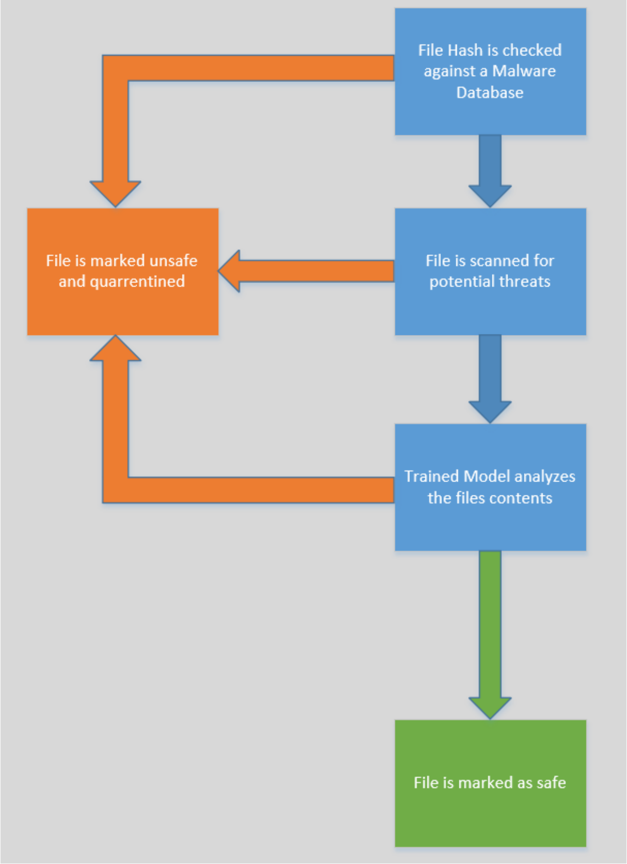
File Hash is checked against a Malware Database
File is marked unsafe and quarrentined
File is scanned for potential threats
Trained Model analyzes the files contents
File is marked as safe

Figure 3. Example of file-analysis anti-ransomware

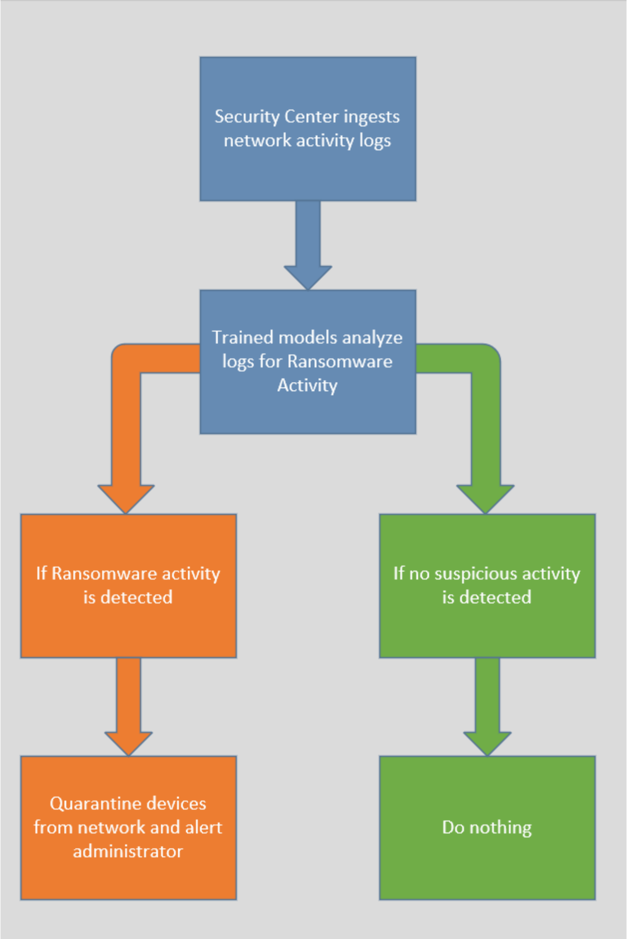
Security Center ingests network activity logs
Trained models analyze logs for Ransomware Activity
If Ransomware activity is detected
If no suspicious activity is detected
Quarantine devices from network and alert administrator
Do nothing

Figure 4. Example of network-based anti-ransomware techniques

the model being trained to detect potential hostile activity on the network. This design allows the model to have oversight over multiple clients at once, and can potentially detect ransomware spreading through a network and react accordingly. This can help reduce the amount of data affected by crypto-ransomware and also stop data leaks from exfiltration focused ransomware strains. It also reduces the amount of compute required per device, instead offloading the processing of these tasks to devices specifically designed for this purpose. The trade off being a singular device, or group of devices, with enough computing power to analyze all traffic on a network which will increase in cost per unit as the network grows. The generalized implementation can be seen in Fig 3.

6. CURRENT STATE OF THE FIELD

The current state of the field is dominated by the application of these techniques in layers of defense. Often the frameworks analyzed in this review suggested as such, and current commercial solutions do so. By having multiple layers of defense and following zero-trust methodology ransomware attacks can be restricted and prevented from causing widespread damages. Using trained AI decision models can detect questionable actions done on the network and speed up the response speed of a network/system. Major security companies and firewall providers such as Microsoft [21] and CrowdStrike [22] officially utilize machine learning techniques in their ransomware detection security suites bundled in their defensive services. Alongside this AI techniques are also used to detect the potentially hostile files and links as another layer of defense.

These methods, however, are designed around the current standards of ransomware. Requiring the models used in these services to be updated as the malware evolves, similar to the current operating standards of anti-malware services. This means that there is a constant battle to update the trained models, and that preventative actions done in network design and user guidance are still critical to avoid ransomware attacks that are yet unknown. For future research the focus should undoubtedly be on upcoming ransomware, new novel versions of ransomware and how to stop them, and researching ways to create models that can retain their accuracy over time or evolve alongside new trends with less oversight and cost.

For this research it would be prudent to design solutions around the idea of multiple layers of defense, which has seen some success in the current commercial marketplace. By designing around this it allows for the techniques to be easily incorporated into existing frameworks and allows for design decisions that can simplify the trained models and make them more efficient. Other avenues of future research that would be useful is studies of local populations like as Ortloff et al.[12] did in Germany to see how each population reacts and understands ransomware to develop frameworks and assist in the education of the community against ransomware.

7. FUTUREWORK

Ransomware is one of the biggest threats in the digital landscape. Ransomware is commonly mentioned in many countries news cycles as major public infrastructure, organizations, and others report successful attacks and the amount of time it will take for issues caused by these attacks to be fixed. Of the new technologies that have exploded in popularity over the last couple years, AI is uniquely positioned to offer assistance when it comes to detecting and reacting to these threats. However, AI is not in a position to completely remove the threat on its own but instead offers another layer of defense to assist in this critical battle. Arguably the most important research into what AI can do against ransomware is to focus on optimizing its detection in ways that take advantage of the already commonly implemented layers of defense.

The classifiers used should be properly chosen to suit the task based on the features used for detection. Traditionally successful classification methods such as random forest, decision tree, and meta methods have consistent results and lend themselves to testing. Alongside this, other methods such as KNN and neural networks should be tested alongside the best performing methods to gather a large sample of performance data.

Alongside this, there should be specific focus in codifying what results are showcased in reported research. In the surveyed sample of work, many papers reported only accuracy and neglected to showcase other key statistics such as recall, f1-score and more. These values are important to give a full understanding of the performance of the model in the experiment performed. Thus, future research should ensure that these key statistics are reported. Alongside this many research efforts still do not provide the datasets used which can make it difficult to re-test results and build off of these experiments. For the best results research into this field should ensure that all data used, and the methodology is clearly shown and easy to re-test and prove, as it will help other researchers test new methods and assist the development of the field.

Furthermore, currently machine learning and artificial intelligence is a relatively new field that has had a massive surge in popularity due to many high-profile investments. To assist in developing this technology, future research could provide the datasets used and full explanations of the processes used to achieve the reported results. By ensuring that the data used is publicly available other researchers could have an easier time creating branches of older ideas and styles to help develop the field and our understanding of it.

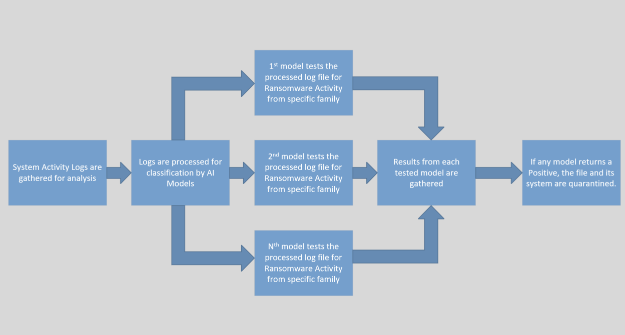
System Activity Logs are gathered for analysis
Logs are processed for classification by AI Models
1st model tests the processed log file for Ransomware Activity from specific family
2nd model tests the processed log file for Ransomware Activity from specific family
Nth model tests the processed log file for Ransomware Activity from specific family
Results from each tested model are gathered
If any model returns a Positive, the file and its system are quarantined.

Figure 5. Example of network-based anti-ransomware techniques

Finally, generalized ransomware models theoretically work but are more prone to error due to the amount of variance in ransomware. To correct for this, models should be specialized to target specific variants of ransomware based on the methods used to execute an attack. Future research could also be done in designing a multi-model solution that analyzes the information through multiple trained models to determine the safety of a file, an example of the method outlined in Fig. 4. Alongside this, other novel methods of detection such as visual recognition should be experimented with to give further understanding into the nature of detecting ransomware using trained AI models.

8. CONCLUSION

A severe ransomware attack is one of the costliest attacks to recover from. By encrypting and locking away an organization's data and systems hackers can completely halt any processes carried out by that organization. This threat affects a wide berth of users, governments, and businesses in similar but different ways. The use of artificial intelligence has been seen as a potential solution to ransomware leveraging the statistical calculations of the computer against such malicious software. There lies potential in the adoption of these techniques as seen in major companies such as Microsoft and CrowdStrike who have added similar techniques into their own security offerings.

Further research should be done to increase the reliability of the models as ransomware changes, how people understand and react to ransomware, and into the study of future ransomware threats. Upcoming research should also do their best to provide their datasets to build time specific captures of common ransomware to assist in the archival of this information and the development of future experiments in the field. Classification methods such as random forest and decision tree should be used for testing due to their high reported performances, and other methods alongside it to ensure multiple styles are analyzed. The results reported should ensure all key statistics are included such as F1-

Score, recall, accuracy, and precision. Alongside this, these future models should be designed in a way that works congruous with current defensive measures to encourage easy adoption of these techniques in a commercial setting.

References

[1] Cloudwards. (2024) Ransomware statistics, trends and facts for 2024 and beyond. [Online]. Available: https://cloudwards.net/ransomware-statistics

[2] K. Bing and S. Kelly. Cyber attack shuts down u.s. fuel pipline 'jugular', biden briefed. [Online]. Available: https://www.reuters.com/technology/colonial-pipeline-halts-all-pipeline-operations-after-cybersecurity-attack-2021-05-08/

[3] D. Goodin. (2024) London hospitals declare emergency following ransomware attack. [Online]. Available: https://arstechnica.com/security/2024/06/london-hospitals-declare-emergency-following-ransomware-attack/

[4] N. Alzahrani and D. Alghazzawi, "A review on android ransomware detection using deep learning techniques," in *Proceedings of the 11th International Conference on Management of Digital EcoSystems*, ser. MEDES '19. New York, NY, USA: Association for Computing Machinery, 2020, p. 330–335.

[5] A. Bertia, S. B. Xavier, G. J. W. Kathrine, and G. M. Palmer, "A study about detecting ransomware by using different algorithms," in *2022 International Conference on Applied Artificial Intelligence and Computing (ICAAIC)*, 2022, pp. 1293–1300.

[6] M. Sneha, A. Arya, and P. Agarwal, "Ransomware detection techniques in the dawn of artificial intelligence: A survey," in *Proceedings of the 2020 9th International Conference on Networks, Communication and Computing*, ser. ICNCC '20. New York, NY, USA: Association for Computing Machinery, 2021, p. 26–33.

[7] N. Thamer and R. Alubady, "A survey of ransomware attacks for healthcare systems: Risks, challenges, solutions and opportunity of research," in *2021 1st Babylon International Conference on Information Technology and Science (BICITS)*, 2021, pp. 210–216.

[8] R. Moussaileb, N. Cuppens, J.-L. Lanet, and H. L. Bouder, "A survey on windows-based ransomware taxonomy and detection mechanisms," *ACM Comput. Surv.*, vol. 54, no. 6, jul 2021. [Online]. Available: https://doi.org/10.1145/3453153

[9] H. Oz, A. Aris, A. Levi, and A. S. Uluagac, "A survey on ransomware: Evolution, taxonomy, and defense solutions," *ACM Computing Surveys*, vol. 54, no. 11s, p. 1–37, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1145/3514229

[10] S. Razaulla, C. Fachkha, C. Markarian, A. Gawanmeh, W. Mansoor, B. C. M. Fung, and C. Assi, "The age of ransomware: A survey on the evolution, taxonomy, and research directions," *IEEE Access*, vol. 11, pp. 40 698–40 723, 2023.

[11] T. McIntosh, A. S. M. Kayes, Y.-P. P. Chen, A. Ng, and P. Watters, "Ransomware mitigation in the modern era: A comprehensive review, research challenges, and future directions," *ACM Comput. Surv.*, vol. 54, no. 9, oct 2021. [Online]. Available: https://doi.org/10.1145/3479393

[12] A.-M. Ortloff, M. Vossen, and C. Tiefenau, "Replicating a study of ransomware in germany," in *Proceedings of the 2021 European Symposium on Usable Security*, ser. EuroUSEC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 151–164.

[13] S. R. Davies, R. Macfarlane, and W. J. Buchanan, "Review of current ransomware detection techniques," in *2021 International Conference on Engineering and Emerging Technologies (ICEET)*, 2021, pp. 1–6.

[14] C. Bansal, P. Deligiannis, C. Maddila, and N. Rao, "Studying ransomware attacks using web search logs," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '20. ACM, Jul. 2020.

[15] I. Almomani, R. Qaddoura, M. Habib, S. Alsoghyer, A. A. Khayer, I. Aljarah, and H. Faris, "Android ransomware detection based on a hybrid evolutionary approach in the context of highly imbalanced data," *IEEE Access*, vol. 9, pp. 57 674–57 691, 2021.

[16] A. H. Lashkari, A. F. A. Kadir, L. Taheri, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark android malware datasets and classification," in *2018 International Carnahan Conference on Security Technology (ICCST)*, 2018, pp. 1–7.

[17] S. Mahdavifar, A. F. Abdul Kadir, R. Fatemi, D. Alhadidi, and A. A. Ghorbani, "Dynamic android malware category classification using semi-supervised deep learning," in *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*, 2020, pp. 515–522.

[18] H. Al-Sahaf and I. Welch, "A genetic programming approach to feature selection and construction for ransomware, phishing and spam detection," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, ser. GECCO '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 332–333.

[19] M. J. M. M., U. S., M. B. P., and S. G. Sandhya, "Detection of ransomware in static analysis by using gradient tree boosting algorithm," in *2020 International Conference on System, Computation, Automation and Networking (ICSCAN)*, 2020, pp. 1–5.

[20] M. S. Hossain, N. Hasan, M. A. Samad, H. M. Shakhawat, J. Karmoker, F. Ahmed, K. F. M. N. Fuad, and K. Choi, "Android ransomware detection from traffic analysis using metaheuristic feature selection," *IEEE Access*, vol. 10, pp. 128 754–128 763, 2022.

[21] M. T. Intelligence. (2023) Ai-driven adaptive protection against human-operated ransomware. [Online]. Available: https://www.microsoft.com/en-us/security/blog/2021/11/15/ai-driven-adaptive-protection-against-human-operated-ransomware/

[22] Crowdstrike. (2023) Ransomware protection: Everything you need to stop ransomware in its tracks. [Online]. Available: https://www.crowdstrike.com/solutions/ransomware-protection

Alexander Veach is a graduate from Eastern Michigan University in Ypsilanti, Michigan with a bachelor's in information technology and a master's in cybersecurity. He is currently pursuing a Doctorate in Technology, focusing on technology studies. His current academic interests are in machine learning, technology adoption, and cloud computing.

Munther Abualkibash is an associate professor in the School of Information Security and Applied Computing at Eastern Michigan University. His interests and expertise include computer and network security, cloud computing, and machine learning. He received his master's degree from the University of Bridgeport, in Bridgeport, Connecticut. There, he also earned his Ph.D. in computer science and engineering.