Using Machine Learning to Detect Ransomware Attacks on Electronic Health Records (EHRs)

Miguel Durán Zamora
Master in Computer Science
Advisor: Dr. Jeffrey Duffany
Polytechnic University of Puerto Rico
Graduate Project EXPO, October 2025

Abstract *— Ransomware attacks have always been a burden for many industries, and nowadays, with the advancement of technology and computers, they have taken over everything. One crucial industry is the health industry, which has been a victim of these attacks for a while. Could we explore ways to detect and mitigate these attacks in critical areas, such as health? That's why in this article, we will discuss how we can use machine learning to detect ransomware attacks on electronic health records.*

Key Terms *— Detect, Electronic Health Records, Machine Learning, Ransomware.*

INTRODUCTION

Healthcare attacks have increased throughout the years, not only because of the advancement of technology but also because of ways to protect and defend against them. According to the NIH, the PIH data breaches went up from 4% to 81%, and surprisingly enough, from 2010 to 2024, ransomware attacks went down from 31% to 11% [1]. You might think the decrease means it is safe, but as stated before, when attacks increase, protection increases, but there is still more we can do to detect and mitigate these attacks even more. With the increase in the use of machine learning (ML) in all industries, we can make sure we catch these early ransomware attacks against healthcare infrastructures and create systems that automatically fight against them without human intervention.

We know there are many systems in place to defend against these attacks, but the future of computer automation will be learning from mistakes and using methods where computers can adapt to different types of attacks, and even if new attacks are created, then computers can understand whether it's an attack or not, according to learned ransomware attack patterns through the years.

BACKGROUND

The reason I picked this project was because, as a cybersecurity student and having spoken with family who are in the healthcare industry, I learned the serious need for more security that protects the integrity and confidentiality of patients. Also, diving more into the world of AI and ML, I felt there is more we can do with these to make the security of EHRs even stronger than before. I strongly believe that we must use new technology and tools that meet industry standards. Also, the automation of processes is key because when we have things like generative AI where you can write a prompt and it generates an answer, imagine that but for ransomware attacks where you type the ransomware specification and automatically create and execute the ransomware without human intervention. That will be a game changer in the field of cybersecurity and that's why it's crucial for us to prepare for those days.

PROBLEM

A problem I found that inspired me to do this project is the lack of automation when it comes to security threats like ransomware, especially in the health industry. I believe that we should implement, with the use of ML, more autonomous detection and mitigation systems, and with the use of ML now, it's possible. ML will help us detect ransomware patterns in these EHRs, and then we can use these ML models in artificial intelligence to create systems that counterattack these attacks autonomously. Not only will this protect the integrity of the patient's

data, but it will also save the industry money and resources.

HARDWARE AND SOFTWARE

For this project, I had to use a combination of ransomware and ML technologies. Also had to use Windows applications to log activities and then turn them into ML-ready datasets.

- **VMWare:** VMWare is a virtual machine, and it was used for this article to simulate an EHR system with around 600 EHRs of different file formats such as XML, CSV, and JSON [2]. These file formats are common on EHR servers. We used Windows 10 as the host computer, and for the VM, we used a Windows 7 VM to safely run the ransomware.
- **Host Computer:** Windows 10 HP Envy, Intel Core i7 8th Gen.
- **Sysmon:** Sysmon is a Windows application. This was used to track ransomware activity and log it for later use for feature extraction, and then ready to train the model.
- **Synthea:** Synthetic EHR data that was downloaded in the VM. We used synthetic data for patient confidentiality reasons [3].
- **Python:** Python was used for feature extraction from Sysmon XML logs and was used to run the Random Forest Classifier.

METHODOLOGY

For this article, we started by creating a VM in a Windows 10 laptop where we simulated ransomware attacks. The sample used for this project was Yashma ransomware, which is a variant of the Chaos ransomware. This ransomware was the sample selected due to its fast encryption and ease of use. The specifications of the test machine are:

Table 1
Test Machine Specifications

CPU	Intel Core i7-8550U 1.80GHz
RAM	16GB
Network	Intel Dual band Wireless AC-7265
SDD	512GB
Operating System	Windows 10

Once we have the VM, then we installed Sysmon so we can track ransomware activities. Then I downloaded the Yashma ransomware builder, which is a ransomware builder to create the Yashma ransomware variant. This is an easy way to create ransomware, but always make sure you are testing it in a private and secure environment, such as a VM. The VM is useful in this case because you can create snapshots, so that in case something goes wrong, you can roll back. After I ran the ransomware in the secure VM then Sysmon will start capturing the activities or events. In this case, the events used were:

Table 2
Sysmon Events Captured

Event	Description
1	Process Created
2	Process changes the file creation time
11	File Creation
12	Modifies registry keys and values
23	Detect file deletion and archive it in C:\Sysmon
26	Same as the previous event, but doesn't archive

Once we captured the event, I created a Python script to extract different features and heuristic rules to not just determine if it's ransomware, but ransomware related to the EHRs. For this article, we are working with a supervised dataset that allows us to categorize each entry into ransomware or benign depending on the features and heuristic characteristics.

Structure of EHRs

What are EHRs or Electronic Health Records? They are an electronic repository of diagnoses, medical history, medications and immunizations, to X-rays, laboratory results, and clinical notes [4]. These are crucial for medical centers and hospitals because they help medical professionals access these

records easily and allow automation and efficiency. The fact that these contain tons of sensitive data protected by GDPR and HIPAA requires the utmost security. That's why we propose in this article ways you can automate these EHRs security using ML. But first, we need to understand the structure of EHRs. Some of the components we can find in EHRs are:

- **Patient Information:** Information about patients, such as medication, history, and diagnosis.
- **Order Entry Systems:** These contain tests, medications, and treatments.
- **Decision Support Systems:** Contain medical recommendations.
- **Secure Protocol:** contains protocols to ensure patient information safety.
- **Communication Tools:** allow medical professionals to communicate with each other.

Another factor in the structure of EHRs is file types and how they are stored. Many EHR files can be found in XML, JSON, and CSV as flat files [2]. But if you have them in a server or database, they can be stored in database formats such as .db or .sql.

Feature Extraction

These are the features and characteristics we extracted to train the models on. In this project we are looking for patterns of ransomware but in EHRs. This means that some features will be extracted directly from the Sysmon logs, but others will have to be calculated using Python in a script. In other words, the ones in the table below that show the "EventId" were extracted from the log directly but suspicious path, directory depth, process length, extension similarity, file entropy and EHR related where calculated using functions. Next, we can see the table of features in more detail.

Table 3
Some of the Features Gathered

Feature	Description
File Deleted	Detects deleted files and backup files (EvendId: 26).
File Created	This is the most common because ransomware when they encrypt a file is categorized as file creation (EventID: 11).
File Create Time Change	This happens when a process changes the creation time of a file (EventId: 2).
Process Create	This happens when a new process is created (EventId:1).
Suspicious Path	This checks if the Image or path from where the process is coming contains known suspicious keywords.
System Executable	Checks if the process is being executed from a known system path or Image as c:\windows\system32.
Path Length	Get the length of the path.
Directory Depth	Check the directory and see how deep in the directory the process is.
Process Length	The length of the process.
Extension Similarity	Checks if extensions of the target files are similar. For example, some ransomwares encrypt the file into .encrypted.
Filename Entropy	This calculates the Shannon entropy of the base name of the path. In other words, it checks the randomness of that path.
EHR Related	It checks if it is EHR related depending on the keywords within the path of the target file.

For this project, I used a dataset containing 7,365 entries, where 3,067 are benign and 4,298 are ransomware. This dataset was collected by running the ransomware repeatedly to collect the ransomware activity and doing normal operations to collect the benign ones. Ransomware activities are all those that when executing the ransomware then it will act upon. For example, encrypting files, deleting files, executing processes, editing registry and suspicious network connection. For benign ones we access the internet, used applications and created different files like ,py and .txt.

ML and Result

After collecting the dataset and extracting the necessary features, I ran the dataset through a Random Forest Classifier, a K-Nearest Neighbor Classifier, and a Support Vector Classifier. Using the labeled feature of ransomware or benign, I

concluded the precision, recall, F1-score, and accuracy of the model. The Shannon entropy formula used for the feature extraction, the recall, precision, accuracy, and F1-score are:

Entropy:

$$-\sum p(x)\, log_2(p(x)) \quad (1)$$

Precision:

$$\frac{TP}{TP+FP} \quad (2)$$

Recall:

$$\frac{TP}{TP+FN} \quad (3)$$

Accuracy:

$$\frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

F1:

$$\frac{2*Precision*Recall}{Precision+Recall} \quad (5)$$

Where:

- **TP:** True Positive
- **TN:** True Negative
- **FP:** False Positive
- **FN:** False Negative

Then I ran the random forest classifier (RF), k-nearest neighbor (KNN), and support vector classifier (SVC), but before you need to run a train-test split that splits the dataset between training data and testing data. Once the classifier ran, I ran a prediction to gather sklearn metrics to see if the model was accurate enough, and these were the results:

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.96	0.98	968
1	0.97	1.00	0.98	1242
accuracy			0.98	2210
macro avg	0.98	0.98	0.98	2210
weighted avg	0.98	0.98	0.98	2210

Figure 1
Random Forest (RF) Prediction Metrics

Classification Report:

	precision	recall	f1-score	support
0	0.99	0.98	0.99	969
1	0.98	1.00	0.99	1241
accuracy			0.99	2210
macro avg	0.99	0.99	0.99	2210
weighted avg	0.99	0.99	0.99	2210

Figure 2
K- Nearest Neighbor (KNN) Prediction Metrics

Classification Report

	precision	recall	f1-score	support
0	0.9858	0.9721	0.9789	645
1	0.9785	0.9891	0.9838	828
accuracy			0.9817	1473
macro avg	0.9822	0.9806	0.9814	1473
weighted avg	0.9817	0.9817	0.9817	1473

Figure 3
Support Vector Classifier (SVC) Prediction Metrics

Then ran to see which were the most important features. This is important because you want to understand which feature is key to detecting ransomware and which aren't. Having non-important features in the case of random forest can cause us to create useless nodes in the trees it generates. Also, to show the Receiving Operating Characteristic (ROC). This graph is important to analyze the rate between the true positive and the false positive of a model. In this case the rate of the TP and the FP are so apart that the line seems like a constant line. These metrics are a good way to see if the models are good and if they are accurate enough to detect what you're looking for.

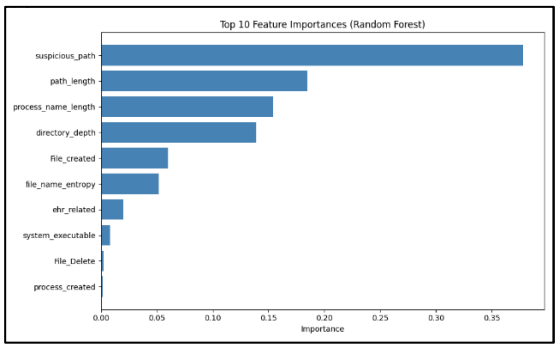

Figure 4
Feature Importance Graph

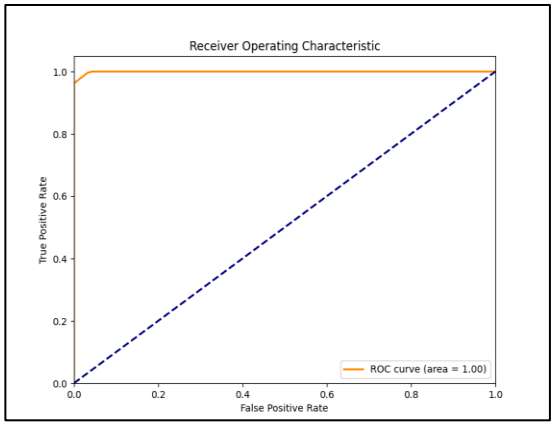

Figure 4
Receiving Operating Characteristic (ROC)

NORMAL RANSOMWARE VS. EHR

When we think about ransomware detection, we think we can detect all ransomware the same, but we need to focus on different use cases for better results. In ransomware detection systems, you will target all files, including system files, to detect, but in EHR, you will focus on the EHR file specifically. Also, ransomware attacks in general may not be paid for since they are not crucial, but ransomware in EHR puts extra stress because it can be deadly in many cases. For example, if ransomware attacks happen in critical infrastructure, such as health, then hospitals are more susceptible to paying for the ransomware. [5] In general, there has been a 300% increase in ransomware attacks since 2015. Another difference is the file types and paths where you can find these.

LEGAL AND ETHICAL IMPLICATIONS

Healthcare has always been a very controversial topic when it comes to privacy and confidentiality. That's why when it comes to the use of ML in EHRs must be done carefully and follow all the legal and ethical policies. For machine learning, you train the model on data, and that's why there's a method to do it with synthetic data and not real data to make sure you protect the confidentiality of patients. For this, the data must be of high quality and reflect real-world data. [6] When working with real data of patients, you need to make sure you follow the guidelines of GDPR and HIPAA to ensure you are following the law.

LIMITATIONS

For the highest quality dataset, you require thousands and thousands of entries, and due to time constraints, I could only get around 7k entries. Other limitations were the host computer in use, which wasn't the best, but it got the job done. Another thing is that there are not many studies about ML ransomware detection in EHRs, but by using datasets like the one from RanSAP [7] and CSU-Ransomware-Data [8], they were a good start.

CONCLUSION

After running the dataset in each of the three algorithms (random forest, k-nearest neighbor, and support vector classifier), we can see that the best one to perform overall was k-nearest neighbor with an 0.99 accuracy. Also, we can see that the most favorable feature was suspicious path, which happens when files are created in sensitive or suspicious directories such as \\appdata\\ or \\temp\\. When analyzing ransomware with ML, we must target these heuristic characteristics as part of the conclusion to check if it's ransomware because that's how the model does the predictions and learns from those patterns.

FUTURE WORK

There is a lot more we can do with ML, Ransomware, and EHRs. For future work, I would like to expand on this topic by making the model better by adding more data and creating a more robust feature extractor. Also, I would like to create software or applications where you can connect to your EHR systems and automatically detect ransomware attack patterns using the ML models created. Ransomware is growing, and so is ML. Using ML to automate things and detect early signs of ransomware in different systems will strengthen the security of crucial industries, such as the healthcare industry.

REFERENCES

[1] J. X. Jiang, J. S. Ross, and G. Bai. (2025, May 14). *Ransomware attacks and data breaches in US Health Care Systems* [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC12079295/#:~:text=Results,decreased%20(Figure%2C%20A). [Accessed: Oct. 3, 2025].

[2] NHI Pragmatic Trials Collaboratory. (n. d.). *Rethinking Clinical Trials* [Online]. Available: https://rethinkingclinicaltrials.org/chapters/conduct/acquiring-real-world-data/data-formats/. [Accessed: Oct. 3, 2025].

[3] J. Walonoski, M. Kramer, J. Nichols, A. Quina, C. Moesel, D. Hall, C. Duffett, K. Dube, T. Gallagher, and S. McLachlan, "Synthea: An approach, method, and software mechanism for generating synthetic patients and the synthetic electronic health care record," in Journal of the American Medical Informatics Association, vol. 25, no. 3, pp. 230–238, March 2018. Available: https://doi.org/10.1093/jamia/ocx079.

[4] ISO. (n. d.). *Electronic Health Records Explained* [Online]. Available: https://www.iso.org/healthcare/electronic-health-records. [Accessed: Oct. 3, 2025].

[5] J. Reed. (n. d.). *When ransomware kills: Attacks on Healthcare Facilities* [Online]. Available: https://www.ibm.com/think/insights/when-ransomware-kills-attacks-on-healthcare-facilities. [Accessed: October 1, 2025].

[6] T. Pham, "Ethical and legal considerations in Healthcare AI: Innovation and policy for safe and Fair use," in *Royal Society Open Science*, vol. 12, no. 5, May 2025. Doi:10.1098/rsos.241873.

[7] M. Hirano, R. Hodota, and R. Kobayashi, "RANSAP: An open dataset of ransomware storage access patterns for training machine learning models," in *Forensic Science International: Digital Investigation*, vol. 40, pp. 301314, Mar. 2022. Doi: 10.1016/j.fsidi.2021.301314.

[8] R. Islam (Creator), "CSU-Ransomware-Data," in *GitHub*, Nov. 2024.