BLACKBOX: A DIGITAL FORENSIC RECORDER

Akshay Achary
Student, MSc. Information and Cybersecurity
Department of Information Technology
Guru Nanak Khalsa College, Mumbai, India

Abstract

Cyber threats such as ransomware, insider attacks, and fileless malware have increased significantly in recent years. Because of this, traditional forensic methods that rely mainly on analyzing incidents after they happen are no longer always effective. Many modern attacks execute quickly and often leave very little trace behind, which makes it difficult to fully understand what actually occurred.

This paper presents **BlackBox**, a real-time digital forensic recording system designed to continuously monitor system activity and preserve important evidence whenever suspicious behavior is detected. The idea is somewhat inspired by flight recorders used in aviation. Instead of recording everything permanently, the system keeps recent activity in circular buffers, including file operations, process execution, network connections, registry changes, PowerShell usage, USB interactions, and even screen captures.

A centralized trigger engine evaluates activity across different modules and calculates a risk score. When certain thresholds are reached, or when critical behavior is detected, the system captures and stores a structured forensic case. This includes logs, metadata, context of the trigger, and integrity hashes. Compared to traditional forensic tools, which are mostly reactive, **BlackBox** takes a more proactive approach while still keeping system performance stable.

Keywords

Digital Forensics, Real-Time Monitoring, Event Correlation, Ransomware Detection, Evidence Preservation, Circular Buffer, Endpoint Monitoring, Cybersecurity

1. INTRODUCTION

Cyberattacks are a problem these days. They are not happening more often but they are also much harder to track and understand than they used to be. The old ways of looking at computer systems after something has gone wrong do not work well anymore. Usually investigators look at logs and other information to try to figure out what happened.. The problem is that this does not always work with the new kinds of threats we are seeing.

For example some kinds of malware can do their damage in a few seconds. They can even delete their tracks or make them hard to find before anyone can start looking into what happened. Because of this we need systems that can watch what is going on all the time and catch any evidence of something happening right away. If we can do this we can respond faster. Do a better job of investigating.

It is not easy to combine watching what is going on all the time with collecting evidence in a way that is useful. Some tools make much data that is not really useful which can make it harder to look at everything. Other tools miss things when they are happening. This is a problem that still needs to be solved.

That is why we made BlackBox. It is a system that combines watching what is going on in time looking at events together and automatically collecting evidence into one solution.

1.1 Research Contribution

Our work is about making a system that can record what is happening on a computer in real time based on triggers. Of looking at different things that are happening on the computer separately we look at them together. This makes it easier to understand what is really going on when something suspicious happens.
Our system looks at a lot of things like what is happening with files, which programs are running what is happening on the network and more. We do all of this in a way that does not slow down the computer. For example we only keep the recent and relevant data instead of keeping everything all the time.

Another important part of our system is how we evaluate events. We use a centralized scoring system to connect things that are happening in parts of the system. This helps us identify patterns that we might otherwise miss. When we detect something the system can automatically start collecting the relevant data. We then. Store this data and make sure it has not been tampered with.

Overall our goal is not to collect much data as possible but to collect the right data at the right time. This makes our system more practical and easier to use in life.

- BlackBox is designed to address the issues of cyberattacks
- It combines real-time monitoring, event correlation and automated evidence capture
- The system looks at monitoring areas, including file system activity and network behavior
- It uses a centralized scoring approach to connect activities from modules
- The system can automatically trigger a case capture process when something unusual is detected
- The focus is on capturing the data at the right time rather than collecting as much data, as possible.

2. LITERATURE REVIEW

Digital forensics has changed a lot over time. This is because cyberattacks have become faster and harder to track. In the past people would look at systems after something had gone wrong. They would gather information like logs and disk images. Try to figure out what had happened.

This way of doing things still works sometimes. It is not always good enough for new types of attacks. Some attacks happen quickly and even try to hide or delete evidence. So just looking at things after they have gone wrong can leave some gaps in our understanding of forensics.

Now people are working on making digital forensics more automatic and real-time. Of waiting for something to go wrong systems are watched all the time and important information is captured as things happen. This is very important for dealing with digital forensics threats.

2.1 Evolution of Digital Forensic Techniques

A time ago digital forensics investigations were based on looking at static information. People would take a snapshot of a system like a disk image. Look at it separately. This helped keep evidence safe. It meant that what was happening at that moment could not be seen.

As systems got more complicated new ways of doing things were developed. Techniques like live forensics and looking at logs made it possible to get more dynamic information. However a lot of the work still had to be done by hand. This can be slow and sometimes important details can be missed, especially when dealing with attacks and digital forensics.

2.2 Automated Digital Forensics and Incident Response

Automation is very important in forensics now. This is because a lot of the work is repetitive. Things like collecting logs and organizing data can take a time if done by hand. There are tools that can help automate some of this work. However most of them only start working after a problem has been found. So they are still just reacting to things. By the time they start collecting data some important evidence may be gone. Digital forensics needs to be able to deal with this.

The BlackBox system does things differently. It combines automation with monitoring. This means that important activity is being recorded all the time even before a problem is formally detected. This can make a difference when analyzing digital forensics.

2.3 Real-Time System Monitoring

Real-time monitoring is about watching what is happening on a system as it happens. This includes things like file changes and network communication. One good thing about this approach is that it reduces the time between an attack and when it is detected. For example if a suspicious file appears it can be found away. Digital forensics can benefit from this.
At the time setting up this kind of monitoring is not always easy. It can create a lot of events. If they are not managed properly they can affect how the system works. Finding a balance between being able to see what is happening and keeping the system working is important for digital forensics.

2.4 Research Gap

When looking at the way things are done now some limitations are clear. Many systems still focus on looking at problems after they have happened not while they are happening. Also tools for monitoring and digital forensics are often developed separately which can make it hard to use them together.

Another issue is that event correlation is not always used well when collecting evidence for forensics. Some solutions are too complicated or use many system resources, which limits their use. Digital forensics needs to be able to deal with these issues.

The BlackBox system is designed to address these concerns. By combining real-time monitoring, event correlation and automated evidence capture it tries to offer a balanced and practical solution, for digital forensics.

3. PROBLEM STATEMENT AND OBJECTIVE

3.1 Problem Statement

When you look at how forensic investigationsre done on Windows systems you will see that a lot of the work is still done by hand. Forensic investigations on Windows systems involve a lot of work. Investigators have to go through logs and check running processes and network activity one by one. Then they try to put everything together.

This way of doing things can work,. It is not always the best way. Sometimes it takes a lot longer than you think it will especially when there is a lot of data to look at. Forensic investigations on Windows systems can be slow. In attacks even a small delay can mean that some important details are lost. It also depends on who's doing the analysis and that can lead to different results in different cases. Different people doing investigations on Windows systems can get different results.

Because of this just relying on steps is not enough anymore. Forensic investigations on Windows systems need a way. It would be better to have a system that can watch what the system is doing as it happens especially when something strange starts to happen. That way you are less likely to miss information. This system would help with investigations on Windows systems.

3.2 Objectives

The goal of this project is to make it better to handle data when the system is still running. One part of this is to make a monitoring system for Windows that can watch what is going on all the time without slowing down the system much. Another part is to bring different kinds of system data. Of looking at each thing separately the system looks at many sources at the same time, which can give you a clearer picture of what is going on with the Windows system.

There is also a focus on finding behavior in a more flexible way. Of just using fixed rules the system uses a risk-based approach to decide when something might be a problem with the Windows system.

Once it finds something with the Windows system the system should be able to collect the relevant information automatically and put it into a structured form. The main goal is to make it faster and more consistent to start investigating investigations, on Windows systems.

4. METHODOLOGY

4.1 System Architecture

The BlackBox system is designed to keep an eye on things all the time and capture evidence when something weird happens. It does not do things the way, where data is collected first and then looked at later. The BlackBox system watches what is happening. Responds right away if something seems unusual.

The BlackBox system has three parts. Each part does something

First there is a part that keeps track of what the BlackBox system's doing. This includes things like what filesre being used what programs are running, who is connected to the network and what changes are being made. The BlackBox system also keeps track of what's happening on the screen and what commands are being used. It does not keep all of this information forever though. The BlackBox system only keeps the recent activity so it does not use up too much space.

The second part of the BlackBox system is like a watchdog. It looks at everything that is happening. Decides if it is okay or not. It does not look at each thing separately. Rather at how they are all related. The watchdog gives each thing a score. Then it adds up all the scores to get a total risk score. If the score gets too high the BlackBox system does something about it.

When the BlackBox system decides to do something it is the job of the case manager to take care of it. The case manager saves all of the information that the watchdog has been looking at. Then it puts it all together into a special file. This file has all of the details like what happened, when it happened and who was involved.

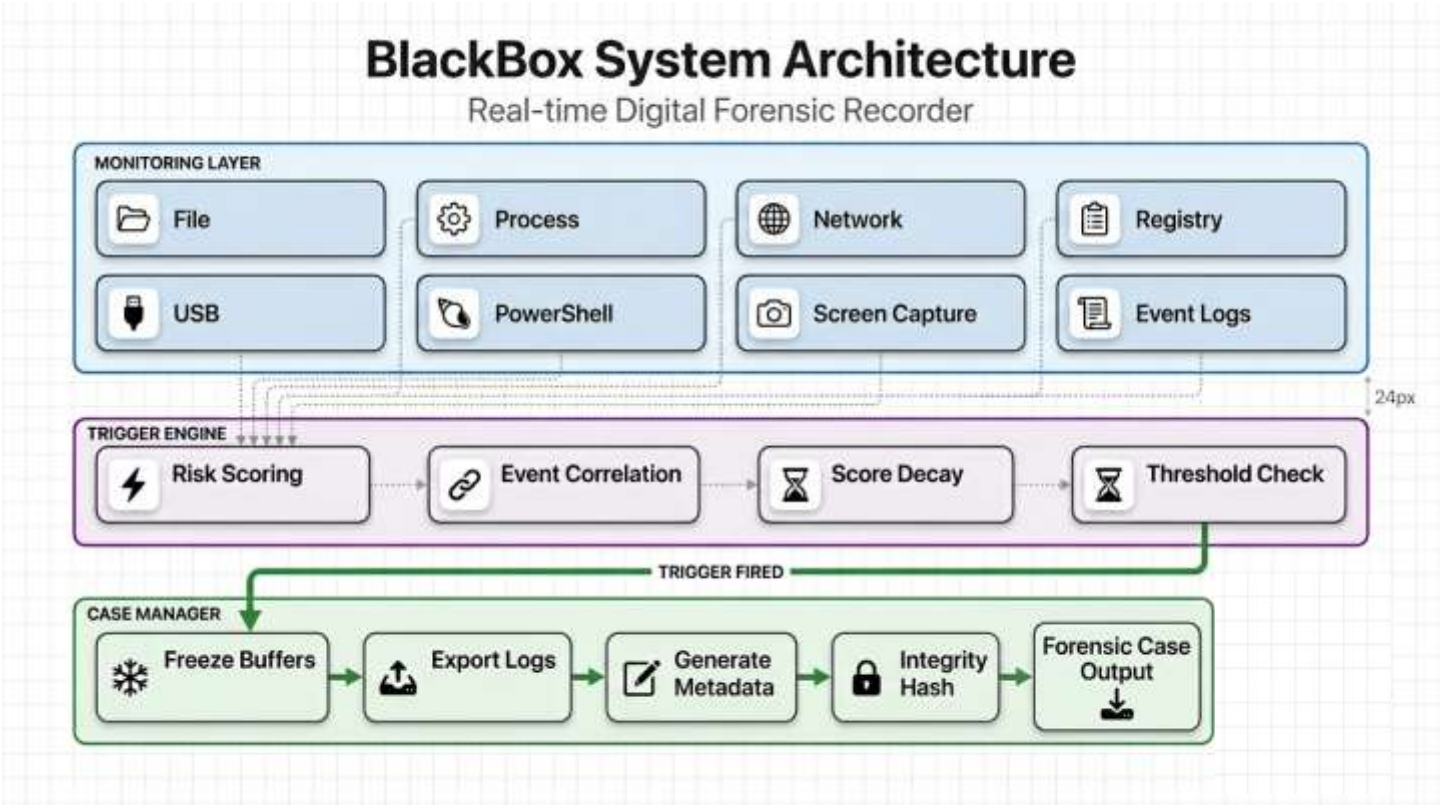

Figure 1 : System Architecture of BlackBox

4.2 Data Collection Components

The BlackBox system collects all kinds of information from all over the place. It looks at what filesre being used what programs are running and who is connected to the network. It also looks at what changesre being made what devices are being used and what commands are being given. The BlackBox system even takes pictures of the screen to help figure out what is going on. All of this information is stored in a format that is easy to look at later.

4.3 Event Correlation and Triggering

One of the things that makes the BlackBox system special is that it does not wait until later to look at all of the information it has collected. It looks at everything in time which means it can see how all of the different things are related. The BlackBox system gives each thing a score. Then it adds up all the scores to get a total risk score. If the score gets too high the BlackBox system does something about it. It can even detect things like ransomware or unauthorized programs.

4.4 Alignment with Forensic Process

The BlackBox system is designed to work like an investigator. It looks at all of the clues collects the evidence and then analyzes it. The BlackBox system does all of this in a cycle, which means it is always watching and always ready to respond. This makes it a powerful tool for keeping the BlackBox system safe.

The BlackBox system follows the basic steps, as a forensic investigator. First it looks for clues. Then it collects the evidence. Next it preserves the evidence so it is not lost. Finally it analyzes the evidence to figure out what happened. The BlackBox system does all of this in a cycle, which means it is always watching and always ready to respond.

5. FINDINGS

The system was tested in different scenarios and some consistent things were observed. Some results were expected, while others became clearer after running many tests.

5.1 Real-Time Monitoring and Detection Speed

One thing that was clear was the impact of monitoring on detection speed. Events were picked up right away after they happened which made the system feel more responsive. This is different from approaches where detection depends on scheduled scans. In those cases there is always a delay even if it is small.

5.2 Event Correlation and Detection Accuracy

Another improvement was noticed when combining signals instead of relying on a single event. Looking at events alone often does not give context.

Some patterns that appeared during testing included:

- File activity with PowerShell usage often indicated higher risk
- Process execution with registry changes looked suspicious

These combinations made detection more reliable and helped reduce false positives.

5.3 Automated Case Generation

The system can generate a case automatically once a trigger condition is met. This removes the need to manually collect data, which can be time-consuming.

It also helps ensure that important details are not missed, in fast-moving scenarios.

5.4 System Performance

A common concern with monitoring is system performance.. In this case the impact was not significant.

With multiple modules running CPU and memory usage stayed within a reasonable range suggesting that the design is lightweight enough for practical use.

5.5 Evidence Integrity

Each generated case includes a hash value, which allows verification of data integrity. This means the collected evidence can be checked to ensure it has not been altered. This is still important for maintaining reliability.

5.6 Experimental Setup

To evaluate the system different scenarios were created, including both suspicious activity.
Some of the test cases included:

- System usage to establish baseline behavior
- Rapid file creation to simulate ransomware-like activity
- Suspicious PowerShell commands
- Processes running from unusual locations

These tests helped in observing how the system reacts under different conditions and whether it triggers case generation.

6. TECHNICAL CHALLENGES

During the development of the BlackBox system, several practical challenges were encountered due to real-time monitoring and multiple modules running simultaneously.

- **Real-Time File Access Issues**

Some files were still in use when the system tried to read or hash them, which caused access errors.
This was handled using retry logic and exception handling so the system continues running without interruption.

- **Handling High Volume of Events**

Real-time monitoring generates a large number of events, especially during rapid file changes.
Filtering and circular buffers were used to store only recent and relevant data, reducing system load.

- **Synchronization Between Modules**

Multiple modules run in parallel, so maintaining consistent data flow to the trigger engine was challenging. This was solved by properly managing shared variables and using live references instead of static copies.

- **Risk Scoring Balance**

Setting the right trigger threshold was difficult.
A decay mechanism was introduced so that only continuous suspicious activity leads to a trigger.

- **Detecting Rapid File Changes**

Rapid file changes were initially detected but did not trigger the system.
This was fixed by assigning higher severity to such patterns so they are treated as critical events.

- **Maintaining System Performance**

Since all modules run continuously, managing performance was important.
Optimizations like controlled polling, reduced logging, and efficient data structures helped keep the system lightweight.

7. CONCLUSION

This research is about the BlackBox system. The BlackBox system is a time digital forensic recording solution for Windows environments. It can continuously monitor system activity. Correlate events across multiple modules. The BlackBox system can also automatically generate a case when it detects suspicious behavior.

The results of this research show that monitoring things in time and using a trigger-based approach can make evidence collection faster and more effective. The BlackBox system does not wait until after something bad happens to look at the data. It captures the data at the exact moment something happens. This reduces the chances of missing information.

Another good thing about the BlackBox system is that it stays lightweight when it is handling many monitoring tasks. This makes it practical to use all the time without slowing down the system.

The BlackBox system has key contributions.

- The BlackBox system is a time multi-module monitoring system.
- It has a risk-based trigger engine.
- It can automatically generate a case with structured logs.
- It keeps evidence safe using hash-based verification.

There are some limitations to the BlackBox system.

- It mainly uses rules to detect things. May miss unknown threats.
- It only works with Windows operating systems now.
- It does not have advanced machine learning models.

There are some things that can be done in the future to make the BlackBox system better.

- The BlackBox system could use machine learning to detect threats.
- It could work with operating systems, not just Windows.
- It could get better at looking at behavior to detect unknown attacks.
- It could. Analyze forensic cases, in the cloud.

8. REFERENCES

[1] B. Carrier, File System Forensic Analysis. Addison-Wesley, 2005.
[2] National Institute of Standards and Technology (NIST), Guide to Integrating Forensic Techniques into Incident Response, NIST SP 800-86.
[3] Microsoft, Windows Event Logging and Event Tracing Documentation.
[4] "Efficient Window Logging for Incident Response," Research Paper.
[5] "Research on Capture of Live Digital Evidence," Research Paper.
[6] "Evaluating Tamper Resistance of Digital Evidence," Research Paper.
[7] S. Tomonaga, "Event Tracing for Windows Internals," Research Paper.
[8] "Anti-Forensic Study," Research Paper.
[9] "Failure to Detect Escalation of Privileges by Common Monitoring Tools," Research Paper.
[10] Python Software Foundation, Python Documentation.
[11] psutil Documentation, https://psutil.readthedocs.io
[12] Watchdog Documentation, https://python-watchdog.readthedocs.io
[13] OpenCV Documentation, https://docs.opencv.org
[14] Microsoft, Windows Registry Documentation.
[15] M. Casey, "Digital Evidence and Computer Crime: Forensic Science, Computers and the Internet," Academic Press.