POLITECNICO DI TORINO

MASTER DEGREE COURSE IN CYBERSECURITY

A.A. 2024/2025

DECEMBER GRADUATION SESSION 2025

Log Analysis and Forensic Implications:

THE IMPORTANCE OF SOC AND DFIR DEPARTMENTS IN CORPORATE CYBERSECURITY AND RELATED CASE STUDY

Supervisors:	**Candidate:**
Andrea Atzeni	Simone Costanzi
Gianluca De Luca Fiscone	
Ciccolunghi Emanuele	
Chiaretti Riccardo	

Ai miei genitori, ai miei zii e ai miei nonni che mi hanno sempre supportato non facendomi mai mancare nulla.

Index

1 Introduction

Over the past few decades, cybersecurity has gone from playing a secondary, almost marginal, role to being considered a fundamental requirement in ICT processes, products, and services for companies. In recent years, standards (such as ISO 27001 and 21434), directives (NIS2), and regulations (DORA, GDPR, Cyber Resilience Act) have been issued to foster a culture of security by design and security by default.

In economic terms, the average cost of a data breach in Italy reached €4.37 million in 2024, a 23% increase compared to 2023, due to increasingly damaging breaches and increased demands on security teams. To minimize attacks, more and more companies are turning to SOC (Security Operation Center) solutions to actively monitor data and promptly detect any anomalies in corporate systems.

The SOC department, in its monitoring and incident response activities, can be supported by the DFIR (Digital Forensics and Incident Response) team. This allows them to rigorously and technically address system compromise cases, attempt to understand how the cyber incident could have occurred, and implement remediation actions to prevent similar incidents in the future.

After this brief introduction, we will first provide an Italian and global overview of the infrastructures most susceptible to attacks and the costs of a cyber attack. Following this economic overview, we will delve deeper into the topic, focusing on the SOC department, log analysis, EDR, XDR, SIEM, and SOAR technologies. This will then lead to a better understanding of how to integrate an incident response team into a company. We will then discuss the DFIR department and its integration with the SOC, and provide an overview of the regulations and standards that must be followed to conduct forensic investigations. This phase will also focus on the MTTR metric and the importance of rapid incident response in limiting the attacker's time in the system.

Finally, after an excursus on malware analysis, some small forensic analysis simulations will be addressed, comparing the monolithic methodology of tools such as Autopsy with a hybrid, more triage-oriented approach such as the Hayabusa-KAPE-Zimmerman Tools suite.

1.1 Italian and global overview

Taking into consideration the Italian situation in today's world, SMEs (Small and Medium Enterprises), Local Health Authorities (ASL) and institutional portals (for example the website of the Ministry, the Carabinieri and some port areas) are increasingly frequent targets of criminal organizations due to their importance in the supply chain and in the processing of sensitive data. Just think of the attacks by the hacker group NoName57(16), which in the last 2 years have cyclically attacked the country with targeted DDoS attacks. At the same time, according to the Cybersecurity & Data Protection Observatory of the Politecnico of Milan, in 2024 Italy has recorded an increase in investments in the cybersecurity sector, with a growth of 15% in 2024, reaching 2.48 billion euros. Forecasts for 2025 indicate a further increase, with many companies making cybersecurity a priority and intending to increase spending in this area.

1.2 Why SMEs and Local Health Authorities?

Local health authorities handle citizens' medical and personal information, which is extremely valuable to cybercriminals and can be used for blackmail or resold on the black market.

- **Critical Infrastructure**: The healthcare sector is considered critical infrastructure, and attacks on hospitals and healthcare systems can have serious consequences for public health and the continuity of services;
- **System vulnerabilities**: Many healthcare systems are outdated and poorly protected, making them easy targets for cyber attacks.

SMEs also represent an attractive target for cybercriminals for the following reasons:

- **Fewer resources dedicated to security**: many SMBs do not have adequate budgets to implement advanced defenses;
- **Lack of awareness**: cybersecurity is not always seen as a priority, leaving plenty of room for attackers;
- **Indirect access:** SMEs can be exploited as a gateway to attack larger companies with which they collaborate (supply chain attacks, see for example the attack on SolarWinds).

The most common attacks include ransomware, phishing, and software vulnerability exploits. In many cases, the damage isn't limited to financial loss: the company's reputation can be severely damaged, with long-term consequences.

1.3 Some data (taken from IBM's Cost of data breach 2024 report)

Globally, 70% of organizations affected by a data breach reported that the breach caused significant or very significant damage. The increase in costs year-over-year is due to lost business and the need for post-breach response by customers and third parties, as collateral damage from data breaches intensified. Recovery took more than 100 days for the 12% of organizations that were able to fully recover from a data breach. Furthermore, law enforcement involvement in ransomware cases saved victims an average of nearly $1 million in breach costs, not including the cost of paying the ransom.

Of these victims, 63% managed to avoid paying the ransom. In Italy, however, analysis of attacks reported by companies confirmed the predominance of phishing attacks, which accounted for 17% of breaches, with an average total cost of €4.18 million per breach. Stolen or compromised credentials followed at 13% (€4.75 million), while social engineering was the most costly entry point (€4.78 million), accounting for 7% of the breaches analyzed. Forty percent of the breaches analyzed in Italy involved data stored across multiple environments (including public cloud, private cloud, and on-prem) and 29% on the public cloud. The data breaches examined that affected multiple environments also resulted in higher costs (€4.49 million on average), while breached data stored on public clouds took longer to identify and contain (254 days).

Regarding the most affected entities, technology companies recorded the most expensive breaches in Italy, with average costs reaching €5.46 million, followed by the industrial sector (€5.13 million) and the pharmaceutical sector (€5.01 million). The three main factors that increased the costs of breaches for Italian organizations were a lack of security expertise (€185,000), the involvement of third parties (€176,000), and the complexity of the security system (€172,000).

Regarding detection times, the Italian companies surveyed took an average of 218 days to identify and contain incidents, 40 days less than the global average of 258 days. This timeframe is further reduced for organizations leveraging AI to detect and contain threats. According to the report, 69% of Italian organizations have adopted AI and automation technologies in their Security Operations Centers (SOCs), an 11% increase from the previous year. More than half of the organizations surveyed globally experienced severe or high-level staffing shortages last year and consequently recorded significantly higher costs per breach ($5.74 million for high levels compared to $3.98 million for low or no levels).

Using the data cited above, it is possible to define the MTTR parameter. Mean Time To Respond (MTTR) is the operational metric that measures the average time between the detection of a security event and its effective mitigation or readiness for mitigation (mitigation-readiness). In the SOC/DFIR context, MTTR represents a direct indicator of infrastructure exposure: shorter response times reduce the window in which an adversary can execute critical phases of the attack (privilege escalation, lateral movement, exfiltration), thus reducing both operational damage (in addition to financial and reputational damage) and the likelihood of lateral propagation. This centrality of MTTR in modern operations is also reiterated by recent studies and reviews (including Kuforiji, 2025) that explicitly link the reduction of response time with damage limitation and the reduction of the attacker's "dwell time." According to Kuforiji, in fact, mean response times (MTTR) are today a critical factor in limiting operational damage and lateral movement phenomena, since "ransomware campaigns, zero-day exploits, and APTs now demand response times measured in seconds rather than hours or days" (Kuforiji, 2025, p. 2)

1.4 Return On Investment (ROI) of the SOC

According to American research firm Statista, the estimated cost of cybercrime in the cybersecurity market is expected to rise steadily from 2024 to 2029, registering an overall increase of 69.41%, reaching a peak of $15.63 trillion by 2029. Therefore, organizations will need to be able to identify investments in cybersecurity solutions to counter threats.

Furthermore, according to the latest data from the Startup Thinking Observatory and the Digital Transformation Academy of the Polytechnic University of Milan, ICT budgets are expected to increase by 1.5% by 2025, maintaining the trend of the last nine years, albeit with a lower growth rate than in 2023 (i.e., +1.9%).

It's important to remember that, in order to achieve adequate budgets, it's also essential to measure the SOC's ROI, considering that it's not just a financial metric, but represents the tangible and intangible benefits of corporate investments in cybersecurity defenses.

1.5 How to Use SOC Metrics and Reports to Calculate ROI

To calculate SOC ROI, it's necessary to compare the benefits and costs of security investments. As mentioned above, benefits include positive outcomes or losses avoided thanks to security activities, such as risk reduction, improved compliance, increased reputation, or productivity. Costs, on the other hand, include the burdens or expenses associated with security activities, such as general operating expenses, capital expenditures, staff training, or incident response. In fact, SOC metrics and reports can be used to estimate both the benefits and costs of security investments, applying formulas, models, or frameworks suited to the specific context and objectives. The Annualized Loss Expectancy (ALE) formula—i.e., ALE = ARO (Annualized Rate of Occurrence) x SLE (Single Loss Expectancy)—represents the expected monetary loss in a year due to the occurrence of a threat. It can be used to estimate potential losses from cyber incidents and compare them with the Annualized Control Cost (ACC) to evaluate the savings resulting from security investments. Furthermore, the ALE is the value that should be compared with annual security spending to calculate the Return on Security Investment (RoSI), which results from the formula RoSI = (Benefits of the Security Investment – Cost of the Security Investment) / Cost of the Security Investment.

According to the IBM Cost of a Data Breach Report 2024, the average cost of a data breach in Italy is 4.37 million euros, with peaks of 5.46 million euros in the technology sector.

To evaluate the suitability of an investment in a Security Operation Center (SOC), it is possible to use the Annualized Loss Expectancy (ALE):

$$ALE = AROxSLE \tag{1}$$

where:

ARO (Annual Rate of Occurrence) = annual probability of experiencing an incident.

SLE (Single Loss Expectancy) = average cost of a single incident.

Suppose an Italian tech company has:

SLE = €5.46 million (IBM data for the tech sector)

ARO without SOC = 20% → ALE without SOC = €1.092 million

ARO with SOC = 6% → ALE with SOC = €327,600

Base internal SOC cost = €300,000/year (estimate adapted from Arctic Wolf, UnderDefense for the Italian context).

Annual benefit: €1.092 million – €327,600 = €764,400

RoSI (Return on Security Investment):

$$RoSI = (BenefitCostSOC)/CostSOC = \frac{764.400 - 300.000}{300.000} = 1,55 \tag{2}$$

The result (+155%) demonstrates that, in this scenario, investing in a SOC is not only justified from a risk reduction perspective, but also brings a positive economic return already in the first year in the absence of multiple incidents.

1.6 A possible remedy: SOC outsourcing

For SMEs, creating an internal SOC can be prohibitive in terms of costs and resources. The need for advanced technological infrastructure, highly qualified personnel, and continuous monitoring poses a significant challenge. Fortunately, an external SOC (SOC-as-a-Service) can offer an effective and scalable solution.

Advantages:

1. Cost reduction: managed services eliminate the need for significant upfront investments, such as purchasing hardware and software or hiring dedicated staff;

2. Access to specialized expertise: external SOC providers have security experts who monitor and respond to threats promptly. Organizations periodically require access to specialized security experts, such as incident responders, malware analysts, and cloud security architects. These skills can be rare and difficult to maintain within the company. A SOC-as-a-Service provider can offer its customers access to qualified cybersecurity specialists when needed;

3. Flexibility: services can be tailored to your specific business needs, including hourly monitoring models;

4. Focus on core business: by outsourcing security, the company can focus on its core activities without worrying about managing complex IT systems.

1.7 The Challenges of SOC-as-a-Service

Despite the numerous advantages of a SOC-as-a-Service offering, outsourcing security isn't always a simple task. Some common challenges organizations opting for managed SOC services encounter are:

- **Onboarding process**: Managed SOC providers typically use their own security stack, and these solutions must be implemented and configured in the customer's environment before the provider can begin offering services. This onboarding process can be time-consuming and can leave the organization vulnerable to cyber threats during the transition;
- **Company data security**: An organization's SOC-as-a-Service provider needs deep insight into the corporate network to identify and respond to potential threats. To gain this insight, the organization must send large amounts of sensitive data to its service provider. This need to cede control of such a large amount of potentially sensitive information can make corporate data security and risk management more challenging;
- **Log delivery cost**: SOC-as-a-Service providers typically manage their cybersecurity solutions on-site, using data feeds and network intercepts from their customers' networks. This means that log files and other alert data are generated and stored on the provider's network and systems. Obtaining access to complete log data from a managed SOC provider can be costly for an organization.

1.8 What does the Security Operation Center do?

A Security Operations Center (SOC) is a team, typically supported by advanced technology tools, dedicated to monitoring, detecting, and responding to cyber threats in real time. Its primary goal is to protect an organization's IT infrastructure, ensuring that any anomalies are identified and addressed before they cause significant damage. Typically, a SOC is designed using a centralized hub-and-spoke configuration. It consists of a Security Information and Event Management (SIEM) system that collects and correlates data coming from security feeds. The main functions of a SOC:

1. Continuous monitoring of network activities;
2. Incident Management: respond quickly to security events, reducing threat exposure times;
3. Vulnerability and patch management: identify vulnerabilities in systems and apply the necessary patches to mitigate risks;
4. Log collection and analysis: examine data generated by IT systems to identify patterns that may indicate an attack;
5. Proactive prevention and response: implement preventative measures, such as firewalls and intrusion detection systems (IDS/IPS), to block threats before they materialize.
6. Routine maintenance and preparation: to maximize the effectiveness of the security tools and measures in place, the SOC performs preventative maintenance, such as applying patches and software updates and continuously updating firewalls, allow and block lists, security policies, and procedures. The SOC can also create system backups or assist in establishing backup policies or procedures to ensure business continuity in the event of a data breach, ransomware attack, or other cybersecurity incident.

1.9 SOC ANALYST

The Security Analyst is responsible for performing tasks that are useful for preventing threats and thus protecting corporate systems and networks. Based on their technical knowledge and skills, Security Analysts are typically classified into three levels, each with different tasks and roles: L1, L2, and L3. Generally, the tasks performed by Security Analysts are as follows:

- Monitoring systems and network activity to detect threats and identify suspicious behavior;
- Data analysis to identify attack patterns and strategies;
- Conducting infrastructure tests to uncover vulnerabilities;

- Configuring cybersecurity tools and active network components;
- Risk assessments to establish the effectiveness of security measures;
- Report generation;
- The diffusion of cyber culture inside the company

Below are the duties performed by each level (note that the tasks assigned to each level may be different in each company).

1.9.1 SOC Analyst L1: The Front Line

The Level 1 Analyst is the SOC's first line of defense and is responsible for monitoring and initial alert triage. Their role is crucial for filtering the enormous volume of notifications generated by security systems and identifying those that require immediate and deep attention.

Main Responsibilities:

- **Continuous monitoring**: They constantly monitor security consoles, particularly the SIEM (Security Information and Event Management), to examine alerts generated in real time;
- **Alert Triage**: They analyze each alert to determine whether it's legitimate activity, a false positive, or a potential threat requiring further investigation. This requires speed and the ability to process hundreds of alerts per day;
- **Initial Investigation**: They perform basic checks, such as analyzing logs and checking IP addresses or file hashes against threat intelligence platforms to enrich the context of the alert;
- **Documentation and Escalation**: They document their findings in a ticketing system. If an alert is confirmed as a potential incident, they escalate it to a Level 2 analyst for more detailed analysis;
- **Playbook management**: They follow standard operating procedures and predefined playbooks to respond to known and common threats, such as user-reported phishing emails.

The main tools that an L1 SOC Analyst uses are the following: SIEM, EDR, XDR, SOAR (all these technologies will be described in more detail later) and threat intelligence platforms.

1.9.2 SOC Analyst L2: The Investigator

When an incident is escalated from Level 1, the Level 2 analyst takes over the investigation. This role requires greater experience and deeper technical expertise to analyze the nature, scope, and root cause of a threat.

Main Responsibilities:

- **In-depth Incident Analysis**: They conduct detailed investigations into escalated incidents. They correlate information from various sources (system logs, network traffic, endpoint data) to reconstruct the attack chain and better understand the underlying alert scenario;
- **Incident Response**: They act as first "incident responders", coordinating actions to contain the threat (e.g. isolating an infected system from the network), limiting the damage and providing direct escalation to the customer (in the event that the company has relied on an external SOC service);
- **Basic Forensic Analysis**: They examine logs, analyze network packets (e.g., with Wireshark), and use sandboxes to analyze malware samples in a controlled environment;
- **Improving Defense Systems**: They provide feedback to the L1 team and recommend tuning SIEM rules to reduce false positives and improve detection accuracy;

- **Reporting**: They produce detailed incident reports, documenting the attack methodology, compromised systems, and remediation actions taken.

As for the tools used by L2 analysts, in addition to knowledge of SIEM, EDR, XDR and SOAR solutions, they also use some basic forensic analysis tools such as Wireshark and TCPDump for network traffic analysis, malware sandboxes and scripting languages (Python, PowerShell) to automate analyses.

1.9.3 SOC Analyst L3: Threat Hunter

The Level 3 Analyst is the most experienced and qualified member of the SOC, often referred to as the "Threat Hunter." They handle the most complex and sophisticated threats, such as Advanced Persistent Threats (APTs), and work proactively to uncover hidden threats within the infrastructure.

Main Responsibilities:

- **Proactive Threat Hunting**: They don't wait for alerts, but actively search for unknown or undetected signs of compromise within the network. They develop hypotheses based on threat intelligence and verify them by analyzing large amounts of data;
- **Advanced Malware Analysis**: They reverse engineer malware to fully understand its functionality, capabilities, and indicators of compromise (IOCs);
- **Critical Incident Management**: They lead the response to the most serious security incidents, developing complex containment, eradication and recovery strategies;
- **Countermeasure Development**: They design and implement new custom detection rules, SIEM correlation policies, and SOAR playbooks, improving the defensive capabilities of the entire organization;
- **Mentorship and Training**: They play a crucial role in training and developing the skills of L1 and L2 analysts, acting as a technical point of reference for the entire team.

Regarding tools and knowledge, in addition to the tools mentioned above, the L3 analyst is familiar with threat hunting platforms, reverse engineering tools (IDA Pro, Ghidra), forensic analysis frameworks, and big data analysis tools.

1.10 Tools used in a SOC

To properly monitor systems and network activity, analysts have access to several tools that allow them to accurately analyze the logs produced by the various devices within the company's IT infrastructure and then transform that raw data into useful information for ensuring IT security. Before describing these tools, it's important to briefly discuss the importance and method of log analysis.

2 Background

To understand the connection between SOC analysts and forensic analysts, it's necessary to proceed step by step, trying to understand how log analysis works and the tools SOC analysts use in their monitoring activities.

Once the tools and their uses are understood, it will be possible to grasp the profound connection between the two departments.

2.1 Log Analysis

Log analysis is the process of reviewing, interpreting, and understanding logs generated by systems, networks, and applications. These logs are like fingerprints of every action that occurs within a system. They contain valuable information that can help us understand what's happening in our IT environment, from identifying potential security threats to troubleshooting performance issues.

2.2 Basics

One of the main benefits of log analysis is related to security. By regularly analyzing logs, you can identify unusual activity that could signal a potential security threat. For example, multiple failed login attempts from a single IP address could indicate a brute force attack. By detecting such threats early, you can intervene before they become serious security breaches.

Log analysis typically includes the following steps:

1. **Data Collection**: The first step in the log analysis process is data collection. This involves gathering log data from various sources, such as servers, network devices, and applications. Data can be collected manually, but it's often more efficient to use automated tools that can collect and centralize logs in one place.

2. **Data Indexing**: Once the data is collected, the next step is indexing. Indexing involves organizing log data for easier search and analysis. This typically involves categorizing data based on various attributes, such as timestamp, source, and event type, followed by normalization and analysis of the order of common fields compared to other log types. Proper indexing is essential for efficient log analysis, as it allows you to quickly locate relevant log entries when needed.

3. **Analysis**: After indexing, the log data is ready for analysis. This is where you can analyze the logs to extract valuable information. You can look for patterns or anomalies that might indicate a security issue or threat. You can also use the logs to try to understand the processes that might be slowing down an application or who accessed certain resources during a specific time period.

4. **Monitoring**: Monitoring is an ongoing part of the log analysis process. It involves monitoring logs to detect any unusual or suspicious activity. This can be done manually, but it's generally more efficient to use log monitoring tools that can alert you when certain conditions occur, such as a sudden increase in error logs or multiple login attempts from an unusual location.

5. **Reporting**: Monitoring is an ongoing part of the log analysis process. It involves monitoring logs for any unusual or suspicious activity. This can be done manually, but it's generally more efficient to use log monitoring tools that can alert you when certain conditions occur, such as a sudden increase in error logs or multiple login attempts from an unusual location.

2.3 A bit of history

Before briefly discussing some aspects of log analysis, it's worth mentioning its evolution.

Until the 1990s, log analysis was performed manually using text-based tools (grep, awk, shell scripts). The focus was primarily on manual troubleshooting and operational support. However, since this activity was time-consuming and there was a real possibility that suspicious events could escape the analyst's attention, a change of direction was needed. The increase in data that began to occur in those years (just consider that with the advent of the Waterfall paradigm, each application began to have its own logs, and even system logs began to become more complete and complex) led to the development of the first log analysis systems to address their lack of contextualization and aggregation.

In the early 2000s (2000–2006), the term SIM/SEM emerged, meaning Security Information Management (SIM) and Security Event Management (SEM) solutions were developed to aggregate and correlate events from firewalls, IDS/IPS, and servers. The goal was to reduce noise and identify relevant events.

Gradually (2006–2013), SIEM solutions began to evolve to be more scalable (indexing, full-text search—e.g., Splunk 2003) and to have operational dashboards. From this point on, SIEM became central to compliance and log-based detection.

2.4 Log Analysis Techniques and Methods

As previously mentioned, log analysis is the fundamental process underlying SOC operations. The main challenge of log analysis lies not only in the importance of the data but also in its nature, particularly the following aspects:

1. Volume: a medium-sized company can generate billions of log events per day, a volume that is unmanageable by hand;
2. Velocity: logs are produced in real time, and during an attack, the velocity of events can skyrocket, requiring immediate analysis for effective containment;
3. Variety: logs come from a variety of sources (firewalls, Windows servers, custom applications, cloud), each in its own unstructured or semi-structured format.

To address this complexity, analysis methodologies have evolved from simple text searches to sophisticated artificial intelligence models. This evolution reflects a paradigm shift: from searching for known threats (Pattern Matching) to discovering anomalous behavior (Anomaly Detection), and finally to semantic understanding of events (AI/LLM).

2.4.1 Classical Approaches: Pattern Recognition and Signatures

Pattern recognition in log analysis is like looking for a needle in a haystack. It involves identifying patterns or trends in log data that could indicate a problem or anomaly. Pattern recognition algorithms are often used to simplify and make this process more efficient. They can help identify common patterns such as repeated failures, unusual activity, or spikes in resource usage. Pattern recognition doesn't just identify problems; it also helps predict future trends. For example, if log data shows regular spikes in server load at certain times of the day, this information can be used to anticipate and manage peak periods.

Here are some techniques you can use:

- Text Search (grep/regex): the most basic approach is to use regular expressions (regex) to search for specific strings within log files (e.g. grep "Failed password" /var/log/auth.log).
- Structured Parsing (Grok): tools like Logstash (part of the ELK stack) use Grok filters, which are essentially predefined, named regexes, to extract structured fields (e.g., source IP, user) from unstructured logs.
- Signature-based rules (YARA/Sigma): this is pattern matching applied to threat detection. Sigma rules, for example, provide a generic, open-source format for describing malicious log patterns (TTPs) that can then be translated into queries for various SIEMs (Splunk, QRadar, Sentinel).

These methodologies have the advantages of speed and accuracy (since exact string searches are computationally efficient and generate very low false positive rates) and clarity due to the binary result (presence or absence of the pattern). On the other hand, previous techniques are not effective against unknown threats. They can only detect "known threats" (documented Indicators of Compromise) and cannot cover zero-day threats or attack variants that slightly modify their signature (e.g., polymorphic malware). For this reason, constant signature and rule updates are required to remain effective.

2.4.2 Statistical Approaches: The Emergence of Anomaly Detection

Because pattern matching fails to detect unknown threats, the next evolutionary step was anomaly detection. The goal here is no longer to search for "evil," but to define "normal" and flag anything that deviates from it. As the survey by Landauer, Onder, Skopik, and Wurzenberger shows, some common techniques include:

- Statistical Baselining: the system monitors metrics over a period of time and builds a baseline (e.g. "Server 'SRV-DB-01' typically has 10 connections per minute and 5MB of traffic"). An alarm is triggered if it detects 1,000 connections or 500MB of traffic;

- Clustering (e.g., K-Means): logs are grouped based on common characteristics. Events that do not fit into any cluster ("outliers") are flagged as anomalies;

- Multidimensional Statistical Analysis (e.g. PCA): techniques such as Principal Component Analysis (PCA) are used to reduce the dimensionality of complex logs and identify the most significant deviations.

Compared to previous models, statistical approaches can detect zero-days, but on the other hand, they introduce (significantly) false positives that must be managed by analysts. In fact, an "anomalous" activity does not necessarily imply malicious activity (a software update or nightly maintenance could generate legitimate anomalies).

2.4.3 Modern Approaches: Machine Learning and Deep Learning

To overcome the limitations of statistical methods (too much noise) and pattern matching (too rigid), research has shifted to machine learning (ML) and, more recently, deep learning (DL) models. Before listing some of the possible techniques, the following premise is necessary. Since log data is generally unstructured, it must be preprocessed in some way before being fed into deep learning systems. The research presented by Landauer, Onder, Skopik, and Wurzenberger shows that there are two main approaches to handling unstructured data. The most common approach is to use parsers, usually called log keys, to extract unique event identifiers. An alternative to parsing is token-based strategies that break log messages into word lists, for example, by separating them into whitespace. It is therefore common to clean the data by lowercaseing all letters and removing special characters and empty words before obtaining the final word vectors. Although these approaches derive less semantic information from individual tokens, they have the advantage of being more flexible as they rely on generally applicable heuristics rather than predefined parsers. After this operation, it is important to perform a grouping operation. Indeed, the goal of deep learning in log analysis is to understand suspicious behavior by knowing the context, and this is possible, for example, by grouping by time windows (which could be sliding or fixed time).

Let us now list the techniques applied by modern approaches:

- Supervised ML (e.g. Random Forest, SVM): these models are trained on labeled datasets ("normal" logs and "malicious" logs). They are very effective at classifying novel events, but require the difficult and expensive creation of training datasets;

- DL for Sequential Analysis (e.g. RNN, LSTM): logs are not isolated events; they are sequences. An attack is a chain of events. As highlighted by academic surveys (e.g., "Deep learning for anomaly detection in log data"), deep learning models such as Long Short-Term Memory (LSTM) are specifically designed to understand context and temporal dependencies in sequential data;

- Unsupervised DL (e.g. Autoencoders): an autoencoder is a DL model that learns to "compress" and "reconstruct" normal logs. When it receives an abnormal log as input (e.g., an obfuscated command), the model fails to reconstruct it accurately, generating a "high reconstruction error" and flagging the anomaly.

The adoption of DL has significantly improved accuracy. Autoencoders, for example, are much more effective than statistical methods in defining "normal" in complex, multidimensional systems. So, for example, the sequence

"Login Failed -> Login Failed -> Login Successful -> User Creation" is highly suspect, while individual events may not be. The advantages include a greater understanding of the context (unlike statistical methods), which leads to greater accuracy (significantly reducing false positives compared to traditional anomaly detection). However, the main disadvantage is the model's lack of explainability (black box) and the high computational and training costs.

2.4.4 The Emerging Frontier: AIOps and Generative AI (LLM)

The latest evolution shifts the focus from automated detection to automated analysis and augmenting the capabilities of human analysts. This is the domain of AIOps and Generative Artificial Intelligence (GenAI) through Large Language Models (LLM).

Techniques:

- AIOps (AI for Operations): apply AI to intelligently cluster and correlate alerts. For a numerical example, perhaps instead of generating 1,000 separate alerts, AIOps groups them into three distinct "Incidents," identifying the likely root cause and reducing noise for the L1 analyst;
- Generative AI (SOC Co-pilots): LLMs are trained not just on logs, but on human language, technical documentation, and threat intelligence reports.

As highlighted by industry leaders like IBM ("How AI-driven SOC co-pilots will change security center operations"), GenAI is redefining SOC operations. Its impact is not only on detection, but on the entire Incident Response (IR) workflow, directly addressing the skills gap:

1. Democratization of Expertise: an L1 analyst can now ask complex natural language questions (e.g., "Show me all users who have had RDP logins followed by new process creation in the last 24 hours") and the LLM generates the query (Splunk SPL, KQL) that would otherwise require an L3 analyst;
2. Incident Summarization: when faced with a complex incident, the LLM can analyze thousands of related logs and provide a natural language summary: "This is a ransomware attack. Initial access occurred via RDP to host X";
3. Investigation Guidance (DFIR): the LLM can suggest next steps for the forensic investigation, acting as a virtual "L3 expert" (e.g., "I suggest acquiring a memory dump from host X and running Volatility to look for suspicious processes").

The introduction of Gen AI certainly helps analysts with their analyses by automating certain tasks and providing greater context and understanding of the log (significantly reducing the time between detection and response, MTTR). On the other hand, the analysis performed by LLM models should not be considered foolproof, and analysts must review everything to avoid the phenomenon of "AI Hallucination" (i.e., the generation of false but convincing information). A final critical issue concerns the security of the logs themselves: the use of cloud-based AI models (APIs) raises significant privacy (GDPR) and security concerns, as sensitive corporate logs are sent to third parties for analysis.

2.4.5 Towards an Augmented SOC

The evolution of log analysis is a clear path from reactive string searching (Pattern Matching), to proactive deviation discovery (Anomaly Detection), to contextual sequence understanding (Deep Learning), to collaboration and understanding meaning (Generative AI). The future of log analysis in the SOC is not a human-free environment, but an augmented SOC, where Artificial Intelligence manages volume and velocity, freeing human analysts (SOC and DFIR) from noise and allowing them to focus on high-value activities: critical analysis, defensive strategy, and proactive threat hunting.

2.5 What is Log Management?

Log management is the practice of managing large volumes of computer-generated log data and messages. Various computer systems and applications generate logs, including:

- Servers
- Databases
- Websites
- Network devices and endpoints

The logs contain valuable information about events that occur on these systems and can be used to troubleshoot potential problems or monitor system performance.

2.6 Importance of log management

Log management allows organizations to track all activities within their IT infrastructure. This can be useful in a variety of business situations:

- Identifying Security Breaches
- Troubleshooting technical issues.
- System performance monitoring.

Additionally, many regulatory bodies require companies to retain log data for a specified period of time as part of compliance regulations. Log management simplifies compliance with these requirements.

2.7 Types of logs

To understand your data, you need to understand the variety of logs you might encounter. Each type of log provides unique, often vital, data. Here are some types of logs you might encounter:

2.7.1 Server logs

Server logs contain crucial data about user activity, system errors, and other operational details. Server logs can help identify performance issues, unauthorized access attempts, and suspicious activity.

2.7.2 Application logs

Application logs are indispensable tools for system administrators, as they provide detailed information on software behavior, user interactions, and potential issues that affect user experience. Application logs can be useful for:

- Identifying anomalies or inconsistencies that impact the performance of an application;
- Understand user behaviors and patterns to improve user experience;
- Troubleshooting and fixing issues in the application;
- Maintain a historical log of software activity for audit and compliance purposes.

2.7.3 Network logs

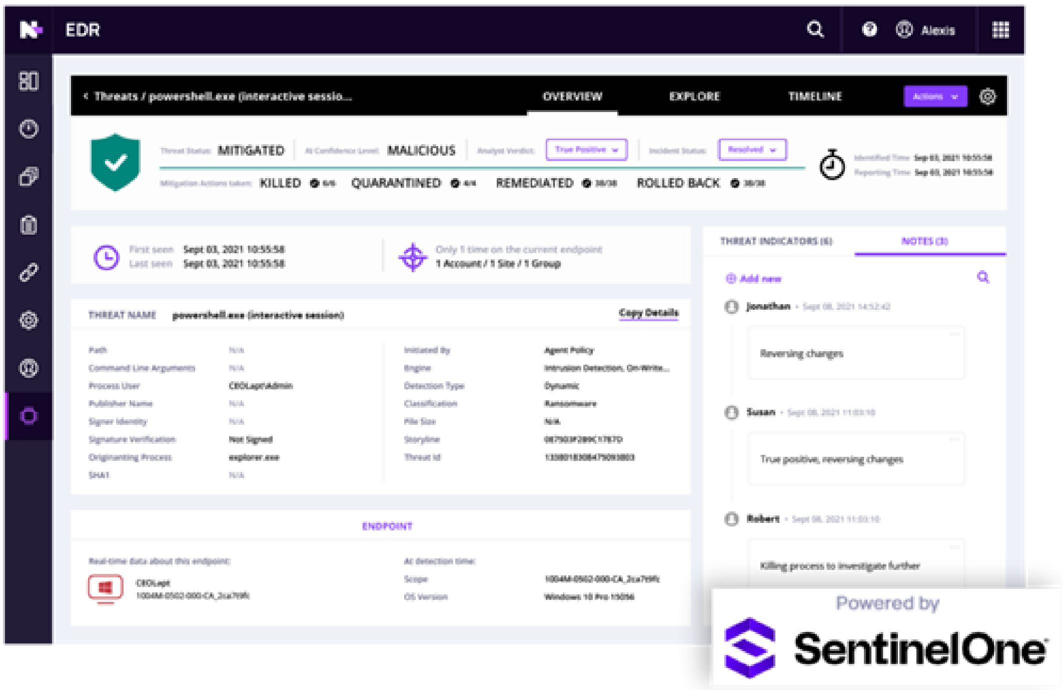
Network logs, which record traffic entering and leaving a network, provide useful information for troubleshooting and identifying potential problems. A network log:

- Contains information about server performance;
- It can help identify unusual patterns or anomalies in network traffic.
- Supports the detection and mitigation of security threats.

Having completed the log analysis narrative, we can now proceed to analyze the tools used by the SOC Analyst during his monitoring activity.

2.8 EDR (Endpoint Detection and Response)

EDR is a cybersecurity technology that continuously monitors endpoints for evidence of threats and takes automated actions to mitigate them. Endpoints are the numerous physical devices connected to a network, such as mobile phones, desktops, laptops, virtual machines, and Internet of Things (IoT) devices, which offer malicious actors multiple entry points for an attack. EDR solutions enable security analysts to detect and remediate threats on endpoints before they can spread across the network. EDR security solutions record endpoint behavior 24/7 and continuously analyze this data to reveal suspicious activity that could indicate threats such as ransomware. They can also take automated actions to contain threats. For example, the EDR developed by SentinelOne uses its behavioral analysis algorithm to automatically terminate and quarantine the process if it detects potentially malicious or dangerous activity, providing analysts with key information for in-depth analysis.



Threat Status: MITIGATED | AI Confidence Level: MALICIOUS | Analyst Verdict: True Positive | Incident Status: Resolved

Mitigation Actions taken: KILLED 6/6 QUARANTINED 4/4 REMEDIATED 38/38 ROLLED BACK 38/38

Identified Time: Sep 03, 2021 10:55:58
Reporting Time: Sep 03, 2021 10:55:58

First seen Sept 03, 2021 10:55:58
Last seen Sept 03, 2021 10:55:58

Only 1 time on the current endpoint
1 Account / 1 Site / 1 Group

THREAT NAME powershell.exe (interactive session) Copy Details

Path	N/A	Initiated By	Agent Policy
Command Line Arguments	N/A	Engine	Intrusion Detection, On-Write...
Process User	CEOLapt\Admin	Detection Type	Dynamic
Publisher Name	N/A	Classification	Ransomware
Signer Identity	N/A	File Size	N/A
Signature Verification	Not Signed	Storyline	0E7503F289C1787D
Originating Process	explorer.exe	Threat Id	1330018308479093803
SHA1	N/A		

ENDPOINT

Real-time data about this endpoint:
CEOLapt
1004M-0502-000-CA_2ca79fc

At detection time:
Scope 1004M-0502-000-CA_2ca79fc
OS Version Windows 10 Pro 15056

THREAT INDICATORS (6) NOTES (3)

Add new

Jonathan · Sept 08, 2021 14:52:42
Reversing changes

Susan · Sept 08, 2021 11:03:10
True positive, reversing changes

Robert · Sept 08, 2021 11:03:10
Killing process to investigate further

Figure 1: Sentinel One EDR Interface Example

2.9 Role of EDR in Cybersecurity

For organizations working to stay safe from a cyberattack, EDR represents a step beyond antivirus technology. An antivirus program is designed to prevent malicious actors from entering a system by checking a database for known threats and taking automatic quarantine actions if any are detected. Endpoint protection platforms are the first line of defense, including advanced antivirus and antimalware protection, and an EDR provides additional protection in the event of a breach by enabling detection and remediation. An EDR can search for unknown threats that slip past the perimeter by detecting and analyzing suspicious behavior, otherwise known as indicators of compromise (IOC). It also provides security teams with the visibility and automation needed to speed up incident response and prevent attacks from spreading to endpoints. Endpoints are accustomed to:

- monitor endpoints and maintain a complete record of activities to detect suspicious activity in real time;
- analyze this data to determine whether the threats warrant investigation and remediation;
- generate prioritized alerts for your security team so they know what needs to be addressed first;
- provide visibility and context into the full history and scope of a breach to aid security teams in their investigations;
- automatically contain or remediate the threat before it can be distributed.

To put things into context, the migration from Antivirus to EDR solutions began in 2010, in fact, as previously mentioned:

- Traditional antivirus software is limited in its ability to combat fileless, escalatory, and living-off-the-land attacks. This creates a need for tools that monitor endpoint activity and behavior;
- Gartner (and analysts) start talking about Endpoint Threat Detection & Response: the term EDR consolidates around 2013 as a category for tools that collect endpoint telemetry, enable investigations and responses.

2.10 How does an EDR work?

While EDR technology may vary by vendor, it works similarly across vendors. An EDR solution:

1. **Continuously monitor endpoints**. When onboarding devices, the EDR solution will install a software agent on each one to ensure the entire digital ecosystem is visible to security teams. Devices with the agent installed are called managed devices. This software agent continuously logs relevant activity on each managed device;
2. **Aggregate telemetry data**. The data entered by each device is sent back by the agent to the EDR solution, which can be in the cloud or on-premises. Logs regarding events, authentication attempts, application usage, and other information are made visible to security teams in real time;
3. **Analyze and correlate data**. The EDR solution identifies IOCs that would otherwise be easily missed. EDR records typically use artificial intelligence and machine learning to apply behavioral analytics based on global threat intelligence to help the team avoid using advanced tactics against the organization;
4. **Detects suspicious threats and takes automatic remediation actions**. The EDR solution flags a potential attack and sends a useful alert to the security team so they can respond quickly. Depending on the trigger, the EDR system can also isolate an endpoint or otherwise contain the threat to prevent its spread while the incident is being analyzed;
5. **Archive data for future use**. EDR technology maintains a forensic record of past events to inform future investigations. Security analysts can use it to consolidate events or gain a comprehensive picture of a sustained or previously undetected attack.

2.11 Impact of EDR on Incident Response

EDR security solutions can help your team implement incident response plans:

- **Containment, elimination and recovery**: EDR solutions allow teams to quickly isolate infected endpoints, block traffic to and from malicious IP addresses, and begin executing any playbook and response plan steps to mitigate the threat. EDR tools continuously capture images of endpoints, making it easy to roll back to a previous unaffected state;

- **Post-event analysis**: EDR provides information about endpoint activity, network connections, user actions, and file changes that can help analysts perform root-cause analysis to identify the source of an event. It also speeds up the analysis and reporting process on what worked well and what didn't, so such events don't recur in the future.

Proactive cyber threat detection is a security exercise that analysts perform to search for unknown threats in networks. EDR solutions support this effort by leveraging forensic data to help analysts decide which IOCs to target, such as specific files, configurations, or suspicious behavior. In a cyber landscape where malicious actors often hide within an undetected environment for months, threat hunting is a valuable way to strengthen security posture and meet compliance requirements. Some EDR solutions allow analysts to create custom rules for targeted threat detection. These rules proactively monitor various events and system states, including suspicious breach activity and misconfigured endpoints. They can be configured to run at regular intervals, generate alerts, and take response actions whenever matches are found.

2.12 How an XDR (Extended Detection and Response) works

The term XDR (Extended Detection and Response) was proposed in 2018 (often cited as having been coined by industry figures, such as Nir Zuk of Palo Alto Networks) to describe an approach that extends telemetry beyond the endpoint (network, email, cloud workloads, identity) and correlates it natively for cross-layer detection and response. An XDR uses artificial intelligence and advanced analytics to monitor numerous domains in an organization's technology environment, identify alerts, correlate them into incidents, and prioritize them based on risk (obviously, the higher the risk level, the higher the priority for event management). An XDR system:

- **Collects and normalizes data**: The system automatically ingests telemetry data from multiple sources. It cleans, organizes, and standardizes the data to ensure consistent, high-quality data is available for analysis;

- **Analyze and correlate data**: The system uses machine learning and other artificial intelligence capabilities to automatically analyze data and correlate alerts with incidents. It can analyze data and identify cyberattacks and malicious behavior in real time much faster than security teams attempting to manually correlate incidents and resolve alerts;

- **Facilitates incident management**: The system prioritizes the severity of new incidents and provides more context, helping security personnel more quickly assess, recognize, and respond to the most significant cyber threats. Depending on current conditions, personnel can respond manually or allow the system to respond automatically, such as quarantining devices or blocking IP addresses and mail server domains. Security analysts can also review incident reports and recommended solutions and take action accordingly;

- **It helps to avoid future unexpected events**: By analyzing extensive threat intelligence, some XDR systems provide detailed information about cyber threats relevant to an organization's specific environment, including attack techniques and recommended remediation actions. Security teams can use this detailed information to proactively protect against cyber threats that pose the greatest risk to the company's core business.

2.13 Common XDR use cases

Today's cyber threats are diverse and can attack different targets (or even the same ones) through different payloads and methodologies. Therefore, it's important to have systems that employ different detection sources and investigation methodologies. Therefore, with XDR, companies have greater flexibility to address a wide range of cybersecurity issues across all IT environments. Below are some common XDR use cases:

- **Cyber threat research**: With XDR, organizations can automate cyber threat detection, proactively hunting for unknown or undetected threats in an organization's security environment;
- **Investigation of Unexpected Security Events**: XDR automatically collects data across different attack surfaces, correlates anomalous alerts, and performs root cause analysis. A central management console provides complex attack views, allowing security teams to determine which unexpected events are potentially damaging and require further investigation;
- **Threat Intelligence and Analysis**: XDR gives organizations the ability to access and analyze high volumes of raw data on emerging or existing threats;
- **Phishing and email malware**: When employees and customers receive suspicious emails that are part of a phishing attack, they often manually forward the emails to IT. IT then forwards the emails via ticket submission to security analysts for manual review, resulting in manual effort and lengthy wait times for ticket processing and handling. With XDR, companies can automatically analyze emails, identify those with malicious attachments, and delete all infected messages across the organization. This increases protection and reduces repetitive tasks. Similarly, XDR automation and AI capabilities can help teams proactively detect and contain malware;
- **Insider threats**: Insider threats, whether intentional or unintentional, can lead to account compromise, data exfiltration, and corporate reputation damage. XDR uses behavioral and other models to identify suspicious online activity, such as credential misuse and large data uploads, which could signal insider threats;
- **Endpoint device monitoring**: With XDR, security teams can automatically perform endpoint health checks, using indicators of compromise and attack to detect ongoing and pending threats. XDR also provides visibility across endpoints, allowing security teams to determine where threats originate, how they spread, and how to isolate and stop them.

2.14 SIEM (Security Information and Event Management)

It is a fundamental cybersecurity tool that enables organizations to detect, analyze, and respond to security threats. SIEM solutions (such as Microsoft's Sentinel, Splunk, IBM's QRadar, and Google's Chronicle) collect data from various sources, analyze it, and correlate it to identify potential threats and anomalies, providing a comprehensive view of an organization's security.

In detail, a SIEM:

- **Collects data**: receives and aggregates logs, events, and other security data from multiple sources within an organization's IT infrastructure, such as servers, applications, firewalls, operating systems, and other security devices;
- **Analyze the data**: processes the collected data to identify anomalies, suspicious behavior, and potential threats;
- **Correlation of events**: combines data from multiple sources to create a more complete and contextualized view of security events;
- **Threat detection**: uses rules, behavioral analytics, and other methods to identify malicious activity and threats in real time;

- **Incident management**: helps respond to security incidents by providing detailed information and context for resolution.

2.15 Advantages of SIEM

Regardless of an organization's size, it's essential to take proactive measures to monitor and mitigate IT security risks. SIEM solutions offer businesses numerous benefits and are a key component in streamlining security workflows and recognizing threats in real time.

SIEM solutions enable centralized compliance auditing and reporting for the entire enterprise infrastructure. Advanced automation streamlines the collection and analysis of system logs and security events to reduce internal resource utilization while meeting rigorous compliance reporting standards.

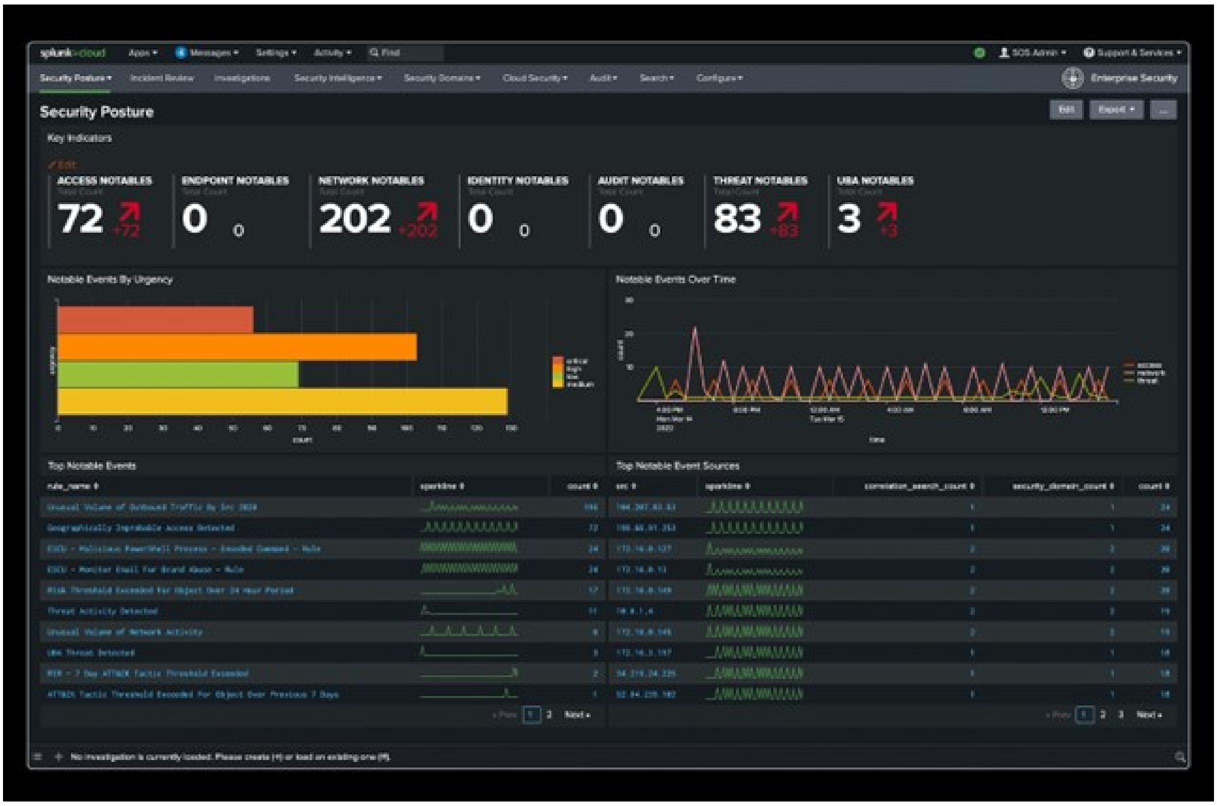
2.16 AI-powered automation

Today's next-generation SIEM solutions integrate with powerful security orchestration, automation, and response (SOAR) systems, saving time and resources for IT teams managing corporate security. SIEM solutions are ideal for conducting cyber forensics investigations when a security incident occurs. SIEM solutions allow organizations to efficiently collect and analyze log data from all digital assets in one place.

This gives them the ability to recreate past incidents or analyze more recent ones to investigate suspicious activity and implement more effective security processes.

With the growing popularity of remote workforces, SaaS applications, and bring-your-own-device (BYOD) policies, organizations need the necessary visibility to mitigate risks from outside the traditional network perimeter.

SIEM solutions track the network activity of all users, devices, and applications, significantly improving transparency across the entire infrastructure and detecting threats regardless of where digital assets and services are accessed.



Security Posture

Key Indicators

✓ Edit

ACCESS NOTABLES	ENDPOINT NOTABLES	NETWORK NOTABLES	IDENTITY NOTABLES	AUDIT NOTABLES	THREAT NOTABLES	UBA NOTABLES
Total Count	Total Count	Total Count	Total Count	Total Count	Total Count	Total Count
72 ↗ +72	0 0	202 ↗ +202	0 0	0 0	83 ↗ +83	3 ↗ +3

Top Notable Events

rule_name ⇕	sparkline ⇕	count ⇕
Unusual Volume of Outbound Traffic By Src 2020		116
Geographically Improbable Access Detected		72
ESCU - Malicious PowerShell Process - Encoded Command - Rule		24
ESCU - Monitor Email For Brand Abuse - Rule		24
Risk Threshold Exceeded For Object Over 24 Hour Period		17
Threat Activity Detected		11
Unusual Volume of Network Activity		8
UBA Threat Detected		5
RIR - 7 Day ATT&CK Tactic Threshold Exceeded		2
ATT&CK Tactic Threshold Exceeded For Object Over Previous 7 Days		1

« Prev 1 2 Next »

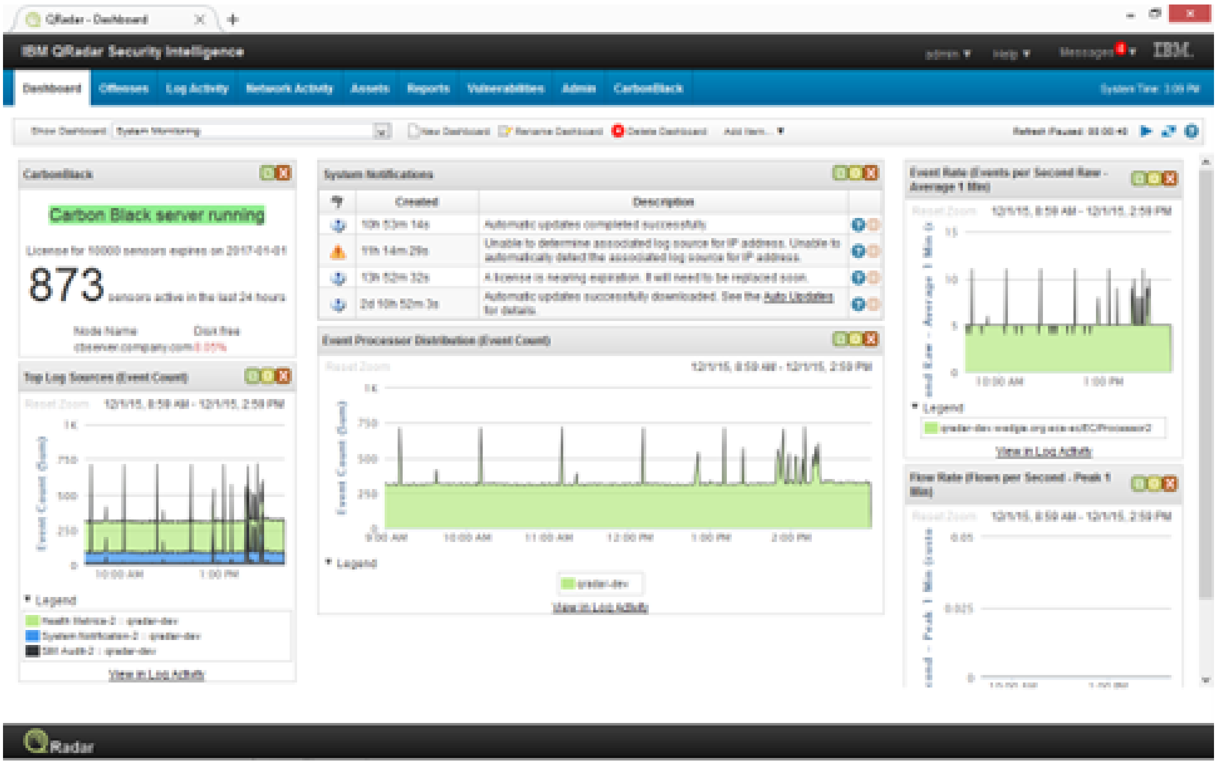
Top Notable Event Sources

src ⇕	sparkline ⇕	correlation_search_count ⇕	security_domain_count ⇕	count ⇕
164.207.83.83		1	1	24
198.46.91.253		1	1	24
172.16.0.127		2	2	20
172.16.0.13		2	2	20
172.16.0.148		2	2	20
10.0.1.6		2	2	19
172.16.0.145		2	2	19
172.16.3.197		1	1	18
54.219.24.229		1	1	18
52.84.226.182		1	1	18

« Prev 1 2 3 Next »

No investigation is currently loaded. Please create (+) or load an existing one (≡).

Figure 2: SIEM example: Splunk


QRadar - Dashboard
IBM QRadar Security Intelligence
admin
Help
Messages
IBM
Dashboard
Offenses
Log Activity
Network Activity
Assets
Reports
Vulnerabilities
Admin
CarbonBlack
System Time: 3:09 PM
Show Dashboard: System Monitoring
New Dashboard
Rename Dashboard
Delete Dashboard
Add Item...
Refresh Paused: 00:00:48
CarbonBlack
Carbon Black server running
License for 10000 sensors expires on 2017-01-01
873 sensors active in the last 24 hours
Node Name
Disk free
Top Log Sources (Event Count)
12/1/15, 8:59 AM - 12/1/15, 2:59 PM
Event Count (Sum)
1K
750
500
250
0
10:00 AM
1:00 PM
Legend
Health Metrics-2 :: qradar-dev
System Notification-2 :: qradar-dev
SIM Audit-2 :: qradar-dev
View in Log Activity
System Notifications
Event Processor Distribution (Event Count)
12/1/15, 8:59 AM - 12/1/15, 2:59 PM
Event Count (Sum)
1K
750
500
250
0
9:00 AM
10:00 AM
11:00 AM
12:00 PM
1:00 PM
2:00 PM
Legend
qradar-dev
View in Log Activity
Event Rate (Events per Second Raw - Average 1 Min)
12/1/15, 8:59 AM - 12/1/15, 2:59 PM
15
10
5
0
10:00 AM
1:00 PM
Legend
View in Log Activity
Flow Rate (Flows per Second - Peak 1 Min)
12/1/15, 8:59 AM - 12/1/15, 2:59 PM
0.05
0.025
0
QRadar

	Created	Description
	10h 53m 14s	Automatic updates completed successfully.
	11h 14m 29s	Unable to determine associated log source for IP address. Unable to automatically detect the associated log source for IP address.
	13h 52m 32s	A license is nearing expiration. It will need to be replaced soon.
	2d 10h 52m 3s	Automatic updates successfully downloaded. See the Auto Updates for details.

Figure 3: SIEM example: QRadar

At the core of the SIEM architecture are five main modules:

1. **Event Generation**:
   - It collects raw data from a wide range of devices and applications, such as operating systems, firewalls, web servers, and routers;
   - They record specific events such as system logins, file changes, or intrusion attempts:
     - **Sensors (event-based)**: they record specific events such as system logins, file changes, or intrusion attempts;
     - **Poller (state-based)**: they periodically check the status of systems and services to detect anomalies or malfunctions.
2. **Events Collection**:
   - It acts as a data centralizer, bringing together events generated by different sensors and pollers.
   - There are two main methods of collection:
     - **Agent-based**: agent software is installed on source devices to filter, aggregate, and normalize data before sending it to the SIEM
     - **Agentless**: the SIEM connects directly to source devices to retrieve data, without the need for agents.
3. **Recording and Storage**:
   - Store collected data in a secure and organized manner, ensuring its integrity and availability for future analysis;
   - Common storage formats include relational databases, text files, and binary files.

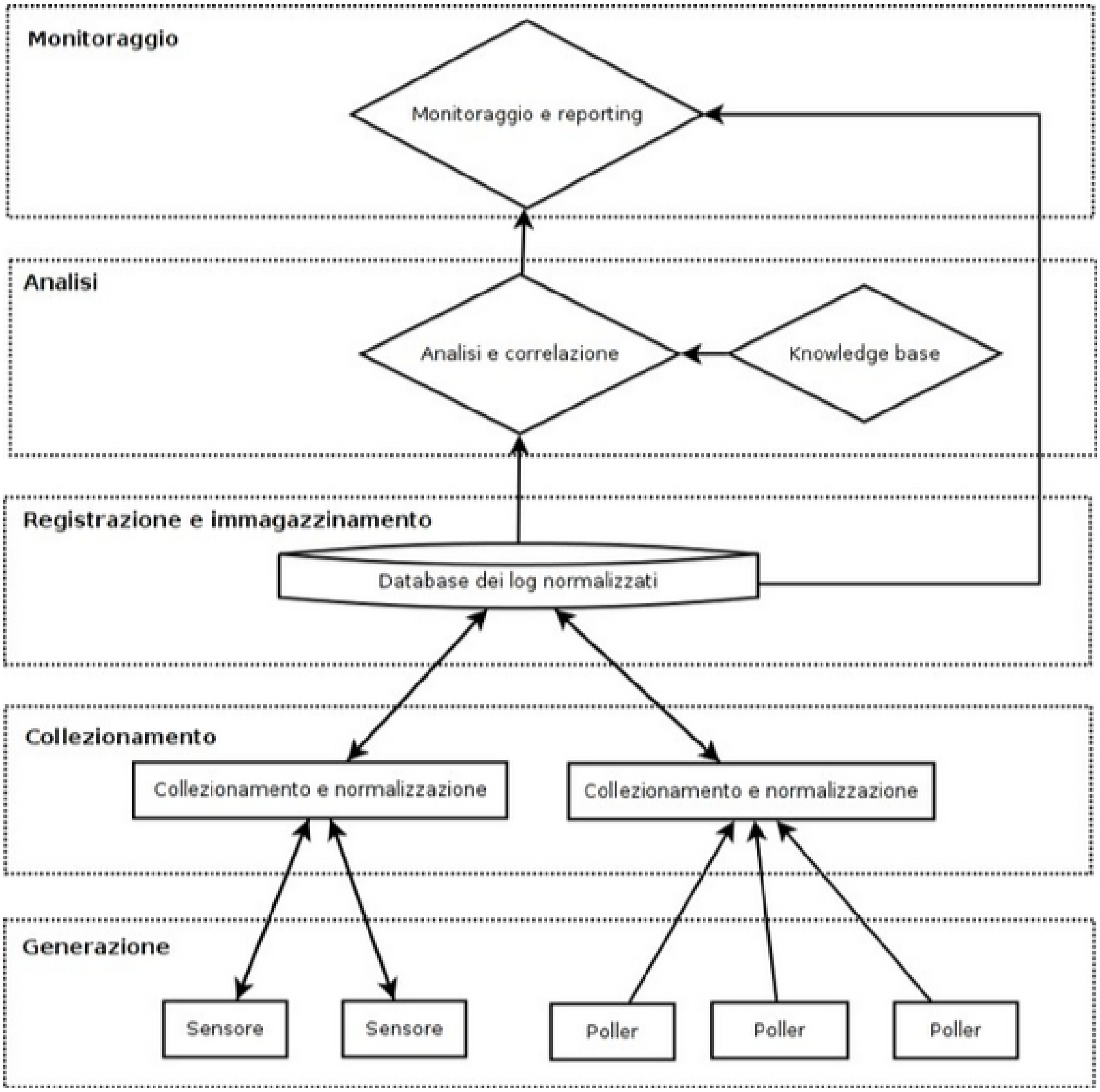
- To ensure data integrity, SIEMs employ techniques such as digests and retention on read-only devices.

4. **Analysis**:
    - The beating heart of the SIEM, where data is transformed into actionable security insights.
    - It uses two key components:
        - **Rule Engine**: apply predefined or custom rules to identify suspicious or anomalous events that may indicate a threat.
        - **Correlation Engine**: correlate events from multiple sources to reconstruct complex scenarios and identify potential attacks.
    - The Knowledge Base, which contains information about vulnerabilities, assets, and security policies, supports analysis by providing context and prioritization of events.
5. **Monitoring and Reporting**:
    - It provides security operators and analysts with the interfaces they need to interact with data and make informed decisions;
    - Typical interfaces include:
        - Real-time monitoring console to view incoming events and filter those of interest;
        - Incident Management to create and manage incident tickets, coordinating response and resolution activities;
        - Statistical analysis to generate reports on security trends, recurring attacks, and the most affected systems;
        - Risk assessment to provide an overview of the overall security posture, vulnerabilities, and potential threats.

2.17 Module integration:

The SIEM architecture is not a rigid sequence, but rather a synergistic system in which the modules interact and exchange information:

- Data generated by sensors flows through collection, recording, and analysis modules, fueling the threat identification process;
- The information extracted from the analyses is used to update the Knowledge Base, improving the accuracy of future analyses;
- Operators interact with monitoring and reporting modules to make data-driven decisions and initiate incident response actions.



Architettura di un SIEM Macro architettura di un SIEM

Figure 4: SIEM architecture

2.18 Open Source vs. Enterprise SIEM Solutions

Having completed the description of SIEM architecture, we can now proceed with a brief comparison of the main open source and enterprise solutions on the market. First, we need to distinguish between open source and enterprise solutions, keeping in mind that:

Using open source solutions certainly offers greater customization for rule creation, but at the same time, we lose speed and, unless you have an experienced team, quality.

For their part, enterprise solutions, in addition to support for use and any issues with the platform, also provide advanced integrated ML/UEBA capabilities, threat intelligence, playbooks, and often "detection content" (ready-made use cases). The Microsoft Sentinel product, for example, integrates XDR and AI for detection/graph analytics, aspects that in open source solutions must be implemented by the internal team.

Therefore, you can choose an open source solution if:

- you are an SME or startup with a limited licensing budget but with in-depth DevOps/SecOps skills;
- full control over data, architecture, and customization is preferred. This allows for flexibility in building customized pipelines, rules, and dashboards;
- you are willing to invest in highly qualified and technical personnel;

Instead, you can choose an enterprise solution if:

- it is necessary to minimize time-to-value, take SLAs into consideration and have official support;
- the SOC department needs ready-made content (detection rules, playbooks) and certified integrations without having to develop everything internally;
- you are in a regulated environment and require certified compliance and prompt reporting;

With these premises in mind, we can proceed briefly with a real comparison between open source and enterprise solutions:

- **ELK Stack (Elastic Stack)**: is an acronym for Elasticsearch, Logstash, and Kibana—the three original components developed by Elastic.co for collecting, indexing, analyzing, and visualizing log data. It is the heart of many open source SOCs, as it is used as:
  - open source SIEM to collect, correlate and visualize security logs (firewall, endpoint, applications);
  - analysis and threat hunting engine. Then its NoSQL database and search engine are used for storing and querying billions of logs at high speed.
- **Elasticsearch**: the NoSQL database and search engine for storing and querying billions of logs at high speed.
- **Logstash (o Beats)**: Ingestion pipeline. Beats (Filebeat, Winlogbeat) are lightweight agents that send logs; Logstash receives them, normalizes them (parses them), enriches them, and sends them to Elasticsearch.
- **Kibana**: the visualization (dashboard, graphs, discovery) and management (alerting, security rules) interface.
- **Wazuh**: often used on top of ELK, it transforms it from a simple log aggregator to a true SIEM/HIDS (Host-based Intrusion Detection System). It provides endpoint agents (which replace or complement Beats), a manager that analyzes logs with predefined rules (rootkit detection, file integrity monitoring, vulnerability scanning), and a customized Kibana interface (the Wazuh plugin).
- **Osquery**: it's an endpoint visibility framework (developed by Meta). It allows you to query an operating system (Windows, Linux, macOS) using SQL queries. It's excellent for granular threat hunting ("Show me all processes listening on non-standard ports") and can feed its logs to ELK or Wazuh.
- **Splunk / QRadar / Sentinel**: designed for large enterprise SOCs: ingestion, indexing, retention, and related features are already optimized; they offer both on-prem and cloud capabilities (Sentinel is cloud-native). They scale with SLAs, but come with ingest/retention costs.

With previous SIEM solutions, it's also possible to introduce SOAR solutions (the concept of which will be explored in more detail in the next section) for automation and incident response.

In open source contexts:

- it is possible to integrate playbooks with open projects (e.g., StackStorm, custom scripts) or use Wazuh response actions. However, this solution requires development and governance;
- licenses are free, but there are significant costs for infrastructure, storage, networking, backup, tuning, and disaster recovery. Costs can also increase rapidly with volume, and a highly specialized team is required to build, operate, maintain, and scale the tool.

In enterprise contexts:

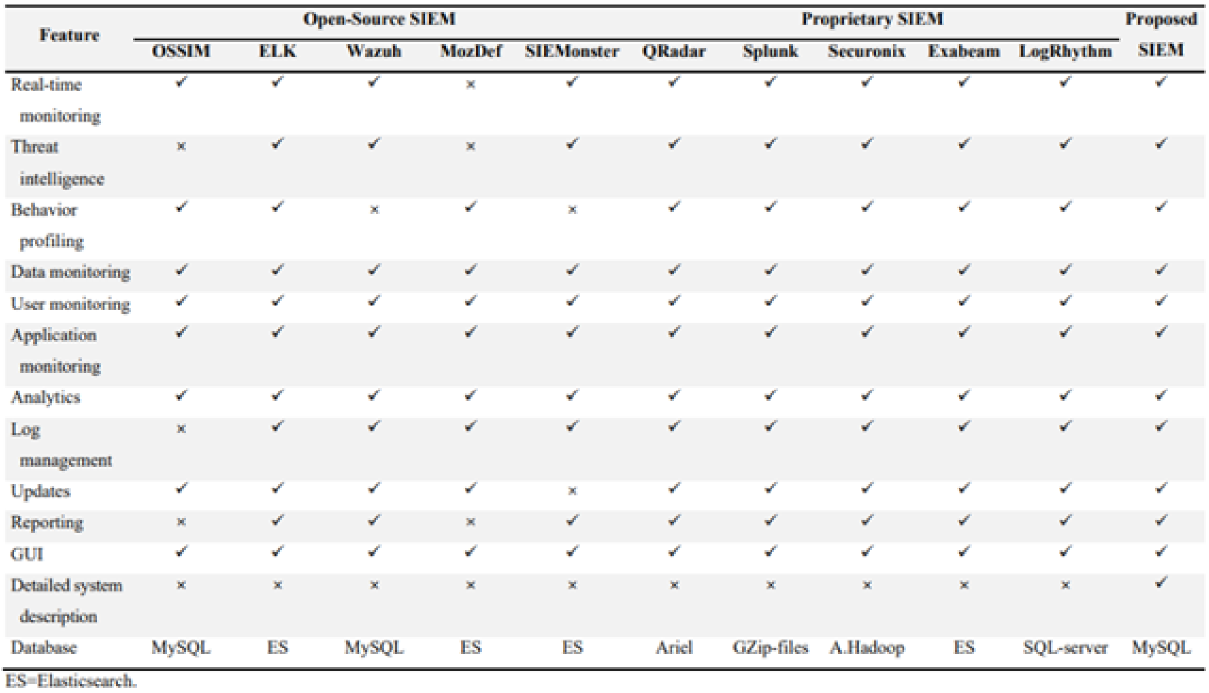
- integrated SOAR solutions or partner products (Splunk SOAR, Sentinel automation rules) are often offered with low-code editors, connectors and ready-made case management
- there are high costs for licensing, ingest/retention, and additional modules, but at the same time There is dedicated support with related SLAs that must be met and ready-made features that reduce operating costs. Overall, for large volumes, enterprise solutions can often be more cost-effective, and the analyst can spend time on analysis (Tier 1, 2, and 3) and threat hunting, not on maintaining the SIEM infrastructure. Updates, patching, and scaling are managed by the vendor (especially in SaaS models like Sentinel).

Now let's sum up pros and cons about open source and enterprise solutions:

- **ELK (Elastic Stack)**
  - Pros: Powerful search, Kibana visualization, Beats ecosystem, great flexibility.
  - Cons: Complex cluster management; infrastructure and storage costs; some advanced features are paid (Elastic license).
- **Wazuh**
  - Pros: Ready-to-use open source SIEM/XDR, agent management, rule set and compliance templates; integration with Elastic/Kibana or OpenSearch.
  - Cons: Depends on underlying infrastructure; scaling and tuning required; some enterprise features require effort.
- **Osquery**
  - Pros: Excellent for host visibility, flexible querying, lightweight, ideal for forensics and auditing.
  - Cons: Not a SIEM; requires pipeline and storage; fleet management/updates and response require additional tools.
- **Splunk**
  - Pros: Mature user experience, powerful search, app/security content ready, SOAR (Splunk Phantom), enterprise support.
  - Cons: High licensing and ingest costs; lock-in; expensive planning for high volumes.
- **IBM QRadar**
  - Pros: Strong network analytics, incident prioritization, and integrated intelligence; proven enterprise solution.
  - Cons: Some strategic product shifts (partnerships/divestitures) may impact the roadmap; on-prem oriented for certain modules.
- **Microsoft Sentinel**
  - Pros: Cloud-native, deep integration with Microsoft/Defender stack, pay-as-you-go cost model, and now Data Lake for retention; strong AI/XDR convergence.

- Cons: Better experience in Microsoft environments; for heterogeneous environments, carefully evaluate connector and ingest costs.

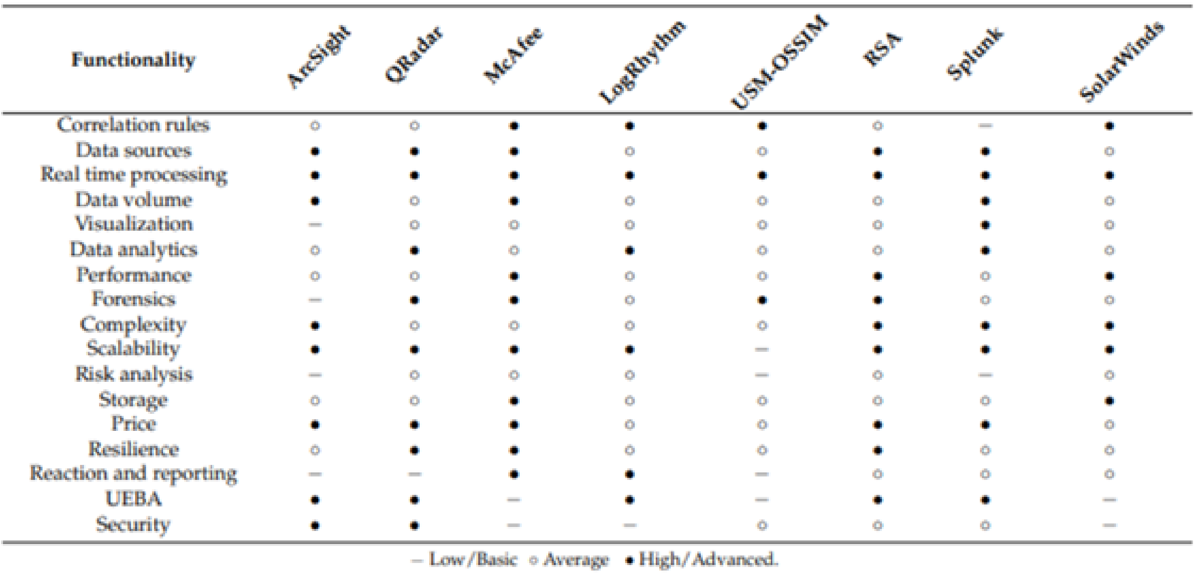
As previously mentioned, various SIEM solutions exist, some open source and some proprietary. When choosing a SIEM for a business environment, the first thing to consider is the budget you want to allocate for monitoring. Once this is understood, you can choose a solution (as briefly analyzed in the Ertuğrul Akbaş paper's) by considering aspects such as correlation rules, real-time processing, threat intelligence integration, data analytics, and user and entity behavior analytics (UEBA). The features advertised by vendors aren't always practical, so lastly, you must also consider your available IT assets.



Feature	Open-Source SIEM					Proprietary SIEM					Proposed
	OSSIM	ELK	Wazuh	MozDef	SIEMonster	QRadar	Splunk	Securonix	Exabeam	LogRhythm	SIEM
Real-time monitoring	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
Threat intelligence	×	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
Behavior profiling	✓	✓	×	✓	×	✓	✓	✓	✓	✓	✓
Data monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
User monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Application monitoring	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Analytics	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Log management	×	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Updates	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓
Reporting	×	✓	✓	×	✓	✓	✓	✓	✓	✓	✓
GUI	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Detailed system description	×	×	×	×	×	×	×	×	×	×	✓
Database	MySQL	ES	MySQL	ES	ES	Ariel	GZip-files	A.Hadoop	ES	SQL-server	MySQL

ES=Elasticsearch.

Figure 5: Open source SIEM vs Enterprise SIEM capabilities



Functionality	ArcSight	QRadar	McAfee	LogRhythm	USM-OSSIM	RSA	Splunk	SolarWinds
Correlation rules	○	○	•	•	•	○	–	•
Data sources	•	•	•	○	○	•	•	○
Real time processing	•	•	•	•	•	•	•	•
Data volume	•	○	•	○	○	○	•	○
Visualization	–	○	○	○	○	○	•	○
Data analytics	○	•	○	•	○	○	•	○
Performance	○	○	•	○	○	•	○	•
Forensics	–	•	•	○	•	•	○	○
Complexity	•	○	○	○	○	•	•	•
Scalability	•	•	•	•	–	•	•	•
Risk analysis	–	○	○	○	–	○	–	○
Storage	○	○	•	○	○	○	○	•
Price	•	•	•	○	○	•	•	○
Resilience	○	•	•	○	○	•	○	○
Reaction and reporting	–	–	•	•	–	○	○	○
UEBA	•	•	–	•	–	•	•	–
Security	•	•	–	–	○	○	○	–

– Low/Basic ○ Average • High/Advanced.

Figure 6: SIEM capabilities

2.19 What is SOAR (Security Orchestration, Automation and Response)?

Security, Automation, Orchestration, and Response (SOAR) technology helps coordinate, execute, and automate the activities of multiple people and tools within a single platform. This enables organizations not only to respond rapidly to cyberattacks but also to observe, understand, and prevent future incidents, thus improving their overall security posture.

A comprehensive SOAR product (the first signs of which date back to 2015), according to Gartner's definition, is designed to encompass three primary software capabilities: threat and vulnerability management, security incident response, and security operations automation.

Threat and vulnerability management (orchestration) covers technologies that help remediate cyber threats, while security operations automation (automation) refers to technologies that enable automation and orchestration within operations. SOARs capture alert data, which in turn trigger playbooks that automate/orchestrate response workflows or activities. Then, using a combination of human and machine learning, organizations can analyze the diverse data to understand and prioritize automated incident response actions, creating a more efficient and effective approach to managing cybersecurity and improving security operations.

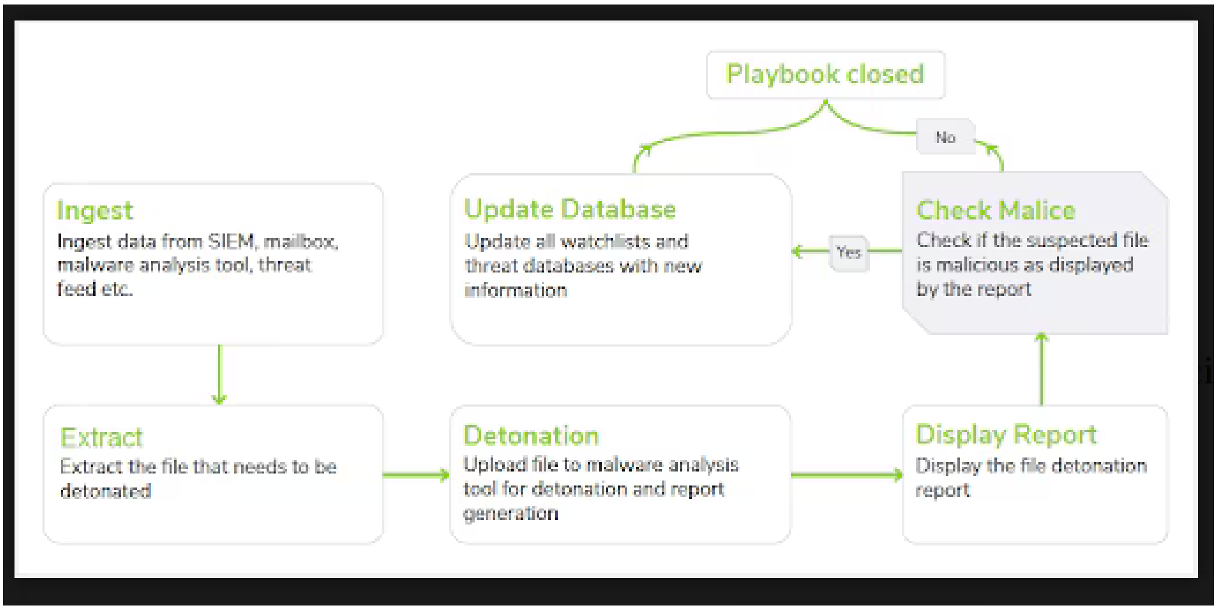

Playbook closed
No
Ingest
Ingest data from SIEM, mailbox, malware analysis tool, threat feed etc.
Update Database
Update all watchlists and threat databases with new information
Yes
Check Malice
Check if the suspected file is malicious as displayed by the report
Extract
Extract the file that needs to be detonated
Detonation
Upload file to malware analysis tool for detonation and report generation
Display Report
Display the file detonation report

Figure 7: Flowchart SOAR

2.20 SOAR vs. SIEM

Many people think of SOAR and SIEM as similar products because they both detect security issues and collect data about the nature of the problem. They also provide notifications that security personnel can use to resolve issues. However, there are significant differences between the two products. SOAR security, however, adds automation and response to the investigation process, using playbooks or automated workflows and artificial intelligence (AI) to learn patterned behaviors, allowing it to predict similar threats before they occur. Because SOAR products, such as Cortex XSOAR, typically capture alerts from sources not covered by SIEMs (for example, vulnerability scan results and security alerts from the cloud and IoT devices), it's easier to deduplicate alerts. Indeed, this is a typical use case for integrating SOAR and SIEM. This reduces the amount of time needed to manually manage alerts, simplifying threat detection and management for IT security personnel.

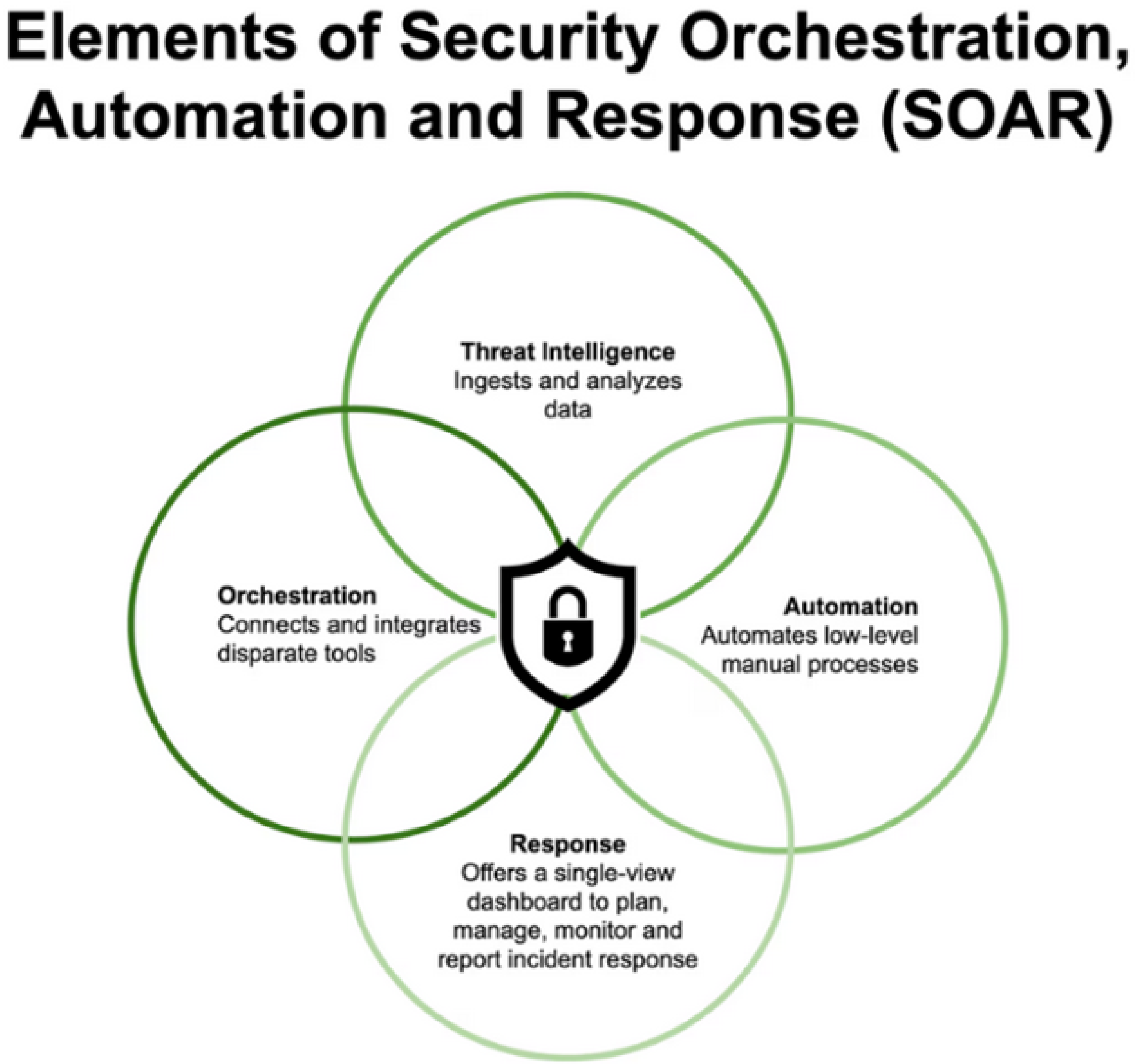

Elements of Security Orchestration, Automation and Response (SOAR)
Threat Intelligence
Ingests and analyzes data
Orchestration
Connects and integrates disparate tools
Automation
Automates low-level manual processes
Response
Offers a single-view dashboard to plan, manage, monitor and report incident response

Figure 8: SOAR

2.21 What are security orchestration and automation?

Security automation is the execution of machine-based security actions that can detect, investigate, and remediate cyber threats without requiring manual intervention from staff. It takes care of much of the repetitive work for the SOC team, freeing them from the hassle of manually addressing every incoming alert. Security automation can:

- Detect threats in your environment;
- Assess potential threats;
- Determining whether to take action regarding an incident;
- Contain and resolve the problem.

Security orchestration is the machine-based coordination of a series of independent security activities, encompassing incident investigation, response, and resolution within a single, complex infrastructure. It ensures that all tools, whether security-related or not, work together, automating product and workflow tasks or manually alerting agents to significant incidents that require further attention.

Security orchestration can:

- **Provide more context on security incidents**. A security orchestration tool aggregates data from multiple sources to provide more detailed information. This provides a more comprehensive view of the entire environment.
- **Enable more in-depth and meaningful investigations**. Security analysts can stop managing alerts and start investigating why incidents occur. Furthermore, security orchestration tools typically feature highly interactive and intuitive dashboards, graphs, and timelines, and these visuals can be very helpful during investigations.

2.22 What is the difference between automation and orchestration?

While security orchestration and automation are terms often used interchangeably, the two platforms have very different roles:

- **Security automation**: reduces the time it takes to detect and respond to repetitive incidents and false positives, so alerts don't go unrecognized for long periods of time.
  - Analysts have more time to focus on strategic activities, such as research and investigations.
  - Each automated playbook addresses a known scenario with a pre-established course of action.
- **The orchestration of security**: it enables easy information sharing, enabling different tools to respond to incidents as a group, even when data is spread across a large network and multiple systems or devices. Security orchestration uses multiple automated tasks to execute a complete, complex process or workflow.

The goal of automation is to simplify and streamline security operations by addressing a series of individual tasks, while orchestration connects all security tools so they feed into each other, creating a fast and efficient workflow process from start to finish. They work best when combined, and security teams can maximize efficiency and productivity by adopting both.

2.23 The importance of having and using SOAR

SOAR is important for businesses and organizations because it minimizes the impact of security incidents of all types and maximizes the value of existing security investments, reducing the overall risk of legal liability and business downtime. SOAR helps companies address and overcome security challenges, enabling them to:

- **Unify existing security systems and centralize data collection** to gain full visibility, greatly improving the company's safety, operational efficiency, and productivity;
- **Automate repetitive manual tasks** and manage all aspects of the security incident lifecycle, increasing analyst productivity and giving them more time to focus on improving security rather than performing manual tasks;
- **Define incident analysis and response procedures** and leverage security playbooks to prioritize, standardize, and scale response processes in a consistent, transparent, and documented manner;
- **Respond to incidents faster**, analysts can quickly and accurately identify and assign incident severity levels to security alerts, reducing alerts and alleviating alert fatigue;
- **Simplify processes and operations** to better identify and manage potential vulnerabilities both proactively and reactively;

- **Support real-time collaboration** and unstructured investigations by routing each security incident to the analyst best suited to respond, providing features that support easy communication and monitoring between teams and their members.

As mentioned, SOAR solutions (but also XDR) can help the analyst perform a first level of complete and exhaustive investigation so that the incident can be closed or escalated to the L2 analyst as quickly as possible. Therefore:

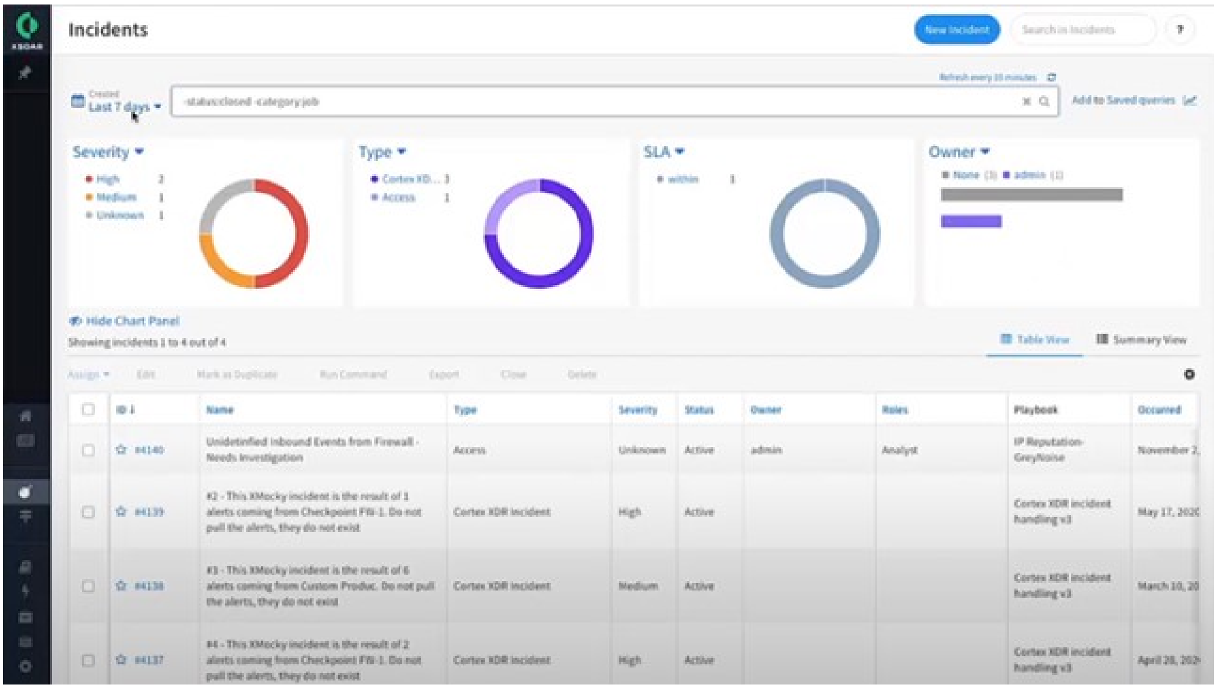
- Using XDR: this technology doesn't just collect logs, but natively correlates events from different security layers (endpoint, network, cloud, email). Instead of generating 10 separate alerts (one from the firewall, one from the EDR, one from the mail server), XDR consolidates them into a single, contextualized incident. This significantly reduces the volume of alerts the analyst has to examine and provides a "complete" view of the scenario right from the start;

- Using SOAR, however, automated triage is possible: SOAR playbooks can independently handle the most common and repetitive alerts. For example, a phishing email alert can be handled entirely by SOAR: it extracts URLs and attachments, analyzes them in a sandbox, and, if malicious, removes them from users' inboxes, closing the ticket without any human intervention.

2.24 SOAR Use Cases

The following table provides examples of common use cases for SOAR:

Use case	Description
Manage security alerts	Phishing Enrichment and Response: Capture potential phishing emails; activate a playbook; automate and execute repeatable tasks, such as triaging and engaging implicated users; extract and verify indicators; identify false positives; and activate the SOC for a standardized response at scale. Endpoint malware infection: Acquire threat feed data from endpoint tools; enrich this data; cross-reference retrieved files and/or hashes with a security information and event management (SIEM) solution; notify analysts; clean endpoints; update the endpoint tool database. Failed user logins: After a predefined number of failed user login attempts, evaluate whether the attempt is genuine or malicious by activating a playbook, engaging users, analyzing their responses, expiring passwords, and closing the playbook. Logins from unusual locations: Identify potentially malicious virtual private network (VPN) login attempts by checking for the presence of VPNs and cloud security brokers (CASBs); match IPs; confirm a breach with the user; activate a block; close the playbook.
Gestire le operazioni di sicurezza	Secure Sockets Layer (SSL) certificate management: Check endpoints to determine which SSL certificates are expired or about to expire; alert users; re-check status a few days later; escalate an issue to the appropriate personnel; close the playbook. Endpoint diagnostics and launch: Check connectivity, including agent connectivity; enrich context; open a ticket; launch agents; close the playbook. Vulnerability management: Gather vulnerability and asset information; enrich endpoint and Common Vulnerabilities and Exposures (CVE) data; query for vulnerability context; calculate severity; transfer control to security analysts for remediation and investigation; close the playbook.
Search for threats and respond to incidents	Indicators of Compromise (IOC) Search: Collect and extract IOCs from attached files; search for IOCs in threat intelligence tools; update databases; close the playbook. Malware Analysis: Collect data from multiple sources; extract and deactivate malicious files; create and view a report; check for malicious events; update databases; close the playbook. Cloud-aware incident response: Consuming data from cloud-centric threat detection and event logging tools; unifying processes across cloud and on-premises security infrastructures; correlating with a SIEM; extracting and enriching indicators; verifying for malicious events; handing over control to analysts for information review; updating the database; and closing the playbook.
Automate data enrichment	IOC enrichment: Acquire data from various sources; extract indicators to deactivate; enrich URLs, IPs, and hashes; check for malicious events; update the database; invite analysts to review and investigate the information; close the playbook. Assign incident severity: Check other products for vulnerability scores and see if existing indicators have been assigned a score; assign severity; check usernames and endpoints to see if they are included in a critical list; assign critical severity; close an incident.



Created Last 7 days

-status:closed -category:job

Showing incidents 1 to 4 out of 4

	ID	Name	Type	Severity	Status	Owner	Roles	Playbook	Occurred
□	☆ #4140	Unidetinfied Inbound Events from Firewall - Needs Investigation	Access	Unknown	Active	admin	Analyst	IP Reputation-GreyNoise	November 2,
□	☆ #4139	#2 - This XMocky incident is the result of 1 alerts coming from Checkpoint FW-1. Do not pull the alerts, they do not exist	Cortex XDR incident	High	Active			Cortex XDR incident handling v3	May 17, 202
□	☆ #4138	#3 - This XMocky incident is the result of 6 alerts coming from Custom Produc. Do not pull the alerts, they do not exist	Cortex XDR incident	Medium	Active			Cortex XDR incident handling v3	March 10, 20
□	☆ #4137	#4 - This XMocky incident is the result of 2 alerts coming from Checkpoint FW-1. Do not pull the alerts, they do not exist	Cortex XDR incident	High	Active			Cortex XDR incident handling v3	April 28, 202

Figure 9: XSOAR

2.25 Differences EDR, XDR, SIEM, SOAR

Now that we have described in detail what EDR, XDR, SIEM and SOAR are, the differences between the previous terms are listed below to clarify any possible doubts:

2.26 XDR vs. SIEM

XDR and SIEM technologies offer different but complementary capabilities. SIEMs aggregate large amounts of data and identify security threats and anomalous behavior. Since they can ingest data from virtually any source, they offer high visibility. They also simplify log management, event and incident management, and compliance reporting. SIEMs can use security orchestration, automation, and response (SOAR) to respond to cyber threats, but they require extensive customization and do not offer automatic attack termination capabilities.

SIEM focuses primarily on log analysis, while XDR extends detection to multiple security layers (endpoint, network, cloud). Furthermore, SIEM generates alerts that require human analysis, while XDR offers an automated or semi-automated response; thus, the former is an analysis tool, while the latter is a detection and response tool.

By combining XDR with SIEM, companies gain comprehensive detection, analysis, and automated response capabilities across every level of the digital estate, as well as a foundation for introducing generative AI capabilities. Companies also gain greater visibility into cyber kill chains, a framework, also known as a cyberattack chain, that outlines the stages of common cybercrimes.

2.27 How is XDR different from SOAR?

Security Orchestration & Automated Response (SOAR) platforms are used by mature security operations teams to build and execute multi-stage playbooks that automate actions through an ecosystem of security solutions connected via APIs. For example, they can automatically trigger a scan of an infected endpoint after receiving an alert from an EDR system, send notifications to security teams, block network traffic, and update the vulnerability management system.

In contrast, XDR focuses on threat detection and response across multiple security domains (endpoint, network, mail, cloud, etc.). XDR provides greater visibility and context into threats; incidents that would otherwise have gone unresolved emerge with a higher level of awareness, enabling security teams to remediate and mitigate any further impact and minimize the scope of the attack.

XDR integrates disparate security controls to provide automated (and thus, in a sense, XDR can be considered a simpler and more intuitive version of SOAR) or one-click response actions within enterprise security, such as disabling user access, forcing multi-factor authentication in the event of suspected account compromise, blocking incoming domains and file hashes, and more, all via custom rules written by the user or logic embedded in the prescriptive response engine.

2.28 Difference between EDR and XDR

While EDR collects and correlates activity across multiple endpoints, XDR extends the scope of detection beyond endpoints to provide detection, analysis, and response across endpoints, networks, servers, cloud workloads, SIEMs, and more. This provides a unified view across multiple tools and attack vectors. This improved visibility provides complete threat context to optimize triage, investigation, and rapid remediation efforts.

XDR automatically collects and correlates data across multiple security vectors, facilitating faster threat detection so security analysts can respond quickly before threats spread throughout the network. With a single pool of raw data encompassing information from across the entire ecosystem, XDR enables faster, more thorough, and more effective threat detection and response than EDR by collecting and comparing data from a broader range of sources. This comprehensive visibility leads to numerous benefits, including:

- **Reduction of mean time to detection (MTTD)** through correlation between data sources;
- **Reduction of Mean Time to Investigation (MTTI)** speeding up triage and reducing investigation time and scope;
- **Reduction of Mean Time To Responde (MTTR)** enabling simple, fast and relevant automation;
- **Improved visibility** across the entire safety area.

Additionally, thanks to artificial intelligence and automation, XDR helps reduce the burden of manual work for security analysts.

EDR systems are designed to monitor and protect individual endpoint devices at scale. EDR capabilities enable security teams to quickly identify and respond to suspicious behavior and malicious activity at the endpoint level.

2.29 Advantages of XDR over EDR

Organizations can implement an EDR or XDR solution to improve visibility, detect cyber threats more efficiently, and respond more quickly. However, because XDR systems can connect to other security environments beyond endpoints, XDR offers several significant advantages over EDR, including:

- improved visibility across different layers of the security stack;
- advanced cyber threat detection across multiple security domains;
- simplified correlation and analysis of unexpected events;
- better scalability and adaptability;
- protection against advanced cyber attacks, such as ransomware.

2.30 Choice of EDR, XDR, SIEM, SOAR

Digital security needs typically vary from company to company. When determining which cyber threat detection and response system is the right choice, it is important to:

- assess the organization's security needs and goals;
- evaluate any relevant budget constraints;
- consider the resources and skills needed to successfully implement EDR or XDR;
- analyze the potential impact of EDR or XDR on existing security infrastructure.

Below is a quick summary of the key differences between SIEM, EDR, SOAR, and XDR:

- **SIEM**
  - Focus: centralization, log management, compliance, rule-based correlation.
  - Provides: long-term storage, search, alerting.
- **EDR**
  - Focus: detailed endpoint telemetry, behavioral detection, response (isolate, kill).
  - Provides: process trees, file/registry/behavioral indicators, agent-driven remediation
- **SOAR**
  - Focus: orchestration of operational flows (playbooks), automation of actions, case management.
  - Provides: repeatable automations, tool-to-tool integration, operational reporting.
- **XDR**
  - Focus: Extended and native cross-layer detection and response; contextual correlation from multiple telemetries.
  - Provides: unified vision, cross-domain insights, and often integrated playbooks; implementations vary widely in their level of integration and openness.

Now that we have the knowledge necessary to distinguish EDR, XDR, SIEM, and SOAR, we can briefly discuss, at the risk of sounding a bit repetitive, their integration and evolution in SOC contexts.

In recent years, Security Operations Centers have undergone a profound transformation, moving from models based on separate tools to an integrated and intelligent detection and response approach. SIEM, XDR, and SOAR technologies, initially designed with distinct functions, are now merging into a single operational ecosystem, supported by artificial intelligence, cloud, and automation.

2.30.1 SIEM: From Log Collector to Analytics Platform

Modern SIEM platforms are no longer simple log collection and correlation systems. They have evolved into full-fledged behavioral analytics engines, capable of:

- apply User and Entity Behavior Analytics (UEBA) algorithms;
- automatically correlate events from multiple domains;
- integrate with threat intelligence and automation (SOAR) modules;
- leverage cloud-native architectures for scalability and real-time analytics.

In the modern SOC, SIEM is the heart of visibility and compliance, providing the database for cross-domain detection and long-term archiving.

2.30.2 XDR: Unified Detection Intelligence

As previously mentioned, XDR was created to overcome the integration limitations between EDR, SIEM, and other tools. It natively combines telemetry from endpoints, networks, cloud, identities, and email, automatically correlating signals and generating contextualized alerts. In the current SOC:

- acts as a detection and prioritization engine;
- it integrates directly with SIEM for historical analysis and with SOAR to trigger automated response.

2.30.3 SOAR: The Automation Orchestrator

SOAR, initially conceived as a separate platform, is now the SOC's operational automation engine. Thanks to preconfigured playbooks, API integrations, and semi-automated workflows, it enables:

- automatic execution of containment and remediation actions (e.g. host isolation, IP blocking);
- contextual enrichment of alerts;
- the coordination of activities between L1–L3 analysts.

Thanks to its capabilities, many vendors are now integrating SOAR modules directly into SIEM or XDR platforms (e.g. Microsoft Sentinel, Cortex XSOAR, Splunk SOAR).

2.30.4 An integrated ecosystem

In the modern SOC, SIEM, XDR and SOAR no longer operate as separate tools but as components of a converged security platform, where:

- XDR detects and correlates threats in real time;
- SIEM retains and analyzes data long-term, enabling hunting and compliance;
- SOAR automates and orchestrates response and remediation actions.

This approach allows the SOC to move from a reactive to a proactive and predictive model, based on artificial intelligence and automation.

Although the solutions just described represent a huge step forward compared to the scenario of 20 years ago, there are still some problems related to the use of these tools:

- **SIEM**:
  1. Alert overload and false positives that burden L1 and reduce the operational effectiveness of the SOC;
  2. Data costs and scaling: massive ingestion (cloud, applications, IoT) increases storage/ingest costs and complicates retention/archiving;
  3. Complex rules maintenance: continuous tuning required to avoid regressions. It requires time and expertise.
- **XDR**:
  1. Vendor lock-in and variable scope: many XDR solutions are "closed ecosystems" that offer deep integration only with vendor components, complicating multi-vendor policies;
  2. Complexity in integrating with SIEM and existing processes: risks of alert duplication/operational contradictions.
- **SOAR**:

1. Process maturity and poorly designed playbooks: automation reproduces faulty processes; without robust processes, SOAR can make errors worse (e.g., automatic actions on false positives);
2. Playbook maintenance: many integrations and conditions change frequently; playbooks age and require ongoing auditing;
3. Over-automation / Incorrect trust model: uncalibrated levels of automation can lead to improper shutdowns, business disruptions, or unnecessary escalations.

To try to counteract and improve these critical issues, the market is trying to evolve in the following way:

- **For SIEM**:
  - Cloud-native SIEM & security data pipelines: moving toward data-lake architectures that separate ingestion, indexing, and analytics (reduces costs and improves scalability);
  - Focused ML/UEBA adoption to reduce false positives.
- **For XDR**:
  - "Open XDR" models and focus on telemetry standards: vendors and initiatives for open connectors and interoperability emerge (to reduce vendor lock-in).
- **For SOAR**:
  - Low-code / no-code playbook editors: facilitate creation and maintenance by analysts; reduces barriers;
  - Adopting AI for enrichment and decision support: as previously mentioned, fully automated solutions risk causing unnecessary bottlenecks in service delivery and unnecessary escalations. Therefore, AI is limited to providing suggestions and context so that analysts can make the right decisions and mitigate actions.

2.31 Indicators of Compromise

An indicator of compromise (IOC) is evidence that someone may have breached an organization's network or endpoint. This forensic data not only indicates a potential threat, but also signals that an attack has already occurred, such as malware, compromised credentials, or data exfiltration. Security professionals search for IOCs in event logs, XDRs, and the SIEM. During an attack, the team uses IOCs to eliminate the threat and mitigate the damage. After recovery, IOCs allow an organization to better understand what happened, so the organization's security team can strengthen security and reduce the risk of another similar incident.

2.32 Examples of IOCs

In IOC security, IT monitors the environment for the following signs of an ongoing attack:

2.32.1 Network traffic anomalies

In most organizations, there are consistent patterns in network traffic entering and leaving the digital environment. When conditions change, such as significantly increased data leaving the organization or activity originating from an unusual location on the network, it could be a sign of an attack.

2.32.2 Unusual login attempts

Like network traffic, people's work habits are predictable. People typically log in from the same locations and at approximately the same times throughout the week. Security professionals can detect a compromised account by paying attention to logins (such as from unusual geographic locations, or from a country where an organization has no offices) or unusual activity (such as activity at night when the user would normally be working during business hours). It's also important to note multiple failed logins from the same account. Repeated failed login attempts may indicate that someone is attempting to access the organization using a stolen account.

2.32.3 Privileged account irregularities

Many malicious users, whether internal or external, are interested in accessing administrative accounts and acquiring sensitive data. Atypical behavior associated with these accounts, such as attempted privilege escalation, may be a sign of a breach.

2.32.4 Changes to system configurations

Malware is often programmed to make changes to system configurations, such as enabling remote access or disabling security software. By monitoring these unexpected configuration changes, security professionals can identify a breach before excessive damage occurs.

2.32.5 Unexpected software installations or updates

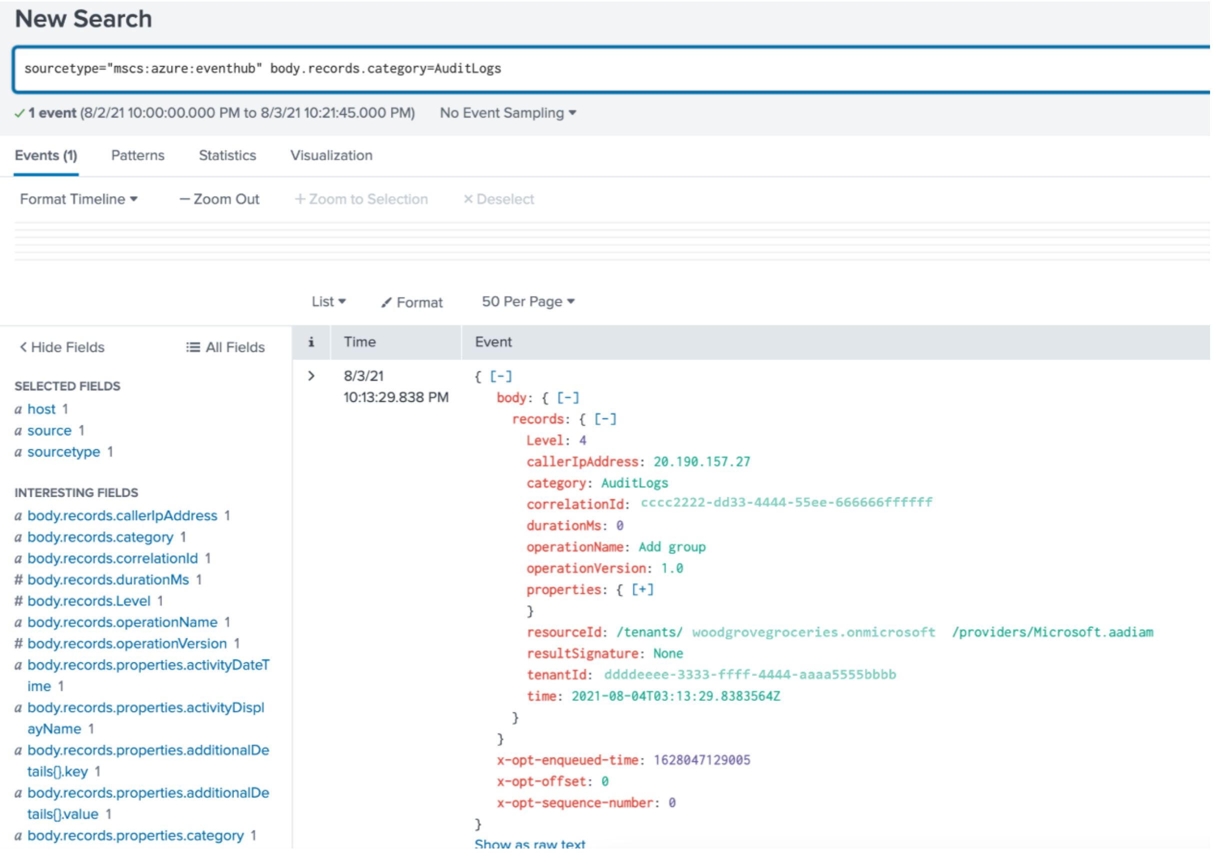
Many attacks begin with the installation of software, such as malware or ransomware, designed to make files inaccessible or allow attackers to gain network access. By monitoring unplanned software installations and updates, organizations can quickly intercept these IOCs.

2.32.6 Unusual Domain Name System Requests

Some attackers use an attack method called "command and control" (C&C). They install malware on an organization's server that establishes a connection to a server they own. They then send commands from the server to the infected computer to attempt to steal data or disrupt operations. Unusual Domain Name System (DNS) requests help IT detect these attacks.

2.33 How to identify IOCs

Signs of a digital attack are recorded in log files. As part of IOC cybersecurity, teams regularly monitor digital systems for suspicious activity. Modern SIEM and XDR solutions simplify this process through artificial intelligence and machine learning algorithms that establish a baseline for events that can normally occur in the organization and then alert the team to any anomalies. It's also important to involve employees outside of security who might receive suspicious emails or accidentally download an infected file. A good security training program helps workers improve their ability to spot compromised emails and gives them the ability to report anything that seems unusual.



```
sourcetype="mscs:azure:eventhub" body.records.category=AuditLogs
```

✓ **1 event** (8/2/21 10:00:00.000 PM to 8/3/21 10:21:45.000 PM) No Event Sampling ▾

SELECTED FIELDS
- a host 1
- a source 1
- a sourcetype 1

INTERESTING FIELDS
- a body.records.callerIpAddress 1
- a body.records.category 1
- a body.records.correlationId 1
- # body.records.durationMs 1
- # body.records.Level 1
- a body.records.operationName 1
- # body.records.operationVersion 1
- a body.records.properties.activityDateTime 1
- a body.records.properties.activityDisplayName 1
- a body.records.properties.additionalDetails{}.key 1
- a body.records.properties.additionalDetails{}.value 1
- a body.records.properties.category 1

i	Time	Event
>	8/3/21 10:13:29.838 PM	

```
{ [-]
   body: { [-]
     records: { [-]
       Level: 4
       callerIpAddress: 20.190.157.27
       category: AuditLogs
       correlationId: cccc2222-dd33-4444-55ee-666666ffffff
       durationMs: 0
       operationName: Add group
       operationVersion: 1.0
       properties: { [+]
       }
       resourceId: /tenants/ woodgrovegroceries.onmicrosoft  /providers/Microsoft.aadiam
       resultSignature: None
       tenantId: ddddeeee-3333-ffff-4444-aaaa5555bbbb
       time: 2021-08-04T03:13:29.8383564Z
     }
   }
   x-opt-enqueued-time: 1628047129005
   x-opt-offset: 0
   x-opt-sequence-number: 0
}
```

Figure 10: IOC Splunk

2.34 How does threat intelligence work?

Threat intelligence platforms analyze large volumes of raw data on emerging and existing threats to enable rapid, informed cybersecurity decisions. A robust threat intelligence solution maps global signals daily and analyzes them to enable proactive responses to the ever-evolving threat landscape.

A cyber threat intelligence platform uses data science to filter out false alarms and prioritize risks that could cause real damage. This data comes from:

- Open Source Threat Intelligence (OSINT)
- Threat Intelligence Feed
- Internal analysis

A simple threat data feed might provide information on recent threats, but it can't make sense of this unstructured data to determine the threats to which you're most vulnerable or to suggest a post-breach action plan. This work would normally fall to human analysts.

A threat intelligence solution, especially one with tools that use artificial intelligence, machine learning, and advanced capabilities like orchestration, automation, and response (SOAR), automates many security functions to help prevent attacks, rather than simply respond. Threat intelligence also allows security professionals to automate remediation actions when an attack is detected, such as blocking malicious files and IP addresses.

2.35 Why is threat intelligence important?

Threat intelligence is important because it helps organizations prioritize the strategies and tactics that will best protect them from a dynamic threat landscape. It's difficult to keep track of the constant flow of information about emerging threats and decide what is relevant and actionable.

Threat intelligence, combined with tools enriched with machine learning and automation, can improve threat detection and response efforts by:

- Remove the masking of potential adversaries and their motivations;
- Explain an attacker's tactics, techniques, and procedures (TTPs);
- Show the different ways in which various attacks can impact the business;
- Identify common indicators of compromise (IOCs) that signal an active breach;
- Suggest a set of actions to take when attacked;
- Automatically block entire attacks;
- Inform broader security strategies and workflows with advanced threat data.

2.36 Benefits of threat intelligence for security teams

Any company can improve its security posture with threat intelligence. It provides small and medium-sized businesses with the information they need to strategically defend themselves against ransomware and other risks. Security teams and company executives also have much to gain from intelligence.

Beyond better utilization of human expertise and faster response to threats, threat intelligence solutions offer new efficiencies for people in many roles:

- **Security and IT analysts**: Achieve and maintain network security;
- **Cyber Threat Intelligence Analysts**: Analyze threats to the organization and develop detailed information that helps inform others about relevant threats;
- **Security Operations Center (SOC)**: Gain context to assess threats and correlate them with other activities to determine the best and most effective response;
- **Computer Security Incident Response Team (CSIRT))**: Gain a deeper understanding of vulnerabilities, exploits against those vulnerabilities, and the methods attackers use to compromise systems;
- **Executive managers**: Understand which threats are relevant to the organization so you can provide data-driven budget recommendations to the CEO and board.

2.37 Types of threat intelligence

Threat intelligence can be divided into four categories:

2.37.1 Strategic

Strategic threat intelligence is a high-level analysis for non-technical stakeholders who care about the business as a whole, such as C-suite executives, IT management, and boards of directors. This department communicates this type of information in a broad, long-term context. Recipients must manage overall risks, such as the evolving threat landscape, how a business decision could introduce new vulnerabilities, how advanced technology helps companies mitigate threats at a lower cost, or the potential financial and operational implications of a breach.

2.37.2 Tactical

Tactical threat intelligence is information cybersecurity professionals need to take immediate action to mitigate threats. It includes technical information on current TTP trends and IoCs and is typically used by IT service managers, security operations center (SOC) employees, and architects. They use this type of intelligence to make decisions about security controls and create proactive defense strategies. This type of information is constantly evolving and can be automated to allow security teams to maintain maximum agility.

2.37.3 Operating

Operational threat intelligence is the knowledge of specific threats and campaigns. It provides incident response teams with specialized information about an attacker's identity, motivations, and methods. Enable your organization's security professionals to receive this type of intelligence more efficiently with a cyber threat intelligence platform that automates data collection and translates foreign language sources when necessary.

2.37.4 Technician

Closely aligned with operational intelligence, technical threat intelligence refers to signals that an attack is underway, such as IoCs. Use an AI-powered threat intelligence platform to automatically analyze these types of known indicators, which may include phishing email content, malicious IP addresses, or specific malware implementations. SOC and incident response teams can quickly respond to this information and prevent damage to the organization.

2.38 What is an incident response team?

An incident response team, also called a CSIRT (computer security incident response team), a cybersecurity incident response team (CIRT), or a computer emergency response team (CERT), is a group of individuals from various organizational functions responsible for implementing the incident response plan. It includes not only those who remove the threat, but also those who make business or legal decisions related to the incident. A typical team includes the following members:

- An incident response manager, often the IT director, oversees all phases of the response and keeps internal stakeholders informed;
- Security analysts analyze the incident to try to understand what's happening. They also document their findings and collect forensic evidence;
- Threat analysts look outside the organization to gather intelligence and provide further context;
- Someone from management, such as a chief information security officer (CISO) or a chief information officer (CIO), provides direction and acts as a liaison to other executives;
- Human resources specialists help manage insider threats;
- The general counsel helps the team understand liability issues and ensure the collection of forensic evidence;
- Public relations specialists coordinate accurate external communications with the media, clients and other stakeholders;

An incident response team can be a subset of a security operations center (SOC), which manages security operations after an incident response.

2.39 Automated incident response

In most organizations, network and security solutions generate a number of security alerts that cannot be objectively managed by the incident response team. To focus on legitimate threats, many companies implement automated incident response. Automation uses artificial intelligence and machine learning to triage alerts, identify incidents, and eradicate threats by executing a response playbook based on programmatic scripts.

As previously mentioned, SOAR technology represents a category of security tools that companies use to automate incident response. These solutions offer the following features:

- Correlate data from multiple endpoints and security solutions to identify incidents for humans to investigate.
- Execute a pre-crafted playbook to isolate and address known incident types;
- Creating an investigative timeline that includes actions, decisions, and forensic evidence that can be used for analysis;
- Introduction of external intelligence relevant to human analysis.

2.40 Incident Response Procedure

There's no single way to approach incident response, and many organizations rely on a security standards organization to guide their approach. Here are some steps that could be used:

- **Preparation**: Before an incident occurs, it is important to reduce vulnerabilities and establish security policies and procedures. In the preparation phase, organizations conduct a risk assessment to determine where weaknesses exist and prioritize resources. This phase includes writing and refining security procedures (creating an Incident Response Plan and creating playbooks to manage the most common types of incidents), defining roles and responsibilities, implementing tools such as SIEM, EDR, XDR, ticketing solutions, and escalation procedures, and updating systems to reduce risk. Most organizations repeat this step regularly to make improvements to policies, procedures, and systems as they learn lessons or as technology changes (also in compliance with the NIS2 directive).
- **Threat identification**: Once an incident has been identified through rules implemented in monitoring systems, the team must delve deeper into the nature of the breach (classifying its severity level) and document its findings, including the source of the breach, the type of attack, and the attacker's objectives. Furthermore, it is necessary to analyze the impacted systems, conduct an in-depth log analysis, perform memory dumps, and use forensic tools to perform a more accurate and precise analysis to better understand the attack. Maintaining the chain of custody for all collected evidence is crucial in case legal action is necessary. At this stage, the team must also inform stakeholders in compliance with the GDPR.
- **Threat containment**: Containing the threat as quickly as possible is the next priority. The longer attackers have access, the more damage they can cause. To prevent lateral movement—the threat's horizontal expansion to other endpoints and servers—it's necessary to quickly isolate the applications or systems under attack from the rest of the network.
- **Elimination of the threat**: Once containment is complete, the team removes the attacker and any malware from the affected systems and resources. Malicious files, backdoors, and any compromised utilities are then removed, necessary security patches are applied, and an attempt is made to clean and rebuild the systems (paying attention to any threat persistence mechanisms or reinfection attempts—for example, through automated scripts stored in main memory, programs that start at boot time, and registry modifications). Document all actions taken;

- **Recovery and restoration**: Recovery from an incident can take several hours. Once the threat is neutralized, the team restores the systems, retrieves data from backup, and monitors the affected areas to ensure the attacker cannot access them again.
- **Feedback and refinement**: Once the incident is resolved, the team analyzes what happened and identifies improvements that can be made to the process (policies, escalation and crisis management procedures, playbooks). Learning from this phase helps the team improve the organization's defenses.

NIST SP 800-61 Rev. 3 also presents an incident response lifecycle, emphasizing "Detect, Respond, and Recover" as the core IR process, with "Govern, Identify, and Protect" as the broader risk management activities that underlie it. A central theme is "continuous improvement." While the incident response lifecycle (Preparation, Identification, Containment, Elimination, Recovery, Feedback) may appear linear in various guidelines, a deeper analysis reveals a profound interdependence, particularly with integrated forensic activities. The "Identification" phase explicitly requires the use of forensic tools and chain of custody maintenance. "Containment" and "Elimination" are not just technical isolation or removal, but rely heavily on forensic analysis to understand the scope of the attack, the attacker's methods, and any persistence mechanisms. "Recovery" is more effective and secure when informed by forensic analyses that confirm complete eradication and identify vulnerabilities to prevent reinfections. More importantly, the "Feedback and Refinement" phase, which drives continuous improvement, depends directly on thorough post-incident forensic analysis (root cause analysis, lessons learned).

2.41 DFIR (Digital Forensics & Incident Response) in the SOC

When a major incident eludes proactive defenses, the SOC employs DFIR (Digital Forensics and Incident Response) practices for advanced analysis. As CrowdStrike explains, digital forensics examines system data and user activity to determine the nature and perpetrator of an attack, while incident response is the structured process for preparing, detecting, containing, and recovering from a breach. From a SOC perspective, DFIR complements and deepens investigations: it provides evidence and details (file system artifacts, memory dumps, network traces) that help reconstruct the attack sequence and determine its scope.

- **Approaches and methodologies**: In a complex incident, DFIR proceeds in phases: evidence collection (acquisition of disk images and memory dumps, export of network logs), forensic analysis (timeline reconstruction, search for hidden malware, artifact correlation), and response (final containment and remediation of vulnerabilities). For example, following an incident response plan, the analyst can clone the compromised host's disk with a forensic acquisition tool (e.g., FTK Imager) and then analyze the image with Autopsy or EnCase. Alternatively, Volatility can be used on a memory dump to identify injected processes or invisible rootkits on disk. Network forensics (e.g., analyzing a PCAP file with Wireshark or Zeek) allows reconstructing lateral movement or data exfiltration.
- **Commercial and open source tools**:
  - **Autopsy e SleuthKit** (open source): Autopsy is a graphical interface for The Sleuth Kit, widely used in investigative settings. It allows you to analyze hard drives and various types of media, offering modules for artifact extraction (timeline, keyword search, deleted file recovery, web analysis, etc.). SleuthKit is a collection of command-line tools and the underlying C library: it allows you to mount disk images and recover files in depth.
  - **EnCase (Guidance Software/OpenText)**: A widely used commercial forensic suite. It enables comprehensive file system analysis, custom scripting, and produces evidence that can be used in court.
  - **FTK (Forensic Toolkit)**: AccessData's software includes a free imaging module (FTK Imager). FTK scans disks for strings and deleted emails, supports decryption, and generates MD5 hashes to ensure integrity.

- **Volatility**: Open source framework for RAM analysis. It is the "world's most widely used" for memory forensics, written in Python. Volatility extracts processes in memory, loaded modules, live network connections, and detects running malware. Volatility v3 is the current branch, maintained by the community (Volatility Foundation).
- **KAPE (Kroll Artifact Parser and Extractor)**: A free triage tool that quickly scans a system for relevant forensic artifacts (Windows logs, registry keys, browser cache, etc.) and extracts them in minutes. KAPE is very useful for collecting data from multiple endpoints across an organization during an investigative campaign.

2.41.1 Operational example

Suppose a ransomware attack has escaped proactive controls. The SOC convenes the DFIR team: they immediately create a RAM dump of the affected server (with winpmem or FTK Imager) and a bit-by-bit disk image (e.g., in E01 format with EnCase). Using Volatility, the analyst identifies the ransomware process in memory and obtains the malware ID. Autopsy/SleuthKit extracts encrypted files and system logs to determine when and how the malware entered. At the same time, the collected suspicious network traffic (PCAP) is analyzed with Wireshark to determine if any data has been exfiltrated. The results allow both blocking the malware activation keys and documenting the incident for the authorities and the company's RTO.

2.41.2 Compliance and legal considerations

DFIR must follow strict chains of custody. In the context of GDPR, analysts must be careful about personal data that inevitably appears in evidence (user files, emails, web pages visited, notes in memory). For example, a forensic image of a company computer might include customer or employee data. It is therefore standard practice to minimize data (extract only relevant artifacts) and ensure the confidentiality of evidence (secure storage, logged access). In many cases, especially if the investigation involves an employee, an internal mandate or legal authorization is required to analyze the data. Furthermore, the ISO/IEC 27037 standards and ENISA guidelines provide best practices for conducting forensic analysis in compliance with European regulations. In short, DFIR increases the visibility of anomalous events at the file system, memory, and network levels, but it must be performed with a balance of security and privacy, documenting each step for potential audits or legal action.

2.42 Escalation Criteria and Scenarios from SOC to DFIR

A Security Operations Center (SOC) operates as a dedicated team whose primary mission is continuous monitoring, real-time detection, and rapid response to cyber threats. Its primary objective is to safeguard an organization's IT infrastructure by identifying and addressing anomalies before they cause significant damage. The DFIR provides specialized expertise to rigorously and technically address system compromises, delve into the root causes of incidents, and formulate effective remediation strategies to prevent future events.

The SOC is the first line of defense, but when an incident becomes too complex, severe, or persistent, the specialized intervention of the DFIR team is required. Escalation is based on well-defined criteria, often related to the category and severity of the incident. An incident requires the specialized intervention of the DFIR in various circumstances. First and foremost, severity and business impact are determining factors. Incidents that severely impact operations, such as the compromise of sensitive data, widespread malware attacks, unauthorized access to critical systems, or Denial of Service (DoS) attacks that impact the entire company, justify escalation.

DFIR evaluates data exfiltration, system integrity, operational disruptions, attack complexity, and persistence to determine the financial, reputational, and regulatory consequences. Sophisticated and stealthy attacks, such as Advanced Persistent Threats (APTs) or the use of Living Off the Land Binaries and Scripts (LOLBAS), are inherently difficult to detect and require thorough investigation. Legal or compliance requirements are non-negotiable. Incidents that may require the collection and preservation of evidence for legal proceedings, regulatory investigations, or to demonstrate compliance with data protection regulations must be managed by a DFIR team to ensure forensic integrity.

Incident categories that typically trigger escalation include:

- **Unauthorized Access (Category 1)**: when an individual gains unauthorized logical or physical access to a network, system, application, or data.
- **Malicious Code (Category 3)** the successful installation of malicious software (viruses, worms, Trojans) that infects an operating system or application, especially if it is widespread or has a significant impact (e.g., ransomware).
- **Investigation (Category 6)**: This category is specific to unconfirmed incidents or potentially harmful activities that, while not yet fully understood, are deemed by the SOC to warrant a thorough review.
- **High Severity Accidents**: Any incident classified as "High Severity" in the severity matrix is an automatic trigger for escalation. Escalation is a recognition of the specialization and depth required. The fragments clearly distinguish roles: SOCs "focus on detection and alerting," while DFIR "is responsible for investigating and responding to identified incidents."

The collaborative workflow between the SOC and DFIR departments and information sharing are well-defined processes. Initial detection is the responsibility of the SOC, which, using EDR, XDR, SIEM, and AI, continuously monitors the environment to detect anomalies and generate alerts. SOC analysts perform initial alert triage and validation to confirm that the incident is legitimate. Escalation from the SOC to the DFIR occurs when an incident exceeds a certain severity, complexity, or impact threshold. Once the incident has been escalated, the DFIR team takes over for a thorough investigation. Using forensic tools, they collect and analyze digital evidence from file systems, memory, networks, and logs. The goal is to reconstruct the attack timeline, identify the root cause, understand the attacker's tactics, techniques, and procedures (TTPs), and determine the extent of the compromise. Based on the findings of the forensic investigation, the DFIR team develops and implements response plans that may include malware removal, vulnerability patching, system recovery, and defense hardening.

Examples:

- In ransomware attacks, the SOC can detect anomalous cryptographic behavior or C2 communications. Escalation to DFIR is crucial for understanding the ransomware variant, initial access, lateral movement, and guiding containment (endpoint isolation, network segmentation) and recovery (restoration from backups).
- In data breaches, the SOC may detect unusual data exfiltration or unauthorized access. DFIR intervenes to determine exactly what data was compromised, how the attacker obtained the credentials, whether code injection or backdoors were used, and to gather evidence for legal and regulatory obligations.
- For insider threats, the SOC, through behavioral analysis (UEBA) and AI, can report anomalous user activity, such as downloading large volumes of sensitive files after hours or accessing systems outside of the role. The DFIR investigates to confirm the insider threat, identify the actor, and determine the intent and impact.

In conclusion, it is possible to summarize the escalation from the SOC and DFIR departments in the following steps:

1. **Phase 1: Detection and Triage (Domain: SOC)** An alert is generated by a monitoring tool (e.g. SIEM, EDR, XDR) following a correlation rule or the detection of an anomaly.
    - **Initial Analysis**: A SOC analyst (L1/L2) takes charge of the alert. They perform a preliminary analysis to confirm that it is a legitimate incident and not a false positive.

- **Severity Assessment**: The analyst classifies the incident using a severity matrix, considering the potential impact on the business..

2. **Phase 2: Decision and Escalation (SOC-DFIR Point of Contact)**

   - **Escalation Criteria**: If the incident exceeds a predefined threshold, escalation is triggered. Criteria include:**High gravity** (e.g., ransomware, confirmed data breach, critical system compromise), **High complexity** (e.g. APT attacks, use of LOLBAS techniques, unknown malware), **Legal or compliance requirements** (e.g. need to collect evidence for a court).
   - **Formalization of Escalation**: Escalation isn't just a phone call. It's formalized by creating (or updating) an incident ticket in a management system.

3. **Phase 3: Handover of Information**: The incident ticket created by the SOC must contain a comprehensive information package for the DFIR team. The specific information passed is:

   - Initial Alert Details: timestamp, rule triggered, original logs;
   - Incident Timeline: a preliminary chronology of events reconstructed by the SOC;
   - Indicators of Compromise (IOC): any identified IOCs (file hashes, IP addresses, malicious domains, modified registry keys);
   - Affected Systems: list of impacted hosts, users, and applications;
   - Actions Taken: any containment actions already performed by the SOC (e.g., isolating a host);
   - Logs and Data Collected: Any log exports or network traffic captures (PCAP) already performed.

4. **Phase 4: In-Depth Investigation (Domain: DFIR)**: The DFIR team accepts the ticket and begins the forensic investigation, using the information received as a starting point. The goal is to reconstruct the timeline, identify the root cause, and determine the extent of the compromise.

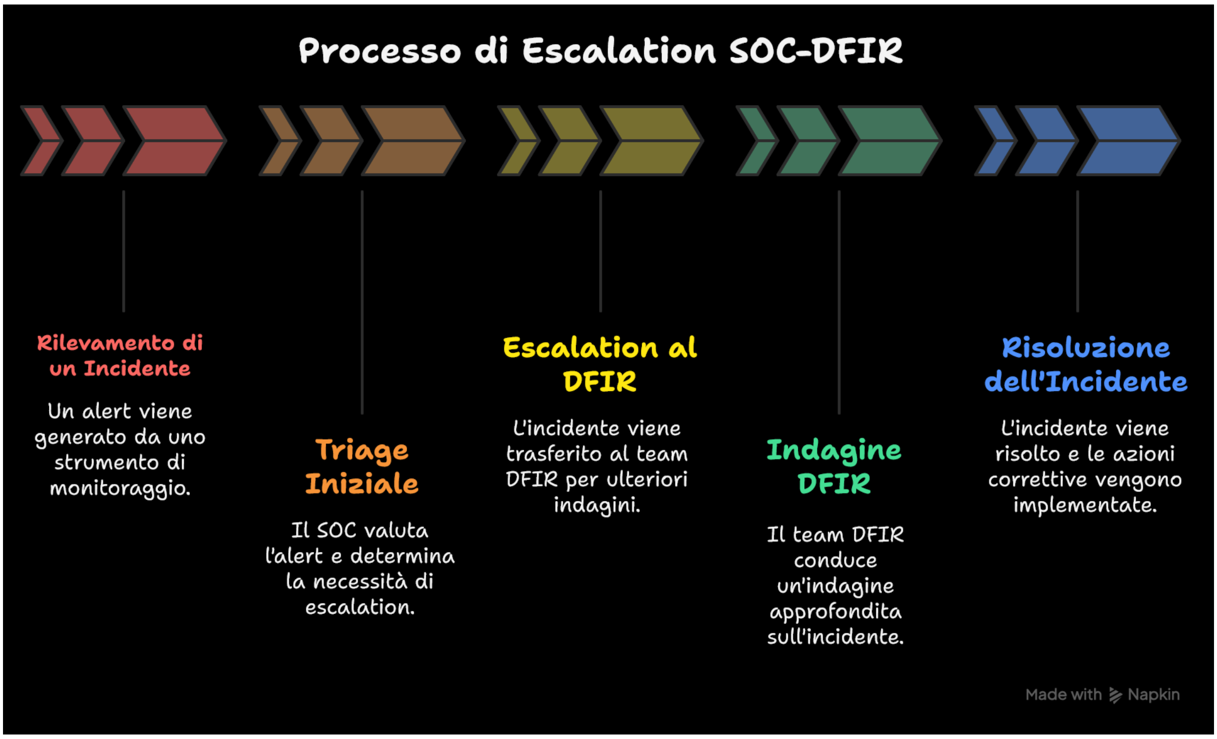

Processo di Escalation SOC-DFIR
Rilevamento di un Incidente
Un alert viene generato da uno strumento di monitoraggio.
Triage Iniziale
Il SOC valuta l'alert e determina la necessità di escalation.
Escalation al DFIR
L'incidente viene trasferito al team DFIR per ulteriori indagini.
Indagine DFIR
Il team DFIR conduce un'indagine approfondita sull'incidente.
Risoluzione dell'Incidente
L'incidente viene risolto e le azioni correttive vengono implementate.
Made with Napkin

Figure 11: Example step Escalation SOC- DFIR

Finally, having a well-equipped SOC and rapid and efficient escalation procedures with the DFIR department is also important in trying to minimize MTTR.

In fact:

- Reducing MTTR decreases the dwell time (time during which the attacker has active access), thus reducing the likelihood of the adversary completing subsequent phases (escalation, lateral movement, exfiltration);
- A shorter time window also reduces the likelihood of forensic artifacts being overwritten or compromised, improving the quality and effectiveness of DFIR actions.

In literature and industry reports, the adoption of automated pipelines (AI + SOAR + playbooks) is correlated with measurable reductions in MTTD/MTTR: empirical examples and case studies show how the automatic orchestration of evidence acquisition and containment actions allows for the transition from times measured in hours to times measured in minutes, with consequent benefits in limiting lateral movements and overall damage. Kuforiji (2025) summarizes these results by arguing that “AI-driven playbooks… drastically improve both Mean Time to Detect (MTTD) and Mean Time to Respond (MTTR)” and that automated actions can be executed “instantaneously… preventing lateral movement and escalation”.

2.43 DFIR Evolution: From Manual Post-Event Analysis to DFIR-as-a-Service and SOC Integration

In the cybersecurity landscape, the discipline of Digital Forensics and Incident Response (DFIR) has undergone profound evolution over the past twenty years, paralleling the growth of digital infrastructures and the complexity of threats. While in the past, DFIR was essentially a reactive practice, confined to forensic laboratories and focused on reconstructing events that had already occurred, today it is a dynamic function, integrated with Security Operations Center (SOC) processes and supported by automation and orchestration tools (SOAR), as well as "as-a-Service" delivery models.

Originally, cyber incident response was characterized by a manual, post-event approach. Investigations were primarily conducted in controlled environments and with tools dedicated to offline analysis of digital media. Techniques such as bit-by-bit disk acquisition, file carving, and manual timeline construction were central to the work of forensic experts, whose primary objective was to ensure the integrity and legal admissibility of evidence. Tools such as The Sleuth Kit, EnCase or FTK represented the operational standard, but required long times, specialized skills and a close connection with the judicial system.

At this stage, DFIR was a post-mortem discipline: it intervened only after the damage had been suffered and with the primary purpose of understanding the causes or supporting legal action (even in those situations of data exfiltration—still widespread today—where former employees leaving the company decide to send confidential material to external storage solutions, whether hardware or cloud). With the advent of the cloud, virtualization, and the widespread deployment of endpoints, this model began to show clear limitations.

Beginning in the 2010s and 2020s, the explosion in data volume and the velocity of cyberattacks forced a paradigm shift. The emergence of tools such as EDR (Endpoint Detection and Response) and SIEM (Security Information and Event Management) introduced the ability to collect telemetry in real time, enabling an initial form of rapid response. During this period, DFIR gradually moved closer to the SOC, becoming no longer just a post-hoc analysis activity but an operational support for incident handling processes. Thanks to centralized log collection and event correlation, SOC analysts could initiate preliminary triage and containment activities, leaving the DFIR team to perform in-depth analysis and formalize the results.

A further turning point came with the spread of SOAR (Security Orchestration, Automation and Response) platforms, which made it possible to automate the initial phases of the DFIR process. Evidence acquisition, volatile data preservation, and initial incident containment can now be executed through automated playbooks, dramatically reducing response times and the risk of evidence loss.

In a modern SOC, SOAR represents the direct point of contact between detection and response: when a potential incident is identified, the system can automatically isolate an endpoint, collect RAM, or generate a forensic disk image, documenting each step in accordance with chain of custody rules. This approach has streamlined collaboration between SOC and DFIR, allowing mitigation and investigation activities to be managed simultaneously.

At the same time, the DFIR department can also leverage machine learning and artificial intelligence to reduce MTTR, automate and optimize acquisition and analysis processes, and focus on where the analyst truly makes a difference.

Indeed, in Kuforiji's (2025) study, the adoption of automated DFIR pipelines allows for reduced mean time to response (MTTR) thanks to "AI-enabled SOAR platforms [that] execute containment actions instantaneously, often without direct human intervention" (p. 7).

Kuforiji also proposes a model of AI-driven DFIR automation in which machine learning, natural language processing (NLP), and robotic process automation (RPA) techniques are integrated into SOAR platforms to reduce both mean time to detect (MTTD) and mean time to respond (MTTR). Specifically, the study shows how automatic log correlation, the execution of response playbooks, and real-time threat correlation improve MTTD and MTTR, implicitly limiting as much as possible the time an attacker can remain in systems where they could engage in privilege escalation and lateral movement.

One way to significantly reduce MTTR, as shown in the study performed by Ali, is to introduce AI and SOAR solutions into Incident Response, which can reduce it (as well as MTTD) by approximately 77%.

However, these models have limitations and risks related to model drift, model bias, adversarial ML, explainability issues, and critical issues in using them in legal proceedings.

And it is also due to the critical issues just listed that it is possible to reduce the MTTR metric by carefully choosing the tools for conducting forensic analysis (as we will see in the Simulation in Chapter 4). Indeed, a solution combining Hayabusa, KAPE, and Zimmerman Tools allows for rapid, efficient, and in-depth triage (in terms of analysis), which optimizes the Incident Response process.

Alongside automation, the market has witnessed the emergence of the DFIR-as-a-Service model, which is increasingly widespread today. The increasing technical complexity of attacks, combined with the shortage of experienced forensic professionals, has driven many organizations to outsource (partially or entirely) the DFIR function. In terms of numbers, the market for digital forensics and incident response solutions is estimated to be worth $10.46 billion in 2025 and is expected to reach $26.43 billion by 2030, with a compound annual growth rate (CAGR) of 20.37%. Providers of these managed services offer continuous availability, specialized expertise, advanced analysis tools, and scalable infrastructure. This allows organizations to rely on professional forensic capabilities even without maintaining a dedicated in-house laboratory.

However, this transformation introduces new challenges in terms of trust, data privacy, and regulatory compliance: sharing sensitive digital artifacts with external providers requires rigorous controls, precise contractual agreements, and structured governance.

Today, modern DFIR is no longer a standalone activity but an integrated function within the enterprise security ecosystem. XDR (Extended Detection and Response) platforms and cloud-native SIEMs provide a constant stream of telemetry from endpoints, networks, and cloud workloads; SOAR systems orchestrate the collection and automation of processes; and the DFIR team intervenes where needed to further investigate, validate, or support the legal process. This "continuous chain" operating model significantly reduces mean time to detect (MTTD) and mean time to respond (MTTR), while maintaining the quality of forensic analysis and traceability of operations. A survey conducted by magnetforensics found the main motivations for automating DFIR processes to be:

- Increase quality through standardized processes (currently, although there are guidelines, tools, and best practices, much freedom is left to the sensitivity of the forensic analyst)
- Improve analyst efficiency and traceability of work performed
- Manage a larger volume of data and analytics

In the most recent context, the trend is toward forensic readiness and cloud forensics: organizations and cloud providers are establishing processes and APIs to securely and compliantly acquire snapshots, logs, and artifacts from virtualized systems. Again, endpoint forensics generated 47% of revenue in 2024, supported by entrenched EDR footprints. However, cloud forensics is the fastest-growing segment, with a CAGR of 28.20%, driven by container-orchestrated environments, where evidence disappears in seconds. The market share of digital forensics and incident response solutions for cloud forensics is expected to reach 31% by 2030. Artificial intelligence and machine learning are increasingly being used to speed up the triage and analysis phase—for example, by automating the extraction of indicators of compromise or identifying anomalous patterns among thousands of events.

The goal is not to replace the forensic expert, but to allow him to focus on high-level decisions, delegating repetitive and low-value-added tasks to machines.

However, significant challenges remain: data volatility in cloud and containerized environments, the fragmentation of telemetry sources, the lack of global standards for acquiring digital evidence, and the difficulties associated with international jurisdiction in cloud forensics. In the latter case, the challenges are significant due to the differing international legislation linking data privacy to investigative activities and international cooperation. This criticality is significant considering that when a user uploads a single file to a cloud service, the provider creates at least two copies on different servers—often in different countries—for redundancy and availability policies.

Furthermore, improved device security (starting with the implementation of security protocols and advanced encryption algorithms designed to protect user data) also complicates extraction and forensic analysis. Furthermore, the spread of anti-forensic techniques has increased the difficulty of investigations.

Data governance also remains a critical issue: the need to reconcile rapid investigations with respect for privacy requires data minimization policies and rigorous access controls. Looking to the future, the world of DFIR is moving toward a hybrid and intelligent model: an ecosystem in which people, automation, and artificial intelligence coexist to provide rapid, scalable, and legally robust responses to security incidents. DFIR-as-a-Service will become the norm for many small and medium-sized businesses, while larger organizations will adopt a "fusion" approach, in which SOC, DFIR, and Threat Intelligence operate as a single entity orchestrated by SOAR and XDR platforms. In this scenario, the challenge will no longer be simply to detect and contain an attack, but to build a resilient infrastructure capable of learning from incidents, automating the response, and ensuring business continuity with immediately available and legally defensible forensic evidence. Ultimately, DFIR is evolving from a specialized, reactive practice to a strategic, proactive function at the heart of modern corporate defense. Its integration with SOC processes, the proliferation of managed services, and the growing use of automation and artificial intelligence clearly outline "where the world is going": toward adaptive, distributed security that is aware of the evidentiary value of every digital event.

2.44 Overview of regulations and standards

Recent years have seen a proliferation of new standards (e.g., ISO 27001, ISO 21434), directives (e.g., NIS2), and regulations (e.g., DORA, GDPR, Cyber Resilience Act). These frameworks are designed to promote the concepts of "security by design" and "security by default," aiming to integrate security from the very beginning of system and service development. The volume and diversity of frameworks—DORA, NIS2, ISO 27035, ISO 27037, ISO 27042, NIST SP 800-61 Rev. 3, and PCI DSS—underscore the critical need for an integrated approach to compliance. Without harmonization, organizations risk duplication of effort, conflicting requirements, and inefficient resource allocation. The multitude of recent regulations and standards, such as DORA, NIS2, GDPR, Cyber Resilience Act, and the ISO 27001 and 21434 series, creates a complex and layered compliance environment, making it almost imperative to address situations such as incident response, data protection, and evidence management in a way that can accommodate multiple frameworks simultaneously.

2.45 Cardinal Principles of Digital Forensics

Digital forensics, as a discipline, "inherently adheres to the most basic forensic principles of preservation, traceability, documentation, and authorization." These principles form the foundation upon which all forensic investigations are built. To maintain the credibility of both the evidence and the analyst in the eyes of legal professionals, any changes made to digital evidence during the analysis and interpretation process "should be traceable and justifiable." This rigorous requirement ensures that the evidence presented is reliable and unaltered.

PCI DSS compliance, particularly in the context of digital forensics, explicitly emphasizes the secure management and storage of digital evidence. This includes the use of secure protocols for their transmission, the implementation of rigorous access controls to limit unauthorized access, and the meticulous maintenance of a chain of custody to track the management and storage of evidence throughout its lifecycle. The integrity of digital evidence is critical to the reliability of forensic analysis. This is achieved through the application of cryptographic hashing algorithms (e.g., represented by the equation $H = \text{hash}(E)$, where H is the hash value, hash is the hashing algorithm, and E is the digital proof), digital signatures to authenticate the source, and secure storage and management procedures designed to prevent any tampering or alteration. Indeed, its ultimate and overarching purpose, especially in a compliance and regulatory context, is to produce evidence that is legally admissible and can withstand rigorous scrutiny in a court of law or by regulatory bodies. This means that every single technical step—from initial collection and acquisition to analysis, storage, and reporting—must be meticulously documented, verified, and performed according to established standards to maintain "forensic reliability."

2.46 ISO/IEC 27035: Information Security Incident Management

ISO/IEC 27035, known as Information Security Incident Management (ISIM), is an international standard that provides best practices and detailed guidelines for establishing an effective incident management process. It covers the entire lifecycle of an incident, from initial detection to closure and post-incident analysis. Its fundamental objective is to provide primary principles for effectively preventing and responding to information security incidents, events, and potential vulnerabilities. The standard is meticulously structured into several parts, each addressing a distinct aspect of incident management:

- Introduction: provides an overview of the key concepts and fundamental principles of information security incident management, outlining the general stages from preparation to continuous improvement;

- Planning and Preparedness: focuses on creating the tools and resources needed for effective incident management. This includes staff training and awareness, developing comprehensive policies and procedures, and implementing incident detection and response technologies.

- Incident Detection and Analysis: describes the methods and tools used to detect and analyze security incidents. The text emphasizes the crucial importance of having effective monitoring systems and well-defined processes to accurately and quickly assess and classify incidents.

- Incident Response: guides organizations in implementing corrective measures to contain, eliminate, and recover from an incident. This section emphasizes the vital role of coordination and communication during the response process.

- Continuous Learning and Improvement: encourages organizations to learn from every incident to strengthen their security posture. It recommends thorough documentation of all incidents, conducting post-incident analyses, and adjusting policies and procedures accordingly to prevent recurrences.

ISO 27035 is presented as a comprehensive framework for information security incident management, outlining a lifecycle that mirrors the Incident Response process described previously (Prepare, Detect, Analyze, Respond, Learn). The standard serves as a strategic blueprint that effectively integrates the highly technical aspects of Digital Forensics and Incident Response (DFIR)—detailed in other ISO standards such as 27037, 27042, 27043, and NIST SP 800-61 Rev. 3—with essential organizational and procedural elements, ensuring that DFIR is not just a technical capability, but a fully integrated and strategically managed organizational function.

2.47 ISO/IEC 27037, 27042, 27043: The Cornerstones of Digital Evidence

The ISO/IEC 270xx series of standards is essential to ensuring that digital evidence is collected, analyzed, and presented in a scientifically valid and legally admissible manner.

- **ISO 27037** (Guidelines for the identification, collection, acquisition and preservation of digital evidence): this standard provides specific guidelines for identifying, collecting, acquiring, and preserving potential digital evidence. It is particularly relevant during the "First Response Process" and the "Digital Evidence Acquisition Process" within the Harmonized Digital Forensic Investigation Process (HDFIP). Initial incident response steps, such as restricting access to affected equipment and removing compromised systems from the network without shutting them down, are critical practices directly aligned with the principles of 27037 for preserving evidence integrity;
- **ISO 27042** (Guidelines for the analysis and interpretation of digital evidence): this standard offers comprehensive guidance on the analysis and interpretation of digital evidence, with an emphasis on ensuring the continuity, validity, reproducibility, and repeatability of the investigative process. It requires meticulous recording of sufficient information to allow independent review of analytical processes. Crucially, any changes made to digital evidence during its analysis and interpretation must be "traceable and accountable" to preserve the evidence's credibility in the eyes of legal professionals.
- **ISO 27043** (HDFIP Model - Harmonized Digital Forensic Investigation Process): this standard promotes good practice methods and processes for the forensic investigation of potential digital evidence. Its primary objective is to ensure acceptance, reliability, usability, and flexibility in digital forensic investigations. The HDFIP model, as part of ISO/IEC 27043, is structured into five comprehensive classes:
  - **Readiness Processes Class**: It focuses on an organization's ability to maximize the use of digital evidence while minimizing investigative costs. This class is optional and primarily involves voluntary organizational participation rather than the role of the investigator;
  - **Initialisation Processes Class**: It deals with the initiation of a digital investigation, in which investigators are physically involved. It includes subprocesses such as: Incident Detection, First Response (incorporating the guidelines of 27035-2 and 27037), Planning and Preparation;
  - **Acquisitive Processes Class**: focuses on the acquisition of potential evidence, covering Incident Scene Documentation, Identification, Acquisition, Transportation and Storage of Digital Evidence;
  - **Investigative Processes Class**: It addresses the core activities of digital evidence discovery and analysis. This includes Digital Evidence Examination and Analysis (guided by 27042), Digital Evidence Interpretation, Report Writing, Investigation Presentation, and Closure;
  - **Concurrent Processes Class**: These processes run in parallel throughout the HDFIP model, ensuring the integrity, confidentiality, availability, efficiency, and admissibility of evidence. Key subprocesses include Authorization, Documentation, Information Flow, Chain of Custody Maintenance, and Interaction with the Physical Investigation Process.

These ISO standards are absolutely critical to ensuring that digital evidence is collected, analyzed, and presented in a scientifically valid and legally admissible manner. ISO 27037, 27042, and 27043 are uniquely positioned within the broader security landscape because they provide granular, technical, and procedural guidance specifically for digital evidence management. The consistent emphasis in these sources on terms such as "continuity, validity, reproducibility, repeatability," "traceable and justifiable changes," and their direct relevance to "legal professionals" and the courts clearly indicates their primary function: ensuring the legal admissibility and integrity of digital evidence. While other frameworks (DORA, NIS2, PCI DSS, NIST) mandate incident response and evidence collection activities, it is the ISO 270xx series that meticulously dictates how these activities must be performed to withstand rigorous legal scrutiny.

2.48 Swiss Regulations in the Digital Forensics Field

Switzerland has developed a robust regulatory framework to address the challenges of cybersecurity and digital forensics, with a particular focus on data protection and evidence management.

At the federal level, the Swiss Constitution of April 18, 1999, protects the right to privacy, specifically the right to protection from the misuse of personal data (Article 13). The collection and use of personal data by private entities are primarily regulated by the Federal Act on Data Protection (FADP) and its ordinances, including the Data Protection Ordinance (DPO). The FADP, effective September 1, 2023, imposes a general requirement to ensure an appropriate level of data security for personally identifiable information, requiring state-of-the-art data security measures without specifying technical standards.

The FADP requires data controllers and processors to notify data security breaches to the Federal Data Protection and Information Commissioner (FDPIC) and, potentially, to data subjects. In the case of automated processing of personal data, additional security and documentation requirements apply, such as the obligation to implement audit trails that must be retained immutably for one year. The Information Security Act (ISA) of December 18, 2020, in force since January 1, 2024, regulates information security practices within the federal government and its administrative bodies. Organizations falling under its scope must report cyberattacks to the National Cybersecurity Center (NCSC) within 24 hours, starting April 1, 2025.

2.49 Criminal Law and Cybercrime

The Swiss Criminal Code (SCC) criminalises various cybersecurity-related activities:

- **Unauthorized Access** (Art. 143bis CPS): Anyone who gains unauthorized access to a data processing system specifically protected against access, using data transmission equipment, is criminally liable. Marketing or making available passwords, programs, or other data intended to commit such a crime (hacking tools) is also punishable;

- **Data Theft (Art. 143 CPS)**: Anyone who, for their own or another's illicit gain, obtains for themselves or others data stored or transmitted electronically or in a similar manner and not intended for them, and which has been specifically protected to prevent access, is criminally liable. This crime requires the actual acquisition of the data and the overriding of security measures;

- **Unauthorized Obtaining of Sensitive Personal Data** (Art. 179novies CPS): anyone who, without authorization, obtains particularly sensitive personal data that is not publicly accessible is liable to prosecution upon complaint;

- **Data Corruption** (Art. 144bis CPS): Anyone who, without authorization, alters, deletes, or renders unusable data stored or transmitted electronically or in any other similar manner is criminally liable. This includes denial-of-service (DoS) attacks, which render data inaccessible, even temporarily. The production, import, marketing, or provision of malicious programs designed to damage data is also punishable;

- **Identity Theft (Art. 179decies CPS)**: Anyone who uses another person's identity without their consent to harm them or to obtain an unlawful advantage for themselves or others is criminally liable.

2.50 US Digital Forensics Regulations

In the United States, the regulatory framework for digital forensics is complex and encompasses both federal and state laws, influencing the collection, analysis, and admissibility of digital evidence.

2.50.1 Electronic Communications Privacy Act (ECPA)

The Electronic Communications Privacy Act of 1986 (ECPA) extended restrictions on government wiretapping from telephone calls to computer-based electronic data transmissions. It includes three main sections:

- **Title I (Wiretap Act)**: Protects wired, oral, and electronic communications in transit by establishing more stringent requirements for search warrants.
- **Title II (Stored Communications Act - SCA)**: It protects communications stored in electronic archives, particularly messages stored on computers. Its protections are weaker than those of Title I, and it does not impose high standards for warrants. For example, the content of emails stored on third-party servers for more than 180 days may be considered "abandoned" by law, and law enforcement may obtain access to them with a written statement certifying their relevance to an investigation, without judicial review, although some federal courts have held that such content is protected by the Fourth Amendment and requires a warrant.
- **Title III (Pen Register and Trap and Trace - PR/TT)**: It allows law enforcement to obtain electronic routing and dialing information, such as email headers and IP addresses, with a court order based on a certification that the information is relevant to an ongoing criminal investigation.

2.50.2 Computer Fraud and Abuse Act (CFAA)

The Computer Fraud and Abuse Act (CFAA), codified under Title 18, Section 1030 of the United States Code, is a foundational law addressing computer crimes. Enacted in 1986 and amended several times, it covers a wide range of conduct, including intentional unauthorized or excessive access to a computer. The CFAA aims to promote privacy and cybersecurity by ensuring the confidentiality, integrity, and availability of information in computer systems.

2.50.3 Health Insurance Portability and Accountability Act (HIPAA)

HIPAA of 1996 is a federal law that establishes national standards for the protection of sensitive patient health information. The HIPAA Privacy Rule protects all "individually identifiable health information" (Protected Health Information - PHI) held or transmitted by a covered entity or its associated businesses, in any form or medium. In the context of digital forensics, HIPAA requires professionals to ensure the confidentiality, integrity, and availability of PHI by limiting access to authorized personnel only and implementing appropriate safeguards. Best practices include using HIPAA-compliant forensic tools and techniques, maintaining a chain of custody for PHI, and implementing secure storage and transmission protocols. Non-compliance can result in significant fines and penalties, as well as reputational damage.

2.51 Criminal Law Framework and Computer Crimes in the Italian Context

Italian criminal law has gradually adapted its provisions to address new forms of crime related to information technology and electronic communications. Legislation has evolved to cover a wide range of unlawful conduct, from computer intrusions and data corruption to fraud and new forms of digital extortion. To avoid an excessively lengthy and legally oriented discussion, the main articles have been summarized in the following table:

Article of the penal code	**Descrizione del reato**	**Penalties (Examples)**	**Notes and aggravating factors**
Art. 615 ter	Unauthorized access to a computer or telematic system	Imprisonment up to 3 years; from 2 to 10 years with aggravating circumstances	Unauthorized access or involuntary retention. Aggravating circumstances: destruction/damage to the system, impediment to data access, attack on systems of public interest (military, public order, healthcare).
Art. 640 ter	Computer fraud	Imprisonment from 6 months to 3 years and a fine; from 2 to 6 years and a fine with aggravating circumstances	Altering the functioning of a computer system for illicit profit, causing harm to others. Does not require misleading. Aggravating circumstance: theft/misuse of digital identity.
Art. 635 ter	Damage to information, data and computer programs of public utility	Increased penalty if the damage occurs	Conduct that is a precursor to damaging state data/programs or public/public utility entities
Art. 635 quater	Damage to computer and telematic systems	Greater penalty than Art. 635 bis	Damage to the integrity of IT/telematics systems as a whole
Art. 615 quater	Unauthorized possession and dissemination of access codes	Imprisonment up to 2 years and a fine of up to €5,164; from 1 to 3 years with aggravating circumstances	Obtaining or distributing access codes for illicit profit
Art. 612 ter	Illicit dissemination of sexually explicit images or videos (Revenge Porn)	Imprisonment from 1 to 6 years and a fine from €5,000 to €15,000	Non-consensual dissemination of sexually explicit material, even if not strictly electronic, often via digital means
(Generic)	Denial-of-Service Attacks (DoS)	Art. 635 bis/quater or interruption of public service	Conduct that makes data or systems inaccessible or unusable fall within

2.52 Code of Criminal Procedure (CPP): Regulation of Computer Forensics

The regulation of digital forensics in the Code of Criminal Procedure (CPP) was significantly introduced by Law No. 479 of December 23, 1999. This law established that digital evidence can be used in criminal proceedings, provided it is acquired in compliance with specific procedural rules to ensure its integrity and reliability. Article 234 of the CPP, in particular, is a broad provision that permits the acquisition of "writings or other documents representing facts, persons, or things through photography, cinematography, phonography, or any other means.". This flexible formulation has allowed for the inclusion of a wide range of digital evidence, such as computer reproductions, videos acquired via mobile devices, audio recordings, and video stills. It is essential that digital evidence be acquired only when strictly necessary to ascertain the crime and if the same information cannot be obtained through other means, in accordance with the principle of subsidiarity. Furthermore, the law requires that such evidence be acquired by "qualified persons" and preserved in a manner that guarantees its integrity. The validity of digital evidence depends heavily on the scientific rigor and transparency of the collection, analysis, and preservation procedures. Therefore, forensic practitioners in Italy must adhere to best practices and international standards, such as ISO/IEC 27042, 27043, and 27050, not only for operational efficiency but also to ensure the validity, reproducibility, and repeatability of the evidence.

2.53 Issues Related to "Live Forensics" and Data Acquisition in Complex Environments (e.g., Cloud)

One of the most pressing challenges is "live forensics," or data acquisition from systems that are running and operational. This practice, while necessary to capture volatile information that would otherwise be lost (such as processes in memory or active network connections), presents a high probability of altering the original data and requires extremely rigorous procedures to ensure the integrity and non-repeatability of the operation. Although cloud storage has become very popular due to its reduced costs, improved performance, and access to new technologies, data acquisition in cloud computing environments introduces additional complexities. The geographical dispersion of data across servers located across multiple countries raises jurisdictional and jurisdictional issues, often requiring international letters rogatory, which can significantly slow down investigations.

Going into more detail:

2.53.1 Technical Complexities in Cloud Data Acquisition

In a traditional investigation, the DFIR analyst has physical control of the asset. In the cloud (IaaS, PaaS, SaaS), the physical infrastructure is managed by the Cloud Service Provider (CSP). Access to the data is exclusively through the APIs and tools provided by the provider (e.g., VM snapshots, log exports). This potentially makes the chain of custody less rigorous, and in a DFIR-as-a-service context, it is important to assist the client in generating a correct forensic image and sending it for analysis without any alteration, using secure and protected sharing tools. The critical points of cloud acquisition are the concept of multitenancy and resource volatility/dynamism. Multitenancy is the economic cornerstone of the cloud. Multiple customers (tenants) share the same physical resources (servers, disks, databases). This creates an obstacle for traditional forensic acquisition, as a provider cannot provide an investigator with a physical image of a server, as it would contain the sensitive data of hundreds of other customers. Consequently, the investigation is forced to limit itself to the data of a single tenant, often accessible only at the logical level (application logs, user files) and not at the physical level (e.g., disk slack space).

Furthermore, modern cloud environments are ephemeral. Resources such as containers (e.g., Kubernetes) or serverless functions can exist for a few minutes or even seconds. Crucial volatile data, such as RAM memory (analyzable with Volatility, as you mentioned), is irretrievably lost upon execution. Live acquisition becomes the only option, but it must be triggered programmatically and instantly, often before the analyst can manually intervene.

These technical challenges demonstrate that the DFIR investigator in the cloud almost never has direct access to the evidence, but must request data from the CSP. It is this act of "request" that transforms a technical issue into a jurisdictional one.

2.53.2 The US vs. EU Conflict of Jurisdiction

Focusing on the jurisdictional conflict, it's worth mentioning the evolution of the US vs. EU scenario. The US and the European Union have different regulations regarding data processing, and this has implications for both cloud technologies and forensic investigations:

- **The US Perspective (e.g. CLOUD Act)**: The United States has laws such as the Clarifying Lawful Overseas Use of Data (CLOUD) Act. This law gives U.S. authorities the power to compel American technology companies (such as Amazon AWS, Microsoft Azure, and Google Cloud) to provide data requested for an investigation, regardless of where that data is physically stored in the world. This creates a principle of extraterritoriality that directly conflicts with the data sovereignty laws of other nations;
- **The European Perspective (e.g. GDPR)**: The European Union, with the General Data Protection Regulation (GDPR), takes the opposite approach. The GDPR places very strict limits on the transfer of personal data outside the EU. A company operating in Europe cannot simply transfer data to a US authority without violating the GDPR unless a specific legal mechanism such as a mutual legal assistance treaty (MLAT) exists.

This legal conflict translates into concrete obstacles for an investigator:

- **Data Access and Delays**: If an Italian company experiences an incident on a cloud infrastructure operated by a US provider with data stored in a data center in Ireland, the investigator can't simply "access" the server. They must make a request to the provider, which is in a difficult position: providing the data to an Italian authority could violate US law, and vice versa. This leads to significant delays, as requests must go through the provider's legal channels, often requiring "international letters rogatory" that can take months.
- **Admissibility of Evidence**: The goal of a DFIR investigation is to produce evidence that is legally admissible in court. If digital evidence is obtained from a cloud provider in a manner that violates the GDPR or local laws, its validity in an Italian criminal trial could be successfully challenged. Chain of custody becomes extremely difficult to ensure when you don't have physical control of the evidence.
- **Data Integrity and Volatility**: Cloud acquisition depends on the tools provided by the provider. These tools may not be designed for forensic purposes and could alter metadata or other digital traces. Furthermore, the dynamic nature of the cloud (with virtual machines that are created and destroyed rapidly) makes live forensics and the acquisition of volatile data (such as RAM) a race against time, further complicated by bureaucratic delays.

As mentioned above, the CLOUD Act (2018) requires a US provider (such as Microsoft, Google, or Amazon) to hand over a user's data, regardless of where that data is physically stored in the world. This directly conflicts with Article 48 of the GDPR:

> "Judgments of a judicial authority and decisions of an administrative authority of a third country requiring a controller or processor to transfer or disclose personal data may be recognised or enforceable in any way only if they are based on an international agreement in force between the requesting third country and the Union or a Member State, such as a mutual legal assistance treaty, without prejudice to other grounds for transfer pursuant to this Chapter."

The legal turning point was the landmark Schrems II ruling (July 2020) by the Court of Justice of the European Union, which invalidated the "Privacy Shield," the agreement that legitimized data transfers between the EU and the US. The Court held that US surveillance laws (specifically FISA 702 and, by extension, the implications of the CLOUD Act) do not provide European citizens with a level of protection "substantially equivalent" to that guaranteed by the GDPR. Specifically, the principle of proportionality was lacking, and, above all, effective judicial protection allowing EU citizens to challenge US government surveillance.

This created a legal vacuum and put US providers in an awkward position. To bridge the gap created by Schrems II, in July 2023 the European Commission adopted the new EU-US Data Privacy Framework (DPF). This new agreement attempts to address the criticisms of the Court of Justice by introducing new protections for EU citizens' data, specifically by limiting US intelligence access to data to what is "necessary and proportionate" and by creating a new redress mechanism, a "Data Protection Review Court" (DPRC), to which EU citizens can appeal.

2.54 Critique of some current forensic investigation methodologies

The purpose of this section is to briefly describe some popular forensic tools, such as Autopsy, Encase, and Velociraptor. These tools are well-known and widely used for performing forensic analyses on systems. However, while they have strengths, they also have weaknesses in performing in-depth analyses when investigating artifacts (a concept that will be discussed later). Therefore, we will now briefly describe the tools and list their main critical issues during analyses. Subsequently, in Chapter 4, through the description of KAPE, Hayabusa, Zimmerman Tools, and simulation, we will gain a detailed understanding of how the hybrid approach based on the use of the latest tools is essential for optimizing and speeding up analyses while increasing their level of depth.

Autopsy and Encase are two of the most popular forensic tools. The former is open source (originally developed by the U.S. Department of Defense and based on The Sleuth Kit (TSK), a collection of command-line tools for file system analysis), while the latter is commercial software developed by OpenText. Both solutions have the following features:

- File system analysis: allows you to examine partitions, directories and files, including deleted or hidden files;
- Timeline Analysis: create timelines of activities (modifications, creations, file accesses);
- Keyword Search: search for keywords across the entire contents of the disk
- Email Analysis: analyze email archives (e.g. Outlook PST, Thunderbird)
- Web Artifacts: extract histories, cookies and cache from web browsers
- Module-based architecture: in both you can add plugins (modules) to expand the functionality:
- Automated reporting: generate reports in HTML, PDF, or XML for legal use

2.54.1 The monolithic approach

In terms of analysis approach, these forensic tools follow a so-called "monolithic approach," the traditional approach that relies on "all-in-one" forensic suites such as EnCase, Forensic Toolkit (FTK), or, in the open-source field, Autopsy. The process involves acquiring a complete disk image (e.g., E01 format) and then ingesting it completely into the suite. The software then parses the entire file system, registry, and artifacts, and indexes each piece of data. Let's now list the weaknesses:

- Speed: this is the main limitation. For example, Autopsy is designed to perform a post-mortem analysis, and the ingestion and parsing process of a modern disk image (hundreds of GB or TB) can take many hours or even days, an unacceptable amount of time in an active Incident Response (IR) scenario;
- Cost: leading solutions like EnCase and FTK have very high licensing and training costs;
- Flexibility: the analyst is tied to the parsing modules provided by the vendor. If a new artifact emerges, they must wait for an official update, while open-source tools like Zimmerman's are updated almost instantly by the community;
- Analysis: Autopsy's graphical interface, while user-friendly, acts as a layer of abstraction. It presents the analyst with high-level categories (e.g., "Executed Programs," "Web History") without forcing them to consider which specific artifacts (e.g., Prefetch, AmCache, SRUM, browser history files) generated that evidence. This makes analysis much more superficial, as it investigates events such as connections, emails, and executed programs without full awareness of all the events (such as authentication, user creation/deletion, password changes) and processes (e.g., PowerShell) occurring on a system. On the other hand, Autopsy excels at browsing and carving out data. The analyst must actively search for evidence, often relying on keyword searches or manual timeline navigation.

Tools like Encase and Autopsy are therefore excellent tools for file system analysis and finding suspicious files, but they are lacking in artifact analysis where the focus is on the action (for example, finding traces in the registry, logs, and Amcache that prove that a given malicious file was executed by a specific user at a given time, how many times it was executed, the exact time of the first and last execution, and all the files and directories accessed by that program).

2.54.2 “Super-Timeline” Approach

Autopsy itself (via the Autopsy Timeline Module), as well as other tools like Plaso/Log2Timeline, allow for a type of approach called "Super-Timeline," whose goal is to collect all the evidence in order to obtain a complete and correlated view of the entire system activity. While useful for gaining a greater overview and context, this type of approach generates a lot of noise (millions of events) and requires advanced filtering techniques.

2.54.3 Velociraptor and Live Forensic

Another approach is Live Forensics, which can be implemented, for example, using the Velociraptor tool. Unlike the monolithic process that requires shutting down the machine to create a forensic copy, this approach allows for live capture of running processes, active network connections, and the operating system state (logged-in users, open files, registry keys in memory, etc.). Velociraptor uses a client-server model in which the server is centralized and available to the forensic analyst, while the clients are all the machines on the network where a tool agent has been installed. From the server, using specific queries, it is possible to interrogate systems in search of IoCs. Although the tool is fantastic for its responsiveness, it still presents many challenges:

- Requires infrastructure: you need to install and configure a server, generate SSL certificates, create agent packages and deploy them to all machines BEFORE an incident occurs;
- Its configuration requires that the machines are active in order to be able to query them and it is necessary to configure the systems well to avoid possible blocks from AV and EDR;
- More in-depth artifact analysis than the monolithic approach but still not sufficient to ensure rigorous analysis;
- Relying on a query language requires mastery of the language since the information returned responds to the written query (in other words: you need to know how to query the agents to obtain valid results);
- The strong point is live forensics and therefore the process of analyzing a mounted disk is expensive and cumbersome.

2.54.4 "Triage-First, Deep-Dive-Second"

The approach that will emerge in the simulation chapter corresponds to a hybrid method that can be described as "Triage-First, Deep-Dive-Second." Specifically, we will leverage open source tools such as KAPE, Hayabusa, and Zimmerman Tools, combining them to mitigate the critical issues of previous models. We will note that thanks to KAPE, the Mean Time to Respond (MTTR) parameter will be significantly reduced thanks to its ability to acquire forensic artifacts in just a few minutes. Thanks to Hayabusa and its Sigma rules, it will be possible to immediately identify high-priority events without having to start with a slow and noisy "super-timeline" like Plaso’s (although it is still used in corporate contexts to gain an even better understanding of timing) or having to wait for a forensic copy to be created (as with EnCase). Furthermore, using the Timeline Explorer tool, it will be possible to correlate Hayabusa alerts but also analyze artifacts and perform groupings, filters (for example in terms of timing and extensions) to gain in-depth knowledge of the impacted machine. Since this methodology makes full use of open source tools, it is possible to integrate other external tools and then analyze the results with Timeline Explorer. Furthermore, since the community is very active, updates to the investigation methodologies (sigma rules, artifacts, features) are much faster than those of a commercial tool.

Let's now briefly discuss emerging directions in the DFIR-SOC space so that we can then begin to move towards malware analysis techniques.

2.55 Emerging Directions in DFIR/SOC: Automation, People, Law, and Queues

The evolution of DFIR and SOC is no longer limited to simple log collection and post-event forensics: we are entering a phase where automation, artificial intelligence, legal constraints, and engineering practices (SOC-as-Code) are redefining processes, responsibilities, and tools. These trends are interconnected and should be considered as a single socio-technical phenomenon: the goal is to achieve faster and more scalable responses without sacrificing forensic rigor and compliance.

Here are some current trends:

1. **Automation and AI in SOCs: AIOps, ML/AI, and Generative AI for Incident Response**

   - SOCs are incorporating AIOps models and artificial intelligence components to improve detection, prioritization, and triage (think, for example, of Crowdstrike and its Charlotte AI, which provides feedback on system vulnerabilities and the necessary remediation). AIOps applies machine learning techniques to correlate events, reduce false positives, and predict operational telemetry anomalies, while generative models and large language models (LLM) are being tested to accelerate reporting, summarize timelines, and suggest remediation actions. Vendors and operators are already demonstrating AI assistant integrations that support analysts in investigations (commercial examples and trials are available on the market). Adoption brings practical benefits: reduced mean time to triage (MTTT), rapid assistance in writing playbooks, and automatic alert enrichment with threat intelligence context. However, concrete risks emerge: reliance on unexplainable models, potential "hallucination" errors (information invented by AI), and the possibility of abuse by adversaries using AI to generate sophisticated attacks. The defense ecosystem will therefore need to combine human-in-the-loop, explicit validation of AI recommendations, and explainability metrics for critical decisions.

     Practically speaking, this means that SOAR playbooks can become "intelligent": no longer just static if-thens but flows that assess risk with ML models, require human approvals for disruptive actions, and learn from analyst feedback.

2. **Outsourcing DFIR**

   - As previously mentioned, one of the aspects (in addition to the fundamental one consisting of the economic one) that favors the outsourcing ((DFIR-as-a-Service / MDR)) of the forensic department in companies is the lack of adequate technical skills to face today's challenges: memory/disk forensics analysis, reverse engineering, cloud forensics, threat hunting based on MITRE ATT&CK.

3. **Legal Challenges in Cloud Forensics**

   - The massive transition to the cloud has made forensics much more complicated from a legal point of view since the servers are located in different countries and it is necessary to comply with the relevant legislation regarding data access and acquisition.

4. **SOC-as-Code: Applying IaC and CI/CD to Security Operations**

   - The "SOC-as-Code" paradigm extends the principles of Infrastructure as Code (IaC) and CI/CD (Continuous Integration / Continuous Deployment) to the world of operational security: SIEM rules, SOAR playbooks, ingestor pipelines, detection configurations, and dashboards become versioned, tested, and deployed code through automated pipelines. Thus:

Detection rules (e.g., Sigma, KQL, SPL formats), log parsing pipelines (Logstash/Grok), SOAR playbooks, and even Kibana/Splunk dashboards are defined in text files (e.g., YAML, JSON, Python). Version Control (Git): These files are stored in a Git repository. CI/CD Pipeline: When an analyst wants to add a new rule, they edit the file, commit it to Git, and an automated pipeline (CI/CD) takes care of:

(a) Test the rule, CI (e.g. correct syntax and can verify that it does not generate false positives in test logs) so that technical and formal requirements are met.

(b) Deploy it: Automatically deploy to production SIEM, CD.

This approach improves repeatability, auditability, and governance (rule versioning, rollback if the rule doesn't behave as expected, automated testing against reference datasets), reduces manual errors, and accelerates time to deploy new detections. However, it requires DevSecOps maturity: rule performance test suites (to assess FPR/TPR impact), playbook sandboxing, and automated rollbacks if deployment generates side effects. The security of the infrastructure code itself (secrets, compromised CI pipeline) also becomes critical: attacking the CI/CD pipeline can allow an adversary to introduce malicious rules or disable detections.

This approach improves repeatability, auditability, and governance (rule versioning, rollback if the rule doesn't behave as expected, automated testing against reference datasets), reduces manual errors, and accelerates time to deploy new detections. However, it requires DevSecOps maturity: rule performance test suites (to assess FPR/TPR impact), playbook sandboxing, and automated rollbacks if deployment generates side effects. The security of the infrastructure code itself (secrets, compromised CI pipeline) also becomes critical: attacking the CI/CD pipeline can allow an adversary to introduce malicious rules or disable detections.

2.56 Malware Analysis

Malware analysis is the process of analyzing malicious software to understand its origin, functionality, and impact. It is a crucial aspect of cybersecurity that helps experts detect, mitigate, and prevent cyber threats. Malware can take many forms, including viruses, Trojans, ransomware, worms, and spyware, each of which can cause significant damage to individuals, businesses, and government organizations. The goal of malware analysis is to determine how a particular piece of malware works, what vulnerabilities it exploits, and how it can be neutralized or removed from an infected system.

There are several key techniques used in malware analysis, each with a specific purpose, but we can classify analysis into two models: static analysis and dynamic analysis.

2.57 Static analysis

Static analysis is the process of analyzing a malware sample without actually running it. This technique involves analyzing the code, file structure, and metadata to extract useful information about the malware's functionality. Security researchers use tools to decompile or disassemble malware code, allowing them to inspect embedded strings, API calls, encryption methods, and network connections. Static analysis is a quick and reliable way to detect known malware signatures, but it has limitations when it comes to obfuscated or polymorphic malware that can modify its code to evade detection.

Static analysis can extract a wealth of information about features, such as PE headers, Application Programming Interface (API) functions, strings, bytes, opcodes, Abstract Syntax Tree (AST), and more. The PE header contains a wealth of program-related information, such as section information, resources, libraries, and lists the API functions used by the program's library information. Additionally, malware can include human-identifiable string information, which in turn can contain malicious behavior information, such as URLs and IP addresses. Byte values can also be extracted from the program, converted to decimal numbers, and set as pixel values to create images useful for comparing malware similarities. A program consists of a set of instructions composed of opcodes and operands, which can be obtained by disassembling the program. The opcode refers to a code that indicates the command to be executed. By using the opcode as a feature, it is possible to approximate the behavioral information intended to be executed. Information about the overall structure is often used to detect script- or document-type malware.

Here are some general steps that could be used:

First, the system collects information about the malware without viewing the code. At this stage, the system essentially distinguishes what a normal file shouldn't do from what a malware file is likely to do, such as generating repetitive activity and unreadable outgoing URLs. Extracting and analyzing metadata, such as file name, type, and size, can provide clues to the nature of the malware. MD5 checksums, an attribute used to verify a file's integrity, or hashes can be compared against the database to verify whether the malware has been previously detected.

Additionally, the system analyzes the code by breaking it down into its various components. Even during this process, the code must not be executed. A file's headers, functions, and strings provide important details about the file's purpose. If malicious intent is present, the system detects it and alerts the user. By examining malware code, security experts can understand the purpose of the file, how it is structured, what actions it might take on a device, and what its overall goal or purpose is. Some common techniques used in static malware analysis include:

- **Disassembling**: In the disassembly phase, static analysis converts the binary code into human-readable assembly language instructions. This allows us to understand the low-level operations and logic employed by the malware.

- **Decompiling**: This technique takes executable files and converts them into high-level code that is easier for people to understand.

- **Format Analysis**: It involves examining the structure and content of a file to ensure it conforms to the expected format. This helps detect malicious alterations, hidden code, or threats lurking within the file.

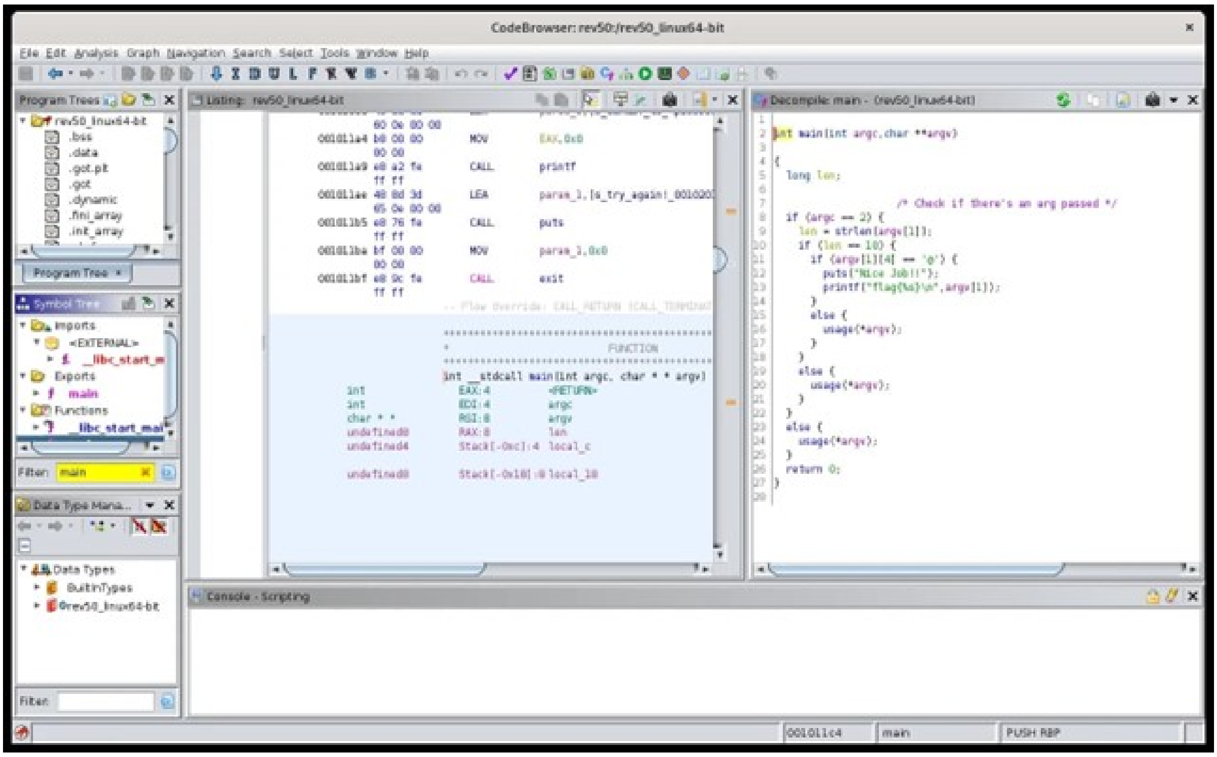
However, malware can go undetected with static analysis if it is sufficiently sophisticated. For example, suppose a PDF file from a third-party vendor generates a string that then downloads a malicious file based on the dynamic string. Since static analysis doesn't execute the code, this PDF file could go undetected and provide access to an attacker impersonating a third-party vendor. In situations like these, dynamic analysis is necessary.

Therefore, static malware analysis helps experts quickly identify and understand malware without the risk of running it on a system. Of course, it has limitations, such as being unable to detect runtime behavior or generate code dynamically. It may be ineffective against heavily obfuscated or encrypted malware, and it requires specialized expertise to accurately interpret the results, which is time-consuming.

Finally, just a few words about the static analysis technique called **reverse engineering**:

Reverse engineering is a more advanced technique that involves decompiling malware code to understand its logic and structure. Security experts use debugging tools to analyze the code and how the malware performs its functions. Reverse engineering is often used to uncover hidden payloads, encryption algorithms, and command-and-control mechanisms used by cybercriminals to control infected systems. This technique requires in-depth knowledge of programming languages, assembly code, and debugging tools, making it one of the most challenging but rewarding methods in malware analysis.

Tools such as IDA Pro, Ghidra, and Radare2 are widely used for reverse engineering and code analysis.



```
                 00 0e 00 00
00101 1a4 b8 00 00       MOV      EAX,0x0
          00 00
00101 1a9 e8 a2 fe       CALL     printf
          ff ff
00101 1ae 48 8d 3d       LEA      param_1,[s_try_again!_0010020
          65 0e 00 00
00101 1b5 e8 76 fe       CALL     puts
          ff ff
00101 1ba bf 00 00       MOV      param_1,0x0
          00 00
00101 1bf e8 9c fe       CALL     exit
          ff ff
                      -- Flow Override: CALL_RETURN (CALL_TERMINAT

                      ****************************************
                      *                  FUNCTION
                      ****************************************
                      int __stdcall main(int argc, char * * argv)
       int              EAX:4          <RETURN>
       int              EDI:4          argc
       char * *         RSI:8          argv
       undefined8       RAX:8          len
       undefined4       Stack[-0xc]:4  local_c

       undefined8       Stack[-0x18]:8 local_18
```

```
int main(int argc,char **argv)

{
  long len;

                  /* Check if there's an arg passed */
  if (argc == 2) {
    len = strlen(argv[1]);
    if (len == 10) {
      if (argv[1][4] == '@') {
        puts("Nice Job!!");
        printf("flag{%s}\n",argv[1]);
      }
      else {
        usage(*argv);
      }
    }
    else {
      usage(*argv);
    }
  }
  else {
    usage(*argv);
  }
  return 0;
}
```

Figure 12: Ghidra Tool

2.58 Dynamic analysis

Dynamic analysis, also known as behavioral analysis, involves running malware in a controlled environment, such as a sandbox, to observe its behavior in real time. By monitoring malware interaction with the operating system, network, and files, analysts can detect malicious activity, including unauthorized data access, registry modifications, and network communications with external servers. This technique is highly effective at identifying new malware strains and zero-day threats that could evade traditional static analysis. However, some sophisticated malware can detect when analyzed in a sandbox and modify its behavior to evade detection.

Unlike static malware analysis, in which malware code is studied without actually running it, dynamic malware analysis reveals malware behavior while it is active. This method provides a clearer picture of the malware's true impact, making it easier to identify and block its harmful effects. Some key aspects that experts look for during dynamic malware analysis might include:

- **Network Activity**: The suspicious file could be a stealer malware that sends the victim's crucial data to a command-and-control server, or a downloader malware that downloads additional malicious files while running. By tracing these connections, analysts can identify the locations (IP addresses) with which the malware communicates and the type of data it sends or receives.

- **File System Changes**:Malware often attempts to hide its presence by creating, modifying, or deleting files on the infected system. During dynamic malware analysis, analysts check whether the malware creates new files, modifies existing ones, or deletes important data. This helps determine whether the malware is attempting to steal information, damage files, or install additional malicious software.

- **Process Manipulation**: Malware often interacts with other processes running on the system, such as system utilities or security software. During dynamic malware analysis, experts observe how the malware interacts with other processes. It may attempt to disguise itself by injecting another process, disabling antivirus programs, or taking control of legitimate system processes to avoid detection.

Malware often interacts with other processes running on the system, such as system utilities or security software. During dynamic malware analysis, experts observe how the malware interacts with other processes. It may attempt to disguise itself by injecting another process, disabling antivirus programs, or taking control of legitimate system processes to avoid detection.

In dynamic analysis, AI may also be used to improve and make analysis more efficient. The AI system monitors the black box environment to see how the malware is modifying it.

2.58.1 Behavioral patterns

The AI system monitors interactions, patterns, and trends between users and entities to identify normal behavior. Anomalies are detected based on this normal behavior. The identified behavioral patterns are classified into one of three categories:

1. **Frequent patterns**: Regular workflows followed by the user on a daily basis and recurring patterns are considered normal.

   Example: Immediately after logging in, the user checks his email, messages, and then moves on to other activities.

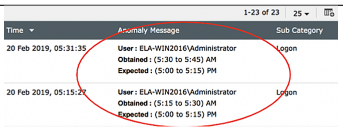
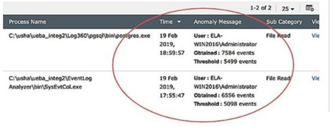
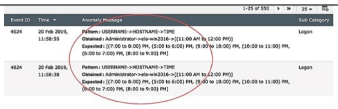
2. **Rare patterns**: These are patterns that occur at regular intervals. They don't happen every day, but from time to time.

   Example: At the end of each month, the user logs in, checks email and messages, and clears the stored cache before moving on to tasks.

3. **Unseen patterns**: These are patterns that the AI system has never detected before. These patterns are red flags and have a high priority. The system immediately sends an alert to the user or security administrator.

   Example: After logging in, the user directly accesses a protected area of the IT environment, rather than checking email. When the user or entity deviates from its normal behavior, our system detects one of these three types of anomalies:

Table 1: AI Pattern

Challenge	Type of anomaly	Solution
Time-based anomalies	Anomaly detected based on the logon time of a user. These users have deviated from their normal behavior by logging on later or earlier than expected by the AI system. The message will now be triggered to IT administrators, who will take appropriate action.	 1-23 of 23 25 Time | Anomaly Message | Sub Category 20 Feb 2019, 05:31:35 | User : ELA-WIN2016\Administrator Obtained : (5:30 to 5:45) AM Expected : (5:00 to 5:15) PM | Logon 20 Feb 2019, 05:15:27 | User : ELA-WIN2016\Administrator Obtained : (5:15 to 5:30) AM Expected : (5:00 to 5:15) PM | Logon
Count-based anomalies	Anomaly detected based on how many times a particular user has accessed an object (in this case, a file).	 1-2 of 2 25 Process Name | Time | Anomaly Message | Sub Category | Vie C:\usha\ueba_integ2\Log360\pgsql\bin\postgres.exe | 19 Feb 2019, 18:59:57 | User : ELA-WIN2016\Administrator Obtained : 7584 events Threshold : 5499 events | File Read | Vie C:\usha\ueba_integ2\EventLog Analyzer\bin\SysEvtCol.exe | 19 Feb 2019, 17:55:47 | User : ELA-WIN2016\Administrator Obtained : 6556 events Threshold : 5098 events | File Read | Vie
Pattern-based anomalies	Anomaly detected based on a pattern of logon times by a user. The user has deviated logon and logoff patterns as depicted	 1-25 of 550 25 Event ID | Time | Anomaly Message | Sub Category 4624 | 20 Feb 2019, 11:58:55 | Pattern : USERNAME->HOSTNAME->TIME Obtained : Administrator->ela-win2016->[(11:00 AM to 12:00 PM)] Expected : [(7:00 to 8:00) PM, (5:00 to 6:00) PM, (9:00 to 10:00) PM, (10:00 to 11:00) PM, (6:00 to 7:00) PM, (8:00 to 9:00) PM] | Logon 4624 | 20 Feb 2019, 11:58:38 | Pattern : USERNAME->HOSTNAME->TIME Obtained : Administrator->ela-win2016->[(11:00 AM to 12:00 PM)] Expected : [(7:00 to 8:00) PM, (5:00 to 6:00) PM, (9:00 to 10:00) PM, (10:00 to 11:00) PM, (6:00 to 7:00) PM, (8:00 to 9:00) PM] | Logon

Depending on the context, these anomalies may indicate an insider threat, account compromise, or data exfiltration. AI, based on the trained model, will provide explanations as to why a particular activity is considered anomalous, helping the SOC team validate the anomalies and make informed decisions.

Finally, here are some tools that could be used for dynamic analysis (to observe malware behavior in an isolated environment): ToThreat.Zone, Joe Sandbox, and Any.Run

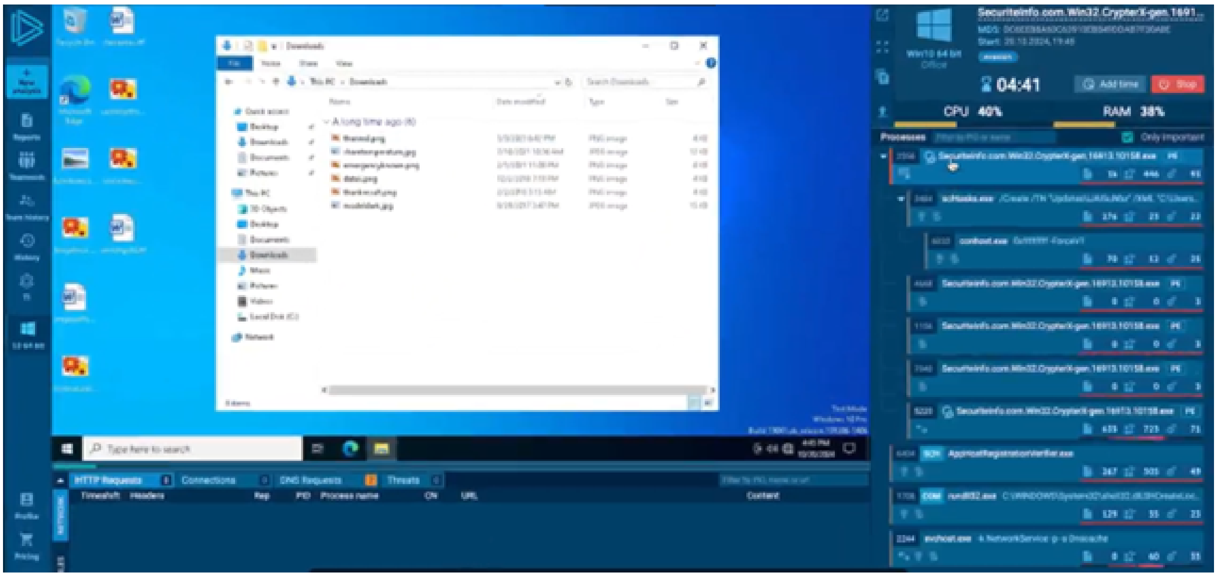


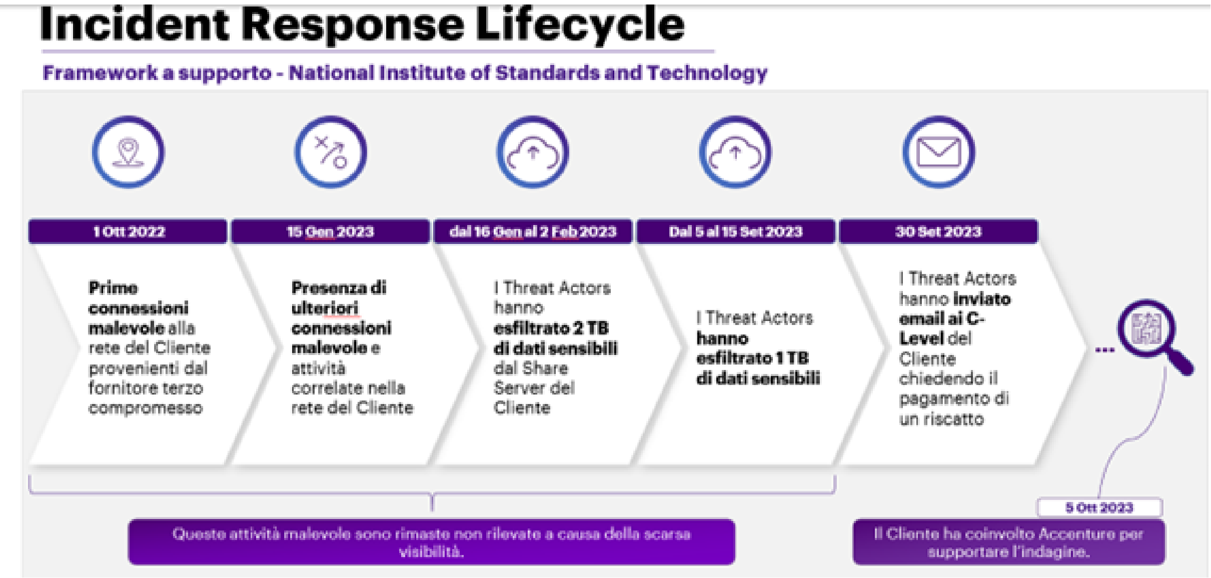
Figure 13: Any.Run

Having completed the discussion of malware analysis, before proceeding with the simulation of a forensic analysis, the following chapter is introduced. Its sole purpose is to bridge the gap between a purely theoretical narrative and the practical aspects. Therefore, the aim is to try to put together some pieces so that the knowledge learned so far can be applied "in the field."

3 Study Case: ProtocolDefense hacking

Let's move on to describe the scenario behind the cyber incident in which ProtocolDefense was involved. (obviously, for privacy and confidentiality reasons, the names and information have been altered so they can be disclosed in this document).

Let's start from the end and show the following timeline:



Incident Response Lifecycle

Framework a supporto - National Institute of Standards and Technology

1 Ott 2022 — **Prime connessioni malevole** alla rete del Cliente provenienti dal fornitore terzo compromesso

15 Gen 2023 — **Presenza di ulteriori connessioni malevole** e attività correlate nella rete del Cliente

dal 16 Gen al 2 Feb 2023 — I Threat Actors hanno **esfiltrato 2 TB di dati sensibili** dal Share Server del Cliente

Dal 5 al 15 Set 2023 — I Threat Actors **hanno esfiltrato 1 TB di dati sensibili**

30 Set 2023 — I Threat Actors hanno **inviato email ai C-Level** del Cliente chiedendo il pagamento di un riscatto

...

Queste attività malevole sono rimaste non rilevate a causa della scarsa visibilità.

5 Ott 2023

Il Cliente ha coinvolto Accenture per supportare l'indagine.

Figure 14: Timeline

In this case study, the affected company contacted the forensics department on October 5, 2023, following an email received from C-level executives demanding a ransom payment to prevent the disclosure of sensitive corporate data that would damage their reputation and business. Analysis revealed that the company lacked an efficient monitoring infrastructure and a targeted logging policy with very short data retention periods, which would allow for the prompt investigation of any suspicious events. Therefore, significant reactive and retrohunting efforts were needed to identify all compromised events involving the company. The analysis revealed that the attack falls within the category of supply chain attacks, in which a third-party supplier is compromised and used to target other companies. In this specific case, retrohunting analysis identified malicious connections to the company network via VPN, which over time resulted in the exfiltration of approximately 3TB of sensitive data from the company's SharePoint.

As previously noted, since the client's infrastructure had very limited logging capabilities, it was necessary to combine various professional figures within the analysis so as not to overlook any details useful for reconstructing the roadmap that led to the cyber incident and avoiding the risk that malicious activity was still ongoing undetected. Once all analyses were completed, the attack vector was identified: it all started with a compromised third-party vendor, which allowed the attackers to infiltrate through an unmonitored VPN and remain undetected for over a year.

Having been contacted following the client's receipt of an email informing them of the compromise, the only possible approach to resolve the company crisis was a reactive investigation of the entire infrastructure (more than 4,000 assets), which revealed that more than 2,000 assets had been compromised. In such a scenario, it is important to analyze the entire infrastructure to verify the presence of other compromises, thus requiring a compromise assessment.

In the field of computer forensics, a compromise assessment is a proactive and in-depth investigation aimed at uncovering active or past cyber threats within an IT infrastructure. Unlike incident response, which is reactive to a known security alert, this assessment assumes that a breach may have already occurred and gone unnoticed (as was the case in this case study). The primary goal is to identify "silent flaws" and adversaries who have already infiltrated systems before they can cause further damage. During a compromise assessment, security analysts actively search for evidence of a compromise, such as unauthorized access, credential theft, lateral movement within the network, hidden malicious software (malware), and data exfiltration. The end result is a detailed report that not only confirms the presence (or absence) of a breach but also provides concrete recommendations for eradicating the threat and strengthening defenses.

Steps of a Compromise Assessment:

- **Step 1: Assess**:

  A compromise assessment begins with forensic data collection, looking for signs of potential compromise in endpoints, network traffic, and logs.

- **Step 2: Analyze**:

  Compromise assessment teams can use the collected data to determine whether an attack has occurred. If so, suspected compromises are validated and the team can develop an analysis of:

  – who is behind the attack
  – because he is targeting an organization
  – what is your goal?
  – how to implement your operational techniques

  This knowledge can be used to anticipate and block your opponent's next moves.

- **Step 3: Assist**:

  Analysts can also use the results of the compromise assessment to respond and remediate identified threats.

- **Step 4: Advise:**

  The compromise assessment concludes when the organization understands how to improve its internal response capabilities and overall security posture so it can prevent or address future incidents.

Therefore, compromise assessment is effective and efficient when the Tactics, Techniques, and Procedures (TTPs) of the MITRE ATT&CK framework are understood. TTPs describe an attacker's modus operandi:

- **Tactics**: They represent the attacker's ultimate goal or strategic objective. Examples include initial access, code execution, privilege escalation, or data exfiltration.
- **Techniques**: These are the specific methods used to achieve a tactical objective. For example, to gain initial access (tactical), an attacker might use a phishing technique or exploit a vulnerability in an application exposed on the Internet.
- **Procedures**: They describe the specific operational steps and tools used to implement a technique. This could be a specific type of phishing email with a specific malicious attachment or the use of an open-source vulnerability scanning tool.

During a compromise assessment, knowledge of adversaries' TTPs actively guides the investigation process. Rather than simply looking for static indicators of compromise (IoCs), such as a file hash or a malicious IP address (which attackers can easily modify), analysts focus on looking for suspicious behavior that matches known TTPs. For example, knowing that a certain threat actor commonly uses the PowerShell tool to move laterally between systems (a technique), the assessment team will actively search for anomalous and suspicious PowerShell usage in system logs. Frameworks like MITRE ATT&CK are essential in this context, as they provide a comprehensive and standardized database of TTPs used by cybercriminal groups. Analysts use this framework to:

- **Threat Hunting**: They formulate hypotheses based on TTPs relevant to the industry or organization and seek evidence of such activities.
- **Contextualizing the results**: When an anomaly is discovered, it can be mapped to a specific TTP, helping to understand the attacker's intent and the stage of the attack.
- **Improve defenses**: Identifying TTPs used in a successful or attempted attack provides valuable insights into which security controls failed and how to strengthen defenses against future similar threats.

In short, a compromise assessment doesn't just ask "have we been breached?" but goes further, asking "how could an adversary breach us, or has already breached us?" The answer to this question lies in actively analyzing and searching for behavioral traces left by attackers' tactics, techniques, and procedures.

To further understand, we can use the following fictional example for a medium-sized company.

Imagine we're a cybersecurity team tasked with performing a compromise assessment for a company. There's no specific alert, but management wants to proactively check for potential intruders. Our approach won't be to search randomly, but to use the MITRE ATT&CK framework to formulate hypotheses and guide our threat hunting.

3.0.1 Step 1: Selecting the Relevant Tactics and Techniques

First, it's necessary to analyze the tools most commonly used by the victim and the attack technique used primarily against companies with similar core businesses. Therefore:

- Victim Profile (Basic): The company uses Microsoft 365 extensively, has remote employees, and on-premise Windows servers.
- Threat Intelligence: We know that cybercriminal groups targeting similar companies often use phishing for initial access and then native Windows tools to avoid detection.

Based on this, it's possible to focus attention and analysis on just a few specific TTPs. For this toy example, we can focus on a common attack chain:

1. Tactics: Initial Access
   - Technique (T1566): Phishing. The adversary sends fraudulent emails to steal credentials.
2. Tactics: Persistence
   - Technique (T1053.005): Scheduled Task/Job: Scheduled Task. The adversary creates a scheduled task to execute malicious code at regular intervals.
3. Tactics: Defense Evasion & Execution
   - Technique (T1059.001): Command and Scripting Interpreter: PowerShell. The adversary uses PowerShell to execute commands and download additional malware, often in "fileless" mode (without writing files to disk).
4. Tactics: Credential Access
   - Technique (T1003): OS Credential Dumping. The adversary attempts to extract password hashes from system memory (e.g., from the LSASS process).

3.0.2 Step 2: Threat Hunting (TTP)

Now having a more specific overview of the attack vehicle and considering a well-defined set of TPPs, it is possible to carry out the following investigations:

- For Phishing (T1566):
  - We don't just look for phishing emails, but their consequences. We analyze authentication logs (e.g., Azure AD) for anomalous logins: logins from countries where the company doesn't operate, late-night logins, or multiple failed logins followed by a successful one from a new geographic location.
  - Result: We find that the account of an employee, mario.rossi, recorded a successful login from a Romanian IP address, followed by valid logins from Italy. This is a suspicious anomaly.
- For Persistence via Scheduled Tasks (T1053.005):
  - On mario.rossi's machine, we don't look for files with suspicious names, but we examine Windows scheduled tasks. We analyze the scheduled task creation logs (Event ID 4698).
  - Result: We find a scheduled task created a few hours after the abnormal login. The task is named "SysUpdate" to appear legitimate, but the action it performs is a hardcoded PowerShell command.
- For malicious use of PowerShell (T1059.001):
  - We enable and analyze PowerShell Script Block logs (Event ID 4104), which record the exact content of executed scripts. We look for obfuscated commands, commands that download content from the Internet (IEX (New-Object Net.WebClient).DownloadString), or that interact with the memory of other processes.
  - Outcome: By decoding the PowerShell command found in the scheduled task, we discover a script that attempts to connect to a URL on Pastebin to download and execute another payload.
- For Credential Dumping (T1003):
  - We monitor access to the lsass.exe process by unauthorized processes. Many Endpoint Detection and Response (EDR) tools have specific rules for this. We also analyze security logs for the use of tools known as Mimikatz.
  - We monitor access to the lsass.exe process by unauthorized processes. Many Endpoint Detection and Response (EDR) tools have specific rules for this. We also analyze security logs for the use of tools known as Mimikatz.

3.0.3 Step 3: Compromise Assessment Results and Conclusions

Using a methodical approach guided by the MITRE ATT&CK framework, we not only answered the question "were we compromised?" but also gained complete visibility into the attack:

- We've confirmed a compromise: mario.rossi's account was compromised, likely through phishing.
- We have identified the attacker's modus operandi: We know exactly what TTPs he used to enter, remain hidden, and escalate privileges.
- We can improve defenses by recommending:
  - implementing multi-factor authentication (MFA) for all accounts to mitigate phishing.
  - the creation of more stringent monitoring rules for the creation of new scheduled activities.
  - limiting the use of PowerShell and improving logging to detect suspicious commands.

  - Configuring an EDR to actively block LSASS access attempts.

In this way, the MITRE ATT&CK framework transforms compromise assessment from a generic and expensive search to a targeted, efficient, and intelligence-driven investigation that delivers concrete and actionable results.

In other words, compromise assessment is a formal, structured project that encompasses threat intelligence (data- and evidence-based knowledge about existing or emerging threats—it seeks to provide answers about who the attackers are, their motivations, and their modus operandi, i.e., tactics, techniques, and procedures), threat hunting (the proactive search for threats within one's network), and retro-hunting (a specialized form of threat hunting. This occurs when new intelligence is acquired—a new indicator of compromise or a newly discovered TTP—and used to search the past).

4 Simulation

The scientific objective of this simulation is to experimentally verify how different DFIR (Digital Forensics and Incident Response) tools, belonging to two technological generations — a traditional one (Autopsy, based on monolithic analysis and complete ingest) and a modern and modular one (KAPE, Hayabusa, Plaso/Log2Timeline, oriented towards triage and targeted correlation) — influence operational performance and the quality of forensic analysis in an Incident Response context.

In order to evaluate these tools, the following metrics will be taken into consideration:

- Operational efficiency: the tool's ability to complete the extraction, parsing, and correlation phases of digital artifacts in a short time, without compromising the quality of the result. It was decided to use the Mean Time To Respond (MTTR), i.e., the average time required to identify, extract, and correlate relevant digital artifacts;
- Quality of evidence: qualitatively assessed based on the number and type of reconstructable artifacts (timestamps, registry keys, application executions, network connections);
- Reliability: Measured in terms of tool consistency. This chapter covers the forensic analysis of malware and a .bat script executed in a command line environment with administrator privileges.

To describe the laboratory experience, it's first necessary to define what we mean by artifacts and some of the tools that will be used in forensic analysis. According to the dictionary, artifacts are defined as: "A work resulting from an intentional human transformation process." Translating this to the forensic field: forensic artifacts are the "fingerprints" that operating systems, applications, and networks leave behind when used. When analyzed, they can reconstruct events or behaviors and be useful for reconstructing an attack timeline.

Some examples:

- System log (Windows Event Log, /var/log/syslog)
- Browser History (Chrome History, Firefox)
- Temporary files and application cache
- File metadata (creation/modification date)

Going into more detail:

1. System logs (EVTX)
   - Windows Event Viewer (logon/logoff, errors, warnings, crashes).
   - They track logins, started processes, network connections.
2. Registry Hives
   - Contains keys relating to installed software, connected peripherals, and logged in users.
   - Files like NTUSER.DAT and SYSTEM.
3. Prefetch files
   - .pf files that record the execution of programs to speed up their startup.
   - They show if and when an application was launched.
   - ShellBags are specific to each user and are located within the user-specific registry hives, NTUSER.dat and UsrClass.dat.
   - Use ShellBags to prove that a folder (or archive) exists (or has existed in the past) in a given location, that a user was aware of a specific folder (or archive), and/or when the folder (or archive) was first or last accessed or modified, and by whom.

4. ShimCache and AmCache

   - AmCache is a Windows registry designed to store information about installed applications, programs running (or present), loaded drivers, and other items. AmCache's primary purpose is to improve application performance and compatibility, ensuring they run correctly in different environments. It also tracks application files, executed programs, driver binaries, Plug and Play (PnP) devices, driver packages, device containers, and application links on systems starting with Windows 10.

     Therefore, AmCache can be used to prove that a file exists (or has existed in the past) in a given location.
   - Shimcache is used to prove that a file exists (or has existed in the past) at a given location. Although in the past, prior to Windows 10, it was considered an "execution evidence" artifact, today it is more accurately described as an "evidence of presence" or "evidence of existence" artifact.

5. Browser history / Cache / Cookie

   - They show visited sites, searches, downloaded files.

6. Jump lists and LNK files

   - Shortcut files (.lnk) are shortcut files created when the file they're associated with is created. They're primarily used by Windows for the metadata they contain.
   - Jump Lists are a collection of link files.
   - Quick links and recently opened files.
   - Useful for understanding which documents the user/attacker has accessed.
   - Link files can be used as evidence of the existence of files that may have existed in the past, but were subsequently deleted from the device.
   - Link files contain the Modified, Accessed, and Created dates of the target file.
   - Link files contain the Modified, Accessed, and Created dates of the target file.
   - The modification date of a Link file indicates the last time the target file was opened.
   - Other key metadata present in a link file includes: destination file path, file size, file attributes, source system name, and source volume information.
   - Jump Lists can be useful for locating files that a particular application has interacted with, as well as obtaining other relevant information.

7. SRUM (System Resource Usage Monitor)

   - Database that tracks app and network usage.
   - SRUM tracks system application usage and user identities via Security Identifiers (SIDs), linking executed applications to user activity. This data can reveal unauthorized or malicious software executions associated with specific accounts, thus aiding investigations.

     The network information stored in SRUM can tell an investigator which networks a given device has connected to; SSID names and incoming and outgoing bytes transferred, along with associated applications, may be available.

     You can use SRUM's network usage data to show data transfers by a given process, but keep in mind that this information is only available in desktop operating systems.

8. RDP and Network artefacts

   - Remote Desktop Connections (e.g., logon type 10 in security logs)
   - Information about remote IPs and sessions.

9. Shellbags
    - They store your browsing history in Windows Explorer (open folders).
10. Shadow Copies and Recycle Bin
    - Previous versions of files and deleted (but recoverable) files

To properly analyze these artifacts, it is necessary to use tools that can uncover them and allow for operations such as selection, filtering, extrapolation, etc.

Although commercial solutions such as Magnet Forensics exist, there are numerous free and open-source alternatives, including Eric Zimmerman Tools and KAPE.

Eric Zimmerman Tools is a suite of free forensic tools developed by Eric Zimmerman, widely used in DFIR (Digital Forensics and Incident Response) for analyzing Windows systems. They have become a de facto standard because:

- They are free and open source (for non-commercial investigative use)
- They cover virtually all major Windows artifacts (mostly in .evtx format)
- have both graphical and command line interfaces
- they are constantly updated.

Below I will very briefly describe some of the Zimmerman Tools that could be used during the simulation:

4.0.1 EvtxECmd — Windows Event Log parser

This tool analyzes Windows .evtx logs (Event Viewer), i.e., all system, security, application, PowerShell, and Sysmon logs, decodes them, and extracts the EventID, Provider, Timestamp, User, and specific fields (Message, CommandLine, Script, etc.) in a readable format (typically generating output in .csv format). It is typically useful for analyzing system event logs, logon/logoff, process creation, RDP, etc.

4.0.2 MFTECmd — Master File Table parser

This tool analyzes the NTFS volume's MFT (Master File Table), decrypting each MFT record (one per file/directory) to generate a complete file system timeline. It is useful for:

- analyze file creation/modification/deletion.
- Check timestamps of suspicious files (e.g. downloaded executables)
- correlate system files and executed .bat or .ps1 scripts

4.0.3 PECmd — Prefetch Parser

This tool analyzes Prefetch (.pf) files, decodes them, and outputs the following information:

- Name of the executed binary (e.g. POWERSHELL.EXE);
- Number of runs (RunCount);
- Last Run Time
- Files and paths accessed during execution

The tool is important for proving that a program was actually executed and for reconstructing the execution timeline and the files accessed.

4.0.4 SBECmd — ShellBags Explorer Command-line parser

It analyzes ShellBags, which are the settings and traces of folders browsed in Windows Explorer. It decodes information about the paths of folders a user has opened so that the following information is output:

- Folder path
- Last time it was viewed
- View type (icons, details, etc.)
- Number of times opened
- Possible presence on removable drives

4.0.5 Timeline Explorer — CSV timeline viewer

It's a graphical user interface (GUI) used to view and filter CSV timelines exported from other tools. It's not a true forensic analysis tool, but it's essential for understanding the results obtained using previous tools.

4.1 KAPE

Kroll Artifact Parser and Extractor (KAPE) is primarily a triage program that targets a device or storage location, finds the most forensically relevant artifacts, and analyzes them within minutes. Thanks to its speed, KAPE allows investigators to identify and prioritize the most critical systems for their case. Furthermore, KAPE can be used to collect the most critical artifacts before the imaging process begins. As imaging completes, the data generated by KAPE can be examined to obtain investigative leads, build timelines, and so on. KAPE performs two main functions:

1. collect files

2. process the collected files with one or more programs

KAPE itself does not perform any of these functions directly; rather, they are accomplished by reading configuration files on the fly and, based on the contents of these files, collecting and processing them. This makes KAPE very extensible when adding or extending functionality.

KAPE uses the concepts of Targets and Modules to accomplish its work. KAPE comes with a set of predefined Targets and Modules for the most common operations needed in most forensic examinations. These can also be used as examples to follow when creating new Targets and Modules.

At a high level, KAPE works by adding file masks to a queue. This queue is then used to find and copy files from a source location. For files that are locked by the operating system, a second pass is performed that bypasses the lock. At the end of the process, KAPE will copy and preserve the metadata of all available files from a source location to a specified directory.

The second, optional, processing phase consists of executing one or more programs on the collected data. This is done by targeting specific files or directories. Multiple programs are run on the files, and the program output is then saved in directories named after a category, such as EvidenceOfExecution, BrowserHistory, AccountUsage, and so on.

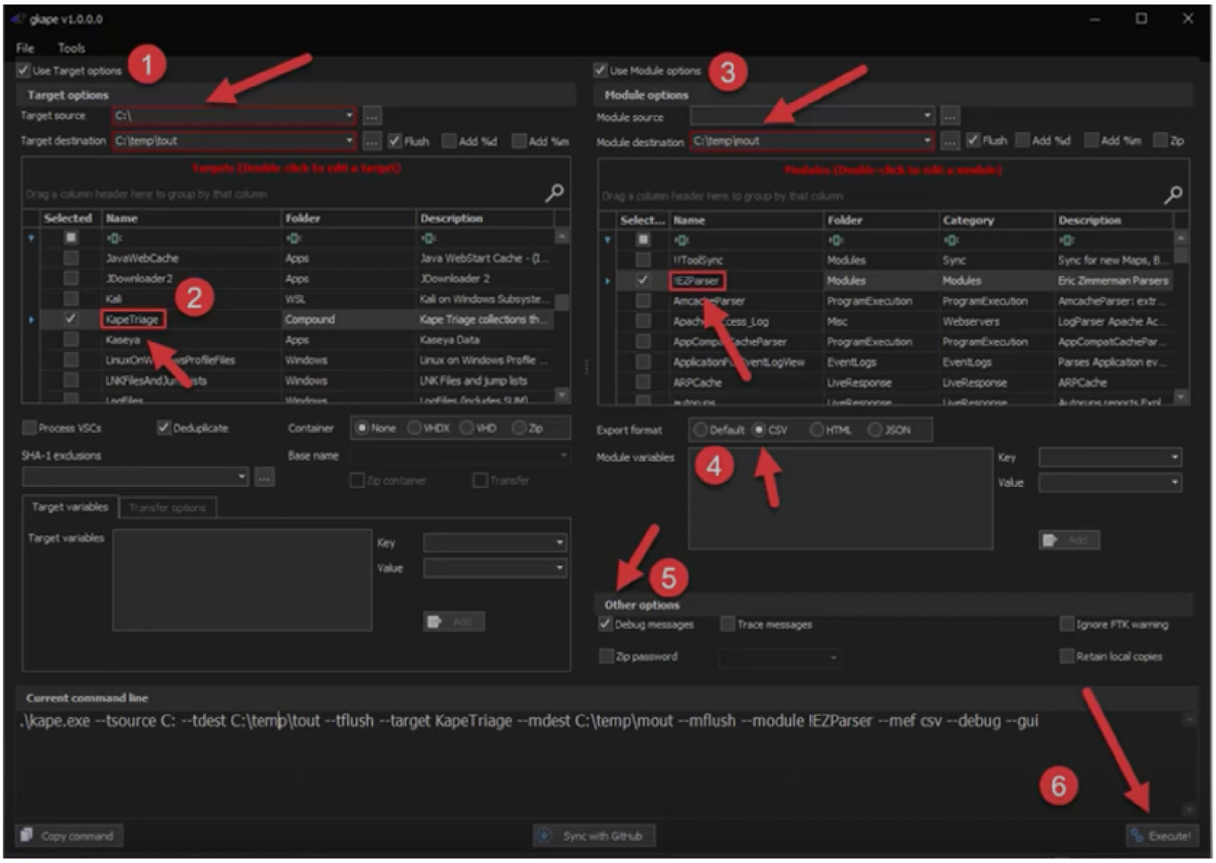
By grouping information by category, analysts at all levels have a means to discover relevant information regardless of the individual artifact from which the information originates. In other words, an analyst no longer needs to know to process Prefetch, ShimCache, Amcache, UserAssist, and so on, in relation to execution evidence artifacts. By thinking categorically and grouping output in the same manner, a broader range of artifacts can be leveraged for each specific requirement.

Targets are essentially collections of file and directory specifications. KAPE knows how to read these specifications and expand them to files and directories that exist in a target location. Once KAPE has processed all the Targets and built a list of files, the list is processed and each file is copied from the source to the destination directory.

For files that are locked by the operating system and therefore cannot be copied using ordinary means, the file is added to a secondary queue. This secondary queue contains all files that were locked or in use.

After the primary queue has been processed, the secondary queue is processed and a different technique, based on raw disk reads, is used to bypass the locks. This allows for a copy of the file as it existed at the source.

Regardless of how the file is copied (regularly or through raw access), the original timestamps of all directories and files themselves are reapplied to the destination files. Metadata is also collected in log files.



gkape v1.0.0.0

☑ Use Target options

Target source: C:\

Target destination: C:\temp\tout ☑ Flush □ Add %d □ Add %m

☑ KapeTriage | Compound | Kape Triage collections th...

□ Process VSCs ☑ Deduplicate Container: ● None ○ VHDX ○ VHD ○ Zip

☑ Use Module options

Module source:

Module destination: C:\temp\mout ☑ Flush □ Add %d □ Add %m □ Zip

☑ !EZParser | Modules | Modules | Eric Zimmerman Parsers

Export format: ○ Default ● CSV ○ HTML ○ JSON

Other options: ☑ Debug messages □ Trace messages □ Ignore FTK warning □ Zip password □ Retain local copies

Current command line

```
.\kape.exe --tsource C: --tdest C:\temp\tout --tflush --target KapeTriage --mdest C:\temp\mout --mflush --module !EZParser --mef csv --debug --gui
```

Figure 15: KAPE graphical interface

4.2 Hayabusa

Another tool we'll use to complete the lab experience is Hayabusa.

As the official GitHub page explains, Hayabusa is a rapid forensic timeline generation and threat hunting tool based on Windows event logs, created by the Yamato Security group in Japan.

Hayabusa currently has over 4,000 Sigma rules and more than 170 built-in detection rules developed specifically for Hayabusa, with new rules being added regularly. It can be used free of charge for both proactive threat hunting on an enterprise scale and for DFIR (Digital Forensics and Incident Response) activities. Analyzing Windows event logs has traditionally been a lengthy and laborious process since Windows event logs are in a difficult-to-parse data format, and most of the information they contain constitutes noise and is not useful for investigations. Hayabusa's goal is to extract only the truly useful data and present it in a concise, easy-to-read format that can be used not only by professional forensic analysts, but also by any Windows system administrator.

To understand how Hayabusa works, it's necessary to take a brief look at YARA and Sigma rules. Sigma rules are a generic, open-source format for describing log patterns. Hayabusa is built around Sigma rules. It uses a large collection of these rules to scan log files and identify specific behaviors that match known attack tactics and techniques (for example, those defined in the MITRE ATT&CK framework).

In essence, Sigma rules are Hayabusa's detection engine. YARA rules are used to identify and classify malware samples. They work by looking for specific patterns (text or binary strings, byte sequences, etc.) within files. The purpose of the tool is therefore to best analyze Windows event logs. Hayabusa can be used at the beginning of a forensic investigation to determine whether the machine (from which we extracted the image) we are analyzing contains traces of suspicious behavior and possible compromises. Unlike other tools that simply parse logs, Hayabusa actively analyzes them for suspicious or malicious activity. It does this using a vast set of detection rules, allowing analysts to quickly identify indicators of compromise (IOCs) such as:

- Abnormal or failed logins;
- Creation of suspicious services;
- Deletion of logs;
- Running suspicious PowerShell commands;
- Exploitation of known vulnerabilities

Using the rules described above, the tool performs correlations and returns matches for potential malicious events. It's important to note that the returned data isn't 100% certain; it's up to the analyst to understand, by interpreting the data obtained, whether the potential malicious match is a normal IT infrastructure activity or a useful clue to reconstructing the chain of events leading to the compromise. Hayabusa classifies these events into four categories: Low, Medium, High, and Critical. By analyzing the events found, it's possible to obtain initial information on how to proceed with the next steps in the investigation. To analyze these matches, the tool provides the csv-timeline functionality, which allows you to pass a file as input (for example, you could analyze an .evtx file) and create a .csv file as output. This can then be analyzed by tools like Timeline Explorer or any spreadsheet program.

4.3 Plaso e Log2Timeline

The following tool will not be used in simulation, but it is widely used in forensic scenarios in corporate environments. Another important tool used to generate a timeline (or supertimeline) useful for reconstructing all the events found in the image and which then allows the forensic expert to apply filters to identify events of particular interest (for example, execution of suspicious processes, unrecognized logins, suspicious connections, etc.) is Log2Timeline. As reported on the GitHub page, while Log2Timeline is the specific tool, Plaso is the framework. Plaso is a Python-based engine used by various tools for the automatic creation of timelines. Plaso's default behavior is to create **super timeline**, but also supports the creation of more targeted timelines.

These timelines help digital forensic investigators and analysts correlate the vast amount of information present in logs and other files found on an average computer.

Plaso's initial goal was to collect all timestamped events of interest on a computer system and aggregate them in a single location for digital forensic analysis (also known as **Super Timeline**).

However, Plaso has become a framework that supports:

- adding new parsers or parsing plugins;
- the addition of new analytics plugins;
- writing custom scripts to automate repetitive tasks in digital forensics or equivalent;

And it's evolving to support:

- adding new general-purpose parsers/plugins that may not have associated timestamps;
- adding more analytical context;
- the ability to assign tags to events;
- a more targeted approach to data collection and parsing.

4.3.1 A Practical Guide to Forensic Analysis with Hayabusa, KAPE, and Timeline Explorer

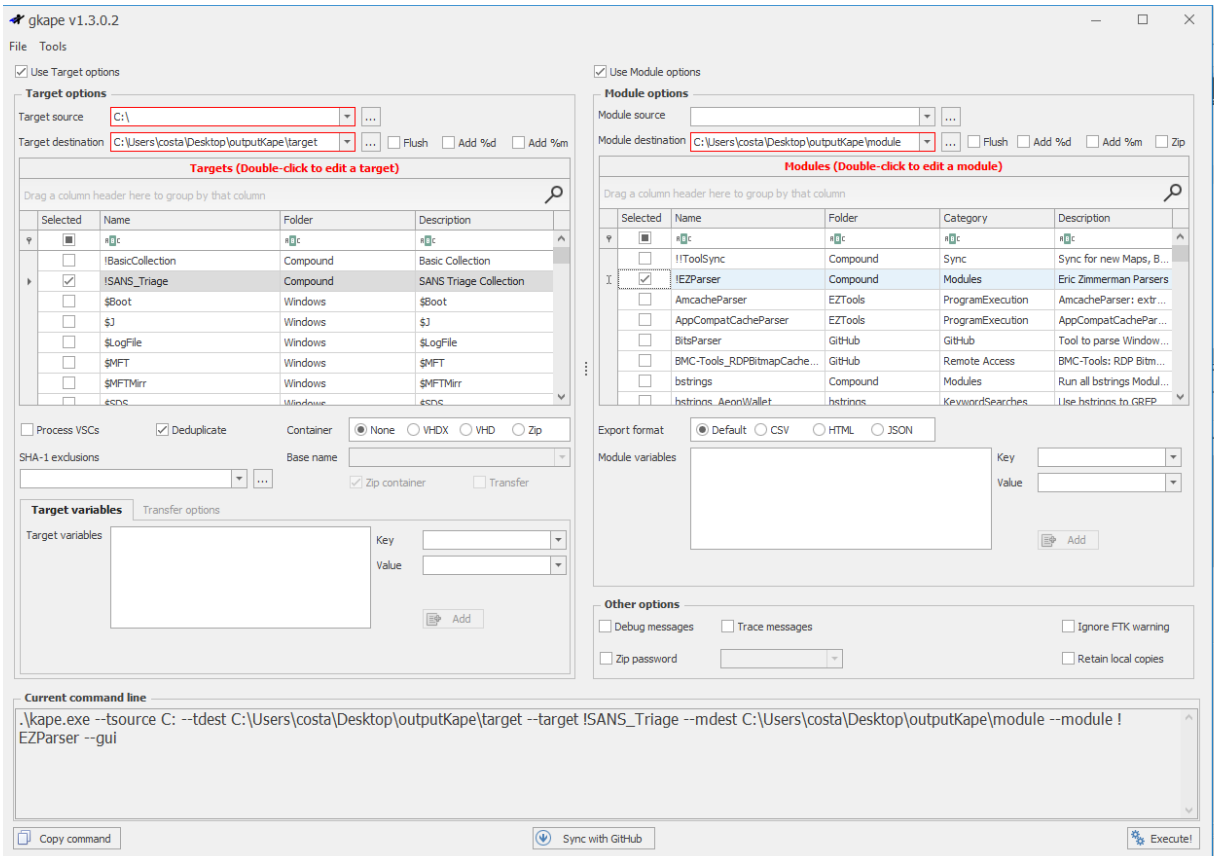
This subsection lists a workflow that can be used in corporate contexts to perform a forensic analysis (and therefore understand what happened) and to build a correct timeline of events where a wide variety of activities (business, development, systems, analysis, and maintenance) are performed on thousands of machines (whether servers, virtual machines, or workstations).

Phase 1: Collecting Artifacts with KAPE

Objective: Quickly and reliably collect all crucial forensic artifacts from the target system.

Once the software is open, you need to choose the resource you want to analyze (if you have an image, you can select the entire mounted image as the Source target). Then you need to choose the destinations (i.e., where Kape will save the detected artifacts and the related .csv files; you will need to create two folders, one for each folder: target and one for each module). Finally, you need to choose the "tools."

In general (as we will do in the simulation), you can enable the KapeTriage options (which will run the entire Zimmerman Tools suite) and !EzParser (simply a parser of Eric Zimmerman's tools).



☑ Use Target options

Target options

Target source: C:\

Target destination: C:\Users\costa\Desktop\outputKape\target ☐ Flush ☐ Add %d ☐ Add %m

Targets (Double-click to edit a target)

Drag a column header here to group by that column

Selected	Name	Folder	Description
☐	!BasicCollection	Compound	Basic Collection
☑	!SANS_Triage	Compound	SANS Triage Collection
☐	$Boot	Windows	$Boot
☐	$J	Windows	$J
☐	$LogFile	Windows	$LogFile
☐	$MFT	Windows	$MFT
☐	$MFTMirr	Windows	$MFTMirr
☐	$SDS	Windows	$SDS

☐ Process VSCs ☑ Deduplicate

Container: ● None ○ VHDX ○ VHD ○ Zip

SHA-1 exclusions

Base name

☑ Zip container ☐ Transfer

Target variables | Transfer options

Target variables

Key

Value

Add

☑ Use Module options

Module options

Module source:

Module destination: C:\Users\costa\Desktop\outputKape\module ☐ Flush ☐ Add %d ☐ Add %m ☐ Zip

Modules (Double-click to edit a module)

Drag a column header here to group by that column

Selected	Name	Folder	Category	Description
☐	!!ToolSync	Compound	Sync	Sync for new Maps, B...
☑	!EZParser	Compound	Modules	Eric Zimmerman Parsers
☐	AmcacheParser	EZTools	ProgramExecution	AmcacheParser: extr...
☐	AppCompatCacheParser	EZTools	ProgramExecution	AppCompatCachePar...
☐	BitsParser	GitHub	GitHub	Tool to parse Window...
☐	BMC-Tools_RDPBitmapCache...	GitHub	Remote Access	BMC-Tools: RDP Bitm...
☐	bstrings	Compound	Modules	Run all bstrings Modul...
☐	bstrings_AeonWallet	bstrings	KeywordSearches	Use bstrings to GREP

Export format: ● Default ○ CSV ○ HTML ○ JSON

Module variables

Key

Value

Add

Other options

☐ Debug messages ☐ Trace messages ☐ Ignore FTK warning

☐ Zip password ☐ Retain local copies

Current command line

```
.\kape.exe --tsource C: --tdest C:\Users\costa\Desktop\outputKape\target --target !SANS_Triage --mdest C:\Users\costa\Desktop\outputKape\module --module !
EZParser --gui
```

Figure 16: KAPE example

Phase 2: Threat Hunting in Logs with Hayabusa

Objective: Quickly identify the most suspicious activities by analyzing the collected event logs. First, the investigation is conducted. This is possible by running a command line similar to the following: hayabusa.exe csv-timeline -d "C:/Windows/System32/winevt/Logs" -o report.csv

The previous command runs the Hayabusa tool and creates a timeline in CSV format. Specifically, -d specifies the source folder from which Hayabusa should read the Windows logs, while -o specifies the file (or rather, the path) where the timeline should be created.

Next, it is necessary to view and analyze the Hayabusa Report: the cmd tool will display the events that triggered a Sigma rule, sorted by criticality level (critical, high, medium, low). To speed up analysis, it's a good idea to first focus on events categorized as critical or high, and then proceed from there using the Timeline Explorer tool to investigate the .csv report produced by Hayabusa. This tool allows you to perform filters and groupings, for example; this part of the investigation is important for getting an initial overview and understanding the severity of the situation.

Step 3: Correlating and Creating the Timeline with Timeline Explorer

Objective: Combine all collected and processed data into a single super-timeline to reconstruct the chain of events. Now, it's necessary to use this tool to investigate all the .csv files created by Kape to perform a complete forensic investigation (for example, it's advisable to look for patterns such as RDP access from strange IP addresses, user creation, attempts to disable security services) and reconstruct the chain of events that led to the cyber incident.

4.4 Premises for Laboratory Simulation

During the theoretical and practical training program on DFIR in a specific corporate context, the following combination of open source tools was chosen for the laboratory experience: KAPE, Hayabusa, Zimmerman Tools, and Timeline Explorer.

KAPE (Kroll Artifact Parser and Extractor) was used primarily for its ability to rapidly and modularly collect and parse digital artifacts from Windows systems, significantly reducing acquisition times compared to manual solutions.

Hayabusa, an open source framework for parsing and correlating Windows Event logs, was chosen for its efficiency in temporally reconstructing suspicious activity (e.g., TTPs mappable according to MITRE ATT&CK) and its ability to produce structured reports.

Zimmerman Tools (specifically EvtxECmd, MFTECmd, PECmd, SBECmd) were used for their precision in parsing specific artifacts (Event Logs, MFT, Prefetch, ShellBags), enabling detailed analysis of host behaviors, executed processes, and process execution traces.

Finally, Timeline Explorer integrated the extracted data into a single timeline, providing a correlated and sequential view of system events. This approach is crucial in the forensic phase, as it allows for understanding the temporal causality of events and validating investigative hypotheses.

Let's now perform a practical simulation, first analyzing the effects of the GhostRat malware on a VM running VMWare.

Acer Aspire A315-53G PC Features:

Processore (CPU)	Intel Core i5-8250U
Scheda grafica (GPU)	NVIDIA GeForce MX130
RAM	4 GB DDR4 (welded) + 16 GB "free"
Archiviazione	2 SSDs of 1 TB each
Sistema Operativo	Windows 10 22H2

Table 2: Acer Aspire A315-53G

VM Features:

OS	Windows 10 22H2
RAM	9 GB
Numero Processori	8
Hard Disk (SSD)	120 GB
WSL	Version 2 with Ubuntu distribution
.NET 6	Useful for running Zimmermann Tools
Kape	installed
Hayabusa	installed
7-Zip	Only for extracting compressed archives

Table 3: VM Features

We are currently running the simulation in a secure, isolated environment (with the network card disconnected in the virtual environment). Before conducting any investigations, a forensic acquisition of the machines involved in the cyber incident is required in a work context. If the forensic analysis is performed by a third-party company, an on-site acquisition can be performed, observing best practices and regulations regarding integrity and chain of custody, or the affected company can be assisted in uploading the acquisitions to secure shared SharePoint servers.

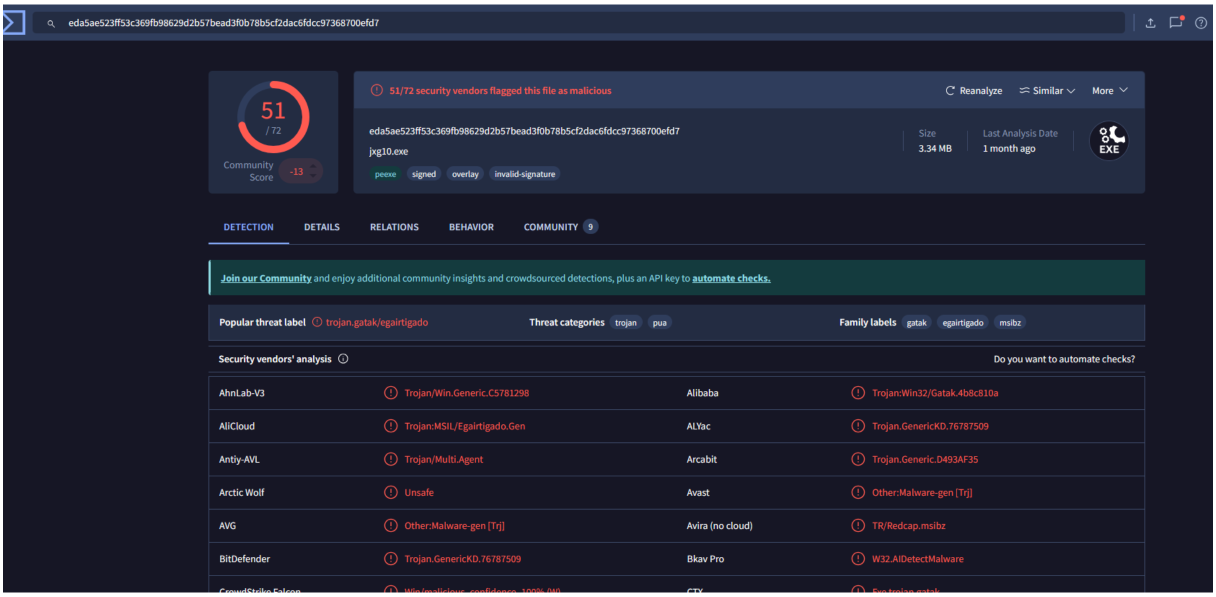
Now let's get to the heart of the simulation. The following is not intended to be a comprehensive discussion of a corporate case study, but rather an experiment on a single host to demonstrate the analysis methodologies and reasoning a forensic analyst might employ.

4.4.1 Simulation with GhostRat malware.

As a first small experiment, the GhostRat malware was downloaded from MalwareBaazar.

As indicated on the website, MalwareBazaar is a platform created by abuse.ch and Spamhaus, dedicated to sharing malware samples with the cybersecurity community, antivirus vendors, and threat intelligence providers.

From this, I chose malware that wasn't too malicious to avoid the risk of it leaking from a "controlled" environment.



eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2dac6fdcc97368700efd7

51/72 security vendors flagged this file as malicious

eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2dac6fdcc97368700efd7
jxg10.exe
peexe signed overlay invalid-signature

Size: 3.34 MB
Last Analysis Date: 1 month ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: trojan.gatak/egairtigado
Threat categories: trojan pua
Family labels: gatak egairtigado msibz

Security vendors' analysis

Do you want to automate checks?

AhnLab-V3	Trojan/Win.Generic.C5781298	Alibaba	Trojan:Win32/Gatak.4b8c810a
AliCloud	Trojan:MSIL/Egairtigado.Gen	ALYac	Trojan.GenericKD.76787509
Antiy-AVL	Trojan/Multi.Agent	Arcabit	Trojan.Generic.D493AF35
Arctic Wolf	Unsafe	Avast	Other:Malware-gen [Trj]
AVG	Other:Malware-gen [Trj]	Avira (no cloud)	TR/Redcap.msibz
BitDefender	Trojan.GenericKD.76787509	Bkav Pro	W32.AIDetectMalware
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	CTX	Exe.trojan.gatak

Figure 17: Ghost malware reputation

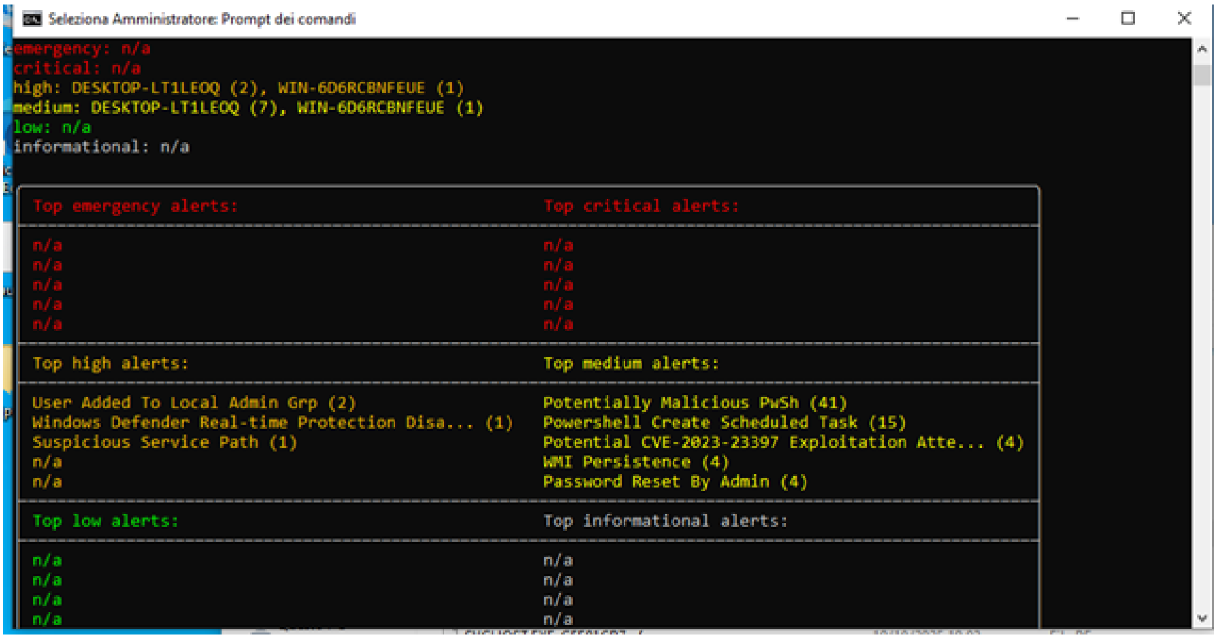
GhostRAT is a remote access trojan for Windows platforms, which originally emerged around 2008. Typical capabilities of GhostRAT (or its variants) include:

- Full remote control of the infected system: remote shell, process listing, file execution.
- Keylogging (keystroke recording), screenshots, webcam/microphone activation
- File download/upload, persistent in the system, and often with anti-analysis capabilities

The infection and operational functioning are as follows:

- Common vector: Phishing attachments, downloads from deceptive web pages, fake installers masquerading as legitimate software
- Once executed, the dropper installs the Gh0st server/agent component, configures the connection to a command & control (C2) server, and establishes the communication channel with the attacker.
- It often uses encrypted (or encoded) communication to the C2, and hides its tracks (packing, anti-VM, kernel drivers).

The malware was executed on October 19, 2025, at 10:02 AM.
The hayabusa tool was first executed using the following command line:
hayabusa.exe csv-timeline -d "C:/Windows/System32/winevt/Logs"
-o report "C:/Users/costa/Desktop/OutputHaya"

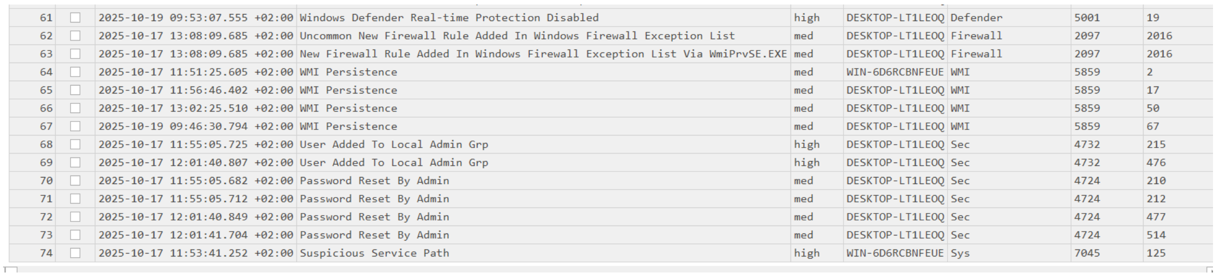


```
emergency: n/a
critical: n/a
high: DESKTOP-LT1LEOQ (2), WIN-6D6RCBNFEUE (1)
medium: DESKTOP-LT1LEOQ (7), WIN-6D6RCBNFEUE (1)
low: n/a
informational: n/a

 Top emergency alerts:                              Top critical alerts:
 ----------------------------------------------------------------------------------------------
 n/a                                                n/a
 n/a                                                n/a
 n/a                                                n/a
 n/a                                                n/a
 n/a                                                n/a
 ----------------------------------------------------------------------------------------------
 Top high alerts:                                   Top medium alerts:
 ----------------------------------------------------------------------------------------------
 User Added To Local Admin Grp (2)                  Potentially Malicious PwSh (41)
 Windows Defender Real-time Protection Disa... (1)  Powershell Create Scheduled Task (15)
 Suspicious Service Path (1)                        Potential CVE-2023-23397 Exploitation Atte... (4)
 n/a                                                WMI Persistence (4)
 n/a                                                Password Reset By Admin (4)
 ----------------------------------------------------------------------------------------------
 Top low alerts:                                    Top informational alerts:
 ----------------------------------------------------------------------------------------------
 n/a                                                n/a
 n/a                                                n/a
 n/a                                                n/a
 n/a                                                n/a
```

Figure 18: Hayabusa on Ghost malware

As you can see from the previous image, events classified as high and medium were detected. We can immediately exclude the "Windows Defender Real-Time Protection Disabled" alert from the analysis because I previously disabled Windows Defender protection to prevent administrator execution from being prevented during pre-execution or detection.

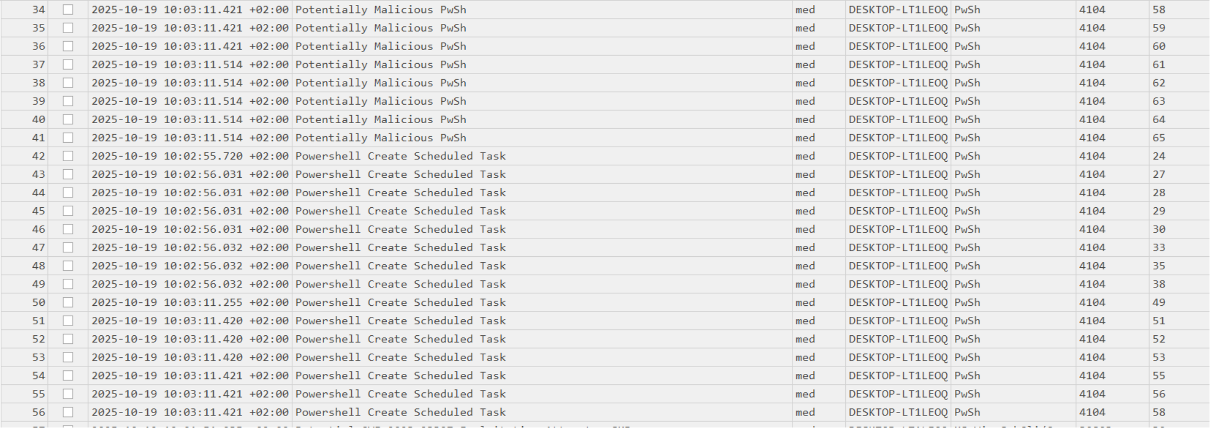
If we filter the alerts presented to us by Jayabusa in Timeline Explorer, we can see that all the activities occurred on October 17th, the day I set up the VM (and therefore, during Windows installation, the user account was created, the password was changed (from the default), and the types of activities detected as suspicious by the systems were performed. Below are some screenshots illustrating this.



61	□	2025-10-19 09:53:07.555 +02:00	Windows Defender Real-time Protection Disabled	high	DESKTOP-LT1LEOQ	Defender	5001	19
62	□	2025-10-17 13:08:09.685 +02:00	Uncommon New Firewall Rule Added In Windows Firewall Exception List	med	DESKTOP-LT1LEOQ	Firewall	2097	2016
63	□	2025-10-17 13:08:09.685 +02:00	New Firewall Rule Added In Windows Firewall Exception List Via WmiPrvSE.EXE	med	DESKTOP-LT1LEOQ	Firewall	2097	2016
64	□	2025-10-17 11:51:25.605 +02:00	WMI Persistence	med	WIN-6D6RCBNFEUE	WMI	5859	2
65	□	2025-10-17 11:56:46.402 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	17
66	□	2025-10-17 13:02:25.510 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	50
67	□	2025-10-19 09:46:30.794 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	67
68	□	2025-10-17 11:55:05.725 +02:00	User Added To Local Admin Grp	high	DESKTOP-LT1LEOQ	Sec	4732	215
69	□	2025-10-17 12:01:40.807 +02:00	User Added To Local Admin Grp	high	DESKTOP-LT1LEOQ	Sec	4732	476
70	□	2025-10-17 11:55:05.682 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	210
71	□	2025-10-17 11:55:05.712 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	212
72	□	2025-10-17 12:01:40.849 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	477
73	□	2025-10-17 12:01:41.704 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	514
74	□	2025-10-17 11:53:41.252 +02:00	Suspicious Service Path	high	WIN-6D6RCBNFEUE	Sys	7045	125

Figure 19: Activities to exclude

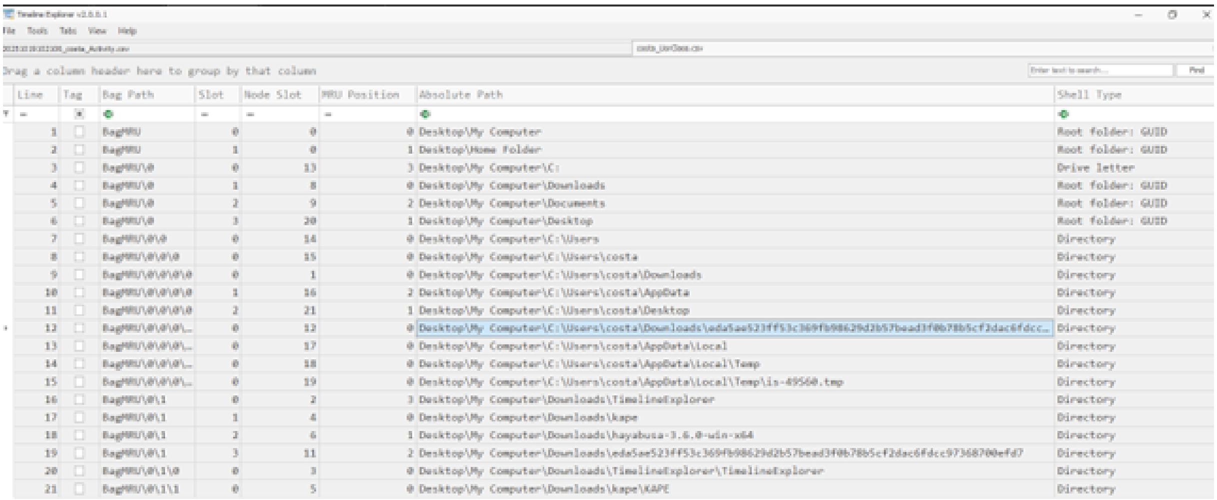
Scrolling through the events detected by Hayabusa it seems that some activities have been performed, in particular powershell activities



34	□	2025-10-19 10:03:11.421 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	58
35	□	2025-10-19 10:03:11.421 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	59
36	□	2025-10-19 10:03:11.421 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	60
37	□	2025-10-19 10:03:11.514 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	61
38	□	2025-10-19 10:03:11.514 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	62
39	□	2025-10-19 10:03:11.514 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	63
40	□	2025-10-19 10:03:11.514 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	64
41	□	2025-10-19 10:03:11.514 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	65
42	□	2025-10-19 10:02:55.720 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	24
43	□	2025-10-19 10:02:56.031 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	27
44	□	2025-10-19 10:02:56.031 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	28
45	□	2025-10-19 10:02:56.031 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	29
46	□	2025-10-19 10:02:56.031 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	30
47	□	2025-10-19 10:02:56.032 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	33
48	□	2025-10-19 10:02:56.032 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	35
49	□	2025-10-19 10:02:56.032 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	38
50	□	2025-10-19 10:03:11.255 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	49
51	□	2025-10-19 10:03:11.420 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	51
52	□	2025-10-19 10:03:11.420 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	52
53	□	2025-10-19 10:03:11.420 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	53
54	□	2025-10-19 10:03:11.421 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	55
55	□	2025-10-19 10:03:11.421 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	56
56	□	2025-10-19 10:03:11.421 +02:00	Powershell Create Scheduled Task	med	DESKTOP-LT1LEOQ	PwSh	4104	58

Figure 20: Malware PowerShell activity

Moving now to analyze the CSV files produced by Kape, it is possible to note that the software was executed as shown in the following screenshot:

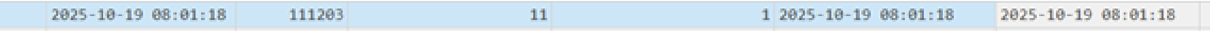


Timeline Explorer v2.0.0.1
File Tools Tabs View Help
costa_UsrClass.csv
Drag a column header here to group by that column
Enter text to search...
Find

Line	Tag	Bag Path	Slot	Node Slot	MRU Position	Absolute Path	Shell Type
1	□	BagMRU	0	0	0	Desktop\My Computer	Root folder: GUID
2	□	BagMRU	1	0	1	Desktop\Home Folder	Root folder: GUID
3	□	BagMRU\0	0	13	3	Desktop\My Computer\C:	Drive letter
4	□	BagMRU\0	1	8	0	Desktop\My Computer\Downloads	Root folder: GUID
5	□	BagMRU\0	2	9	2	Desktop\My Computer\Documents	Root folder: GUID
6	□	BagMRU\0	3	20	1	Desktop\My Computer\Desktop	Root folder: GUID
7	□	BagMRU\0\0	0	14	0	Desktop\My Computer\C:\Users	Directory
8	□	BagMRU\0\0\0	0	15	0	Desktop\My Computer\C:\Users\costa	Directory
9	□	BagMRU\0\0\0\0	0	1	0	Desktop\My Computer\C:\Users\costa\Downloads	Directory
10	□	BagMRU\0\0\0\0	1	16	2	Desktop\My Computer\C:\Users\costa\AppData	Directory
11	□	BagMRU\0\0\0\0	2	21	1	Desktop\My Computer\C:\Users\costa\Desktop	Directory
12	□	BagMRU\0\0\0\...	0	12	0	Desktop\My Computer\C:\Users\costa\Downloads\eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2dac6fdcc...	Directory
13	□	BagMRU\0\0\0\...	0	17	0	Desktop\My Computer\C:\Users\costa\AppData\Local	Directory
14	□	BagMRU\0\0\0\...	0	18	0	Desktop\My Computer\C:\Users\costa\AppData\Local\Temp	Directory
15	□	BagMRU\0\0\0\...	0	19	0	Desktop\My Computer\C:\Users\costa\AppData\Local\Temp\is-49560.tmp	Directory
16	□	BagMRU\0\1	0	2	3	Desktop\My Computer\Downloads\TimelineExplorer	Directory
17	□	BagMRU\0\1	1	4	0	Desktop\My Computer\Downloads\kape	Directory
18	□	BagMRU\0\1	2	6	1	Desktop\My Computer\Downloads\hayabusa-3.6.0-win-x64	Directory
19	□	BagMRU\0\1	3	11	2	Desktop\My Computer\Downloads\eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2dac6fdcc97368700efd7	Directory
20	□	BagMRU\0\1\0	0	3	0	Desktop\My Computer\Downloads\TimelineExplorer\TimelineExplorer	Directory
21	□	BagMRU\0\1\1	0	5	0	Desktop\My Computer\Downloads\kape\KAPE	Directory

Figure 21: BAG MRU

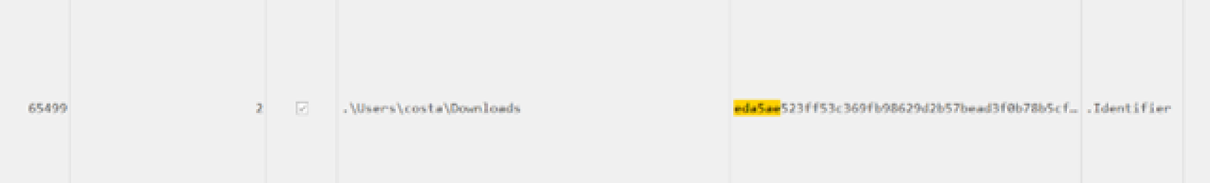
If I continue with the line I can find the MFT entry, this suggests that the malware was actually executed and entered the system processes table.



2025-10-19 08:01:18	111203	11	1	2025-10-19 08:01:18	2025-10-19 08:01:18

Figure 22: MFT entry

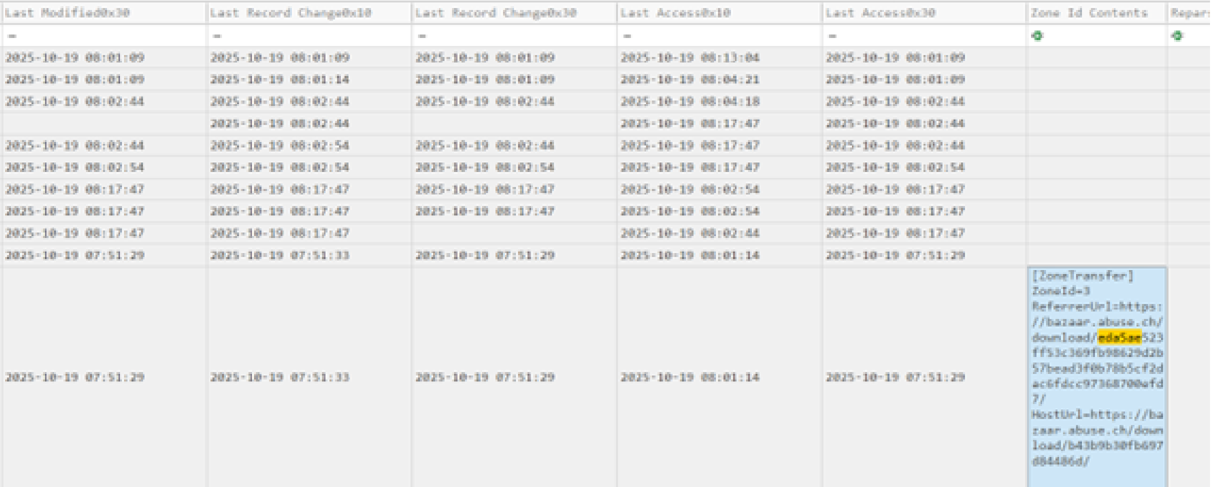
Now if I move to the .csv file related to the MFT and apply a filter looking for part of the malware hash



65499		2	☑	.\Users\costa\Downloads	eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf…	.Identifier

Figure 23: Identifier extension

you will notice an ".identifier" extension and if we continue down the line we can see where the file comes from:



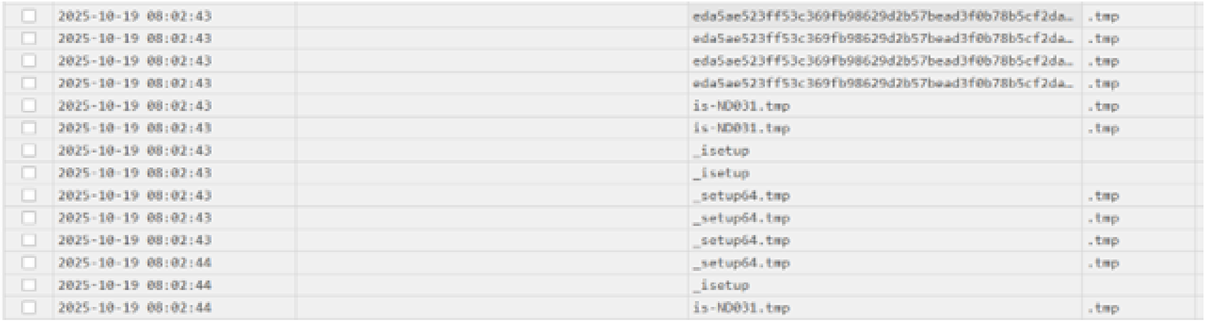
Last Modified0x30	Last Record Change0x10	Last Record Change0x30	Last Access0x10	Last Access0x30	Zone Id Contents	Repar
=	=	=	=	=	⚙	⚙
2025-10-19 08:01:09	2025-10-19 08:01:09	2025-10-19 08:01:09	2025-10-19 08:13:04	2025-10-19 08:01:09		
2025-10-19 08:01:09	2025-10-19 08:01:14	2025-10-19 08:01:09	2025-10-19 08:04:21	2025-10-19 08:01:09		
2025-10-19 08:02:44	2025-10-19 08:02:44	2025-10-19 08:02:44	2025-10-19 08:04:18	2025-10-19 08:02:44		
	2025-10-19 08:02:44		2025-10-19 08:17:47	2025-10-19 08:02:44		
2025-10-19 08:02:44	2025-10-19 08:02:54	2025-10-19 08:02:44	2025-10-19 08:17:47	2025-10-19 08:02:44		
2025-10-19 08:02:54	2025-10-19 08:02:54	2025-10-19 08:02:54	2025-10-19 08:17:47	2025-10-19 08:02:54		
2025-10-19 08:17:47	2025-10-19 08:17:47	2025-10-19 08:17:47	2025-10-19 08:02:54	2025-10-19 08:17:47		
2025-10-19 08:17:47	2025-10-19 08:17:47	2025-10-19 08:17:47	2025-10-19 08:02:54	2025-10-19 08:17:47		
2025-10-19 08:17:47	2025-10-19 08:17:47		2025-10-19 08:02:44	2025-10-19 08:17:47		
2025-10-19 07:51:29	2025-10-19 07:51:33	2025-10-19 07:51:29	2025-10-19 08:01:14	2025-10-19 07:51:29		
2025-10-19 07:51:29	2025-10-19 07:51:33	2025-10-19 07:51:29	2025-10-19 08:01:14	2025-10-19 07:51:29	[ZoneTransfer] ZoneId=3 ReferrerUrl=https://bazaar.abuse.ch/download/eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2dac6fdcc97368700afd7/ HostUrl=https://bazaar.abuse.ch/download/b43b9b30fb697d84486d/	

Figure 24: Malware origin

The time range should not be misleading, since the logs are in UTC, while the initial execution time was expressed in UTC+2.

Now analyzing the MFTECmd$J file,

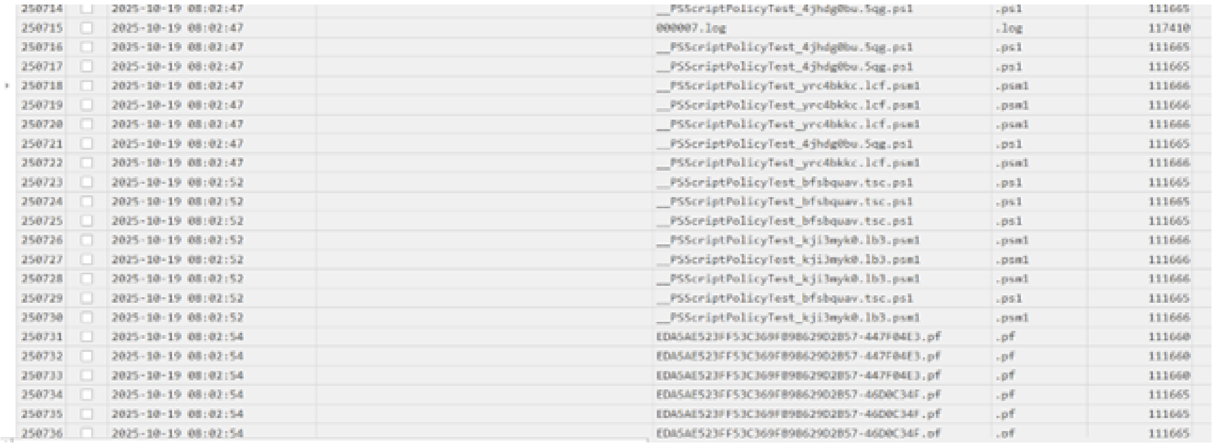
I can see the presence of some temporary files at the time the malware was running:



□	2025-10-19 08:02:43		eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp
□	2025-10-19 08:02:43		eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp
□	2025-10-19 08:02:43		eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp
□	2025-10-19 08:02:43		eda5ae523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp
□	2025-10-19 08:02:43		is-ND031.tmp	.tmp
□	2025-10-19 08:02:43		is-ND031.tmp	.tmp
□	2025-10-19 08:02:43		_isetup	
□	2025-10-19 08:02:43		_isetup	
□	2025-10-19 08:02:43		_setup64.tmp	.tmp
□	2025-10-19 08:02:43		_setup64.tmp	.tmp
□	2025-10-19 08:02:43		_setup64.tmp	.tmp
□	2025-10-19 08:02:44		_setup64.tmp	.tmp
□	2025-10-19 08:02:44		_isetup	
□	2025-10-19 08:02:44		is-ND031.tmp	.tmp

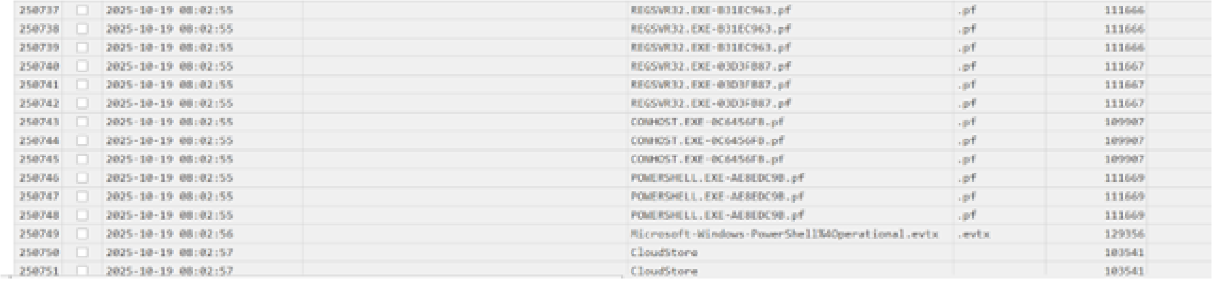
Figure 25: Temporary files

Policy changes can also be observed via scripts with the .psm1 extension, and registry prefetches (.pf) can be found. This suggests that the malware attempted to modify registries and policies, likely for persistence and to conceal its activities.



250714	□	2025-10-19 08:02:47		__PSScriptPolicyTest_4jhdg0bu.5qg.ps1	.ps1	111665
250715	□	2025-10-19 08:02:47		000007.log	.log	117410
250716	□	2025-10-19 08:02:47		__PSScriptPolicyTest_4jhdg0bu.5qg.ps1	.ps1	111665
250717	□	2025-10-19 08:02:47		__PSScriptPolicyTest_4jhdg0bu.5qg.ps1	.ps1	111665
250718	□	2025-10-19 08:02:47		__PSScriptPolicyTest_yrc4bkkc.lcf.psm1	.psm1	111666
250719	□	2025-10-19 08:02:47		__PSScriptPolicyTest_yrc4bkkc.lcf.psm1	.psm1	111666
250720	□	2025-10-19 08:02:47		__PSScriptPolicyTest_yrc4bkkc.lcf.psm1	.psm1	111666
250721	□	2025-10-19 08:02:47		__PSScriptPolicyTest_4jhdg0bu.5qg.ps1	.ps1	111665
250722	□	2025-10-19 08:02:47		__PSScriptPolicyTest_yrc4bkkc.lcf.psm1	.psm1	111666
250723	□	2025-10-19 08:02:52		__PSScriptPolicyTest_bfsbquav.tsc.ps1	.ps1	111665
250724	□	2025-10-19 08:02:52		__PSScriptPolicyTest_bfsbquav.tsc.ps1	.ps1	111665
250725	□	2025-10-19 08:02:52		__PSScriptPolicyTest_bfsbquav.tsc.ps1	.ps1	111665
250726	□	2025-10-19 08:02:52		__PSScriptPolicyTest_kji3myk0.1b3.psm1	.psm1	111666
250727	□	2025-10-19 08:02:52		__PSScriptPolicyTest_kji3myk0.1b3.psm1	.psm1	111666
250728	□	2025-10-19 08:02:52		__PSScriptPolicyTest_kji3myk0.1b3.psm1	.psm1	111666
250729	□	2025-10-19 08:02:52		__PSScriptPolicyTest_bfsbquav.tsc.ps1	.ps1	111665
250730	□	2025-10-19 08:02:52		__PSScriptPolicyTest_kji3myk0.1b3.psm1	.psm1	111666
250731	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660
250732	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660
250733	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660
250734	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-46D0C34F.pf	.pf	111665
250735	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-46D0C34F.pf	.pf	111665
250736	□	2025-10-19 08:02:54		EDA5AE523FF53C369FB98629D2B57-46D0C34F.pf	.pf	111665

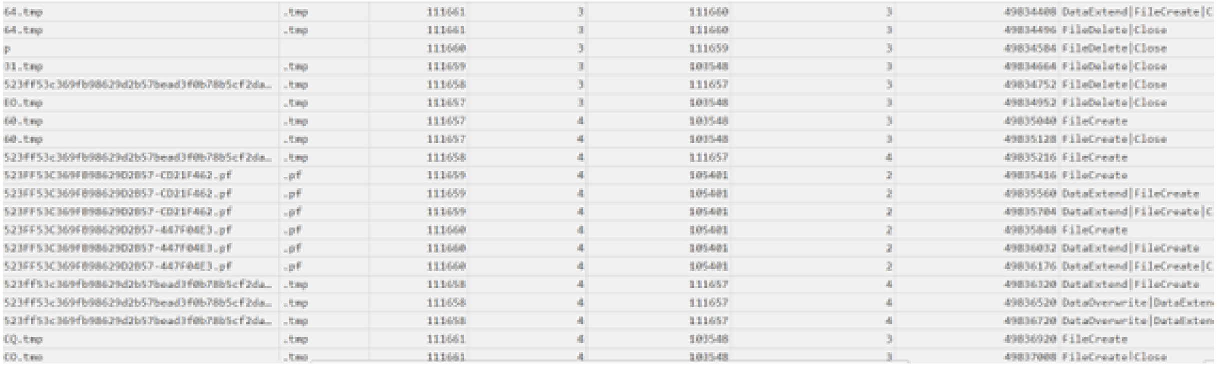
Figure 26: Script .psm1



250737	□	2025-10-19 08:02:55		REGSVR32.EXE-831EC963.pf	.pf	111666
250738	□	2025-10-19 08:02:55		REGSVR32.EXE-831EC963.pf	.pf	111666
250739	□	2025-10-19 08:02:55		REGSVR32.EXE-831EC963.pf	.pf	111666
250740	□	2025-10-19 08:02:55		REGSVR32.EXE-03D3F887.pf	.pf	111667
250741	□	2025-10-19 08:02:55		REGSVR32.EXE-03D3F887.pf	.pf	111667
250742	□	2025-10-19 08:02:55		REGSVR32.EXE-03D3F887.pf	.pf	111667
250743	□	2025-10-19 08:02:55		CONHOST.EXE-0C6456FB.pf	.pf	109907
250744	□	2025-10-19 08:02:55		CONHOST.EXE-0C6456FB.pf	.pf	109907
250745	□	2025-10-19 08:02:55		CONHOST.EXE-0C6456FB.pf	.pf	109907
250746	□	2025-10-19 08:02:55		POWERSHELL.EXE-AE8EDC9B.pf	.pf	111669
250747	□	2025-10-19 08:02:55		POWERSHELL.EXE-AE8EDC9B.pf	.pf	111669
250748	□	2025-10-19 08:02:55		POWERSHELL.EXE-AE8EDC9B.pf	.pf	111669
250749	□	2025-10-19 08:02:56		Microsoft-Windows-PowerShell%4Operational.evtx	.evtx	129356
250750	□	2025-10-19 08:02:57		CloudStore		103541
250751	□	2025-10-19 08:02:57		CloudStore		103541

Figure 27: Modified registries

Finally, you can notice some file modification actions:

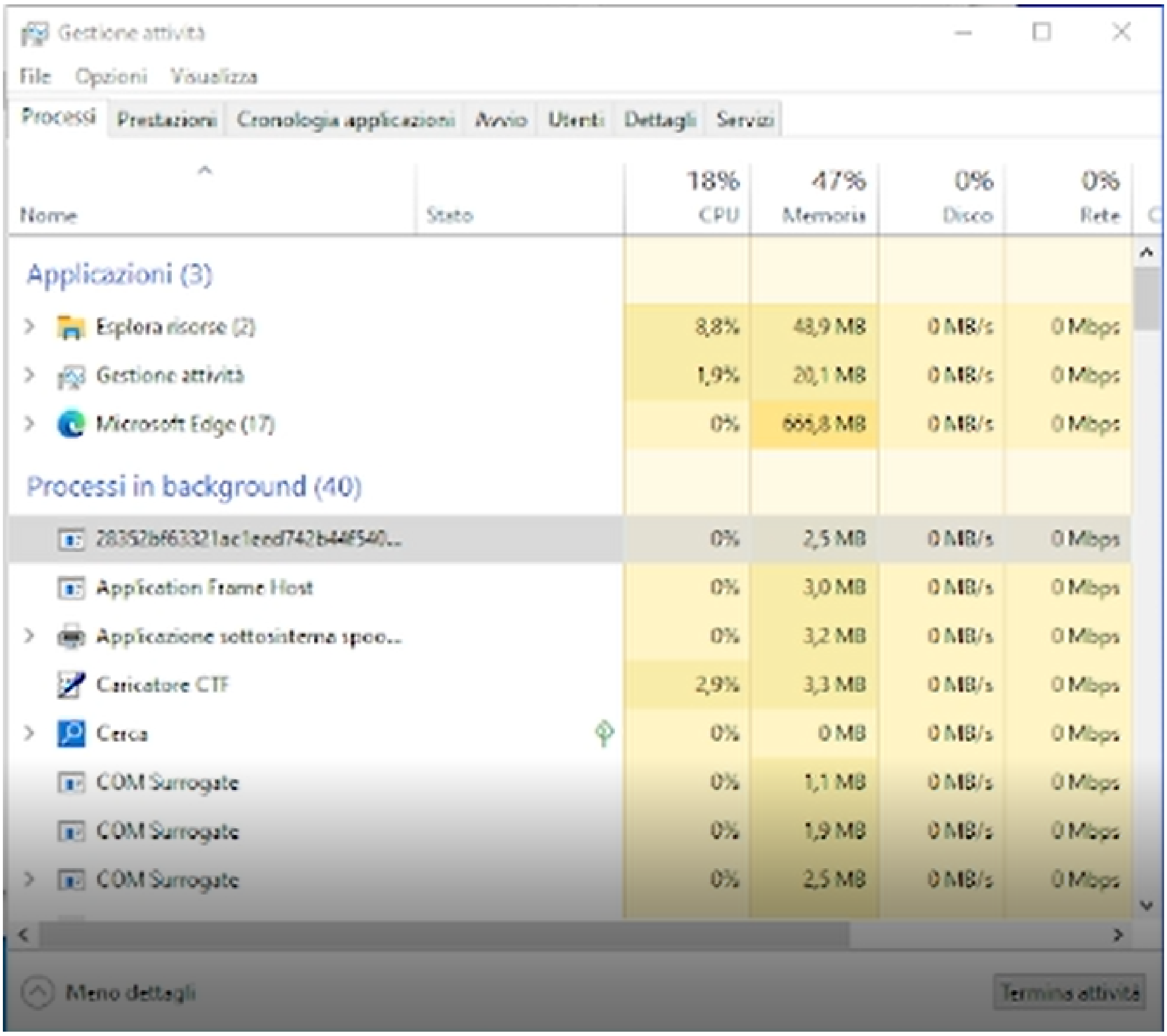


64.tmp	.tmp	111661	3	111660	3	49834408	DataExtend|FileCreate|C
64.tmp	.tmp	111661	3	111660	3	49834496	FileDelete|Close
p		111660	3	111659	3	49834584	FileDelete|Close
31.tmp	.tmp	111659	3	103548	3	49834664	FileDelete|Close
523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp	111658	3	111657	3	49834752	FileDelete|Close
EO.tmp	.tmp	111657	3	103548	3	49834952	FileDelete|Close
60.tmp	.tmp	111657	4	103548	3	49835040	FileCreate
60.tmp	.tmp	111657	4	103548	3	49835128	FileCreate|Close
523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp	111658	4	111657	4	49835216	FileCreate
523FF53C369FB98629D2B57-CD21F462.pf	.pf	111659	4	105401	2	49835416	FileCreate
523FF53C369FB98629D2B57-CD21F462.pf	.pf	111659	4	105401	2	49835560	DataExtend|FileCreate
523FF53C369FB98629D2B57-CD21F462.pf	.pf	111659	4	105401	2	49835704	DataExtend|FileCreate|C
523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660	4	105401	2	49835848	FileCreate
523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660	4	105401	2	49836032	DataExtend|FileCreate
523FF53C369FB98629D2B57-447F04E3.pf	.pf	111660	4	105401	2	49836176	DataExtend|FileCreate|C
523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp	111658	4	111657	4	49836320	DataExtend|FileCreate
523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp	111658	4	111657	4	49836520	DataOverwrite|DataExten
523ff53c369fb98629d2b57bead3f0b78b5cf2da…	.tmp	111658	4	111657	4	49836720	DataOverwrite|DataExten
CQ.tmp	.tmp	111661	4	103548	3	49836920	FileCreate
CO.tmp	.tmp	111661	4	103548	3	49837008	FileCreate|Close

Figure 28: File Activity

Based on the malware's characteristics, although the network connection was disabled in the VM settings to prevent it from spreading across the network and potentially leaking the virtualization environment, I would have expected to find logs of attempted connections to suspicious domains and C&C, but this did not happen.

Therefore, I tried to check the malware's activity in Task Manager. As you can see, the malware appears to be running, and over time, I noticed that its resources were being used consistently. Therefore, it's reasonable to assume that the malware requires a connection to run, or that the malware detected a virtual environment and stopped (also given the lack of connectivity).

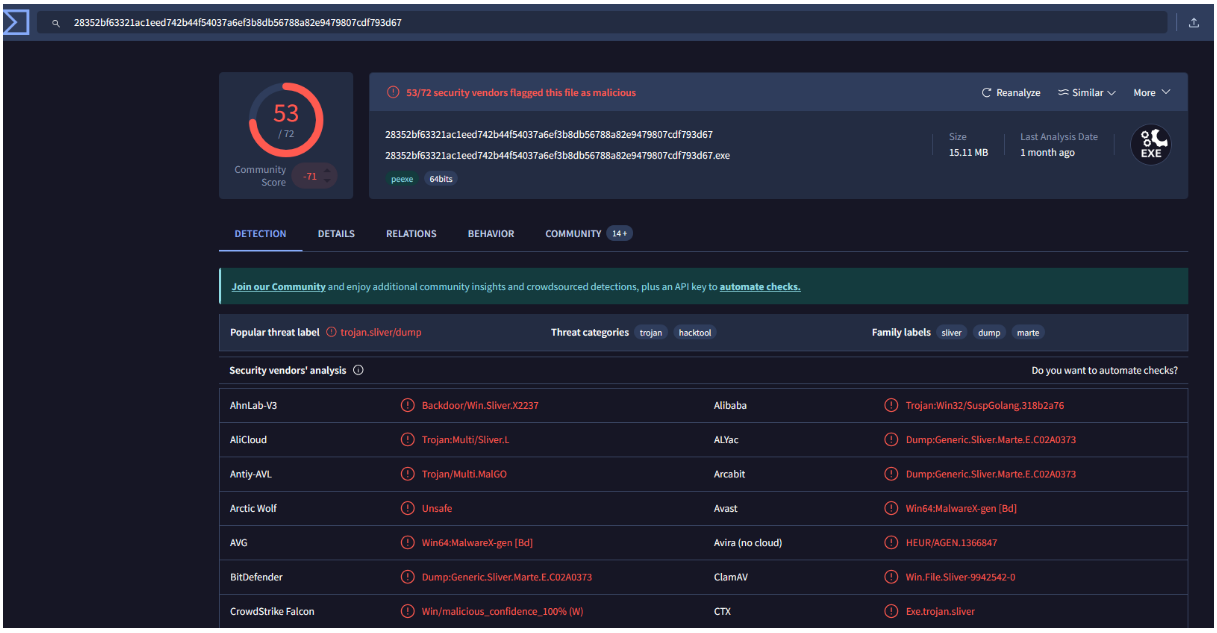


Gestione attività
File Opzioni Visualizza
Processi Prestazioni Cronologia applicazioni Avvio Utenti Dettagli Servizi
Meno dettagli
Termina attività

Nome	Stato	18% CPU	47% Memoria	0% Disco	0% Rete
Applicazioni (3)					
Esplora risorse (2)		8,8%	48,9 MB	0 MB/s	0 Mbps
Gestione attività		1,9%	20,1 MB	0 MB/s	0 Mbps
Microsoft Edge (17)		0%	666,8 MB	0 MB/s	0 Mbps
Processi in background (40)					
28352bf63321ac1eed742b44f540...		0%	2,5 MB	0 MB/s	0 Mbps
Application Frame Host		0%	3,0 MB	0 MB/s	0 Mbps
Applicazione sottosistema spoo...		0%	3,2 MB	0 MB/s	0 Mbps
Caricatore CTF		2,9%	3,3 MB	0 MB/s	0 Mbps
Cerca		0%	0 MB	0 MB/s	0 Mbps
COM Surrogate		0%	1,1 MB	0 MB/s	0 Mbps
COM Surrogate		0%	1,9 MB	0 MB/s	0 Mbps
COM Surrogate		0%	2,5 MB	0 MB/s	0 Mbps

Figure 29: Task Manager

I tried to create a new VM and tried, under the same conditions, to launch the Sliver malware



28352bf63321ac1eed742b44f54037a6ef3b8db56788a82e9479807cdf793d67

53 / 72

Community Score -71

53/72 security vendors flagged this file as malicious

28352bf63321ac1eed742b44f54037a6ef3b8db56788a82e9479807cdf793d67
28352bf63321ac1eed742b44f54037a6ef3b8db56788a82e9479807cdf793d67.exe

peexe 64bits

Size 15.11 MB

Last Analysis Date 1 month ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label trojan.sliver/dump

Threat categories trojan hacktool

Family labels sliver dump marte

Security vendors' analysis

Do you want to automate checks?

AhnLab-V3	Backdoor/Win.Sliver.X2237	Alibaba	Trojan:Win32/SuspGolang.318b2a76
AliCloud	Trojan:Multi/Sliver.L	ALYac	Dump:Generic.Sliver.Marte.E.C02A0373
Antiy-AVL	Trojan/Multi.MalGO	Arcabit	Dump:Generic.Sliver.Marte.E.C02A0373
Arctic Wolf	Unsafe	Avast	Win64:MalwareX-gen [Bd]
AVG	Win64:MalwareX-gen [Bd]	Avira (no cloud)	HEUR/AGEN.1366847
BitDefender	Dump:Generic.Sliver.Marte.E.C02A0373	ClamAV	Win.File.Sliver-9942542-0
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	CTX	Exe.trojan.sliver

Figure 30: Sliver malware reputation

But I noticed the same activity, and so the suspicions remained.

To make the forensic investigation more efficient for dissemination purposes, we decided to change our approach and leverage artificial intelligence to generate some Windows scripts that aren't malicious in intent but should raise awareness of the potential effects. The purpose of these simulations is to demonstrate the significant reduction in MTTR using the Triage-First, Deep-Dive-Second methodology, compared to a monolithic approach using tools like Encase or Autopsy.

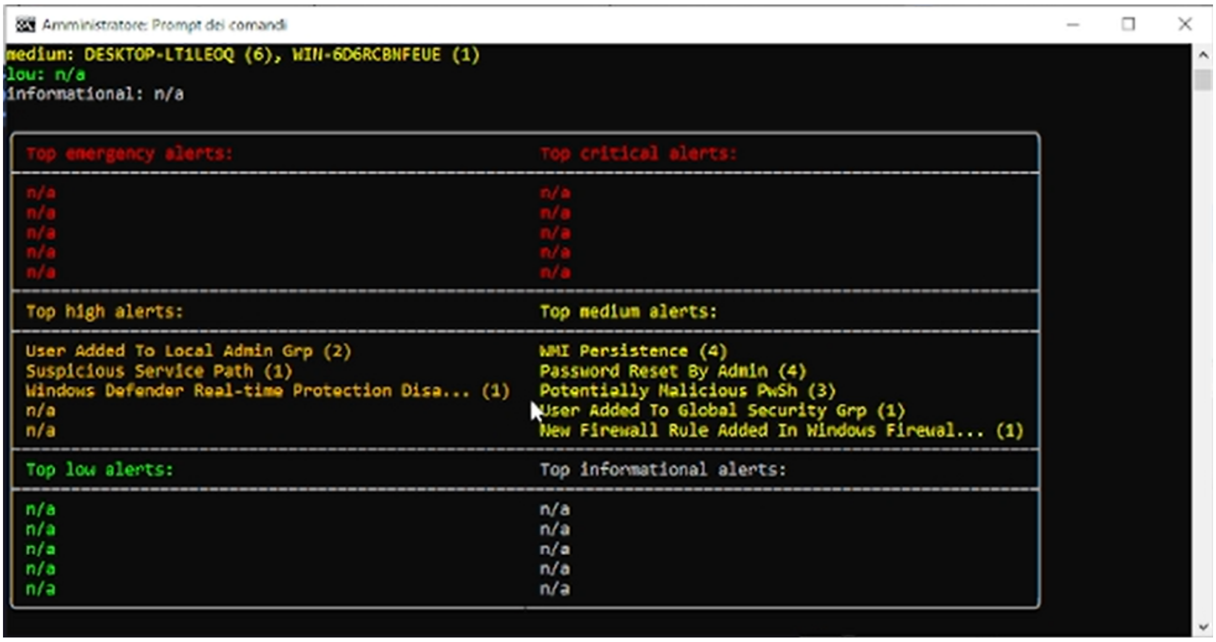
4.5 Script windows

As mentioned above, a .bat script has been generated using artificial intelligence to run in administrator mode that has the following features:

- Notepad++ installer download
- Opening Notepad (Blocco Note)
- Creating a "Test" user
- Creating a text file on the Desktop
- Creating a temporary file on the Desktop that will be immediately deleted and put in the Recycle Bin

The script was executed on October 19, 2025, at 4:29 PM. 10 minutes later, the Hayabusa tool was run, followed by Kape to begin the forensic analysis.

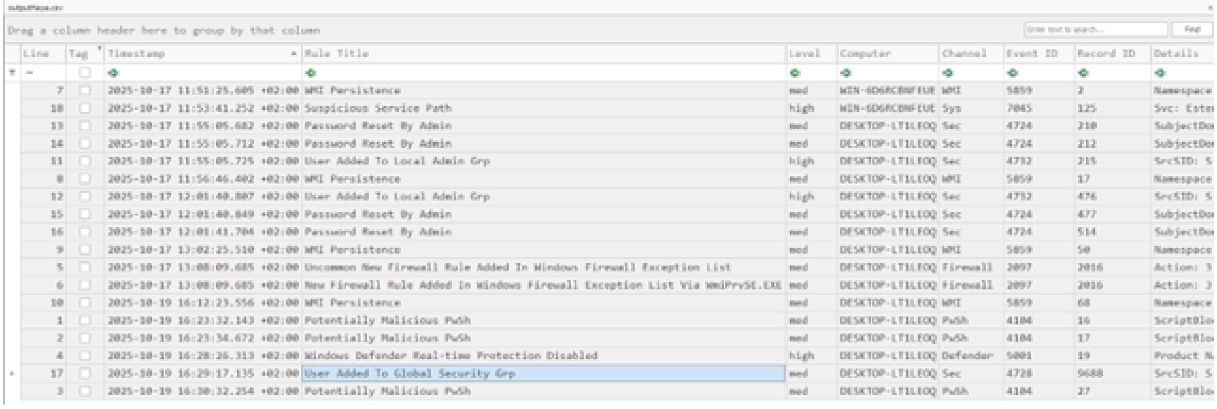
As before, Hayabusa returns similar results (still related to the VM creation and first login activity), but with a small difference regarding the suspicious scripts detected.



```
medium: DESKTOP-LT1LEOQ (6), WIN-6D6RCBNFEUE (1)
low: n/a
informational: n/a

Top emergency alerts:                          Top critical alerts:

n/a                                            n/a
n/a                                            n/a
n/a                                            n/a
n/a                                            n/a
n/a                                            n/a

Top high alerts:                               Top medium alerts:

User Added To Local Admin Grp (2)              WMI Persistence (4)
Suspicious Service Path (1)                    Password Reset By Admin (4)
Windows Defender Real-time Protection Disa... (1)  Potentially Malicious PwSh (3)
n/a                                            User Added To Global Security Grp (1)
n/a                                            New Firewall Rule Added In Windows Fireal... (1)

Top low alerts:                                Top informational alerts:

n/a                                            n/a
n/a                                            n/a
n/a                                            n/a
n/a                                            n/a
n/a                                            n/a
```

Figure 31: Output Hayabusa for the script

Analyzing Hayabusa's output via Timeline Explorer, you can see that a user was added to the Global group and potentially malicious PowerShell scripts were running at the same time I ran the .bat script:



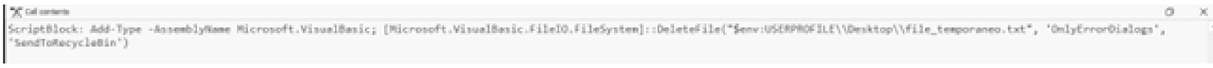
Line	Tag	Timestamp	Rule Title	Level	Computer	Channel	Event ID	Record ID	Details
7	□	2025-10-17 11:51:25.605 +02:00	WMI Persistence	med	WIN-6D6RCBNFEUE	WMI	5859	2	Namespace
18	□	2025-10-17 11:53:41.252 +02:00	Suspicious Service Path	high	WIN-6D6RCBNFEUE	Sys	7045	125	Svc: Exte
13	□	2025-10-17 11:55:05.682 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	210	SubjectDo
14	□	2025-10-17 11:55:05.712 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	212	SubjectDo
11	□	2025-10-17 11:55:05.725 +02:00	User Added To Local Admin Grp	high	DESKTOP-LT1LEOQ	Sec	4732	215	SrcSID: S
8	□	2025-10-17 11:56:46.402 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	17	Namespace
12	□	2025-10-17 12:01:40.807 +02:00	User Added To Local Admin Grp	high	DESKTOP-LT1LEOQ	Sec	4732	476	SrcSID: S
15	□	2025-10-17 12:01:40.849 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	477	SubjectDo
16	□	2025-10-17 12:01:41.704 +02:00	Password Reset By Admin	med	DESKTOP-LT1LEOQ	Sec	4724	514	SubjectDo
9	□	2025-10-17 13:02:25.510 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	50	Namespace
5	□	2025-10-17 13:08:09.685 +02:00	Uncommon New Firewall Rule Added In Windows Firewall Exception List	med	DESKTOP-LT1LEOQ	Firewall	2097	2016	Action: 3
6	□	2025-10-17 13:08:09.685 +02:00	New Firewall Rule Added In Windows Firewall Exception List Via WmiPrvSE.EXE	med	DESKTOP-LT1LEOQ	Firewall	2097	2016	Action: 3
10	□	2025-10-19 16:12:23.556 +02:00	WMI Persistence	med	DESKTOP-LT1LEOQ	WMI	5859	68	Namespace
1	□	2025-10-19 16:23:32.143 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	16	ScriptBlo
2	□	2025-10-19 16:23:34.672 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	17	ScriptBlo
4	□	2025-10-19 16:28:26.313 +02:00	Windows Defender Real-time Protection Disabled	high	DESKTOP-LT1LEOQ	Defender	5001	19	Product N
17	□	2025-10-19 16:29:17.135 +02:00	User Added To Global Security Grp	med	DESKTOP-LT1LEOQ	Sec	4728	9688	SrcSID: S
3	□	2025-10-19 16:30:32.254 +02:00	Potentially Malicious PwSh	med	DESKTOP-LT1LEOQ	PwSh	4104	27	ScriptBlo

Figure 32: Output Hayabusa

By analyzing the logs, you can see the log for the created user:

User: SrcSID: S-1-5-21-3620393304-754872592-685525921-1002 ¦ TgtGrp: NONE ¦ LID: 0x49c8c

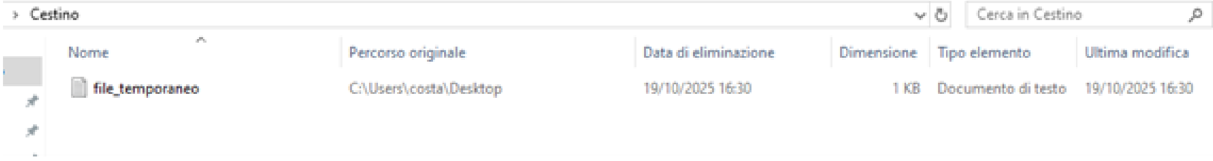
The logs also suggest that a file named file_temporaneo.txt was deleted from the desktop environment at 4:30 PM.



```
ScriptBlock: Add-Type -AssemblyName Microsoft.VisualBasic; [Microsoft.VisualBasic.FileIO.FileSystem]::DeleteFile("$env:USERPROFILE\\Desktop\\file_temporaneo.txt", 'OnlyErrorDialogs',
'SendToRecycleBin')
```

Figure 33: Temporary file

Actually looking at the trash it would appear that there is a deleted file at this timestamp:



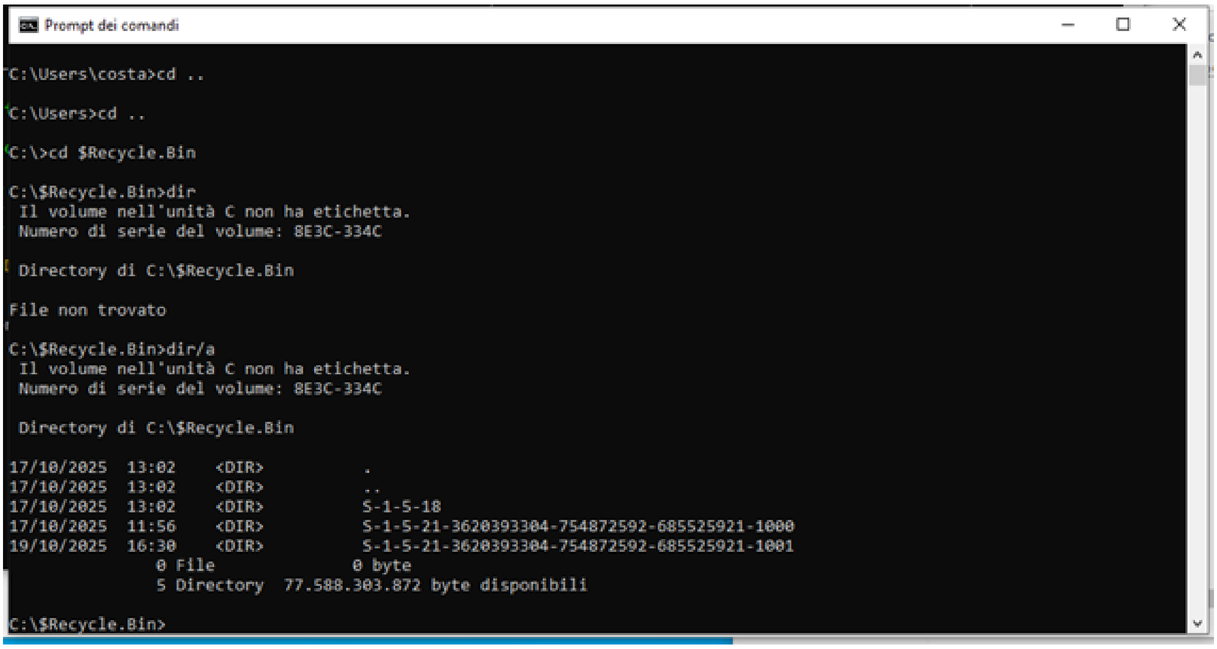
› Cestino

Cerca in Cestino

Nome	Percorso originale	Data di eliminazione	Dimensione	Tipo elemento	Ultima modifica
file_temporaneo	C:\Users\costa\Desktop	19/10/2025 16:30	1 KB	Documento di testo	19/10/2025 16:30

Figure 34: Bin

Let's investigate the Recycle Bin events from the Command Prompt

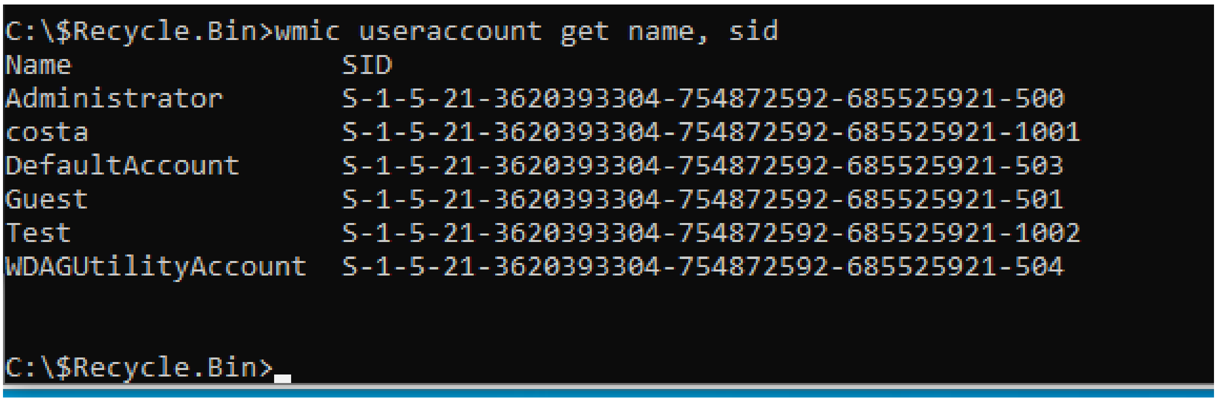


Prompt dei comandi

```
C:\Users\costa>cd ..

C:\Users>cd ..

C:\>cd $Recycle.Bin

C:\$Recycle.Bin>dir
 Il volume nell'unità C non ha etichetta.
 Numero di serie del volume: 8E3C-334C

 Directory di C:\$Recycle.Bin

File non trovato

C:\$Recycle.Bin>dir/a
 Il volume nell'unità C non ha etichetta.
 Numero di serie del volume: 8E3C-334C

 Directory di C:\$Recycle.Bin

17/10/2025  13:02    <DIR>          .
17/10/2025  13:02    <DIR>          ..
17/10/2025  13:02    <DIR>          S-1-5-18
17/10/2025  11:56    <DIR>          S-1-5-21-3620393304-754872592-685525921-1000
19/10/2025  16:30    <DIR>          S-1-5-21-3620393304-754872592-685525921-1001
               0 File              0 byte
               5 Directory  77.588.303.872 byte disponibili

C:\$Recycle.Bin>
```

Figure 35: CMD Bin

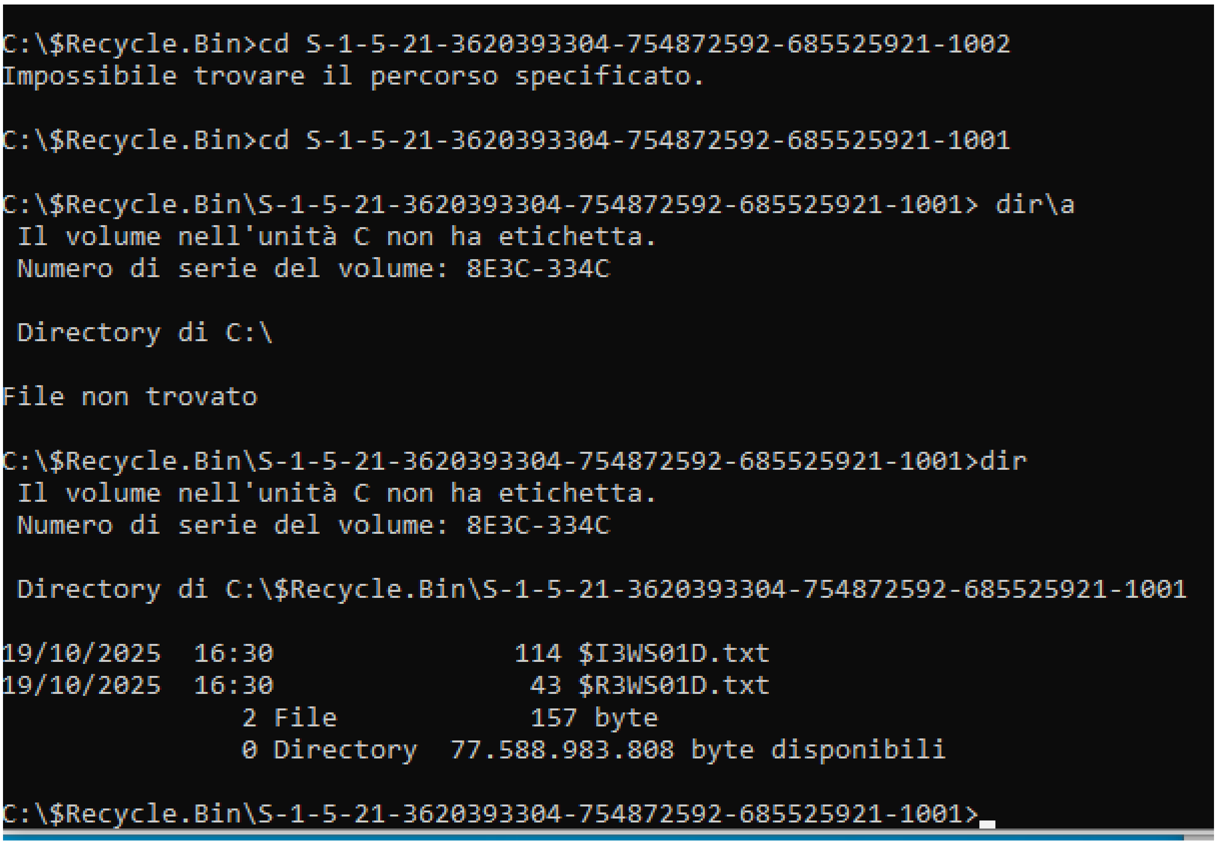
It looks like we have 2 users (and that seems to be in line with what hayabusa told us)



```
C:\$Recycle.Bin>wmic useraccount get name, sid
Name                  SID
Administrator         S-1-5-21-3620393304-754872592-685525921-500
costa                 S-1-5-21-3620393304-754872592-685525921-1001
DefaultAccount        S-1-5-21-3620393304-754872592-685525921-503
Guest                 S-1-5-21-3620393304-754872592-685525921-501
Test                  S-1-5-21-3620393304-754872592-685525921-1002
WDAGUtilityAccount    S-1-5-21-3620393304-754872592-685525921-504


C:\$Recycle.Bin>
```

Figure 36: User SID

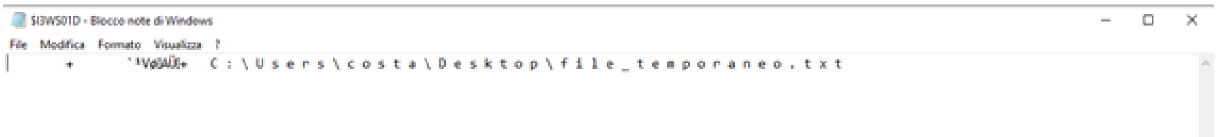
Looking at the users and their associated SIDs, we discovered that in addition to the standard users (Administrator, DefaultAccount, Guest, WDAGUtilityAccount) generated by Windows and the user created on the VM (coast), we have a new user called "Test," with its associated SID. There doesn't seem to be a real connection to the created user (a separate instance), so I can't find the path (it's likely, as we'll find out later, that the user was created but no login was performed). However, with my user's SID, an entry appears to have been deleted at 4:30 PM:



```
C:\$Recycle.Bin>cd S-1-5-21-3620393304-754872592-685525921-1002
Impossibile trovare il percorso specificato.

C:\$Recycle.Bin>cd S-1-5-21-3620393304-754872592-685525921-1001

C:\$Recycle.Bin\S-1-5-21-3620393304-754872592-685525921-1001> dir\a
 Il volume nell'unità C non ha etichetta.
 Numero di serie del volume: 8E3C-334C

 Directory di C:\

File non trovato

C:\$Recycle.Bin\S-1-5-21-3620393304-754872592-685525921-1001>dir
 Il volume nell'unità C non ha etichetta.
 Numero di serie del volume: 8E3C-334C

 Directory di C:\$Recycle.Bin\S-1-5-21-3620393304-754872592-685525921-1001

19/10/2025  16:30               114 $I3WS01D.txt
19/10/2025  16:30                43 $R3WS01D.txt
               2 File            157 byte
               0 Directory  77.588.983.808 byte disponibili

C:\$Recycle.Bin\S-1-5-21-3620393304-754872592-685525921-1001>
```

Figure 37: Items in the bin

By opening the metadata file I can see the file name and the original path before it was deleted:

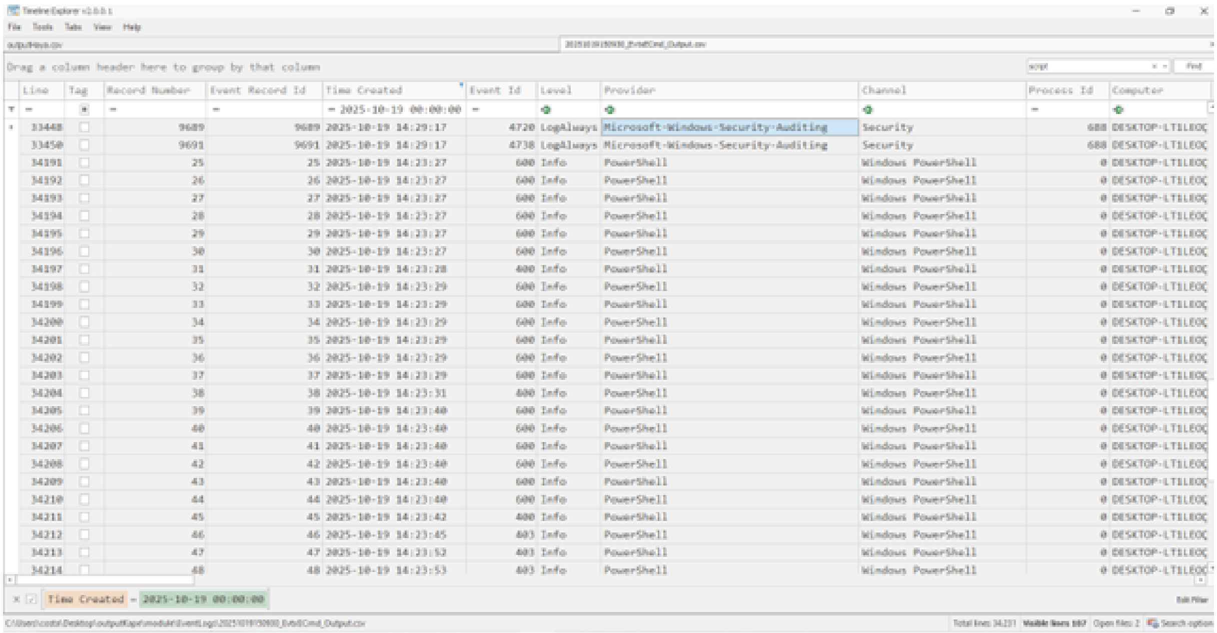


$I3WS01D - Blocco note di Windows
File Modifica Formato Visualizza ?

```
     +     ¹³VøÅÛ+   C:\Users\costa\Desktop\file_temporaneo.txt
```

Figure 38: Metadata

By investigating the EvtxECmd file, you can see the time the user was created: 2025-10-19 14:29:17 UTC:

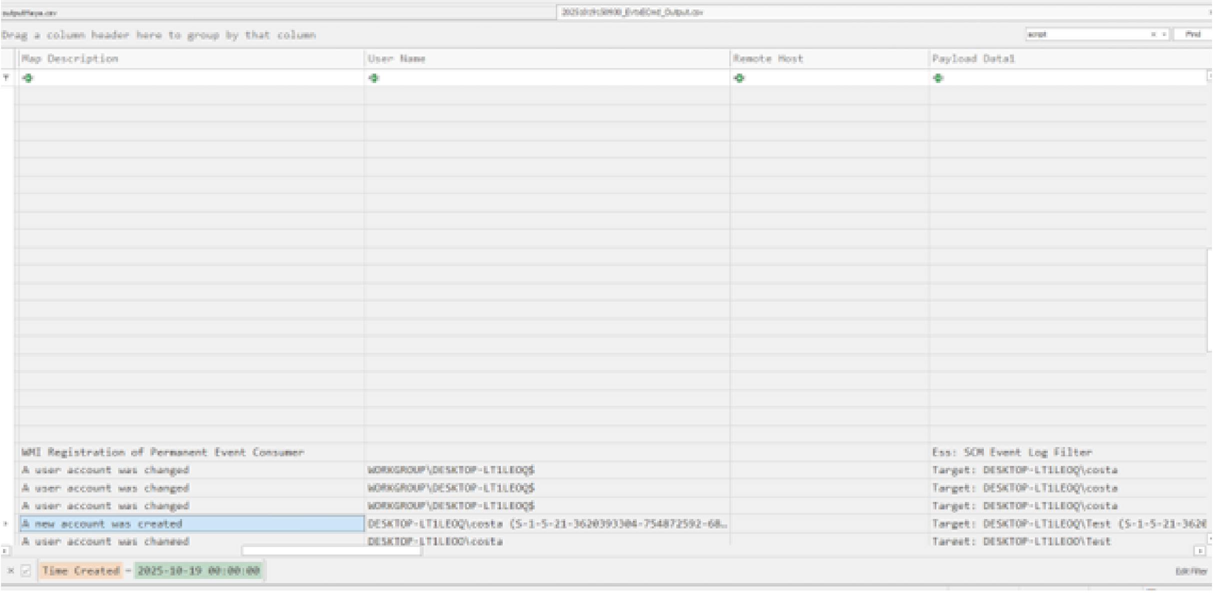


Drag a column header here to group by that column

Line	Tag	Record Number	Event Record Id	Time Created	Event Id	Level	Provider	Channel	Process Id	Computer
33448	□	9689	9689	2025-10-19 14:29:17	4720	LogAlways	Microsoft-Windows-Security-Auditing	Security	688	DESKTOP-LT1LEOQ
33450	□	9691	9691	2025-10-19 14:29:17	4738	LogAlways	Microsoft-Windows-Security-Auditing	Security	688	DESKTOP-LT1LEOQ
34191	□	25	25	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34192	□	26	26	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34193	□	27	27	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34194	□	28	28	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34195	□	29	29	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34196	□	30	30	2025-10-19 14:23:27	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34197	□	31	31	2025-10-19 14:23:28	400	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34198	□	32	32	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34199	□	33	33	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34200	□	34	34	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34201	□	35	35	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34202	□	36	36	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34203	□	37	37	2025-10-19 14:23:29	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34204	□	38	38	2025-10-19 14:23:31	400	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34205	□	39	39	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34206	□	40	40	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34207	□	41	41	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34208	□	42	42	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34209	□	43	43	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34210	□	44	44	2025-10-19 14:23:40	600	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34211	□	45	45	2025-10-19 14:23:42	400	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34212	□	46	46	2025-10-19 14:23:45	403	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34213	□	47	47	2025-10-19 14:23:52	403	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ
34214	□	48	48	2025-10-19 14:23:53	403	Info	PowerShell	Windows PowerShell	0	DESKTOP-LT1LEOQ

Time Created = 2025-10-19 00:00:00

Figure 39: User creation log



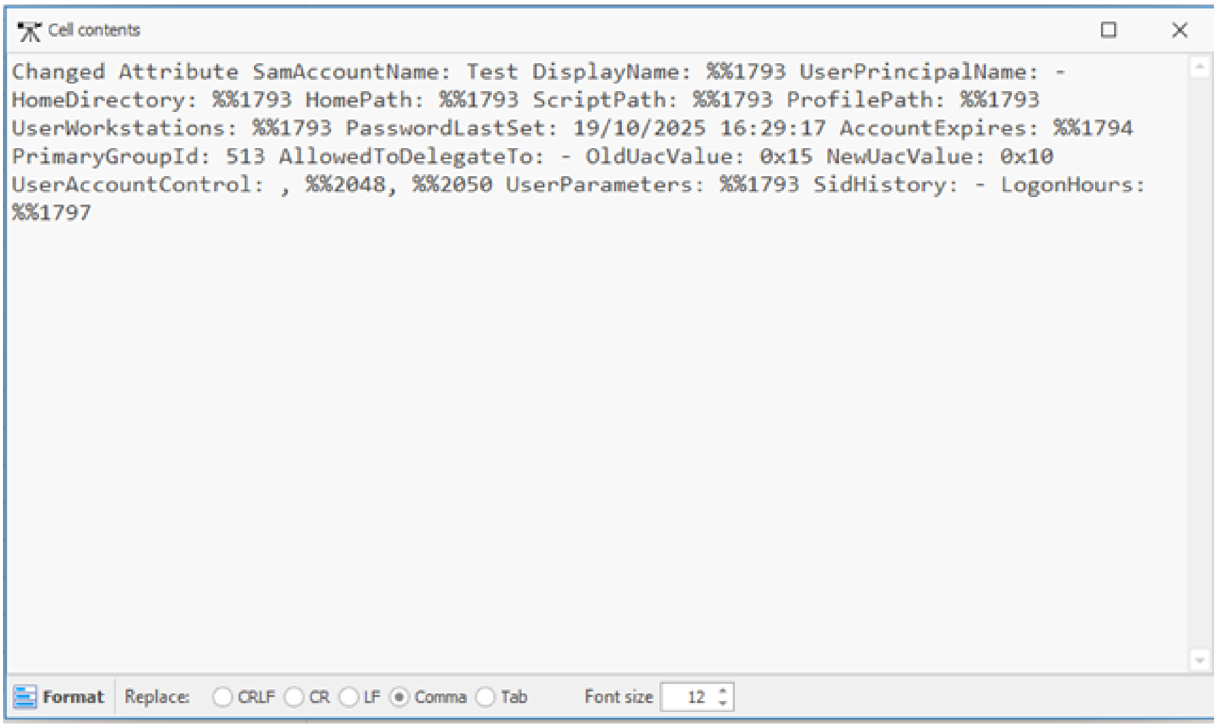
Drag a column header here to group by that column

Map Description	User Name	Remote Host	Payload Data1
WMI Registration of Permanent Event Consumer			Ess: SCM Event Log Filter
A user account was changed	WORKGROUP\DESKTOP-LT1LEOQ$		Target: DESKTOP-LT1LEOQ\costa
A user account was changed	WORKGROUP\DESKTOP-LT1LEOQ$		Target: DESKTOP-LT1LEOQ\costa
A user account was changed	WORKGROUP\DESKTOP-LT1LEOQ$		Target: DESKTOP-LT1LEOQ\costa
A new account was created	DESKTOP-LT1LEOQ\costa (S-1-5-21-3620393304-754872592-68…		Target: DESKTOP-LT1LEOQ\Test (S-1-5-21-362
A user account was changed	DESKTOP-LT1LEOQ\costa		Target: DESKTOP-LT1LEOQ\Test

Time Created = 2025-10-19 00:00:00

Figure 40: User log

Expanding the Payload Data2 field:

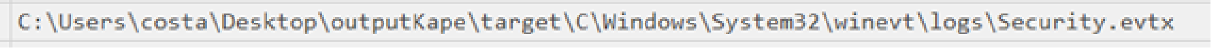


```
Changed Attribute SamAccountName: Test DisplayName: %%1793 UserPrincipalName: -
HomeDirectory: %%1793 HomePath: %%1793 ScriptPath: %%1793 ProfilePath: %%1793
UserWorkstations: %%1793 PasswordLastSet: 19/10/2025 16:29:17 AccountExpires: %%1794
PrimaryGroupId: 513 AllowedToDelegateTo: - OldUacValue: 0x15 NewUacValue: 0x10
UserAccountControl: , %%2048, %%2050 UserParameters: %%1793 SidHistory: - LogonHours:
%%1797
```

Figure 41: Payload Data2

We note that the "Test" user created is not a critical user with Administrator characteristics, but by examining the OldUacValue value, we discover that 0x10 corresponds to a normal user (and not a system or machine user) and NewUacValue 0x15: Normal user account (enabled) that requires a home directory and runs a logon script, suggesting that the user is now active.

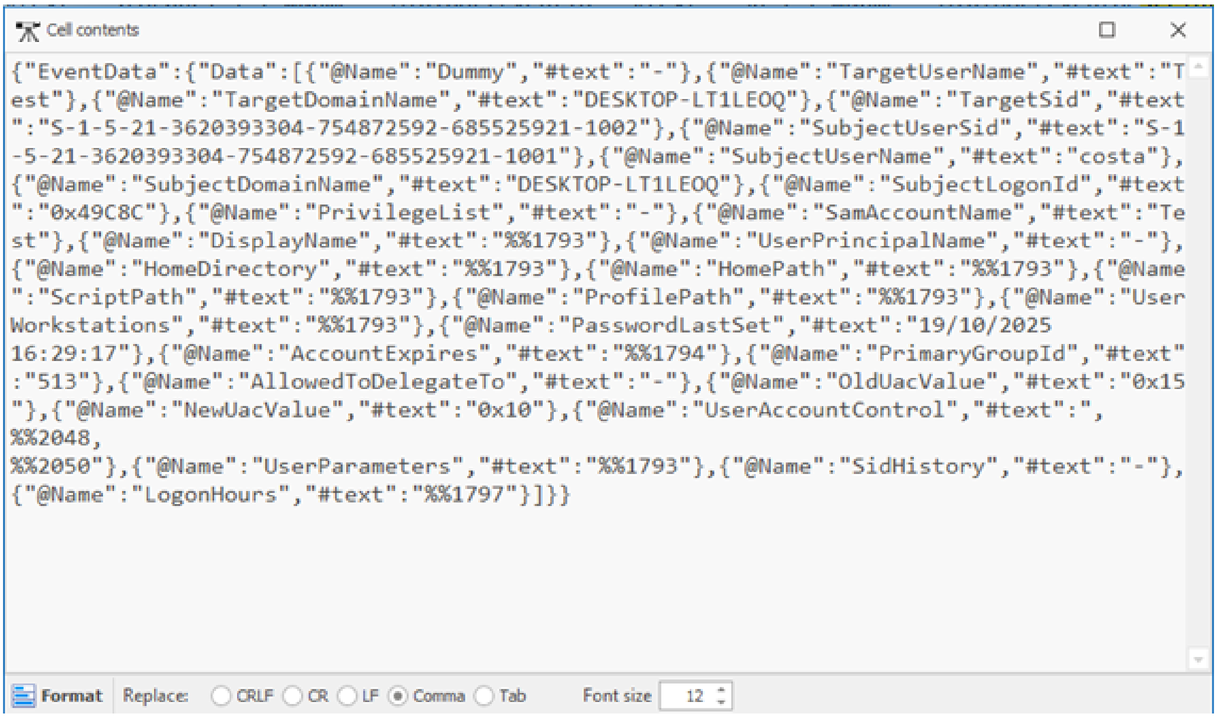
This type of event (being user activity) is found in the Security.evtx artifacts. . .



```
C:\Users\costa\Desktop\outputKape\target\C\Windows\System32\winevt\logs\Security.evtx
```

Figure 42: Security log

Payload:

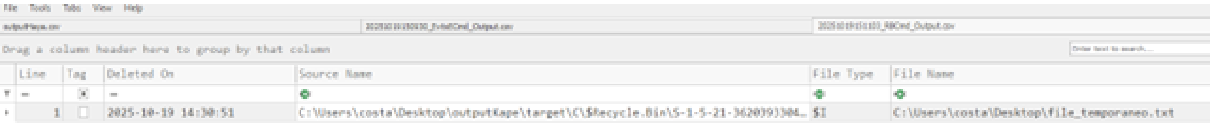


```
{"EventData":{"Data":[{"@Name":"Dummy","#text":"-"},{"@Name":"TargetUserName","#text":"T
est"},{"@Name":"TargetDomainName","#text":"DESKTOP-LT1LEOQ"},{"@Name":"TargetSid","#text
":"S-1-5-21-3620393304-754872592-685525921-1002"},{"@Name":"SubjectUserSid","#text":"S-1
-5-21-3620393304-754872592-685525921-1001"},{"@Name":"SubjectUserName","#text":"costa"},
{"@Name":"SubjectDomainName","#text":"DESKTOP-LT1LEOQ"},{"@Name":"SubjectLogonId","#text
":"0x49C8C"},{"@Name":"PrivilegeList","#text":"-"},{"@Name":"SamAccountName","#text":"Te
st"},{"@Name":"DisplayName","#text":"%%1793"},{"@Name":"UserPrincipalName","#text":"-"},
{"@Name":"HomeDirectory","#text":"%%1793"},{"@Name":"HomePath","#text":"%%1793"},{"@Name
":"ScriptPath","#text":"%%1793"},{"@Name":"ProfilePath","#text":"%%1793"},{"@Name":"User
Workstations","#text":"%%1793"},{"@Name":"PasswordLastSet","#text":"19/10/2025
16:29:17"},{"@Name":"AccountExpires","#text":"%%1794"},{"@Name":"PrimaryGroupId","#text"
:"513"},{"@Name":"AllowedToDelegateTo","#text":"-"},{"@Name":"OldUacValue","#text":"0x15
"},{"@Name":"NewUacValue","#text":"0x10"},{"@Name":"UserAccountControl","#text":",
%%2048,
%%2050"},{"@Name":"UserParameters","#text":"%%1793"},{"@Name":"SidHistory","#text":"-"},
{"@Name":"LogonHours","#text":"%%1797"}]}}
```

Figure 43: User creation payload

From the previous image, it is clear that the "costa" account on the "DESKTOP-LT1LE0Q" device has created the global account "Test".

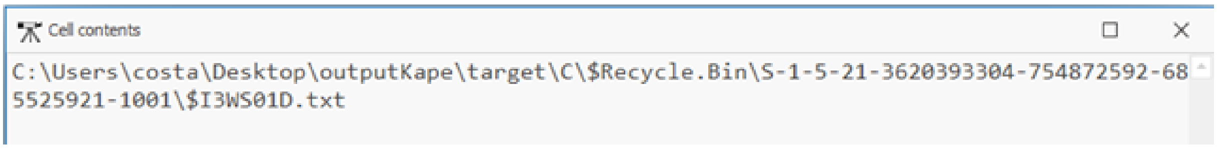
Now examining the RBCmd file, the file in the Recycle Bin whose information we found earlier is confirmed:



File Tools Tabs View Help
Drag a column header here to group by that column

Line	Tag	Deleted On	Source Name	File Type	File Name
1	□	2025-10-19 14:30:51	C:\Users\costa\Desktop\outputKape\target\C\$Recycle.Bin\S-1-5-21-3620393304…	$I	C:\Users\costa\Desktop\file_temporaneo.txt

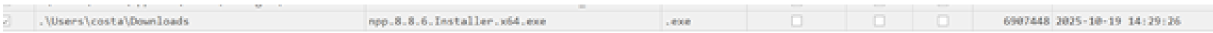
Figure 44: Confirmation of the bin



```
C:\Users\costa\Desktop\outputKape\target\C\$Recycle.Bin\S-1-5-21-3620393304-754872592-68
5525921-1001\$I3WS01D.txt
```

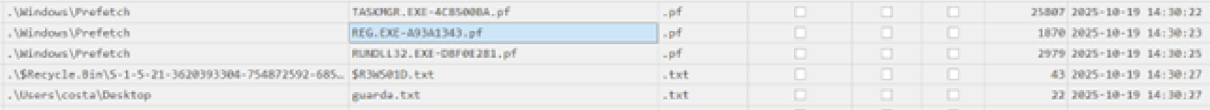
Figure 45: Log bin

Moving on to investigate the MFT file I noticed that there is an entry regarding an installer related to the npp application:



```
.\Users\costa\Downloads    npp.8.8.6.Installer.x64.exe    .exe    □    □    □    6907448 2025-10-19 14:29:26
```

Figure 46: MFT

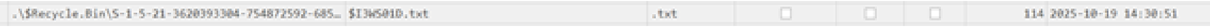


.\Windows\Prefetch	TASKMGR.EXE-4C8500BA.pf	.pf	□	□	□	25807	2025-10-19 14:30:22
.\Windows\Prefetch	REG.EXE-A93A1343.pf	.pf	□	□	□	1870	2025-10-19 14:30:23
.\Windows\Prefetch	RUNDLL32.EXE-D8F0E281.pf	.pf	□	□	□	2979	2025-10-19 14:30:25
.\$Recycle.Bin\S-1-5-21-3620393304-754872592-685…	$R3WS01D.txt	.txt	□	□	□	43	2025-10-19 14:30:27
.\Users\costa\Desktop	guarda.txt	.txt	□	□	□	22	2025-10-19 14:30:27

Figure 47: MFT

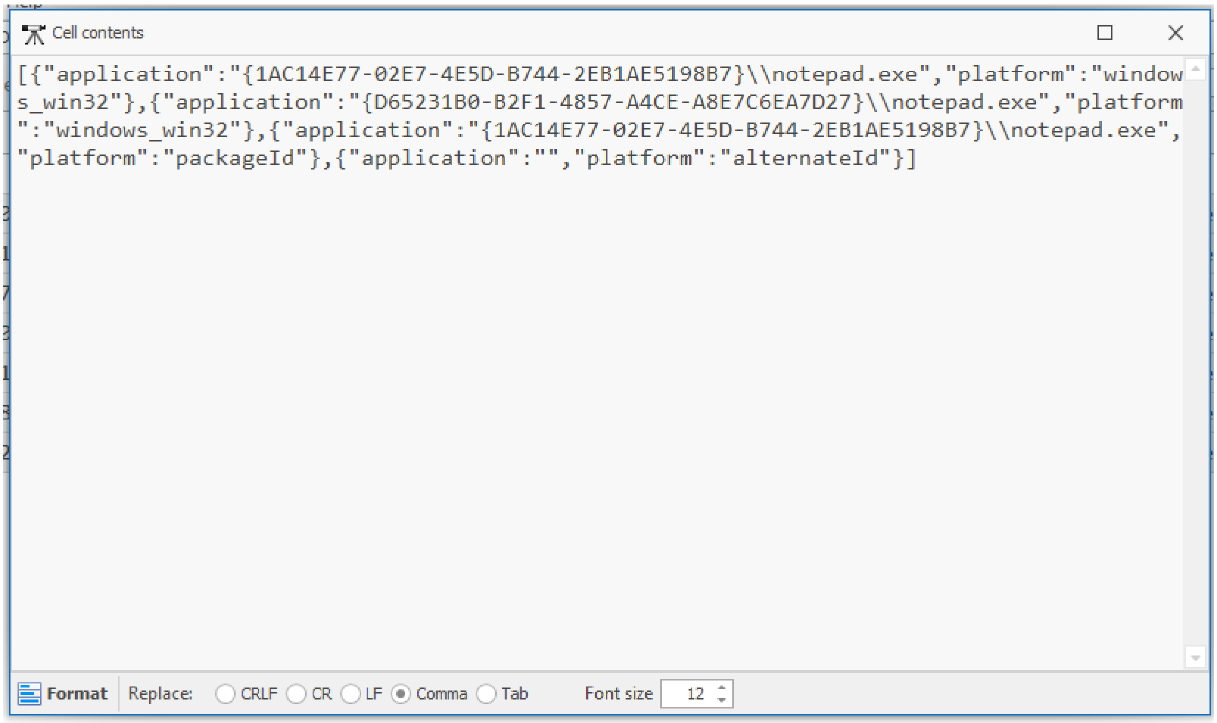
Then the download of Notepad++ is confirmed (by loading the executable on Total Virus, it's possible to trace the file type and other information from the OSINT sources... in our case, we find that the hash associated with that installer matches the Notepad++ application). We also note that some type of registry activity has been performed (probably activity on the users), and the file's appearance in the Recycle Bin and on the Desktop is confirmed.

A few moments later, the metadata file was detected in the Recycle Bin:



.\$Recycle.Bin\S-1-5-21-3620393304-754872592-685…	$I3WS01D.txt	.txt	□	□	□	114	2025-10-19 14:30:51

Figure 48: Bin metadata

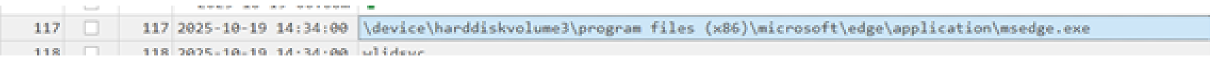


```
[{"application":"{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\\notepad.exe","platform":"windows_win32"},{"application":"{D65231B0-B2F1-4857-A4CE-A8E7C6EA7D27}\\notepad.exe","platform":"windows_win32"},{"application":"{1AC14E77-02E7-4E5D-B744-2EB1AE5198B7}\\notepad.exe","platform":"packageId"},{"application":"","platform":"alternateId"}]
```

Format Replace: ○ CRLF ○ CR ○ LF ● Comma ○ Tab Font size 12

Figure 49: Notepad metadata log

By looking at the network consumption (SrumECmd_NetworkUsages) around the script at most it is possible to detect the use of edge



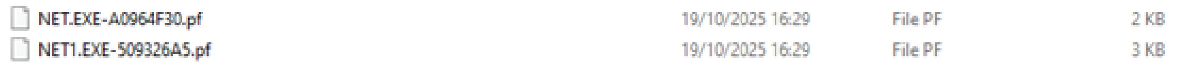
117	□	117	2025-10-19 14:34:00	\device\harddiskvolume3\program files (x86)\microsoft\edge\application\msedge.exe
118	□	118	2025-10-19 14:34:00	wlidsvc

Figure 50: Network consumption

But otherwise, no other noteworthy activity is evident.

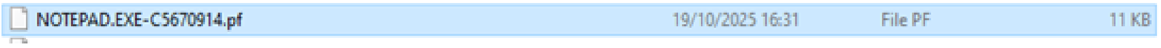
A look at the Prefetch folder reveals the use of edge. Considering that the script was executed at 4:29 PM and, as we've seen, there were two searches on edge, what we saw from the prefetch folder is consistent.

The following, however, is consistent with Notepad++ download activity:



NET.EXE-A0964F30.pf	19/10/2025 16:29	File PF	2 KB
NET1.EXE-509326A5.pf	19/10/2025 16:29	File PF	3 KB

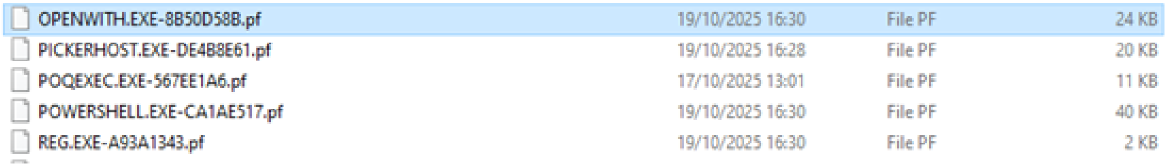
Figure 51: Net notepad



NOTEPAD.EXE-C5670914.pf	19/10/2025 16:31	File PF	11 KB

Figure 52: Notepad executable highlight

Further evidence of Edge usage and changes to user and file settings on the desktop can be found here in the prefetch folder:

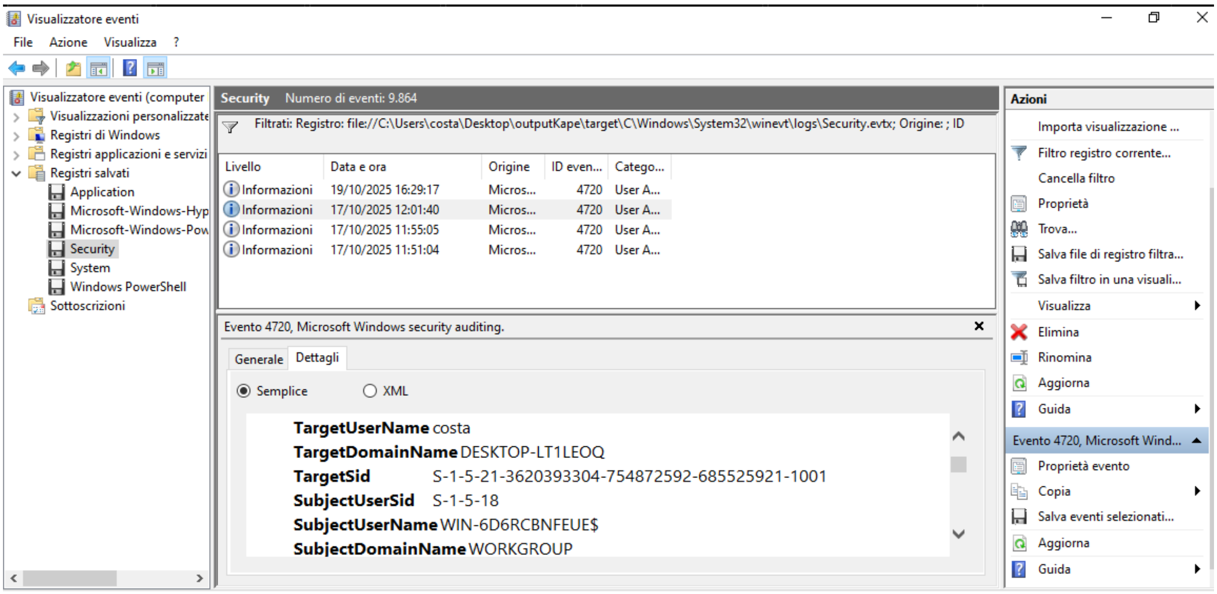


OPENWITH.EXE-8B50D58B.pf	19/10/2025 16:30	File PF	24 KB
PICKERHOST.EXE-DE4B8E61.pf	19/10/2025 16:28	File PF	20 KB
POQEXEC.EXE-567EE1A6.pf	17/10/2025 13:01	File PF	11 KB
POWERSHELL.EXE-CA1AE517.pf	19/10/2025 16:30	File PF	40 KB
REG.EXE-A93A1343.pf	19/10/2025 16:30	File PF	2 KB

Figure 53: User changes

While running the script, we were asked which browser to use to perform a task (knowing the script, we had reached the step where browser searches needed to be performed), since the VM had been recently created, and therefore the presence of Openwith is normal.

Moving now to the Windows Event Viewer, a further analysis was performed to understand the effects of the "Test" user created. Viewing the Windows Security events and filtering by event ID 4720 (user account creation), we noticed some activity on October 17 (the day I created and configured the VM).



Security Numero di eventi: 9.864

Filtrati: Registro: file://C:\Users\costa\Desktop\outputKape\target\C\Windows\System32\winevt\logs\Security.evtx; Origine: ; ID

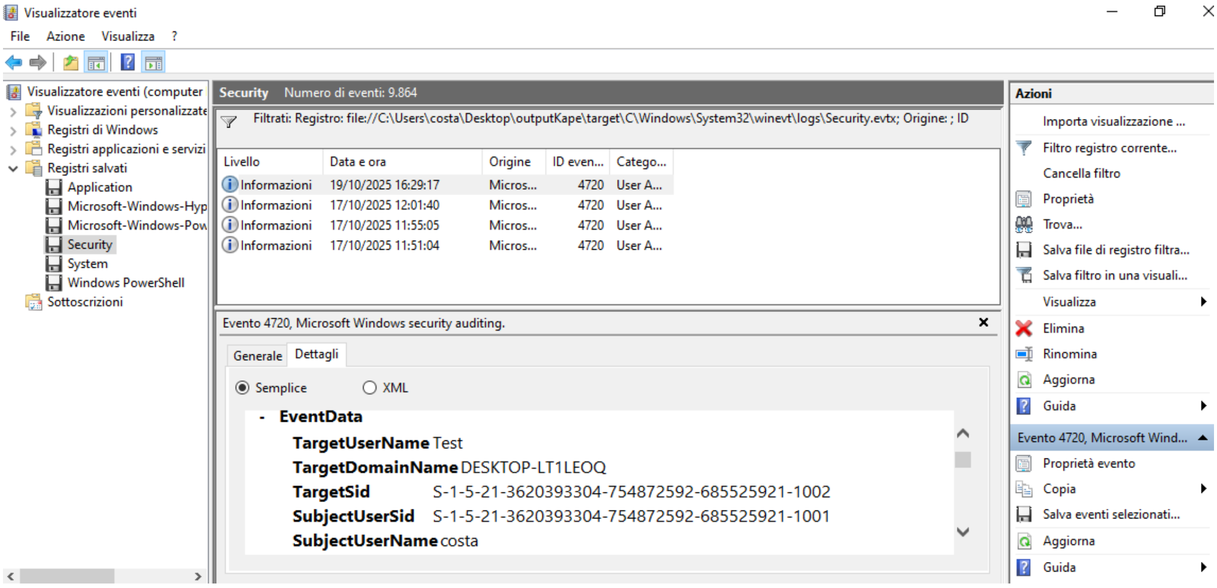
Livello	Data e ora	Origine	ID even...	Catego...
Informazioni	19/10/2025 16:29:17	Micros...	4720	User A...
Informazioni	17/10/2025 12:01:40	Micros...	4720	User A...
Informazioni	17/10/2025 11:55:05	Micros...	4720	User A...
Informazioni	17/10/2025 11:51:04	Micros...	4720	User A...

Evento 4720, Microsoft Windows security auditing.

TargetUserName costa
TargetDomainName DESKTOP-LT1LEOQ
TargetSid S-1-5-21-3620393304-754872592-685525921-1001
SubjectUserSid S-1-5-18
SubjectUserName WIN-6D6RCBNFEUE$
SubjectDomainName WORKGROUP

Figure 54: Security

And on 10/19, the day I took the tests



Security Numero di eventi: 9.864

Filtrati: Registro: file://C:\Users\costa\Desktop\outputKape\target\C\Windows\System32\winevt\logs\Security.evtx; Origine: ; ID

Livello	Data e ora	Origine	ID even...	Catego...
Informazioni	19/10/2025 16:29:17	Micros...	4720	User A...
Informazioni	17/10/2025 12:01:40	Micros...	4720	User A...
Informazioni	17/10/2025 11:55:05	Micros...	4720	User A...
Informazioni	17/10/2025 11:51:04	Micros...	4720	User A...

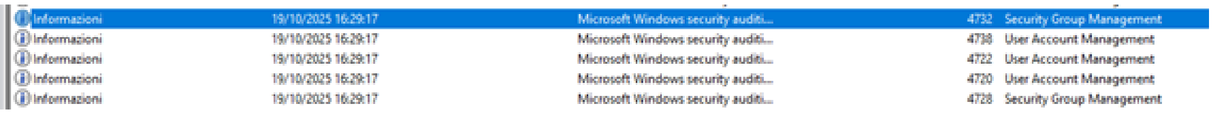
Evento 4720, Microsoft Windows security auditing.

- EventData

TargetUserName	Test
TargetDomainName	DESKTOP-LT1LEOQ
TargetSid	S-1-5-21-3620393304-754872592-685525921-1002
SubjectUserSid	S-1-5-21-3620393304-754872592-685525921-1001
SubjectUserName	costa

Figure 55: Filtered events

By delving deeper into the evtx files, in particular the Security.evtx file, it was possible to find the following events linked to the “Test” user:



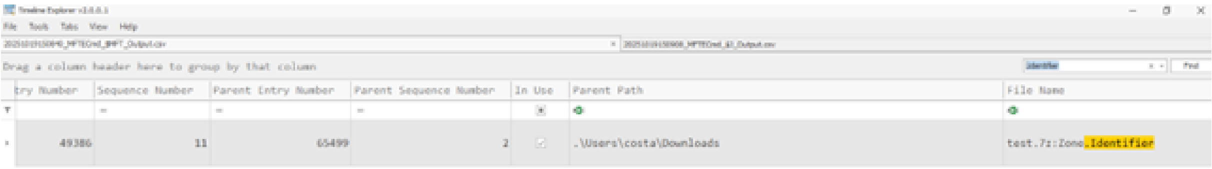
Informazioni	19/10/2025 16:29:17	Microsoft Windows security auditi...	4732	Security Group Management
Informazioni	19/10/2025 16:29:17	Microsoft Windows security auditi...	4738	User Account Management
Informazioni	19/10/2025 16:29:17	Microsoft Windows security auditi...	4722	User Account Management
Informazioni	19/10/2025 16:29:17	Microsoft Windows security auditi...	4720	User Account Management
Informazioni	19/10/2025 16:29:17	Microsoft Windows security auditi...	4728	Security Group Management

Figure 56: Security.evtx

4722 — “A user account was enabled” Event ID 4738 — A user account was changed (changes to passwords, group memberships, profile paths, user attributes, or account flags) Event ID 4732 — A member was added to a security-enabled local group (Event 4732 is logged when a user or computer account is added to a security-enabled local group on a domain controller (for domain local groups) or a local system.)

Regarding the script, further analysis is needed to determine where the Notepad tool originates (or, more precisely, where it was downloaded).

From initial analysis, it doesn’t appear possible to find information regarding the origin of the Notepad++ file. However, we do find information regarding the 7zip tool, which was downloaded to unzip the zip file containing the script.

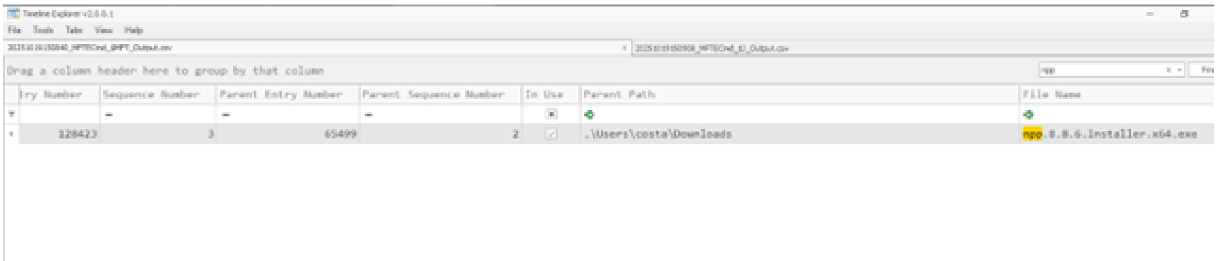

Timeline Explorer
File Tools Tabs View Help
20251019150940_MFTECmd_$MFT_Output.csv
20251019150908_MFTECmd_$J_Output.csv
Drag a column header here to group by that column
identifier
Find

try Number	Sequence Number	Parent Entry Number	Parent Sequence Number	In Use	Parent Path	File Name
49386	11	65499	2	☑	.\Users\costa\Downloads	test.7z:Zone.Identifier

Figure 57: 7zip identifier

By filtering the MFT for the .identifier extension (as in the previous case, the .identifier extension allows us to trace the URL from which the executable originated, as in the case of the previous malware), we can see that in this screenshot we only find information regarding the extension of the folder containing the executed script.

If, however, we try to search for the initials of the downloaded software, we only find the entry relating to the presence of the executable in the download folder:

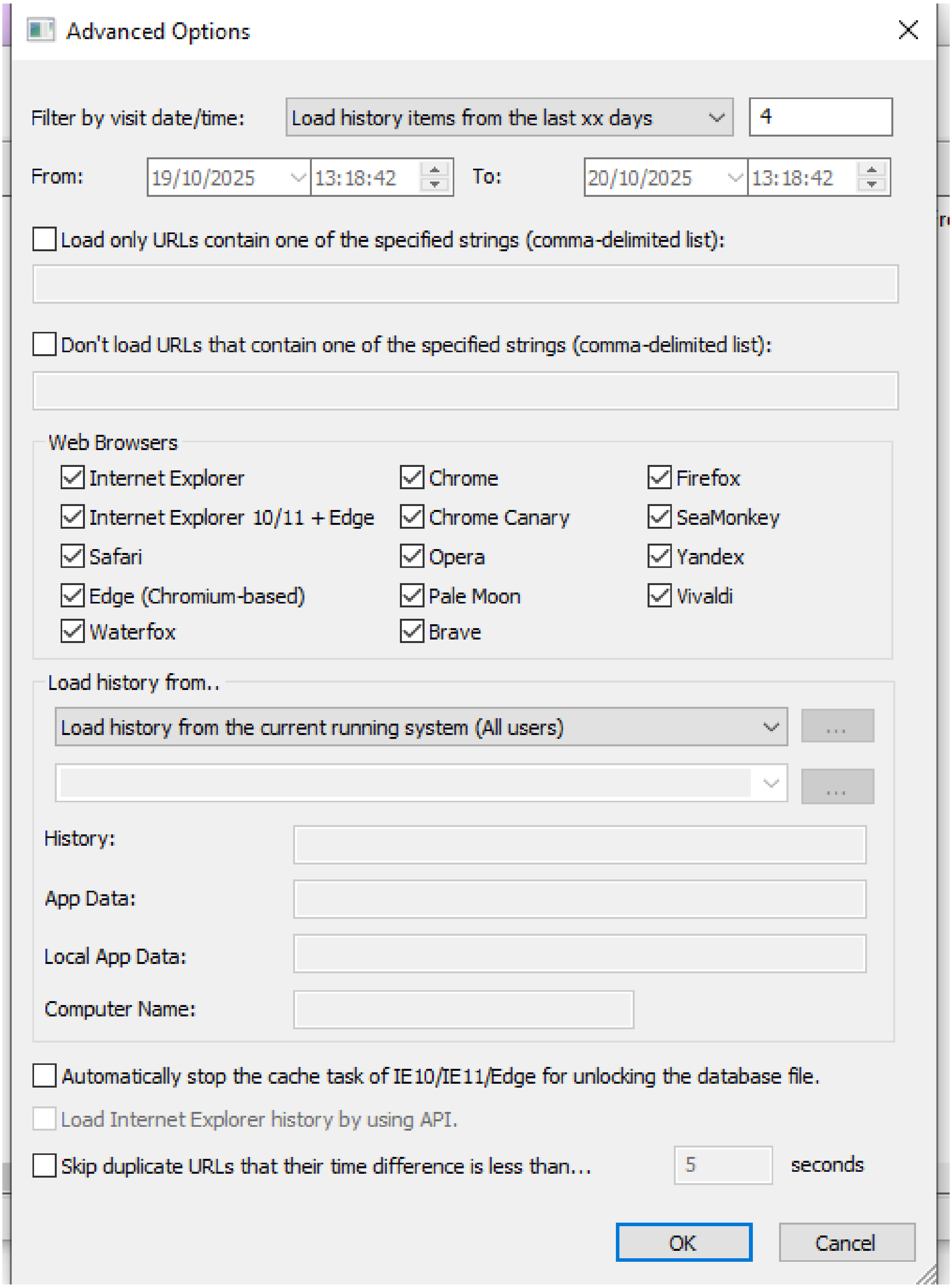

Timeline Explorer v2.0.0.1
File
Tools
Tabs
View
Help
Drag a column header here to group by that column
npp

try Number	Sequence Number	Parent Entry Number	Parent Sequence Number	In Use	Parent Path	File Name
128423	3	65499	2	✓	.\Users\costa\Downloads	npp.8.8.6.Installer.x64.exe

Figure 58: Evidence of npp

This situation brings to mind the possibility of using the BrowsingHistoryView tool.

BrowsingHistoryView is a utility that reads browsing history data from various web browsers (Mozilla Firefox, Google Chrome, Internet Explorer, Microsoft Edge, Opera) and displays the history of all these browsers in a single table. The browsing history table includes the following information: visited URL, page title, date and time of visit, number of visits, web browser used, and user profile. BrowsingHistoryView allows you to view the browsing history of all user profiles on a running system, as well as retrieve history from an external hard drive.



Advanced Options

Filter by visit date/time: Load history items from the last xx days 4

From: 19/10/2025 13:18:42 To: 20/10/2025 13:18:42

☐ Load only URLs contain one of the specified strings (comma-delimited list):

☐ Don't load URLs that contain one of the specified strings (comma-delimited list):

Web Browsers

☑ Internet Explorer ☑ Chrome ☑ Firefox
☑ Internet Explorer 10/11 + Edge ☑ Chrome Canary ☑ SeaMonkey
☑ Safari ☑ Opera ☑ Yandex
☑ Edge (Chromium-based) ☑ Pale Moon ☑ Vivaldi
☑ Waterfox ☑ Brave

Load history from..

Load history from the current running system (All users)

History:

App Data:

Local App Data:

Computer Name:

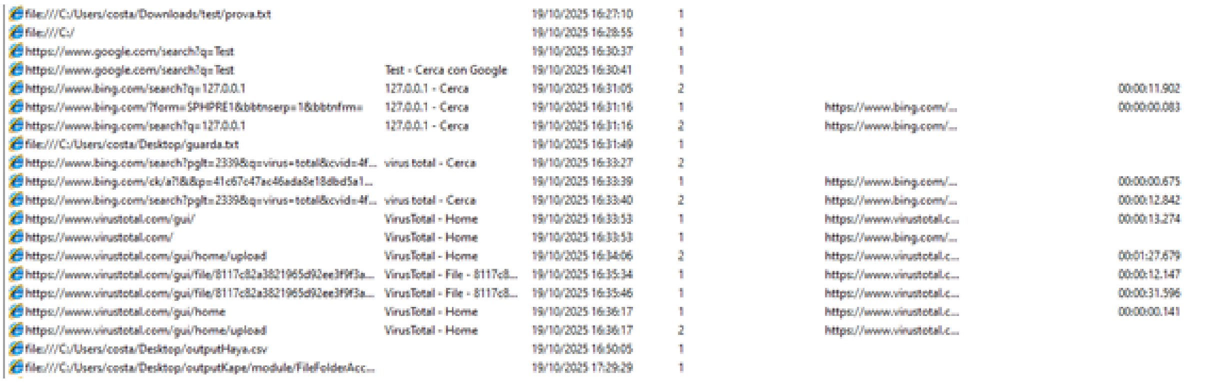
☐ Automatically stop the cache task of IE10/IE11/Edge for unlocking the database file.

☐ Load Internet Explorer history by using API.

☐ Skip duplicate URLs that their time difference is less than... 5 seconds

Figure 59: Browser analysis tool

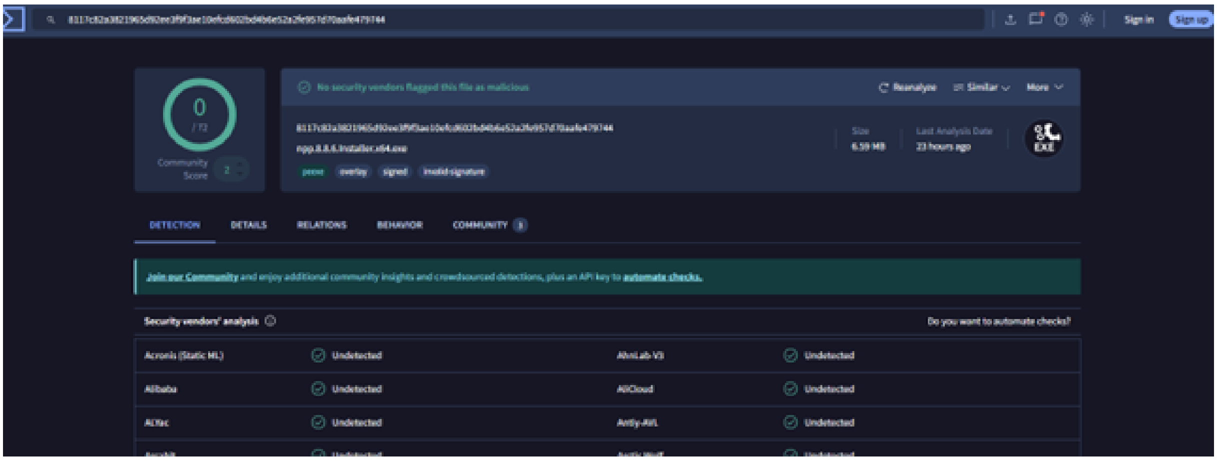
When starting the program, you must choose the analysis timing. Since the VM is unused and I ran the analysis over several days, I set the last four days:



file:///C:/Users/costa/Downloads/test/prova.txt		19/10/2025 16:27:10	1		
file:///C:/		19/10/2025 16:28:55	1		
https://www.google.com/search?q=Test		19/10/2025 16:30:37	1		
https://www.google.com/search?q=Test	Test - Cerca con Google	19/10/2025 16:30:41	1		
https://www.bing.com/search?q=127.0.0.1	127.0.0.1 - Cerca	19/10/2025 16:31:05	2		00:00:11.902
https://www.bing.com/?form=SPHPRE1&bbtnserp=1&bbtnfrm=	127.0.0.1 - Cerca	19/10/2025 16:31:16	1	https://www.bing.com/...	00:00:00.083
https://www.bing.com/search?q=127.0.0.1	127.0.0.1 - Cerca	19/10/2025 16:31:16	2	https://www.bing.com/...	
file:///C:/Users/costa/Desktop/guarda.txt		19/10/2025 16:31:49	1		
https://www.bing.com/search?pglt=2339&q=virus+total&cvid=4f...	virus total - Cerca	19/10/2025 16:33:27	2		
https://www.bing.com/ck/a?!&&p=41c67c47ac46ada8e18dbd5a1...		19/10/2025 16:33:39	1	https://www.bing.com/...	00:00:00.675
https://www.bing.com/search?pglt=2339&q=virus+total&cvid=4f...	virus total - Cerca	19/10/2025 16:33:40	2	https://www.bing.com/...	00:00:12.842
https://www.virustotal.com/gui/	VirusTotal - Home	19/10/2025 16:33:53	1	https://www.virustotal.c...	00:00:13.274
https://www.virustotal.com/	VirusTotal - Home	19/10/2025 16:33:53	1	https://www.bing.com/...	
https://www.virustotal.com/gui/home/upload	VirusTotal - Home	19/10/2025 16:34:06	2	https://www.virustotal.c...	00:01:27.679
https://www.virustotal.com/gui/file/8117c82a3821965d92ee3f9f3a...	VirusTotal - File - 8117c8...	19/10/2025 16:35:34	1	https://www.virustotal.c...	00:00:12.147
https://www.virustotal.com/gui/file/8117c82a3821965d92ee3f9f3a...	VirusTotal - File - 8117c8...	19/10/2025 16:35:46	1	https://www.virustotal.c...	00:00:31.596
https://www.virustotal.com/gui/home	VirusTotal - Home	19/10/2025 16:36:17	1	https://www.virustotal.c...	00:00:00.141
https://www.virustotal.com/gui/home/upload	VirusTotal - Home	19/10/2025 16:36:17	2	https://www.virustotal.c...	
file:///C:/Users/costa/Desktop/outputHaya.csv		19/10/2025 16:50:05	1		
file:///C:/Users/costa/Desktop/outputKape/module/FileFolderAcc...		19/10/2025 17:29:29	1		

Figure 60: Browser History analysis

From the screenshot you can find the searches performed by the script (Test, 127.0.0.1), the searches performed by me to analyze the hash and reputation from osint sources of the executable downloaded through a script and, subsequently, the activities performed by the Kape and Hayabusa tools.



8117c82a3821965d92ee3f9f3ae10efcd602bd4b6e52a2fe957d70aafe479744

No security vendors flagged this file as malicious

8117c82a3821965d92ee3f9f3ae10efcd602bd4b6e52a2fe957d70aafe479744

npp.8.8.6.Installer.x64.exe

peexe overlay signed invalid-signature

Size: 6.59 MB

Last Analysis Date: 23 hours ago

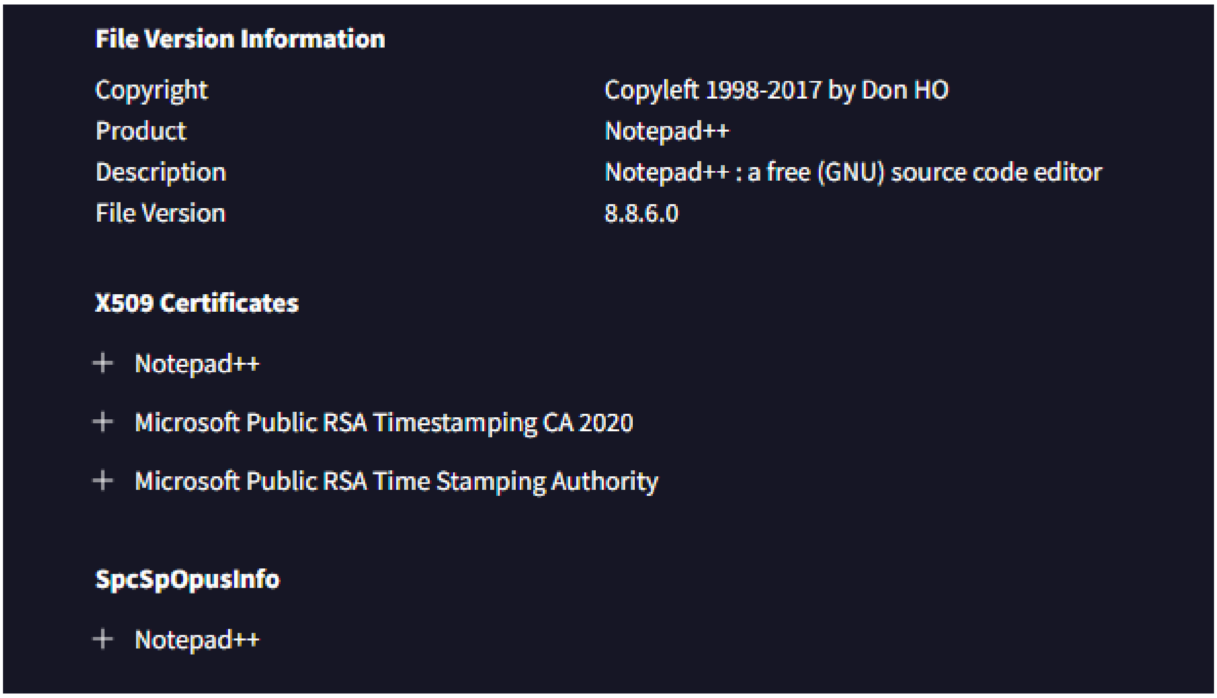
Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis

Do you want to automate checks?

Acronis (Static ML)	✓ Undetected	AhnLab-V3	✓ Undetected
Alibaba	✓ Undetected	AliCloud	✓ Undetected
ALYac	✓ Undetected	Antiy-AVL	✓ Undetected
Arcabit	✓ Undetected	Arctic Wolf	✓ Undetected

Figure 61: Reputation notepad



File Version Information

Copyright	Copyleft 1998-2017 by Don HO
Product	Notepad++
Description	Notepad++ : a free (GNU) source code editor
File Version	8.8.6.0

X509 Certificates

+ Notepad++

+ Microsoft Public RSA Timestamping CA 2020

+ Microsoft Public RSA Time Stamping Authority

SpcSpOpusInfo

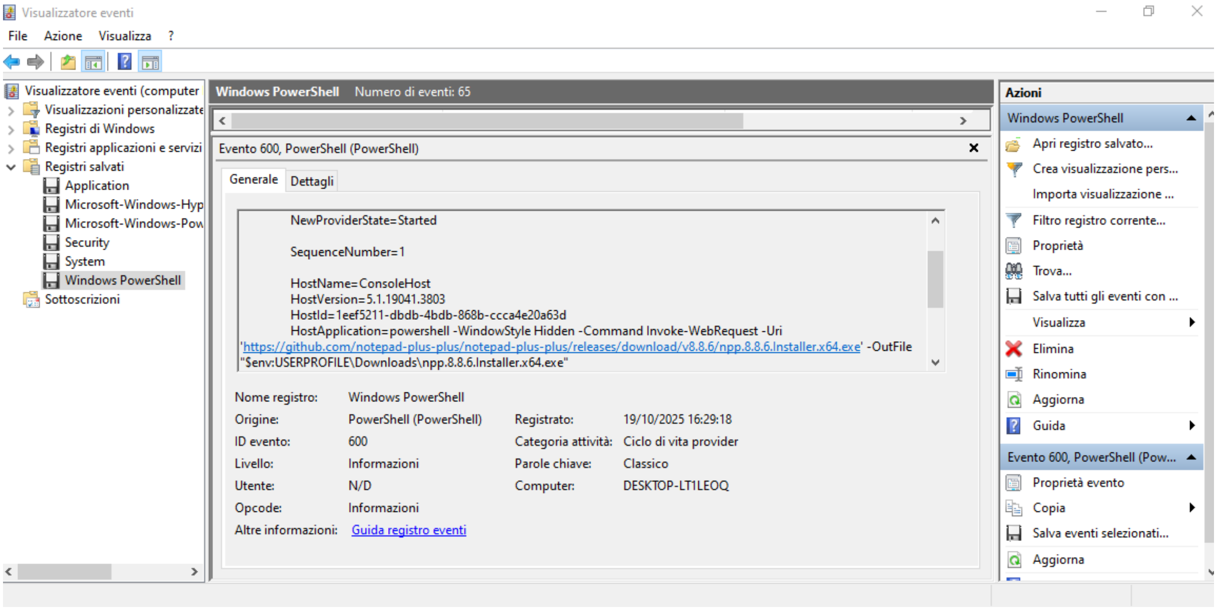
+ Notepad++

Figure 62: Information notepad

Therefore, we can assume that the script downloaded the executable without going through a browser.

Since it's a standard VM, there are no specific logging features, and sysmon hasn't been set. So, one idea that comes to mind is to try to see if I can find anything in the PowerShell logs:

From the Windows event viewer, I open the Windows PowerShell artifact that the Zimmermann suite generated for me via Kape, and by opening the first event, I immediately notice a GitHub URL that mentions Notepad: bingo!

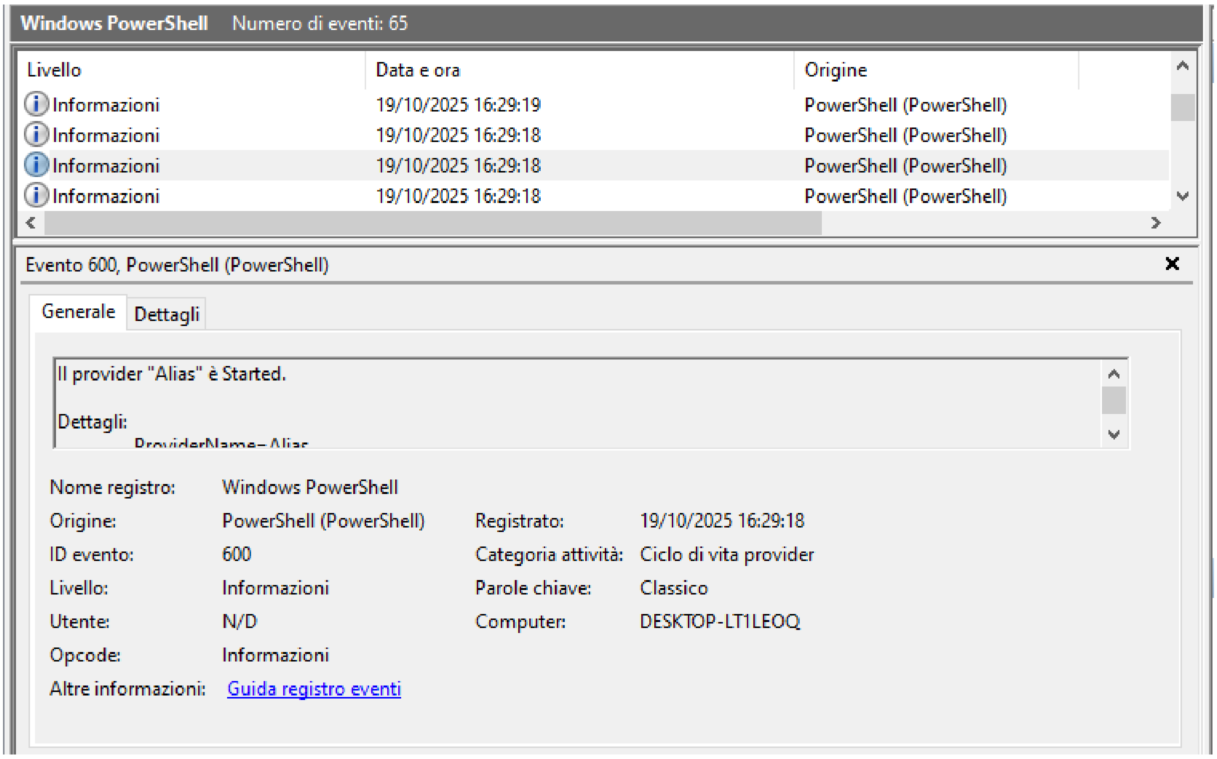


Evento 600, PowerShell (PowerShell)

```
        NewProviderState=Started

        SequenceNumber=1

        HostName=ConsoleHost
        HostVersion=5.1.19041.3803
        HostId=1eef5211-dbdb-4bdb-868b-ccca4e20a63d
        HostApplication=powershell -WindowStyle Hidden -Command Invoke-WebRequest -Uri
'https://github.com/notepad-plus-plus/notepad-plus-plus/releases/download/v8.8.6/npp.8.8.6.Installer.x64.exe' -OutFile
"$env:USERPROFILE\Downloads\npp.8.8.6.Installer.x64.exe"
```

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:18
ID evento:	600	Categoria attività:	Ciclo di vita provider
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 63: Powershell



Windows PowerShell Numero di eventi: 65

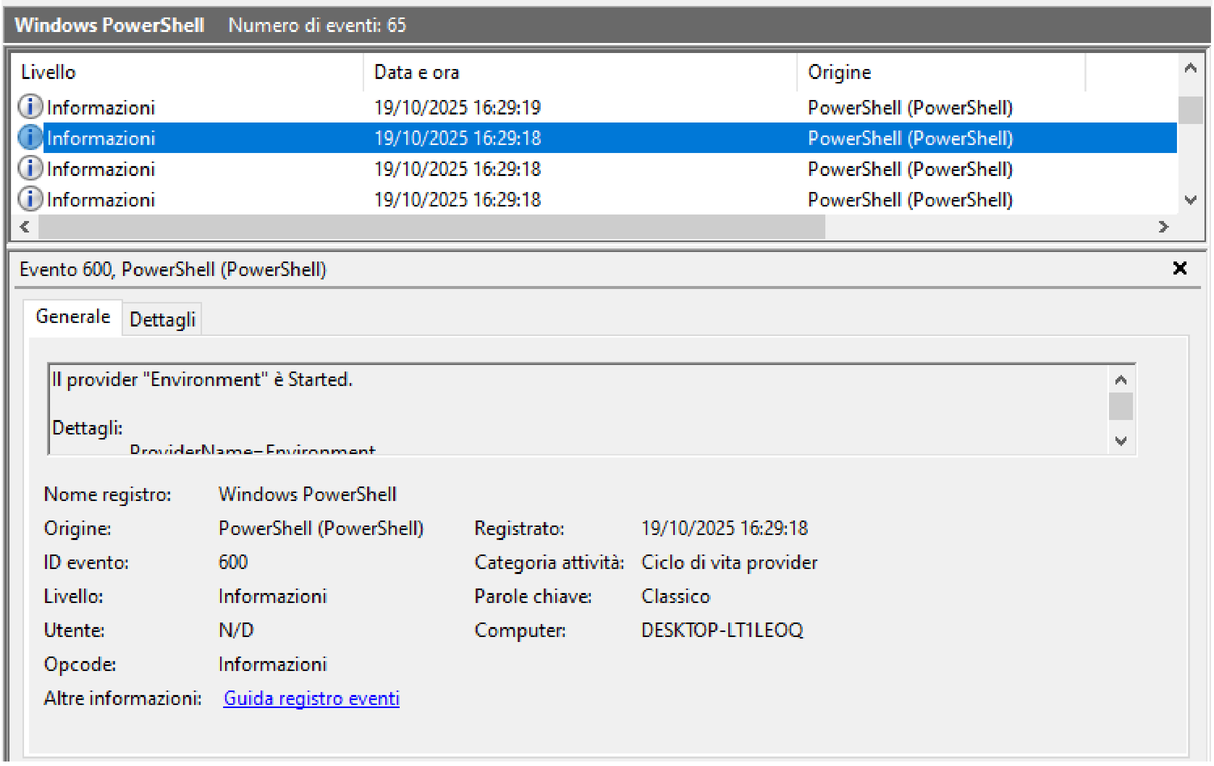
Livello	Data e ora	Origine
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)

Evento 600, PowerShell (PowerShell)

```
Il provider "Alias" è Started.

Dettagli:
	ProviderName=Alias
```

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:18
ID evento:	600	Categoria attività:	Ciclo di vita provider
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 64: Alias



Windows PowerShell Numero di eventi: 65

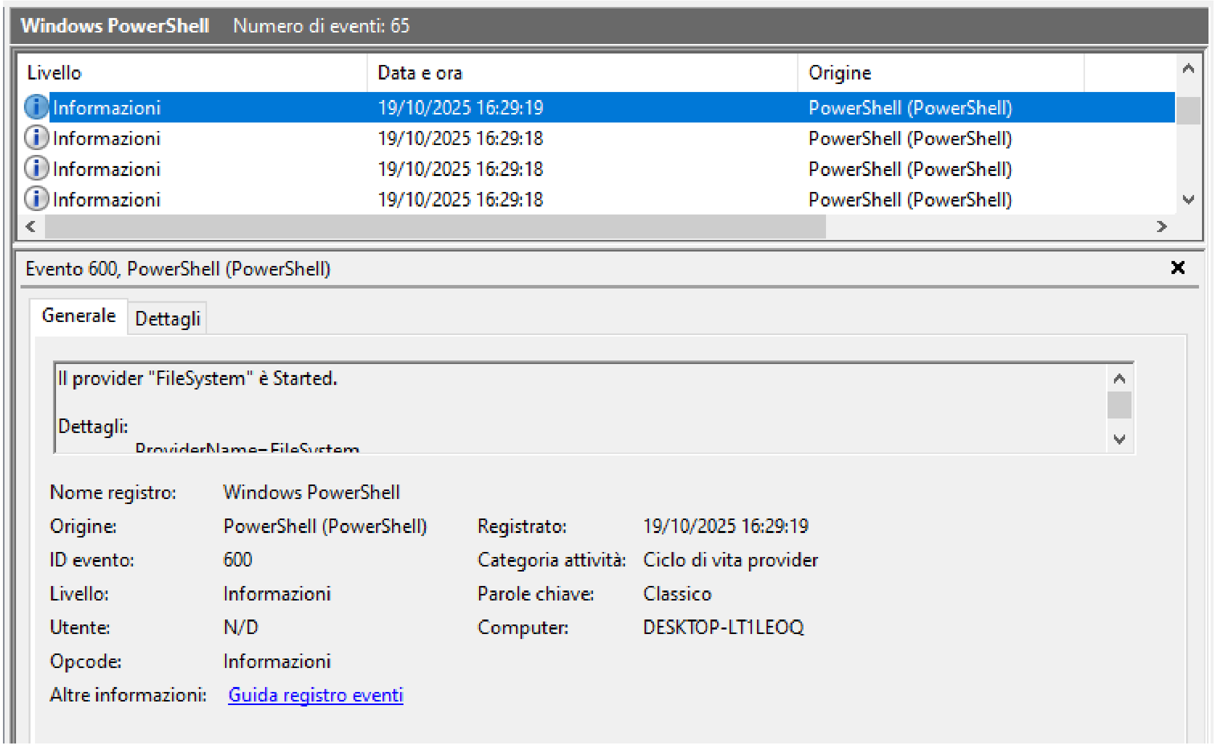
Livello	Data e ora	Origine
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)

Evento 600, PowerShell (PowerShell)

```
Il provider "Environment" è Started.

Dettagli:
	ProviderName=Environment
```

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:18
ID evento:	600	Categoria attività:	Ciclo di vita provider
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 65: Environment



Windows PowerShell Numero di eventi: 65

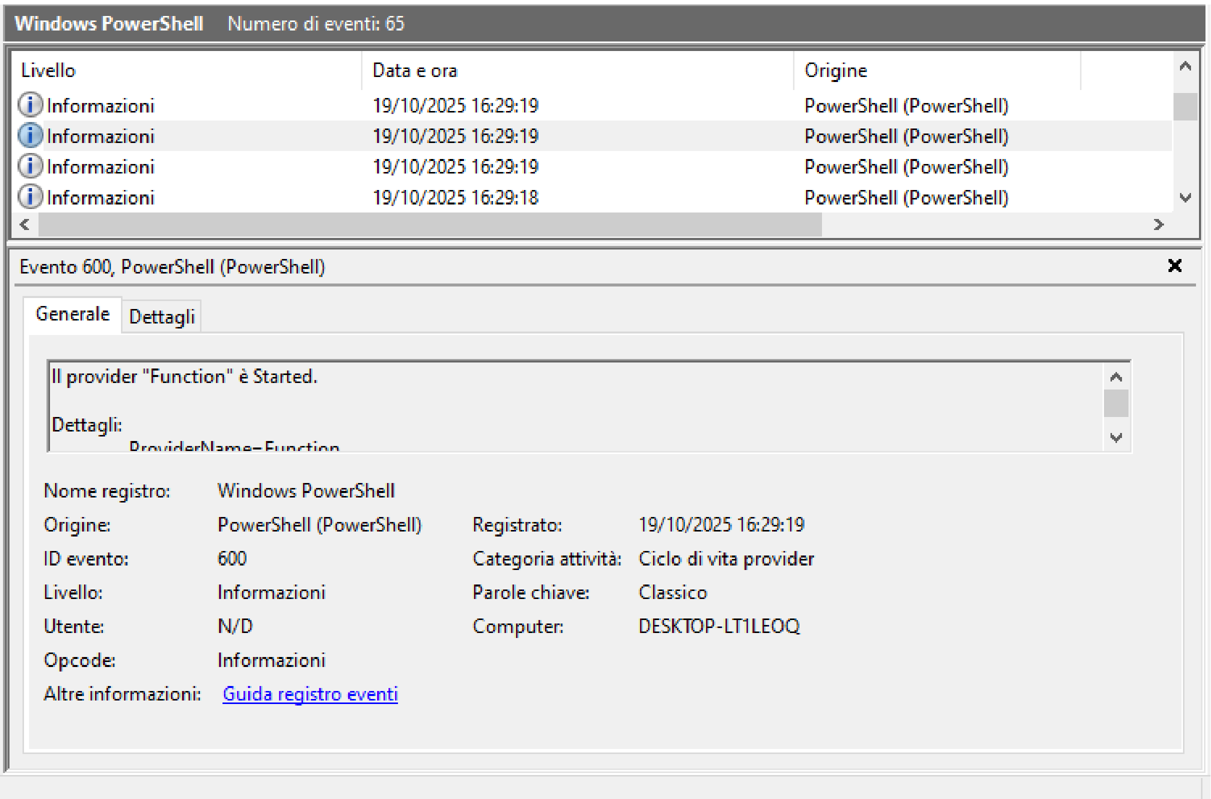
Livello	Data e ora	Origine
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)

Evento 600, PowerShell (PowerShell)

```
Il provider "FileSystem" è Started.

Dettagli:
        ProviderName=FileSystem
```

Nome registro: Windows PowerShell
Origine: PowerShell (PowerShell)
Registrato: 19/10/2025 16:29:19
ID evento: 600
Categoria attività: Ciclo di vita provider
Livello: Informazioni
Parole chiave: Classico
Utente: N/D
Computer: DESKTOP-LT1LEOQ
Opcode: Informazioni
Altre informazioni: Guida registro eventi

Figure 66: Filesystem



Windows PowerShell Numero di eventi: 65

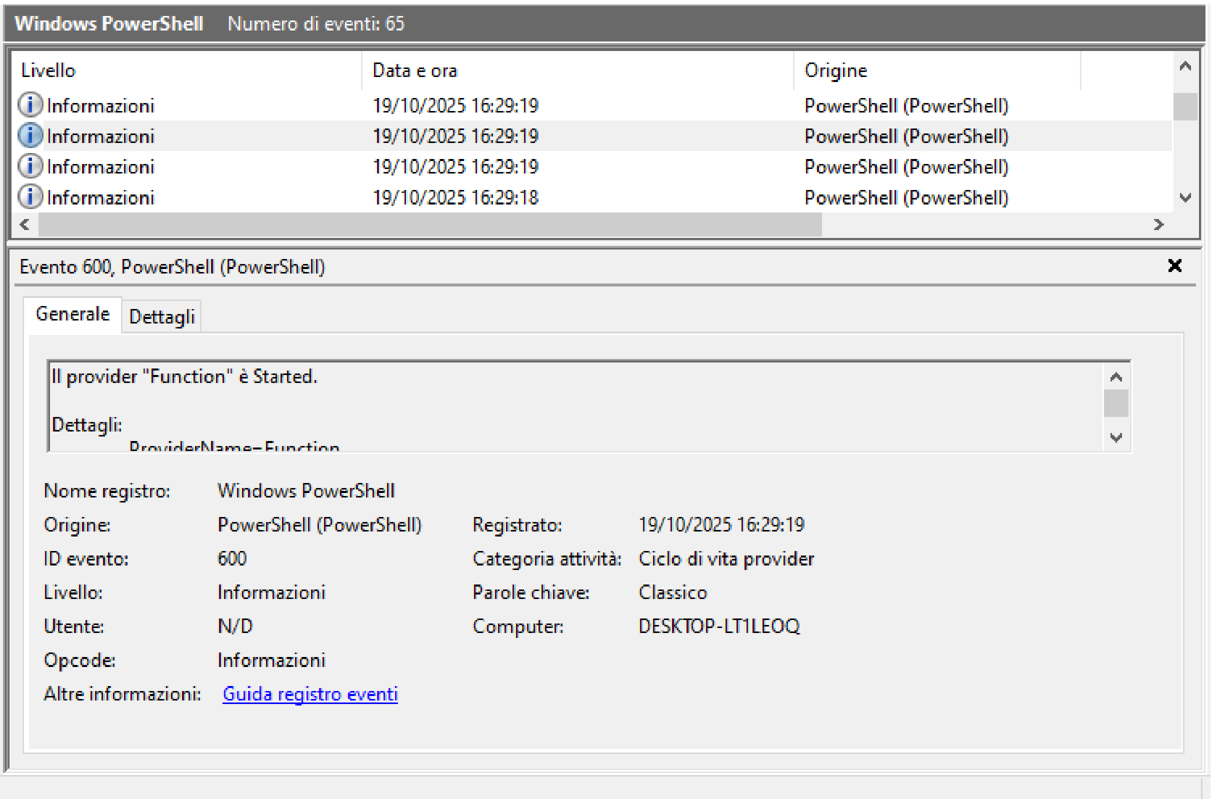
Livello	Data e ora	Origine
ⓘ Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
ⓘ Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
ⓘ Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
ⓘ Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)

Evento 600, PowerShell (PowerShell)

```
Il provider "Function" è Started.

Dettagli:
        ProviderName=Function
```

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:19
ID evento:	600	Categoria attività:	Ciclo di vita provider
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 67: Function



Windows PowerShell Numero di eventi: 65

Livello	Data e ora	Origine
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:18	PowerShell (PowerShell)

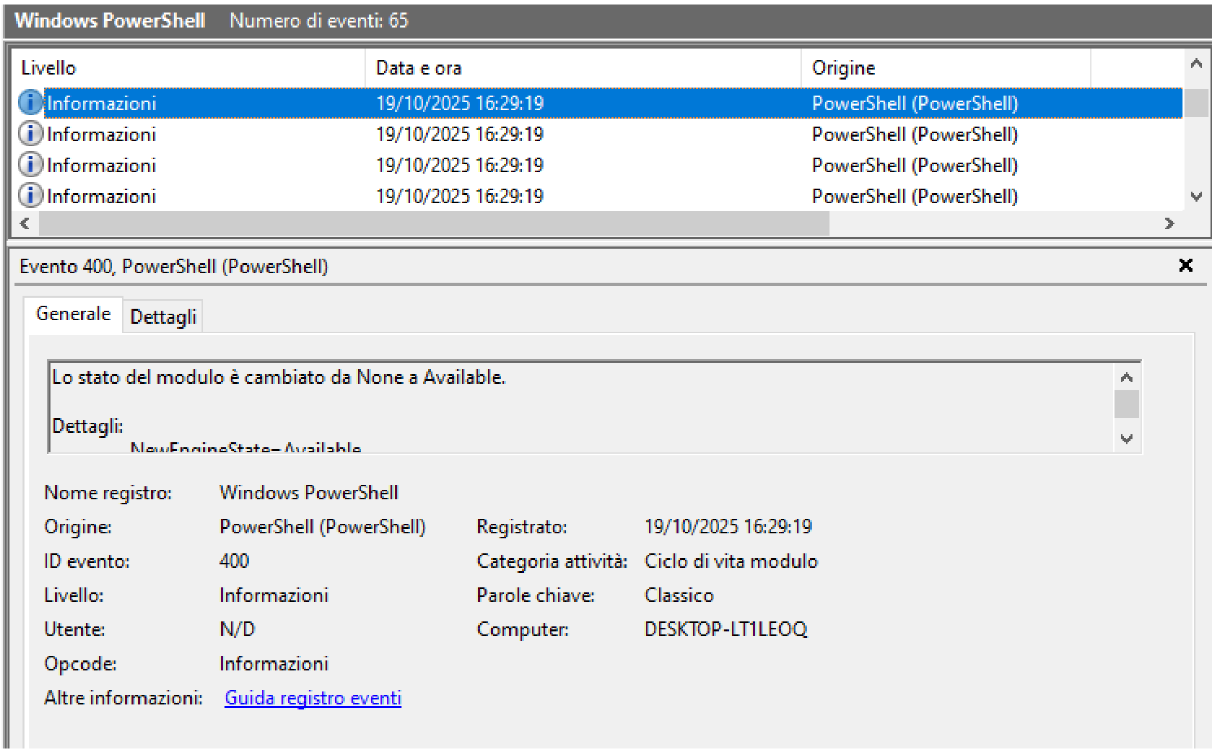
Evento 600, PowerShell (PowerShell)

Il provider "Function" è Started.

Dettagli:
ProviderName=Function

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:19
ID evento:	600	Categoria attività:	Ciclo di vita provider
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 68: Variable



Windows PowerShell Numero di eventi: 65

Livello	Data e ora	Origine
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)

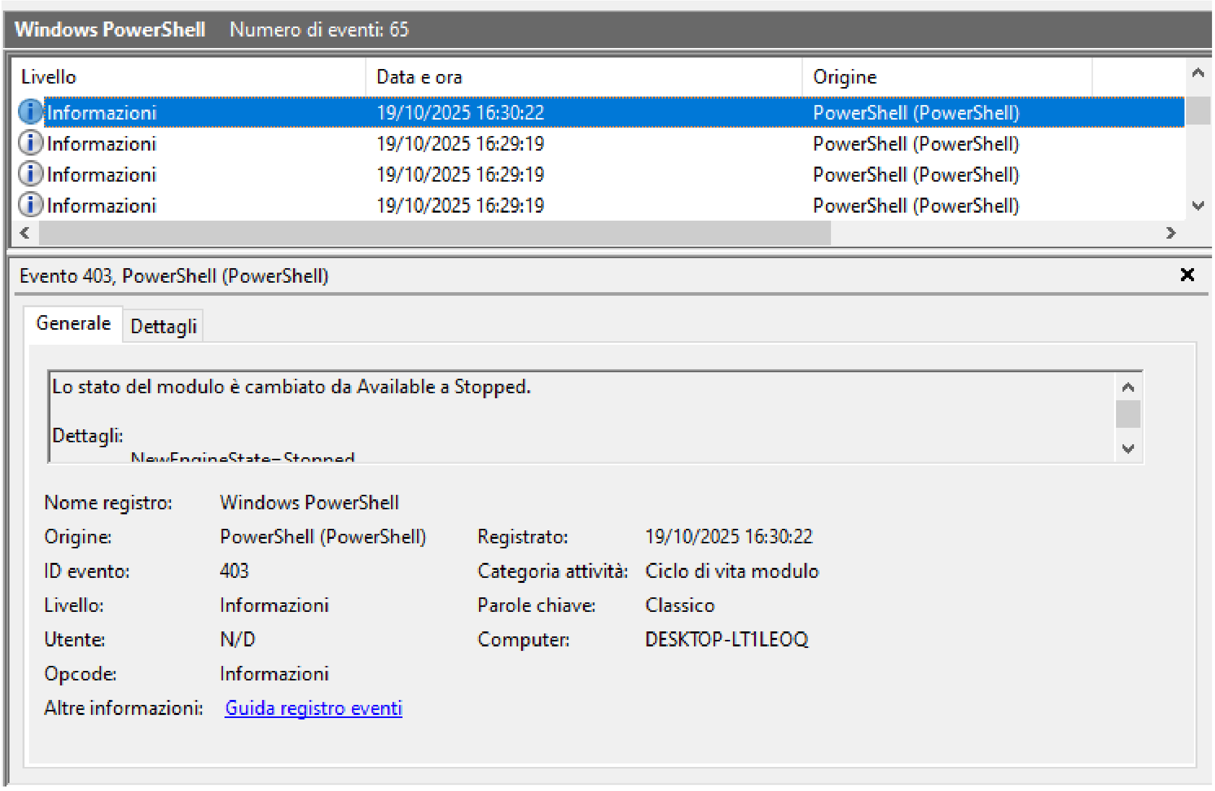
Evento 400, PowerShell (PowerShell)

Lo stato del modulo è cambiato da None a Available.

Dettagli:
NewEngineState=Available

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:29:19
ID evento:	400	Categoria attività:	Ciclo di vita modulo
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 69: Changed



Windows PowerShell Numero di eventi: 65

Livello	Data e ora	Origine
Informazioni	19/10/2025 16:30:22	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)
Informazioni	19/10/2025 16:29:19	PowerShell (PowerShell)

Evento 403, PowerShell (PowerShell)

Lo stato del modulo è cambiato da Available a Stopped.

Dettagli:
NewEngineState=Stopped

Nome registro:	Windows PowerShell		
Origine:	PowerShell (PowerShell)	Registrato:	19/10/2025 16:30:22
ID evento:	403	Categoria attività:	Ciclo di vita modulo
Livello:	Informazioni	Parole chiave:	Classico
Utente:	N/D	Computer:	DESKTOP-LT1LEOQ
Opcode:	Informazioni		
Altre informazioni:	Guida registro eventi		

Figure 70: Stopped

In all the above screenshots, which appear identical, you can see the PowerShell execution flow. Note that PowerShell was activated at this time: 10/19/2025 4:29:18 PM and finished executing at the following time: 10/19/2025 4:30:22 PM. At these times, we find event IDs 600, 400, and 403.

Event ID 600 corresponds to Provider Lifecycle "Provider Initialized" and is present every time PowerShell initializes an internal provider (Registry, Alias, Environment, FileSystem, Variable, Function, etc.).

Event ID 400 corresponds to Engine Lifecycle "PowerShell Engine Started," meaning it starts a new PowerShell session (engine start). It displays the version, host, culture, and initial state of the modules; in other words, it tells us when PowerShell started.

The event id 403 Engine Lifecycle “PowerShell Engine Stopped” signals the closure of the session, and therefore indicates when PowerShell has terminated.

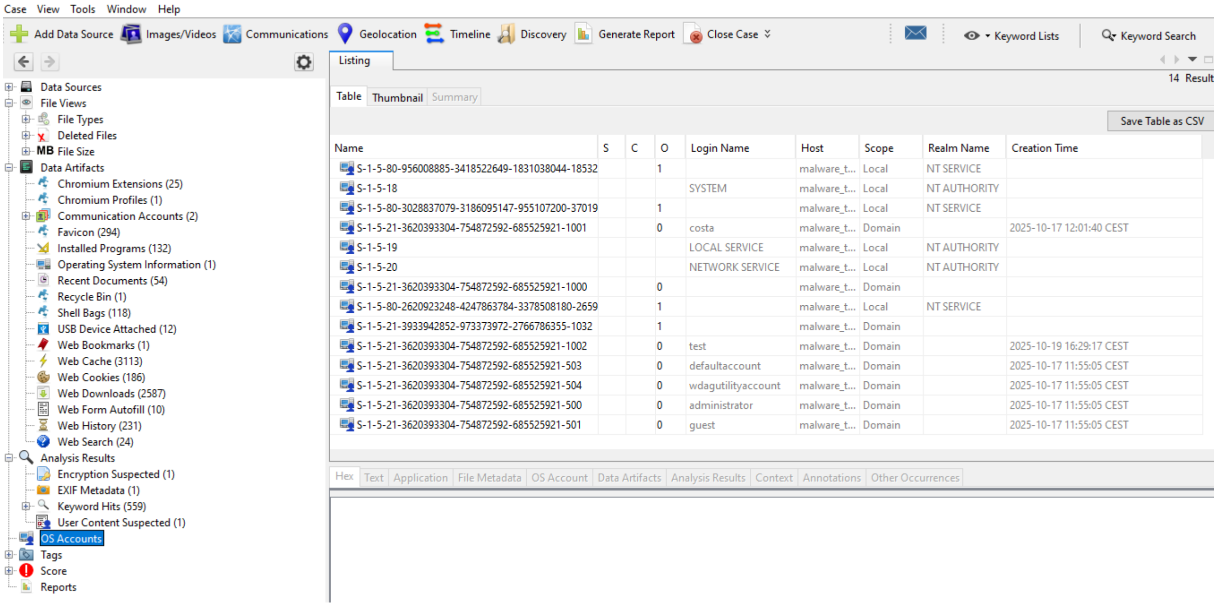
We just mentioned that Event ID 600 in PowerShell corresponds to Provider Lifecycle; a provider is a "logical interface" that allows access to system resources as if they were file systems.

The following providers can be seen from the screenshots: Registry: Access to the registry as a logical path; Alias: PowerShell alias (e.g., dir = Get-ChildItem); Environment: Environment variables; File System: Access to files and folders Variable: Session variables ($env, $PSVersionTable, etc.) Function: Functions defined in the session.

To further appreciate the potential of these forensic tools and this investigation method, an analysis comparison with Autopsy was performed. Two additional toy scenarios were then analyzed, demonstrating the superiority of the Kape+Zimmerman Tools+Hayabusa suite in terms of performance (depth of analysis) and efficiency (timeframes) compared to tools like Autopsy.

4.6 Comparison of the first simulation with Autopsy

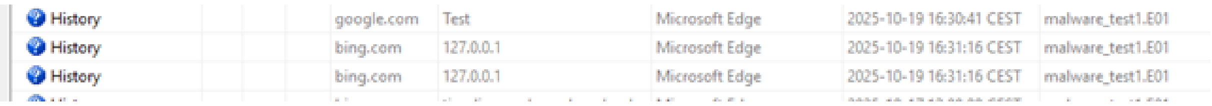
Using Autopsy, we were able to quickly identify the "Test" user. In fact, by going to OS Accounts, in addition to the "standard" system users, we were able to find the "costa" user, the account holder, and the "Test" user, which was created by the script. From this screenshot, we can only see the time of creation, but we don't know what caused it (due to the lack of specific artifacts).



Name	S	C	O	Login Name	Host	Scope	Realm Name	Creation Time
S-1-5-80-956008885-3418522649-1831038044-18532			1		malware_t...	Local	NT SERVICE	
S-1-5-18				SYSTEM	malware_t...	Local	NT AUTHORITY	
S-1-5-80-3028837079-3186095147-955107200-37019			1		malware_t...	Local	NT SERVICE	
S-1-5-21-3620393304-754872592-685525921-1001			0	costa	malware_t...	Domain		2025-10-17 12:01:40 CEST
S-1-5-19				LOCAL SERVICE	malware_t...	Local	NT AUTHORITY	
S-1-5-20				NETWORK SERVICE	malware_t...	Local	NT AUTHORITY	
S-1-5-21-3620393304-754872592-685525921-1000			0		malware_t...	Domain		
S-1-5-80-2620923248-4247863784-3378508180-2659			1		malware_t...	Local	NT SERVICE	
S-1-5-21-3933942852-973373972-2766786355-1032			1		malware_t...	Domain		
S-1-5-21-3620393304-754872592-685525921-1002			0	test	malware_t...	Domain		2025-10-19 16:29:17 CEST
S-1-5-21-3620393304-754872592-685525921-503			0	defaultaccount	malware_t...	Domain		2025-10-17 11:55:05 CEST
S-1-5-21-3620393304-754872592-685525921-504			0	wdagutilityaccount	malware_t...	Domain		2025-10-17 11:55:05 CEST
S-1-5-21-3620393304-754872592-685525921-500			0	administrator	malware_t...	Domain		2025-10-17 11:55:05 CEST
S-1-5-21-3620393304-754872592-685525921-501			0	guest	malware_t...	Domain		2025-10-17 11:55:05 CEST

Figure 71: Users

In just a few seconds, it is possible to obtain information about web browsers (without the need for external tools such as Kape+Zimmerman), caches and cookies (useful for detecting any connections to suspicious domains or any unauthorized tracking activities), and in fact we immediately discover the connections to the URLs expected by the script:



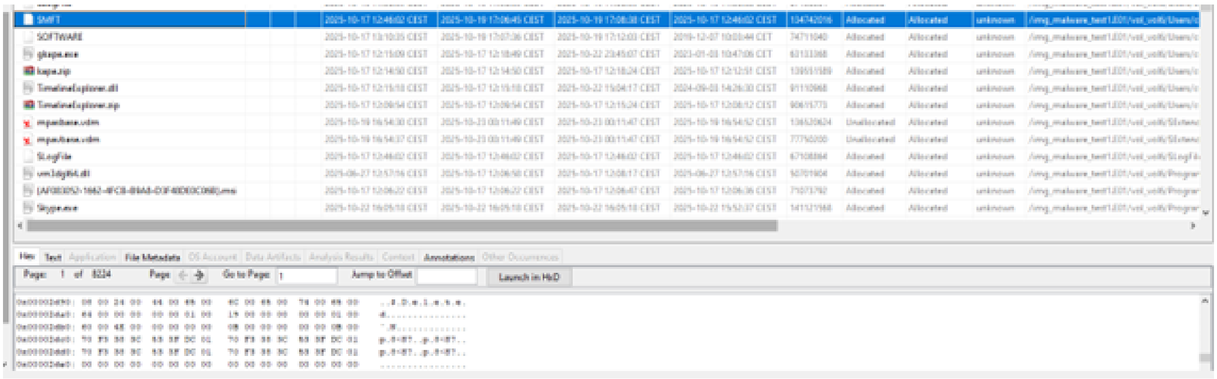
History	google.com	Test	Microsoft Edge	2025-10-19 16:30:41 CEST	malware_test1.E01
History	bing.com	127.0.0.1	Microsoft Edge	2025-10-19 16:31:16 CEST	malware_test1.E01
History	bing.com	127.0.0.1	Microsoft Edge	2025-10-19 16:31:16 CEST	malware_test1.E01

Figure 72: Url

The tool has a very limited view of shallbags and was unable to detect that a script in a folder had been executed (it didn't detect that I had navigated to the folder to execute the script during the simulation (this is visible with KAPE thanks to Zimmerman Tools).

Next, I looked at the MFT:

Analyzing the MFT isn't as intuitive as with KAPE+Zimmerman. However, it's possible to examine byte by byte what can be gleaned. In our case, we were able to determine (not with much help from the GUI) that a file had been deleted:

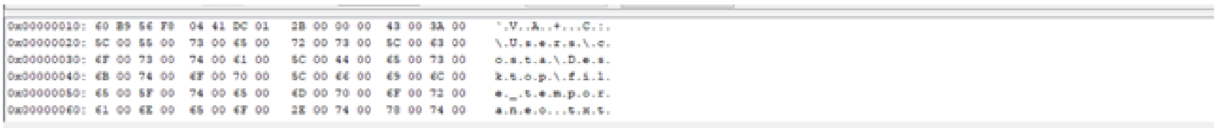


$MFT	2025-10-17 12:46:02 CEST	2025-10-19 17:06:45 CEST	2025-10-19 17:08:38 CEST	2025-10-17 12:46:02 CEST	134742016	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
SOFTWARE	2025-10-17 13:10:35 CEST	2025-10-19 17:07:36 CEST	2025-10-19 17:12:03 CEST	2019-12-07 10:03:44 CET	74711040	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
gkape.exe	2025-10-17 12:15:09 CEST	2025-10-17 12:18:49 CEST	2025-10-22 23:45:07 CEST	2023-01-03 10:47:06 CET	63133368	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
kape.zip	2025-10-17 12:14:50 CEST	2025-10-17 12:14:50 CEST	2025-10-17 12:18:24 CEST	2025-10-17 12:12:51 CEST	139551589	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
TimelineExplorer.dll	2025-10-17 12:15:10 CEST	2025-10-17 12:15:10 CEST	2025-10-22 15:04:17 CEST	2024-09-03 14:26:30 CEST	91110968	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
TimelineExplorer.zip	2025-10-17 12:09:54 CEST	2025-10-17 12:09:54 CEST	2025-10-17 12:15:24 CEST	2025-10-17 12:08:12 CEST	90615775	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Users/c
mpasbase.vdm	2025-10-19 16:54:30 CEST	2025-10-23 00:11:49 CEST	2025-10-23 00:11:47 CEST	2025-10-19 16:54:52 CEST	136520624	Unallocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/$Extend
mpasbase.vdm	2025-10-19 16:54:37 CEST	2025-10-23 00:11:49 CEST	2025-10-23 00:11:47 CEST	2025-10-19 16:54:52 CEST	77750200	Unallocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/$Extend
$LogFile	2025-10-17 12:46:02 CEST	2025-10-17 12:46:02 CEST	2025-10-17 12:46:02 CEST	2025-10-17 12:46:02 CEST	67108864	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/$LogFi
vm3dgl64.dll	2025-06-27 12:57:16 CEST	2025-10-17 12:06:50 CEST	2025-10-17 12:08:17 CEST	2025-06-27 12:57:16 CEST	50701904	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Progra
{AF0E0052-1662-4FCB-B9A8-D3F40DE0C068}.msi	2025-10-17 12:06:22 CEST	2025-10-17 12:06:22 CEST	2025-10-17 12:06:47 CEST	2025-10-17 12:06:36 CEST	71073792	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Progra
Skype.exe	2025-10-22 16:05:10 CEST	2025-10-22 16:05:10 CEST	2025-10-22 16:05:10 CEST	2025-10-22 15:52:37 CEST	141121568	Allocated	Allocated	unknown	/img_malware_test1.E01/vol_vol6/Progra

```
0x00002d90: 08 00 24 00  44 00 65 00  6C 00 65 00  74 00 65 00   ..$.D.e.l.e.t.e.
0x00002da0: 64 00 00 00  00 00 01 00  19 00 00 00  00 00 01 00   d...............
0x00002db0: 60 00 48 00  00 00 00 00  0B 00 00 00  00 00 0B 00   `.H.............
0x00002dc0: 70 F3 38 9C  63 3F DC 01  70 F3 38 9C  63 3F DC 01   p.8<c?..p.8<c?..
0x00002dd0: 70 F3 38 9C  63 3F DC 01  70 F3 38 9C  63 3F DC 01   p.8<c?..p.8<c?..
0x00002de0: 00 00 00 00  00 00 00 00  00 00 00 00  00 00 00 00   ................
```

Figure 73: Autopsy MFT

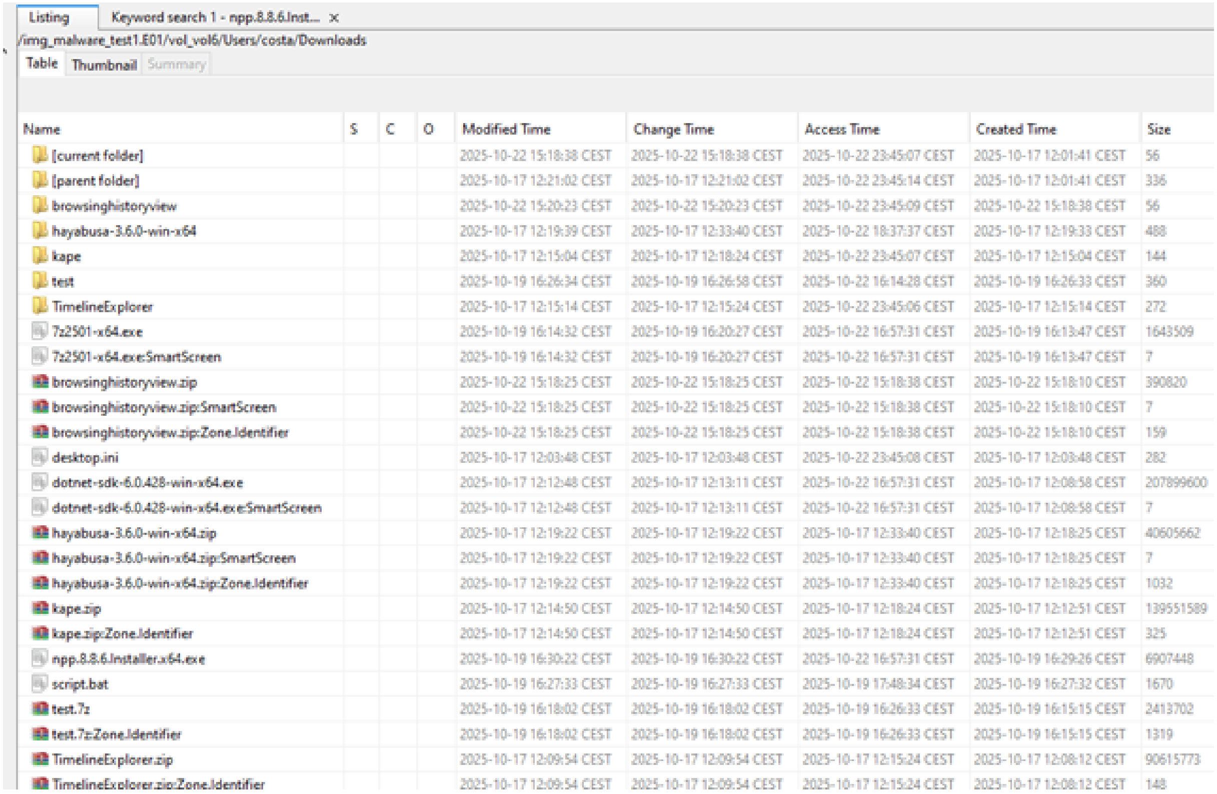
But we don't have any clear confirmation that it's the same file discovered previously. The real strength of this tool is its detailed analysis of the system's file system, allowing you to discover all the partitions present and analyze the memory allocation and specific location in detail at the byte level. For example, analyzing the Recycle Bin provided detailed information about the deleted file (which matches the information in the script):



```
0x00000010: 60 B9 56 F8  04 41 DC 01  2B 00 00 00  43 00 3A 00   `.V..A..+...C.:.
0x00000020: 5C 00 55 00  73 00 65 00  72 00 73 00  5C 00 63 00   \.U.s.e.r.s.\.c.
0x00000030: 6F 00 73 00  74 00 61 00  5C 00 44 00  65 00 73 00   o.s.t.a.\.D.e.s.
0x00000040: 6B 00 74 00  6F 00 70 00  5C 00 66 00  69 00 6C 00   k.t.o.p.\.f.i.l.
0x00000050: 65 00 5F 00  74 00 65 00  6D 00 70 00  6F 00 72 00   e._.t.e.m.p.o.r.
0x00000060: 61 00 6E 00  65 00 6F 00  2E 00 74 00  78 00 74 00   a.n.e.o...t.x.t.
```

Figure 74: Autopsy bin

Going through the folders in the file system we notice that the notepad file is present:

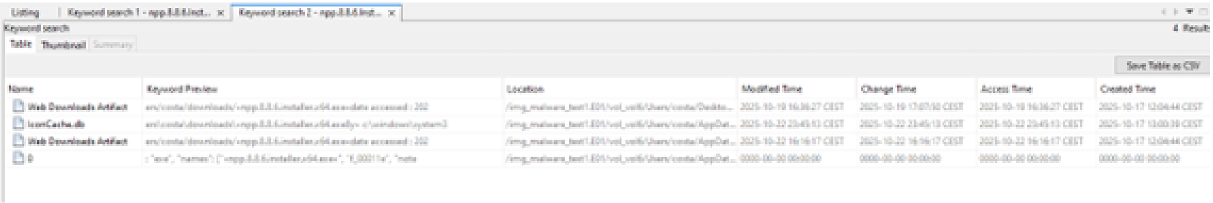


/img_malware_test1.E01/vol_vol6/Users/costa/Downloads

Name	S	C	O	Modified Time	Change Time	Access Time	Created Time	Size
[current folder]				2025-10-22 15:18:38 CEST	2025-10-22 15:18:38 CEST	2025-10-22 23:45:07 CEST	2025-10-17 12:01:41 CEST	56
[parent folder]				2025-10-17 12:21:02 CEST	2025-10-17 12:21:02 CEST	2025-10-22 23:45:14 CEST	2025-10-17 12:01:41 CEST	336
browsinghistoryview				2025-10-22 15:20:23 CEST	2025-10-22 15:20:23 CEST	2025-10-22 23:45:09 CEST	2025-10-22 15:18:38 CEST	56
hayabusa-3.6.0-win-x64				2025-10-17 12:19:39 CEST	2025-10-17 12:33:40 CEST	2025-10-22 18:37:37 CEST	2025-10-17 12:19:33 CEST	488
kape				2025-10-17 12:15:04 CEST	2025-10-17 12:18:24 CEST	2025-10-22 23:45:07 CEST	2025-10-17 12:15:04 CEST	144
test				2025-10-19 16:26:34 CEST	2025-10-19 16:26:58 CEST	2025-10-22 16:14:28 CEST	2025-10-19 16:26:33 CEST	360
TimelineExplorer				2025-10-17 12:15:14 CEST	2025-10-17 12:15:24 CEST	2025-10-22 23:45:06 CEST	2025-10-17 12:15:14 CEST	272
7z2501-x64.exe				2025-10-19 16:14:32 CEST	2025-10-19 16:20:27 CEST	2025-10-22 16:57:31 CEST	2025-10-19 16:13:47 CEST	1643509
7z2501-x64.exe:SmartScreen				2025-10-19 16:14:32 CEST	2025-10-19 16:20:27 CEST	2025-10-22 16:57:31 CEST	2025-10-19 16:13:47 CEST	7
browsinghistoryview.zip				2025-10-22 15:18:25 CEST	2025-10-22 15:18:25 CEST	2025-10-22 15:18:38 CEST	2025-10-22 15:18:10 CEST	390820
browsinghistoryview.zip:SmartScreen				2025-10-22 15:18:25 CEST	2025-10-22 15:18:25 CEST	2025-10-22 15:18:38 CEST	2025-10-22 15:18:10 CEST	7
browsinghistoryview.zip:Zone.Identifier				2025-10-22 15:18:25 CEST	2025-10-22 15:18:25 CEST	2025-10-22 15:18:38 CEST	2025-10-22 15:18:10 CEST	159
desktop.ini				2025-10-17 12:03:48 CEST	2025-10-17 12:03:48 CEST	2025-10-22 23:45:08 CEST	2025-10-17 12:03:48 CEST	282
dotnet-sdk-6.0.428-win-x64.exe				2025-10-17 12:12:48 CEST	2025-10-17 12:13:11 CEST	2025-10-22 16:57:31 CEST	2025-10-17 12:08:58 CEST	207899600
dotnet-sdk-6.0.428-win-x64.exe:SmartScreen				2025-10-17 12:12:48 CEST	2025-10-17 12:13:11 CEST	2025-10-22 16:57:31 CEST	2025-10-17 12:08:58 CEST	7
hayabusa-3.6.0-win-x64.zip				2025-10-17 12:19:22 CEST	2025-10-17 12:19:22 CEST	2025-10-17 12:33:40 CEST	2025-10-17 12:18:25 CEST	40605662
hayabusa-3.6.0-win-x64.zip:SmartScreen				2025-10-17 12:19:22 CEST	2025-10-17 12:19:22 CEST	2025-10-17 12:33:40 CEST	2025-10-17 12:18:25 CEST	7
hayabusa-3.6.0-win-x64.zip:Zone.Identifier				2025-10-17 12:19:22 CEST	2025-10-17 12:19:22 CEST	2025-10-17 12:33:40 CEST	2025-10-17 12:18:25 CEST	1032
kape.zip				2025-10-17 12:14:50 CEST	2025-10-17 12:14:50 CEST	2025-10-17 12:18:24 CEST	2025-10-17 12:12:51 CEST	139551589
kape.zip:Zone.Identifier				2025-10-17 12:14:50 CEST	2025-10-17 12:14:50 CEST	2025-10-17 12:18:24 CEST	2025-10-17 12:12:51 CEST	325
npp.8.8.6.Installer.x64.exe				2025-10-19 16:30:22 CEST	2025-10-19 16:30:22 CEST	2025-10-22 16:57:31 CEST	2025-10-19 16:29:26 CEST	6907448
script.bat				2025-10-19 16:27:33 CEST	2025-10-19 16:27:33 CEST	2025-10-19 17:48:34 CEST	2025-10-19 16:27:32 CEST	1670
test.7z				2025-10-19 16:18:02 CEST	2025-10-19 16:18:02 CEST	2025-10-19 16:26:33 CEST	2025-10-19 16:15:15 CEST	2413702
test.7z:Zone.Identifier				2025-10-19 16:18:02 CEST	2025-10-19 16:18:02 CEST	2025-10-19 16:26:33 CEST	2025-10-19 16:15:15 CEST	1319
TimelineExplorer.zip				2025-10-17 12:09:54 CEST	2025-10-17 12:09:54 CEST	2025-10-17 12:15:24 CEST	2025-10-17 12:08:12 CEST	90615773
TimelineExplorer.zip:Zone.Identifier				2025-10-17 12:09:54 CEST	2025-10-17 12:09:54 CEST	2025-10-17 12:15:24 CEST	2025-10-17 12:08:12 CEST	148

Figure 75: Downloads

By filtering through the file we identified from the downloads folder, only the following is found:



Listing
Keyword search 1 - npp.8.8.6.Inst... ×
Keyword search 2 - npp.8.8.6.Inst... ×
Keyword search
Table
Thumbnail
Summary
4 Result
Save Table as CSV

Name	Keyword Preview	Location	Modified Time	Change Time	Access Time	Created Time
Web Downloads Artifact	ers/costa/downloads/<npp.8.8.6.installer.x64.exe>date accessed : 202	/img_malware_test1.E01/vol_vol6/Users/costa/Deskto...	2025-10-19 16:36:27 CEST	2025-10-19 17:07:50 CEST	2025-10-19 16:36:27 CEST	2025-10-17 12:04:44 CEST
IconCache.db	ers\costa\downloads\<npp.8.8.6.installer.x64.exe>lyn c:\windows\system3	/img_malware_test1.E01/vol_vol6/Users/costa/AppDat...	2025-10-22 23:45:13 CEST	2025-10-22 23:45:13 CEST	2025-10-22 23:45:13 CEST	2025-10-17 13:00:38 CEST
Web Downloads Artifact	ers/costa/downloads/<npp.8.8.6.installer.x64.exe>date accessed : 202	/img_malware_test1.E01/vol_vol6/Users/costa/AppDat...	2025-10-22 16:16:17 CEST	2025-10-22 16:16:17 CEST	2025-10-22 16:16:17 CEST	2025-10-17 12:04:44 CEST
0	: "exe", "names": ["<npp.8.8.6.installer.x64.exe>", "f_00011e", "note	/img_malware_test1.E01/vol_vol6/Users/costa/AppDat...	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00	0000-00-00 00:00:00

Figure 76: Filter

Therefore, it's plausible that the file was downloaded from the internet (unfortunately, there's no evidence of PowerShell processes). The tool still allows us to analyze the artifacts, thanks to the presence of the entire file system, but at this point, tools that can display them to the analyst are needed.

4.7 Results

Below are the results obtained by performing analyses with the suite (KAPE + Zimmerman Tools) and Autopsy.

Autopsy: On a VM with approximately 100GB of space, of which approximately 20GB was occupied by the Win10 OS, default applications, and a few other installed tools, it took about 15 minutes to create a forensic copy in E01 format using FTK Imager and then import it with Autopsy. Therefore, on a corporate workstation where an employee performs daily tasks, where EDR software, SIEM agents, corporate authentication processes, and software updates are running in the background, and where the disk contains many files, it's easy to imagine that acquisition times can be extended to several hours, and if server acquisition were necessary, it could even take days. Furthermore, another slowdown in analysis was observed when using the Autopsy modules: the use of Keyword Indexing, Hash Lookup, Picture Analyzer, and Extension Mismatch Detector, although they allow for a deeper level of investigation at the file system level and assist the analyst with keyword searches, significantly slows down the start of the investigation. Indeed, even simply disabling them took about 4 hours before the ingest operation was completely completed. While it is still possible to work and perform initial investigations, the ingest operation of the modules significantly slows down Autopsy's responsiveness. Therefore, considering the time factor alone, we note that with a tool like Autopsy, the Incident Response process begins to become burdensome, and the MTTR increases dramatically. With the open-source toolkit, however, we encounter a small initial delay due to the time it takes to install the tools on the infected system, but after a few minutes, forensic analysis can begin immediately.

Regarding stability and efficiency:

During testing, the Autopsy tool proved slightly less stable than the suite. At times, Java library issues were encountered during startup, and when performing keyword searches—and the entire file system menu was "exploded"—slight graphics rendering issues were encountered, slowing down the forensic analysis process. As previously mentioned, if you perform initial investigations while modules are being ingested, the tool's responsiveness decreases, slowing down the analysis. Although Autopsy focuses on byte-by-byte file system analysis, to obtain highly detailed information, it's still necessary to analyze Windows artifacts and rely on external tools to view them. With the suite, however, I have everything at my fingertips (at the expense of losing byte-level information).

To try to test the actual advantage (in terms of timing, efficiency and depth of analysis) of the suite compared to tools such as Autopsy, it was chosen to test two other case studies:

1. Browsing and anomalous network activity: The goal was to test the tools' ability to identify history, cache, DNS, connections, and links to remote resources. This scenario was used to simulate file downloads and contacts with external domains (these steps can be considered similar to exploit kits/drive-by campaigns: attacks that exploit browser vulnerabilities to force the download/exec of payloads when the victim visits compromised pages).

2. Persistence and benign obfuscation: The goal was to test the tools' ability to find persistence artifacts (Run key, Startup shortcut, Scheduled Task), execution traces (Prefetch, LNK), and PowerShell commands that leave traces in logs. This behavior is similar to that employed by criminal groups attempting (via self-running scripts and obfuscation) to maintain long-term access unnoticed.

To avoid an excessively verbose treatment, only the results obtained will be reported and the scripts used and generated using AI will be reported in the appendix (so that anyone can test these scenarios).

4.7.1 Abnormal Navigation Scenario

Using the suite, it was possible to immediately identify some of the activity performed (URL searches, PowerShell processes, nslookup, and ping). Using the PowerShell logs, it was possible to quickly obtain confirmation of the download of a file (no longer present), while the MFT revealed the moment in which a .txt file was created on the desktop (containing the results of nslookup and ping). By analyzing the $J file (USN Journal, Update Sequence Number Journal, which allows you to reconstruct what happened in the NTFS—useful when an entry is not stored in the MFT for some reason—it was possible to find the creation and deletion timeframe of the sample.bin file. Using the external tool BrowsingHistoryView, it was then possible to analyze the pages contacted by the script.

Moving on to Autopsy, as with script 1, before proceeding with the analysis, it's necessary to perform forensic image acquisition using FTK Imager, then ingest the modules selected on Autopsy. Once this lengthy preliminary phase is complete (even eliminating the noisiest modules requires about 4 hours), the analysis can proceed. Web history analysis is immediate and doesn't require an external tool (which, however, is negligible for analysis purposes). Even with Autopsy, we didn't detect any web resource downloads. However, it's possible to analyze cookies to understand any tracking, which might be considered illicit, but that's not the case here. Analyzing Recent Documents, it's possible to note the network.log.txt file, but without an access date... a sign that the file was created but the user hasn't opened it. As with the suite, analyzing prefetches (possible thanks to file system analysis) allows us to detect Notepad, PowerShell, nslookup, ping, and browser activity. No traces of the sample.bin file were found; it would be necessary to investigate the $J file using a dedicated parser, just as to find the PowerShell process, it would be necessary to analyze the related artifact using a parser.

4.7.2 Persistence Scenario

Using the suite allowed us to quickly detect the creation of the Analyst user name, the detection of a PowerShell script (with the associated Engine state changes), and by opening the logs (without delving into the .evtx files), we can see that an addition was made to the Notepad application shortcuts. These activities are also confirmed by the prefetch folder where the PowerShell process is located. We can see browser navigation processes, the svchost process (related to authentication), and schtasks (the latter allows us to get an idea of some task scheduling, not necessarily due to the script).

From the MFT analysis, in addition to the previous information, we note that a folder called persistenceTest has been created under the Documents path, containing two files (activitylog.txt and appendlog.ps1). We also note and confirm our previous suspicions regarding the creation of a task called PersistenceTest, since this entry is present in the Windows/System32/Tasks path. Finally, there is an entry relating to an https connection to the example.com domain.

In the $J artifacts, it's possible to discover that within a few moments, two text files were also present that were immediately deleted (tempdel1.txt and tempdel2.txt). Since no trace of them was found in the prefetches and caches, it's possible to deduce that these files were created and deleted almost instantly without any interaction (modification, execution, or opening), which also explains their absence from the MFT.

Returning to Autopsy, as long as you always take into account the slowdowns due to forensic acquisition and module ingest:

Web history analysis is immediate and doesn't require an external tool (which, however, is negligible). Even with autopsy, we didn't find any downloads of web resources. It's possible, however, to analyze cookies to understand some tracking, which might be considered illicit, but that's not the case here.

By going to the Run Programs section, you can see the execution of cmd, powershell, net, reg, and edge processes, which can only indicate with certainty that processes have been executed (without knowing their contents), internet activity has been performed (net and edge), and some type of registry activity (reg). Recent Documents provides evidence of the creation of the Documents folder with its files and whether the user opened the folder and files. By browsing the file system, you can find the system startup tasks in Windows/System32/Tasks and you can see our PersistenceTest with its contents. Accessing the MFT reveals information about some deleted files, but to be sure what they are, you need to analyze the MFT and $J using specific tools to extract and view their contents. To analyze the powershell process (if not obfuscated), you need to view its evtx, and therefore you need to use a tool to extract its contents.

4.8 What indications can be drawn?

The previous simulation using malware and "toy" scripts aimed to uncover an investigative methodology that allows the analyst working on a compromised system to uncover all the activities performed. This scientific method allows us to start with a blank sheet of paper and, little by little, by analyzing the logs, gaining in-depth knowledge of the artifacts and the valuable information they provide, rigorously reconstruct the chain of events and all the activities, along with their timeline, performed by the attacker without having to waste precious time acquiring a forensic image or using systems that rely heavily on query language capabilities.

Referring to what was discussed in Chapter 2 and now with all the theoretical and practical knowledge at our disposal, we can reach the following conclusions: The adoption of the KAPE, Hayabusa, and Zimmerman Tools toolchain, with Timeline Explorer as the aggregator, is not just an operational preference, but a methodological choice that addresses specific critical issues present in other industry solutions, such as Autopsy, Encase, and Velociraptor.

The hybrid approach used in this simulation overcomes previous limitations in these key areas:

1. Speed and Triage (Incident Response): KAPE is specifically designed for triage. Instead of acquiring an entire image, KAPE selectively extracts only the high-forensic artifacts (logs, log files, prefetch, etc.) in just a few minutes. This dramatically reduces the Mean Time To Respond (MTTR)—a metric that explodes when considering Autopsy;

2. Active and Guided Threat Hunting: This is Hayabusa's main advantage. Instead of passive analysis, Hayabusa actively analyzes event logs (collected by KAPE) using a vast set of Sigma rules. It immediately returns a report classifying suspicious activity by criticality level (e.g., High, Critical);

3. Artifact Depth and Updates: Eric Zimmerman's tools (EZ Tools) are constantly maintained and updated. When Microsoft adds a new artifact or modifies the format of an existing one, EZ Tools is almost always the first to support it. Large commercial platforms (like Encase) can take months (or longer) to update their parsers. Often, the parsers integrated into suites don't extract all the data fields that a specialized tool like AmcacheParser does;

4. Transparency: Command line tools (CLI) like EZ Tools and Hayabusa generate output in open, readable formats (CSV, JSON, TSV). The analyst has full control and visibility of the raw and parsed data. Platforms, on the other hand, often store data in proprietary databases, making verification and export more difficult.

5. Flexibility and Scripting: As command-line tools, they can be easily incorporated into scripts and automated processes. You can create a chain where KAPE collects data, automatically passes it to EZ Tools and Hayabusa for parsing, and finally uploads the resulting CSVs to a timeline tool.

Focusing further on the artifact aspect: the use of Zimmerman Tools is artifact-centric. The analyst consciously runs, for example, PECmd.exe to parse prefetch files, or EvtxECmd.exe to parse logs. This forces a deeper understanding.

This provides a more granular understanding of how the operating system records actions and teaches how to manually correlate events: the analyst sees a "High" alert in Hayabusa at 10:30 and can immediately filter the PECmd and MFTECmd output in Timeline Explorer to see which files were executed and created at that exact time.

Furthermore, while tools like Autopsy and EnCase provide a slightly more simplified "top-down" view, the combined use of KAPE, Hayabusa, and Zimmerman Tools forces the analyst to operate at a lower, more technical level, rewarding him with a more solid understanding of the evidence and a much faster (since there is no need to perform forensic image acquisition and subsequent ingest of the modules) and more efficient threat hunting process, fundamental for modern Digital Forensics and Incident Response (DFIR).

In particular, using the suite resulted in a 30% reduction in MTTR. As previously mentioned, the suite's great power lies in its ability to perform rapid and efficient triage, allowing analysts to immediately have the tools to conduct their investigations. Using Autopsy, in addition to lower efficiency in triage analyses, requires long image acquisition and module ingest times to perform the analysis, and the dramatic reduction in investigation times is immediately apparent.

The MTTR reduction was empirically observed during the analyses, however, if we wanted to expand on this, there are factors that can impact and vary the obtained value. Starting from the previous 30%, if we consider factors that could influence the percentage, such as greater analyst experience and a more performing hardware workstation, the reduction in MTTR increases and therefore 30-35% can be considered a realistic estimate applicable in multiple contexts.

The following table summarizes the differences between using the suite and Autopsy in the three simulated scenarios.

Table 4: Comparison between KAPE+Hayabusa+Zimmerman and Autopsy suites in the three simulation scenarios

Scenario / Script	Instruments	Total analysis time (min)	Artifacts identified	Analytical depth	Main observations
Script 1 – Abnormal Activity (browser history analysis, user creation, software downloads, file creation and deletion)	KAPE + Hayabusa + Zimmerman	≈ 90	browser history, prefetch, shellbags, Recycle Bin activity, user log, notepad download, MFT, Security.evtx, $J, Amcache	Alta	Very fast analysis; selective artifact extraction; immediate parsing with Timeline Explorer; requires manual correlation and deep expertise on the part of the analyst to investigate and understand the information coming from the artifacts. All the evidence in the script was found by simply adding a browser history tool.
	Autopsy	≈ 45 (+ 240 per ingest)	(browser history, cache, browser customizations, entire file system byte-for-byte including temporary files)	Media	Full image ingest; accurate identification but long indexing and parsing times; GUI subject to lag. To complete the images by finding all the evidence of the script, tools to analyze the artifacts are required.
Script 2 – Abnormal Navigation (browser history analysis, prefetch, network activity, file downloads)	KAPE + Hayabusa + Zimmerman	≈ 90	Browser history, prefetch, ping and nslookup, shellbags, MFT, processes indicating resource download, instant creation and deletion of two text files	High	Very fast analysis; selective artifact extraction; immediate parsing with Timeline Explorer; all script evidence was found by simply adding a browser history tool
	Autopsy	≈ 30 (+ 240 per ingest)	Browser history including cache, filled forms, entire file system byte-for-byte	full image media	ingest; accurate identification but slow indexing and parsing; GUI prone to lag.
Script 3 – Persistence (user creation, Run/Services registry keys, scheduled tasks)	KAPE + Hayabusa + Zimmerman	≈ 90	Tasks found: New Notepad Open Task, Browser History, Prefetch, Registry Activity, New User and Network Activity, Shellbags, User Creation Log, MFT, Folder Creation and PowerShell Process Activity, Security.evtx, $J, 2 File Creation and Deletion, Amcache	High	Excellent Registry and System File Change Detection; MFTECmd + EvtxECmd detected Account Creation; Immediate Results.
	Autopsy	≈ 60 (+ 240 per ingest)	browser history including cache, filled forms, entire file system byte-for-byte	Media	By browsing the file system you can find the new task but you need to use tools to fully analyze the MFT data

4.9 Autopsy Inefficiencies

The comparative analysis reveals substantial architectural differences between the tools analyzed. In particular, Autopsy stands out for its monolithic and centralized approach, while tools such as KAPE, Hayabusa, and Zimmerman Tools adopt a more modular and task-oriented approach.

Autopsy's lower performance can be explained by a combination of technical and architectural factors, as follows:

- Java Architecture and Sleuth Kit Framework: Autopsy is developed in Java and relies on the Sleuth Kit framework, a set of C/C++ libraries invoked via JNI (Java Native Interface). This design ensures portability and interoperability across different operating systems, but introduces computational overhead due to constant bridging between native code and the JVM environment, dynamic memory management (garbage collection), which can introduce significant latencies when parsing large disk images, I/O stream abstraction, which reduces efficiency in bulk parsing operations compared to natively compiled CLI tools, and the full ingest mechanism.
- Autopsy uses a full data ingestion paradigm: the entire forensic image is analyzed, indexed, and stored in an internal database (typically PostgreSQL or SQLite). This approach, while ensuring completeness and consistency of the evidentiary chain, drastically penalizes processing time (Mean Time To Respond), as even artifacts irrelevant to investigative purposes are analyzed, full-text indexing and the creation of unique hashes for each file generate high CPU and I/O load, and the graphical interface waits for the conclusion of some tasks to update, creating GUI blocking phenomena.

As a result, Autopsy is poorly suited for rapid triage, where the goal is to extract key indicators of compromise (IOCs) in just a few minutes.

Regarding the GUI and I/O optimization, the following is noted:

- Autopsy's Swing GUI, while intuitive and useful for training or teaching purposes, negatively impacts performance. Many operations (loading forms, updating panels, refreshing results) occur on the GUI's main thread, temporarily blocking parallel processing and preventing efficient use of multicore CPUs. In contrast, CLI tools like KAPE or Hayabusa take full advantage of parallelism, dividing parsing across multiple independent threads;
- Autopsy is designed to operate on large forensic images (E01, RAW, AFF), accessing the files through an abstraction layer that translates logical offsets into physical offsets. This abstraction ensures forensic traceability, but results in lower I/O throughput than tools that operate directly on exported log files.

Autopsy performs hashing, carving, keyword search, timeline, and artifact correlation functions in a single integrated environment and also rigorously ensures digital chain of custody, a key aspect in forensic investigations.

Autopsy's lower operational efficiency is not a flaw but the result of an architectural choice. In reactive SOC or DFIR contexts, where the goal is to reduce MTTR, modular, command-line-based tools like KAPE or Hayabusa are more appropriate. However, in forensic contexts, where evidentiary integrity and comprehensive documentation are required, Autopsy remains preferable despite its inefficiency.

5 Conclusion

This discussion has highlighted how log analysis, the integration of SOC and DFIR departments, and the adoption of advanced detection technologies are now fundamental pillars of corporate cybersecurity. The increasing complexity of IT infrastructures, combined with the growing sophistication of cyber threats, requires organizations to adopt a structured and multidisciplinary approach: the simple adoption of technical solutions is no longer sufficient without well-defined incident response processes and specialized security event management personnel.

The modern SOC goes beyond passive monitoring to become a decision-making center based on automation, artificial intelligence, and proactive analysis. In this context, collaboration with DFIR teams takes on a strategic role, enabling a security incident to be transformed into a source of useful knowledge for continuously improving the organization's defensive posture.

The adoption of tools such as EDR, XDR, SIEM, and SOAR allows for the correlation of data from heterogeneous sources, the automation of responses, and the reduction of detection and remediation times. At the same time, Digital Forensics practices ensure the proper collection and preservation of digital evidence, allowing incidents to be managed in compliance with current regulations and with methodological rigor.

The case study demonstrated how the integration of log analysis, forensic investigations, and threat hunting allows for the identification of compromise patterns that are difficult to detect with traditional tools, demonstrating the effectiveness of a DFIR approach driven by intelligence and automation.

Furthermore, the simulations presented in Chapter 4 experimentally quantified how tool choice directly impacts Mean Time To Respond (MTTR), a metric now considered critical in SOC and DFIR. The modular approach based on KAPE, Hayabusa, and Zimmerman Tools has been shown to dramatically reduce the time required to identify, extract, and correlate artifacts compared to monolithic solutions like Autopsy, which require comprehensive and much slower ingestion processes. This time reduction—in some scenarios exceeding 30%—is not only an operational advantage, but also directly impacts the organization's ability to limit the attacker's window of action, reduce the risk of lateral movement, and contain the incident in its initial stages. The results thus confirm what has also emerged in the most recent scientific literature: a lower MTTR corresponds to greater resilience and a more effective response to modern threats.

Looking ahead, the evolution of corporate cybersecurity will inevitably hinge on the combination of human expertise and the analytical capabilities of artificial intelligence. Investing in training, processes, and integrated technologies is no longer an option, but a prerequisite for ensuring organizations' digital resilience and business continuity.

6 Appendix

Below we will report the DFIR best practices on Windows systems (especially considerations and event IDs to pay attention to during the analysis) and, subsequently, the .bat scripts used.

6.1 DFIR Windows Best Practices

The most important Windows artefacts to be collected in a centralised manner are Windows event logs and Sysmon events. Windows event logs can be split into four main categories:

- Security: records access control and security settings
- System: records events related to services, system components, drivers, etc.
- Application: records events logged by applications
- Custom: records special events such as PowerShell, Windows Management Instrumentation, task scheduler, etc.

The relevant EIDs for this use case allow investigations to track suspect/compromised account activities.

- EID 4624/4634: “An account was successfully logged on” and ”An account was logged off” - this EID pair reveals a logon/logoff session. 4624 also contains the logon type that the user performed, the most relevant of which include:
  - Type 10: logged for new remote desktop connections – remote interactive
  - Type 12: logged for new remote desktop connections - cached credentials
  - Type 7: logged for existing remote desktop connections – workstation was unlocked
  - Type 2: logged for activities performed from consoles, command lines (e.g., runas command), and virtual clients
  - Type 3: logged for network logons, such as SMB activities, some RDP connections These EIDs usually contain the Account Name, SID, and Logon ID that allow identification of the timeframe a user was active on a host.
- EID 4634 is not always consistent in showing users’ logon/logoff timeframe activity – e.g., during network connections (type 3) such as to a file server, 4634 might be logged many times a day due to the way services handle idle connection termination. Another example of an inconsistency is when a user turns off the computer, in which case EID 4634 is recorded only after the system restarts resulting in misleading evidence since the timestamp does not reflect the exact time the user logged off
- EID 4625: “An account failed to log on” - shows logon failures, making it useful to check for evidence of password guessing attempts or brute force attacks
- EID 4648: “A logon was attempted using explicit credentials” - reveals commands with explicit credential usage e.g., runas command or applications executed with administrative credentials entered by the user, as well as outgoing connections; this EID is therefore useful in investigating lateral movement and privilege escalation activities
- EID 4672: “Special privileges assigned to new logon” - tracks logons with administrative privileges and can also be paired with EID 4624.

Account creation and permission Amidst their malicious activity, adversaries might try to create and eliminate accounts, respectively:

- EID 4720: “A user account was created”
- EID 4726: “A user account was deleted” A local admin can create local accounts, whereas a domain admin can create domain-wide accounts of any type to evade defences, including sleeper accounts (dormant accounts that have not been used in a long time but maintain any privileges they were given originally).

Account auditing is tied to several useful EIDs that allow adversaries to maintain their persistence and move around the environment; such EIDs include:

- EID 4722: “a user was enabled”
- EID 4724: “password reset attempt”
- EID 4728:” a member was added to a security-enabled global group”
- EID 4732: “a member was added to a security-enabled local group”
- EID 4735: “a security-enabled local group was changed”
- EID 4738: “a user account was changed”
- EID 4756: “an account was added to a security-enabled universal group”

6.1.1 Credential authorisation

In Windows domain environments, most user accounts are domain accounts. Their credentials are stored remotely on the domain controller. When a user logs on to their workstation, their credentials are validated remotely using either NTLM or Kerberos authentication protocols:

- Where NLTM is used: EID 4776: “The domain controller attempted to validate the credentials for an account”, is generated for both success and failure logon events on the domain controller;
- Where Kerberos protocol is used, EID 4768: “A Kerberos authentication ticket (TGT) was requested” or 4769: “A Kerberos service ticket was requested” - generated for success events EID 4771: “Kerberos pre-authentication failed” - logged for failures All the above events are useful during investigations in identifying and confirming account abuse for lateral movement and similar purposes. However, if the user logs in using a local account (not part of the domain), the above events for credential authorisation will be logged locally. This is generally rare in an enterprise environment and might be worth investigating as it may be indicative of rogue accounts on the local system

6.1.2 Remote desktop connection

Malicious actors frequently perform connections and lateral movement using the Remote Desktop Protocol, which can work to an investigator’s advantage in identifying the timeframe of the attacker’s activity. There is a relatively large number of EIDs that can be checked. The Remote Desktop Connections, Security.evtx, EIDs 4778 (Session reconnected) and 4779 (session disconnected) together with EID 4624 (Account usage) help to identify any “stale” RDP session (already existing connections). A parameter which can be useful to reconstruct the chain of events is the logon id. When a user establishes a new RDP connection, there are several artefacts that allow an investigator to rebuild the history of the connection. On the RDP destination host, the EIDs to be collected include:

- Microsoft-Windows-TerminalServices-RemoteConnectionManager/Operational.evtx o EID 261: “Listener X received a connection”

- EID 1149: "User authentication succeeded" - despite the description, this refers only to successful network authentication (e.g., a user attempted to RDP into the target machine, which successfully responded and displayed a login screen to enter credentials). If NLA is not enabled, this EID is not recorded
- EID 1158: "Remote Desktop Services accepted a connection from IP address" (displayed)
- EID 21: "Remote Desktop Services: Session logon succeeded" - provides the remote IP and session ID
- EID 22: "Remote Desktop Services: Shell start notification received" - provides the remote IP and session ID
- EID 23: "Remote Desktop Services: Session logoff succeeded:" - provides the remote IP and session ID
- EID 24: "Remote Desktop Services: Session has been disconnected" - provides the remote IP and session ID
- EID 25: "Remote Desktop Services: Session reconnection succeeded" - provides the remote IP and session ID
- EID 39: "Session has been disconnected by session " - shows a user that has formally disconnected from an RDP session with the explicit Disconnect option (i.e.., via the Windows Start Menu Disconnect option). When the Session ID of differs from , this may indicate a separate RDP session due to disconnection/reconnection activities
- EID 40: "Session has been disconnected, reason code "
- EID 131: "The server accepted a new TCP connection from client SOURCE IP:PORT".
- EID 140: "A connection from the client computer with an IP address of SOURCE IP failed because the username or password is not correct" – this event is recorded only when the failure is due to a non-existent account
- EID 1024: "RDP ClientActiveX is trying to connect to the server " - shows the connection attempt (both successful and failed) and the destination hostname
- EID 1026: "RDP ClientActiveX has been disconnected (Reason=x)" - shows the end of the outgoing RDP connection and the reason for disconnection
- EID 1029: "Base64(SHA256(UserName)) is = ", it shows the Base64 for the hashed username of the account that is attempting to connect. Useful to track down a suspicious account's activity
- EID 1102: "The client has initiated a multi-transport connection to the server ." - Useful to identify the remote IP address and whether the connection was successful

6.1.3 Scheduled task activity

Attackers can often be seen abusing scheduled tasks in several cases, for example they can use them to spread and execute malware such as ransomware across the environment to execute encryption jobs remotely, with the goal of compromising as many endpoints as possible in the shortest amount of time. There are two main sources to collect for scheduled tasks: Security.evtx and Microsoft-Windows-Task Scheduler/Operational.

- Task Scheduler/Operational
  - EID 106: "Scheduled Task created"

  - EID 140: “Scheduled Task updated”
  - EID 141: “Scheduled Task deleted”
  - EID 200: “Scheduled Task executed” – it is not present in Security.evtx, shows task name and full path of the service/application executed
  - EID 201: “Scheduled Task completed” – it is not present in Security.evtx, shows task name and full path of the service/application executed

- Security.evtx, shows complete information about task proprieties, trigger conditions, account name, full path of the service/application to be executed in the following EIDs,
beginitemize

- EID 4698: “Scheduled Task created” o EID 4699: “Scheduled Task deleted”

- EID 4700: “Scheduled Task enabled”

- EID 4701: “Scheduled Task disabled”

- EID 4702: “Scheduled Task updated

6.2 Script1.bat

Listing 1: Windows Automation Batch Script

```
@echo off
REM 0. Crea un utente Windows chiamato "Test"
echo Creazione utente Windows "Test"...
net user Test /add /active:yes

REM 0.1 Scarica Notepad in automatico
echo Download di Notepad in corso...
powershell -WindowStyle Hidden -Command "Invoke-WebRequest -Uri ’https://github.
    ↪ com/notepad-plus-plus/notepad-plus-plus/releases/download/v8.8.6/npp
    ↪ .8.8.6.Installer.x64.exe’ -OutFile \"$env:USERPROFILE\Downloads\npp
    ↪ .8.8.6.Installer.x64.exe\""

REM 1. Apre il Blocco Note
echo Avvio Blocco Note...
start notepad.exe

REM 2. Apre il browser di default ed esegue una ricerca su Google
echo Avvio ricerca Google per "Test"...
start "" "https://www.google.com/search?q=Test"

REM Crea file temporaneo sul Desktop
echo Questo file verra’ spostato nel Cestino. > "%USERPROFILE%\Desktop\
    ↪ file_temporaneo.txt"

REM Crea file temporaneo sul Desktop
echo Guarda nel cestino. > "%USERPROFILE%\Desktop\guarda.txt"

REM Sposta il file nel Cestino usando PowerShell
powershell -WindowStyle Hidden -Command "Add-Type -AssemblyName Microsoft.
    ↪ VisualBasic; [Microsoft.VisualBasic.FileIO.FileSystem]::DeleteFile(\"$env
    ↪ :USERPROFILE\\Desktop\\file_temporaneo.txt\", ’OnlyErrorDialogs’, ’
    ↪ SendToRecycleBin’)"
REM 5. Simula una connessione a 127.0.0.1 tramite Bing
```

```
echo Connessione a 127.0.0.1 da Bing...
start "" "https://www.bing.com/search?q=127.0.0.1"

echo -----------------
echo Script completato.
echo -----------------
pause
```

6.3 Script2.bat

```
    @echo off
echo ===== SCENARIO 2: Attivit  di rete =====

REM 1. Simula ricerche e accessi web
start "" "https://www.wikipedia.org/"
timeout /t 3
start "" "https://www.torproject.org/"
timeout /t 3
start "" "https://github.com/"
timeout /t 3

REM 2. Esegue ping e nslookup
ping google.com > "%USERPROFILE%\Desktop\network_log.txt"
nslookup github.com >> "%USERPROFILE%\Desktop\network_log.txt"

REM 3. Scarica file eseguibile (benigno)
powershell -WindowStyle Hidden -Command "Invoke-WebRequest 'https://nbg1-speed.
    ↪ hetzner.com/100MB.bin' -OutFile \"$env:USERPROFILE\Downloads\sample.bin
    ↪ \""

REM 4. Cancella il file dopo pochi secondi
timeout /t 5
del "%USERPROFILE%\Downloads\sample.bin"

REM 5. Naviga verso IP diretto
start "" "http://142.250.185.196"

echo ----- Scenario 2 completato -----
pause
```

6.4 Script3.bat

```
    @echo off
echo ===== SCENARIO 3: Persistenza e offuscamento benigno =====

REM 0. Crea utente "Analista"
net user Analista /add /active:yes
REM non aggiungiamo ai gruppi di admin per minimizzare impatto

REM 1. Crea cartella per script benigni
mkdir "%USERPROFILE%\Documents\persistenza_test"
echo Log attivit  di persistenza > "%USERPROFILE%\Documents\persistenza_test\
    ↪ activity_log.txt"

```

```
REM 2. Crea un piccolo script PowerShell che scrive sul log (file .ps1)
set PSFILE=%USERPROFILE%\Documents\persistenza_test\append_log.ps1
echo Add-Content -Path "%USERPROFILE%\Documents\persistenza_test\activity_log.txt" -Value ("Eseguito a: " + (Get-Date)) > "%PSFILE%"

REM 3. Crea un Scheduled Task che esegue lo script ogni giorno (persistenza innocua)
schtasks /Create /SC ONCE /TN "PersistenzaTest" /TR "powershell -ExecutionPolicy Bypass -File \"%PSFILE%\"" /ST 23:59 /F

REM 4. Aggiungi chiave Run nel registro per avviare notepad all'accesso (persistenza semplice)
reg add "HKCU\Software\Microsoft\Windows\CurrentVersion\Run" /v "NotepadPersist" /t REG_SZ /d "notepad.exe" /f

REM 5. Crea shortcut nella cartella Startup dell'utente che punta a notepad
set STARTUP="%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup"
powershell -Command "$s=(New-Object -COM WScript.Shell).CreateShortcut('%APPDATA%\\Microsoft\\Windows\\Start Menu\\Programs\\Startup\\open_notepad.lnk'); $s.TargetPath='C:\\Windows\\System32\\notepad.exe'; $s.Save()"

REM 6. Esegue un comando PowerShell "offuscato" benigno (base64) che crea un file marker (simula tecniche di offuscamento)
set MARKER=%USERPROFILE%\Documents\persistenza_test\marker.txt
powershell -EncodedCommand JABwID0gIkhlbGxvLCB0aGlzIGlzIGEgYmVuaWduIG1hcmtlciI7IFtTeXN0ZW0uS
U8uRmlsZV06OkNvbnRlbnRbQS1dID0gJHBzOyAkcHMgPiA%3D


REM 7. Lancia qualche applicazione per generare Prefetch / LNK / Amcache artefacts
start notepad.exe
start "" "https://www.example.com"
timeout /t 2

REM 8. Crea e cancella alcuni file per lasciare tracce in $MFT e possibile carving
echo file temporaneo 1 > "%USERPROFILE%\Documents\temp_del_1.txt"
echo file temporaneo 2 > "%USERPROFILE%\Documents\temp_del_2.txt"
del "%USERPROFILE%\Documents\temp_del_1.txt"
del "%USERPROFILE%\Documents\temp_del_2.txt"

REM 9. Aggiorna il file di log locale
echo Scenario persistente eseguito >> "%USERPROFILE%\Documents\persistenza_test\activity_log.txt"

echo ----- Scenario 3 (persistenza & offuscamento) completato -----
pause
```

7 Glossary

SOC: Security Operation Center
DFIR: Digital Forensics and Incident Response
EDR: Endpoint Detection and Response
XDR: Extended Detection and Response
SIEM: Security Information and Event Management
SOAR Security orchestration, automation and response
PMI: Piccole e Medie Imprese
ASL: Azienda Sanitaria Locale
DDoS: Distributed Denial of Service
ROI: Return On Investment. Metric that measures the profitability of an investment, calculating how much profit was generated compared to the capital initially spent
ALE: Annualized Loss Expectancy. Estimate in euros of the economic damage expected each year from a specific risk, such as a cyber attack
ARO: Annualized Rate of Occurrence. Estimate of the probability that a risk or accident will occur in a year.
SLE: Single Loss Expectancy. It is the estimated economic damage for a single risk event and is calculated by multiplying the asset value (AV) by the exposure factor (EF), which represents the percentage of expected loss.
IDS: Intrusion Detection System
IPS: Intrusion Prevention System
APT: Advanced Persistent Threats
IOC: Indicator Of Compromise
IoT: Internet of Things
BYOD: Bring Your Own Device
SaaS: Software as a Service. Cloud computing model in which a provider hosts applications and makes them available to customers over the Internet, often through a subscription
VPN: Virtual Private Network
CASB: Cloud Access Security Broker. A security solution that sits between users and cloud services to monitor traffic, enforce security policies, and protect application data.
CVE: Common Vulnerabilities and Exposures
MTTD: Mean Time To Detect
MTTI: Mean Time To Identify
MTTI: Mean Time To Detect e To Respond
C&C: Command and Control. To simplify, these are servers managed by a hacker to send commands to devices compromised by malware and receive stolen data (exfiltration)
OSINT: Open Source Intelligence
CSIRT: Computer Security Incident Response Team. It is an entity dedicated to the management of cyber security incidents
CIRT: Cyber Incident Response Team
CERT: Computer Emergency Response Team
CISO: Chief Information Security Officer. The CISO manages an organization's cybersecurity strategy and its implementation to ensure that systems, services, and digital assets are adequately secure and protected.
CIO: Chief Information Officer. He is the overall person responsible for the strategy and management of information technology (IT) of a company.
GDPR (General Data Protection Regulation): European Union Regulation on the processing of personal data and privacy of 2016
NIS2: European Cybersecurity Directive that aims to strengthen cybersecurity in the EU by unifying standards for a large number of critical and highly critical sectors
ENISA: European Union Agency for Cybersecurity. It is the EU agency that aims to achieve a high

common level of cybersecurity in Europe.
UEBA: User and Entity Behavior Analytics (User and Entity Behavior Analysis) is a cybersecurity technology that uses algorithms and machine learning to identify anomalous and suspicious behavior within a network.

References

[] *AI log analysis.* https://logz.io/blog/ai-log-analysis/.

[] *AI Malware Analysis: Types of Malware.* https://www.youtube.com/watch?v=RULzj0DpGfw.

[] *AI malware detection rates.* https://www.infosecurity-magazine.com/news/ai-malware-detection-rates/.

[] *AI-based malware detection.* https://www.manageengine.com/academy/ai-based-malware-detection.html.

[] *ArXiv Article.* https://arxiv.org/html/2409.11313v1.

[] *Autopsy Documentation.* https://www.sleuthkit.org/autopsy/docs/.

[] *Autopsy GitHub.* https://github.com/sleuthkit/autopsy.

[] *Computer forensics - IBM.* https://www.ibm.com/it-it/topics/computer-forensics.

[] *Cortex XSOAR + Greynoise Integration.* https://www.greynoise.io/resources/video-demo-cortex-xsoar-greynoise-integration.

[] *Cost of computer intrusions in Italy.* https://www.innovationpost.it/tecnologie/industrial-security/quanto-costano-le-intrusioni-informatiche-in-italia-ben-437-milioni-di-euro-in-media-oltre-5-milioni-per-le-aziende-industriali/.

[] *Crowdstrike Malware Analysis.* https://www.crowdstrike.com/en-us/cybersecurity-101/malware/malware-analysis/.

[] *Cyber Kill Chain.* https://www.microsoft.com/it-it/security/business/security-101/what-is-cyber-kill-chain.

[] *Cyber threat hunting.* https://www.microsoft.com/it-it/security/business/security-101/what-is-cyber-threat-hunting.

[] *Cybersecurity: Differences Between EDR, SIEM, SOAR, and XDR.* https://www.rbrverona.it/2024/01/09/cybersecurity-quali-differenze-tra-edr-siem-soar-e-xdr/.

[] *Difference Between SOAR vs XSOAR vs SIEM vs XDR.* https://medium.com/@cloud_tips/what-is-the-difference-between-soar-vs-xsoar-vs-siem-vs-xdr-e7d0dca8059.

[] *Digital Forensics and Incident Response.* https://www.crowdstrike.com/en-us/cybersecurity-101/exposure-management/digital-forensics-and-incident-response-dfir/.

[] *EDR vs XDR.* https://www.microsoft.com/it-it/security/business/security-101/edr-vs-xdr.

[] *How AI is shaping malware analysis.* https://blog.virustotal.com/2023/11/how-ai-is-shaping-malware-analysis.html.

[] *How AI-generated malware is changing cybersecurity.* https://www.impactmybiz.com/blog/how-ai-generated-malware-is-changing-cybersecurity/.

[] *How to be a security analyst.* https://www.cybersecurity360.it/outlook/come-diventare-security-analyst-carriera/.

[] *IBM - What is SIEM.* https://www.ibm.com/it-it/topics/siem.

[] *IBM MaaS360 BYOD.* https://www.ibm.com/it-it/products/maas360/byod.

[] *IBM QRadar Architecture.* https://syedhasan010.medium.com/ibm-qradar-the-architecture-e09721ba3205.

[] *IBM Security Orchestration Automation Response.* https://www.ibm.com/it-it/topics/security-orchestration-automation-response.

[] *Incident Response.* https://www.microsoft.com/it-it/security/business/security-101/what-is-incident-response.

[] *Indicators of Compromise (IOC).* https://www.microsoft.com/it-it/security/business/security-101/what-are-indicators-of-compromise-ioc.

[] *Information security privacy risks and AI.* https://www.linkedin.com/pulse/information-security-privacy-risks-ai-andre-gorvel-fciis-0soae/.

[] *Integrated threat prevention.* https://www.cybersecurity360.it/whitepapers/prevenzione-integrata-dalle-minacce/.

[] *Italy in the crosshairs of cybercriminals.* https://www.innovationpost.it/attualita/italia-sempre-piu-nel-mirino-dei-criminali-informatici-attacchi-in-crescita-del-65/.

[] *Log analysis process and best practices.* https://www.exabeam.com/explainers/log-management/what-is-log-analysis-process-techniques-and-best-practices/.

[] *Log monitoring.* https://www.splunk.com/en_us/blog/learn/log-monitoring.html.

[] *Logs and SIEM: IT Monitoring Solutions.* https://www.cybersecurity360.it/soluzioni-aziendali/log-e-siem-soluzioni-per-dotarsi-un-efficiente-sistema-di-monitoraggio-dei-sistemi-it/.

[] *Malware Analysis.* https://www.xcitium.com/malware-analysis/.

[] *Malware Analysis | Static and Dynamic.* https://www.youtube.com/watch?v=dfNeB5AixMw.

[] *Malware Behaviour Analysis.* https://www.researchgate.net/publication/220673433_Malware_behaviour_analysis.

[] *Microsoft Defender - behavior monitor.* https://learn.microsoft.com/it-it/defender-endpoint/behavior-monitor.

[] *Palo Alto Cortex XSOAR.* https://www.paloaltonetworks.it/cortex/cortex-xsoar.

[] *SANS Blog.* https://www.sans.org/blog.

[] *SentinelOne – differenze EDR SIEM SOAR XDR.* https://cio.florence-consulting.it/sentinelone-differenze-edr-siem-soar-xdr.

[] *SIEM - Checkpoint.* https://www.checkpoint.com/it/cyber-hub/cyber-security/what-is-siem-security-information-and-event-management/.

[] *SIEM Model Architecture.* https://vitolavecchia.altervista.org/architettura-del-modello-siem-log-sicurezza/.

[] *SOC Analyst: competenze.* https://www.buckler.it/buckler-journal/articoli/soc-analyst-chi-e-quali-competenze.html.

[] *Splunk Enterprise Security.* https://vietsunshine.com.vn/en/solution/splunk-enterprise-security-sl38.

[] *Static vs Dynamic Malware Analysis - Comparison Chart.* https://www.malwation.com/blog/static-malware-analysis-vs-dynamic-malware-analysis-comparison-chart.

[] *Strategie di sicurezza nel SOC.* https://www.cybersecurity360.it/outlook/strategie-di-sicurezza-e-ritorno-degli-investimenti-nel-soc/.

[] *Stream logs to Event Hub.* https://learn.microsoft.com/it-it/entra/identity/monitoring-health/howto-stream-logs-to-event-hub?tabs=splunk.

[] *The importance of SOC.* https://www.saibaseky.com/importanza-soc-sicurezza-informatica-pmi-soluzioni-flessibili.

[] *Understanding the Difference Between EDR, SIEM, SOAR and XDR.* https://it.sentinelone.com/blog/understanding-the-difference-between-edr-siem-soar-and-xdr/.

[] *What is a cyberattack.* https://www.microsoft.com/it-it/security/business/security-101/what-is-a-cyberattack.

[] *What is a DDoS attack.* https://www.microsoft.com/it-it/security/business/security-101/what-is-a-ddos-attack.

[] *What is a SOC.* https://www.ibm.com/it-it/topics/security-operations-center.

[] *What is a SOC - Checkpoint.* https://www.checkpoint.com/it/cyber-hub/threat-prevention/what-is-soc/.

[] *What is a SOC?* https://www.microsoft.com/it-it/security/business/security-101/what-is-a-security-operations-center-soc.

[] *What is an endpoint?* https://www.microsoft.com/it-it/security/business/security-101/what-is-an-endpoint.

[] *What is CTI?* https://www.sekoia.io/en/glossary/what-is-cti/.

[] *What is cybersecurity.* https://www.microsoft.com/it-it/security/business/security-101/what-is-cybersecurity.

[] *What is insider threat.* https://www.microsoft.com/it-it/security/business/security-101/what-is-insider-threat.

[] *What is malware.* https://www.microsoft.com/it-it/security/business/security-101/what-is-malware.

[] *What is ransomware.* https://www.microsoft.com/it-it/security/business/security-101/what-is-ransomware.

[] *What is SIEM.* https://www.fortinet.com/resources/cyberglossary/what-is-siem.

[] *What is SIEM.* https://www.microsoft.com/it-it/security/business/security-101/what-is-siem.

[] *What is SOAR.* https://www.paloaltonetworks.it/cyberpedia/what-is-soar.

[] *What is SOAR?* https://www.microsoft.com/it-it/security/business/security-101/what-is-soar.

[] *What is vulnerability management.* https://www.microsoft.com/it-it/security/business/security-101/what-is-vulnerability-management.

[21] *Malware Detection using Artificial Intelligence.* https://www.researchgate.net/publication/357163392_Malware_Detection_and_Prevention_using_Artificial_Intelligence_Techniques. 2021.

[Akb24] Ertuğrul Akbaş. "Evaluating SIEM RADAR: A New Metric for Enhancing Regulatory and Compliance Efficiency". In: *Vol. 23 No. 1 (2024): Proceedings of the 23rd European Conference on Cyber Warfare and Security /* 23.1 (2024), pp. 1–9. DOI: https://doi.org/10.34190/eccws.23.1.2346.

[Ali25] Md Liakat Ali. *AI-Powered Incident Response and Automation.* https://www.researchgate.net/publication/396818673_AI-Powered_Incident_Response_and_Automation_Enhancing_Cybersecurity_Resilience_Through_Machine_Learning_and_Orchestration. 2025.

[Kuf25] J. Kuforiji. "Digital Forensics and Incident Response (DFIR) Automation". In: *Journal of Data Analysis and Critical Management* 1.4 (2025), pp. 1–19. DOI: 10.64235/tsvfvz27.