




# Effectiveness of Malware Incident Management in Security Operations Centres: Trends, Challenges and Research Directions

Dakouri Gazo<sup>1</sup><sup>a</sup>, Asma Patel<sup>1,2</sup><sup>b</sup> and Mohammad Hasan<sup>1</sup><sup>c</sup>

<sup>1</sup>*School of Digital Technologies and Arts, Staffordshire University, Stoke-on-Trent, Staffordshire, U.K.*

<sup>2</sup>*Department of Operations and Information Management, Aston University, Birmingham, U.K.*

**Keywords:** Malware, Incident, SOC, Security Operations Centre, Static Challenges, Dynamic Challenges.

**Abstract:** In the ever-changing realm of cybersecurity, protecting digital assets requires constant awareness and rapid incident response in security operations centre (SOC), where security professionals employ cutting-edge threat-fighting strategies. The battle becomes more intense in the face of ever-more complex adversaries, such as advanced and persistent malware. The riddle of malware incidents, on the other hand, provides distinct obstacles, requiring steadfast specialised competence and innovative strategies. Effective incident handling is essential for protecting organisational digital assets, given the ongoing evolution and rising sophistication of cyberattacks. This paper reviews the literature that explores the complexities of the current state of malware event-handling solutions and identifies challenges by delving into SOC operations. It provides the recommendations and guidance necessary to SOC researchers and security professionals, empowering them to tackle malware incidents and strengthen cybersecurity defences.

## 1 INTRODUCTION


Cybersecurity reports (Malwarebytes, 2020) revealed that companies are exposed to multiple risks such as damage to the brand, significant losses, industrial espionage, etc. As a security defence entity, a security operation centre (SOC) is a team of security professionals, who constantly protect an organisation's networks and systems against cyberattacks. SOC's primary role is to coordinate the actions of all other security-related departments to handle cyber incidents and mitigate threats and risks. There is no standard definition or terminology to describe a SOC; other commonly used terms are *Cyber Security Operations Centre (CSOC)*, *Computer Security Incident Response Team (CSIRT)*, *Network Operations Centre (NOC)*, *Network Security Intelligence Centre (NSIC)*. SOC is a combination of technologies, people, and processes (Vielberth, et. al., 2020), its operational goals and objectives vary depending on the specific organisation but generally include protecting assets and managing cyber incidents to secure business operations and services for the organisation.


Therefore, this paper aims to carry out a study on malware incident handling and related challenges in SOCs and highlight emerging research directions. This review objectively focused on the state-of-the-art literature on incident management and malware handling in a SOC to critically analyse the most recent progress and difficulties in the industry based on predetermined standards for rigor and pertinence. The defined research question for establishing the literature search keywords and the inclusion criteria is: What are the trends, challenges, and emerging research directions on the effectiveness of malware incident management within SOCs?


The rest of the paper is organised as, Section 2, 3 and 4 investigate the state-of-the-art malware incident management in a SOC. Section 5 presents related challenges, and Section 6 highlights research directions, followed by the conclusion.

## 2 SOC INCIDENT MANAGEMENT

Malware handling lifecycle in a SOC is a continuous process of detecting, assessing, responding to, and

<sup>a</sup> <https://orcid.org/0009-0005-5204-7526>

<sup>b</sup> <https://orcid.org/0000-0003-1636-5955>

<sup>c</sup> <https://orcid.org/0000-0003-0458-4536>

recovering from security incidents and using lessons learned to improve overall incident management (Jaramillo, 2019). The incident management process encompasses either preparation, detection, analysis, containment, eradication, recovery, and post-incident or identification, protection, detection, response, and recovery capabilities.

**1. Preparation/Identify and Protect** reflects the preparatory measures include obtaining necessary tools and resources, developing and retaining malware-related skills within the incident response team, and enabling communication and coordination in the organisation- also known as the identify and protect phase (Barrett, 2018). It identifies and manages the security risks related to the systems, assets, people, data, and capabilities, developing suitable safeguards to guarantee the delivery of identified critical services.

**2. Detection and Analysis** phase in SOC organisation is to detect and confirm malware incidents rapidly to reduce the number of infected hosts and the amount of damage a business sustains. It involves identifying the incident characteristics (malware category, ports, protocols, exploited vulnerabilities, malicious filenames, etc.), identifying the infected hosts (from a network device, DNS, application server logs, IPS/IDS sensors, manually), engaging incident response, and investigating malware (Souppaya et al., 2013).

**3. Containment, Eradication, and Recovery / Response** phase determines the organisation’s action plan depending on the type of malware incident. This action plan of response includes containment, eradication, and recovery (Ozer, M. et al., 2020).

- **Containment.** It stops the spread of malware and avoids further damage to the network or hosts, for example, by isolating the malware or disconnecting or shutting down the infected host(s).
- **Eradication.** It removes malware from the infected hosts or mitigates the weakness, runs an up-to-date antivirus scan on the infected host(s), applies relevant patches to remove vulnerabilities.
- **Recovery.** Recovering implies restoring the functionality and data of the infected host(s), such as rebuilding the host(s) in the event of considerable damage, recovering a huge number of corrupt/encrypt data and system files or wipe out hard drives by malware.

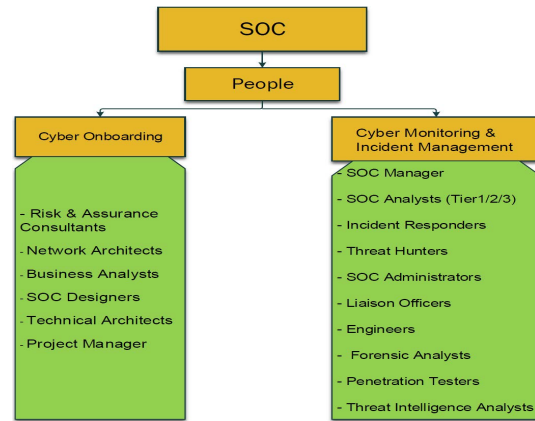


Figure 1: Categories of People in a SOC.

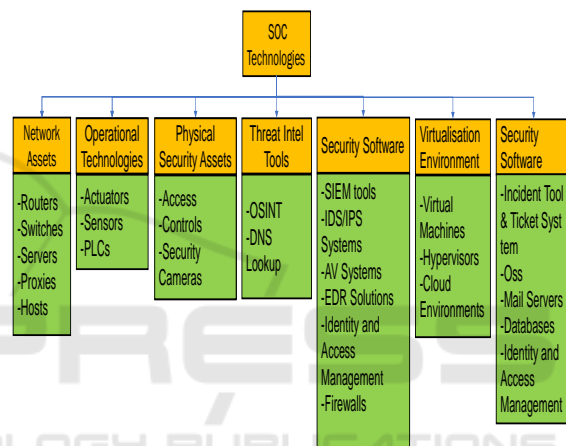


Figure 2: Categories of Technologies in a SOC.

### 3 SOC ARCHITECTURE

This section investigates the three main components of a SOC architecture: people, process, and technology, and their importance in establishing an effective SOC.

#### 3.1 People

People play an important role in the security of businesses. SOC teams are responsible for detecting, addressing support tickets, implementing, configuring, and managing their security infrastructure. From the analyst to the manager, various roles can be identified, whereby a SOC must handle of staffing and recruitment. SOC people can be split into two distinct categories (Onwubiko and Ouazzane, 2019): cyber onboarding people, and SOC monitoring and incident management personnel, as shown in Fig 1. For effective SOC architecture, it is essential to underline areas

```

00011ef0: 8e3e c6d0 d1c4 d1c7 8e3d c6c1 c221 c4c8 .>.....=...!..
00011f00: dac6 d6c2 8e91 9191 918e 4a8c 8b1b aa19 .....J.....
00011f10: 994a 2baa 1b1b c8ce d5ce dcc5 0000 0000 .J+.....
00011f20: 807c 393c 32ba b680 f3b9 b434 b834 3900 .|9<2.....4,49.
00011f30: fcbf 34ba 7cba 3436 b9bc ba3c 807c 393c .4,|46...<|9<
00011f40: 82ba bb76 ba34 3cb9 bfb7 8f30 b3b9 3c32 2..v.4<...0.<2
00011f50: 2012 9751 1556 11a3 5495 55aa b39d a587 ..Q.V..T.U....
00011f60: 91a7 ba85 b393 8d9d bd00 0000 0000 0000 .....
00011f70: 9c85 8927 8b9c 8589 278b 9c85 8927 8b9c .....
00011f80: 8589 278b 9c85 8927 8b9c 8589 270d fd3c .....<

```

```

strings:
$a1 = { 80 7C 39 3C 32 BA BB 80 F3 89 B4 34 BB 34 39 80 }
$a2 = { FC BF 34 BA 7C BA 34 36 B9 BC BA 3C 80 7C 39 3C }
$a3 = { 32 BA BB 76 BA 34 3C B9 BF B7 8F 30 B3 B9 3C 32 }
$b1 = { 9C 85 89 27 8B 9C 85 89 27 8B 9C 85 89 27 8B 9C }
condition:
Macho and filesize < 200KB and all of them

```

Figure 3: Signature-based Detection for Ocean-Lotus malware.

where improvement is needed such as team dynamics, communication patterns, and organisational culture.

### 3.2 Technologies

A categorised and non-exhaustive list of critical technologies related to the SOC is given in Fig 2, which enables a SOC to monitor, detect, and respond to security problems effectively. These technologies provide workflows for incident response, detection of abnormalities and potential risks, and aggregation and correlation of security events. The main characteristics of a SOC are log management, event visualisation, and incident reporting. These three features are intricately related since the collected logs provide input for visualisation, later used to report incidents. Various data collection techniques can be organised into four categories: partial/full collection, real-time/historical, push/pull, and distributed/centralised (Vielberth, et. al., 2020). The collected data/logs are fed into a security information and event management (SIEM) tool (Hossain, et. al., 2021). A non-exhaustive list of technologies related to the SOC can be categorised as in Fig 2. To comprehensively understand SOC architecture, SOC technologies need integration with other elements to reflect the latest advancements.

### 3.3 Processes

Some studies present SOC models and theoretical structures with more endorsement of actual SOC situations; hence, they need real-world confirmation. For example, the cyber incident playbook process (Onwubiko and Ouazzane, 2019) focuses on the importance of developed procedures and instructions to ensure organised and coordinated response actions. It addresses teamwork and communication, incident triage, documentation and reporting, handling workflow, and tools and technologies. The incident response process focuses on incident identification, containment, analysis, mitigation, and post-incident activities. Future research should address these limitations by considering a broader scope, conducting real-world validation, and including pragmatic SOC implementation considerations.

Table 1: Non-exhaustive List of Tools for Malware Analysis.

Category	Tool
Virtualisation	VMWare, VirtualBox Cuckoo Sandbox
Dynamic Analysis	Process Hacker, RegShot, Wireshark, ProcDOT, Wireshark, Fiddler
Static Analysis	(property: PeStudio, Strings, Yara) (code: Ghidra, IDA, OllyDbg)
Memory Analysis	WinPMEM, BelkaSoft Live RAM Capturer, Volatility Framework, ReKall

## 4 MALWARE INCIDENT HANDLING IN SOC ENVIRONMENT

This section delves into the complex terrain of managing malware incidents in SOCs, outlining crucial elements such as automated detection and response, analysis, and detection.

### 4.1 Malware Detection

Malware detection can be performed either automatically or manually. Malware detection methods comprise three types of methods (Guo, et. al., 2020): signature-based, static, and dynamic. Static detection disassembles the malware and analyses the opcodes, static API sequences, and execution logic without running it. Dynamic detection, on the other hand, acquires behavioural features (network activity, system calls, file operations, etc.) by executing the file sample. Signature-based detection works by extracting common characteristics (byte sequence, file size, file hash, imported/exported functions, offsets, strings) for each file and matching them with known signatures that have been collected before. Fig 3 shows a sample of code of the malware *Ocean-Lotus* and its corresponding YARA signature, showing that any file with a size less than 200KB with the type ‘Macho’ containing the strings *a1*, *a2*, *a3*, and *b1* should identify as the threat actor *Ocean-Lotus*. VirusTotal provides a more comprehensive elucidation of YARA rules (VirusTotal, 2022; Coscia et al., 2023).

```

undefined8  Stack[-0x38..local_38  XREF[2]: 14000124a(W),
                                     140001232(W)
FUN_140001150 XREF[1]: FUN_1400027c0:140002808
140001150  PUSH  REX
140001152  PUSH  RDI
140001153  SUB   RSP, 0x48
140001157  XOR   EEX, EEX
140001159  XOR   ECX, ECX
14000115b  MOV   dword ptr [RSP + local_res8], EEX
14000115f  CALL  qword ptr [->XERRNL32.DLL::GetModuleHandleA]
140001165  MOV   RDI, RAX
140001168  TEST  RAX, RAX
14000116b  JNZ   LAB_14000118f
14000116d  CALL  qword ptr [->XERRNL32.DLL::GetLastError]
140001173  MOV   EEX, EAX
140001175  TEST  EAX, EAX

```

Figure 4: Assembly Instructions - Malware brbbot.

## 4.2 Malware Analysis

Malware analysis is the investigation of malware behaviour to identify its mechanisms depending on its type, such as Trojan, viruses, worms, ransomware, rootkits, key loggers, spam, adware, spyware, fileless malware, and backdoors (Wazid, et al., 2019). Generally, there are two methods to perform malware analysis: static and dynamic; on top, the hybrid method or memory analysis are added. A set of malware analysing tools are summarised in Tab 1 that can be used on a standard operating system or the virtual environment (Mohanta and Saldanha, 2020; Pachhala, et. al., 2021).

**1. Static Analysis.** It concentrates on the signature of extracted portable executable (PE) file types such as exe, DLL, documents, assembly code, byte code, etc. (Pachhala, et. al., 2021). This is the triage phase to determine if the sample is malware, how bad it is, how to detect it, and how to analyse it. The next stage is advanced static analysis, which investigates the static structure and features of a programme without executing it. This analysis stage provides instructions that define the intended purpose of a programme by using a debugger and a disassembler. Executable files (.BAT, .COM, .EXE, .BIN, etc.) represent a series of hexadecimal values for corresponding bytes of a binary file and are used to fulfil various functions or operations on a computer. Analysts identify static patterns (Sihwail, et. al., 2018; Wei, et al., 2019) to detect the intent of malicious code. APIs with malicious behaviour (Murthy, et. al., 2019) are listed in Tab 2. Analysts can determine whether a file is malicious by its API calls, some of which are characteristic of certain types of malware. For instance, *NtReadFile*, *NtWriteFile*, *LdrGetProcedureAddress*, *RegQueryValueExW*, *NtClose* are API calls invoked by the ransomware *JigsawLocker*. The APIs in malware PE files are kept in IATs (Import Address

Table 2: Example of API call sequences per malicious behaviour.

Malicious Behaviour	API Call Sequence
Modify File Attribute	SetFileAttribute
Modify Time of File	GetFileTime, SetFileTime
Load Register	RegSetValue, RegCloseKey
Enumerate all process	Process32First, Process32Next
Privilege Escalation	LookupPrivilegeValueA
Terminate Process	TerminateProcess
Screen Capture	GetDC, CreateCompatibleDC
Hooking	SetWindowsHookA
Downloader	URLDownloadToFile, WinExec
Enumerate all process	Process32First, Process32Next
Anti debugging	IsDebuggerPresent
Synchronization	CreateMutexA
Key Logger	FindWindowA, RegisterHotKey
Dropper	FindResource, LoadResource

Table) and can be obtained using reverse engineering tools such as *IDA*. Every API contains a sequence of assembly instructions, such as *push*, *sub*, *xor*, *mov*, *test*, *jnz*, *call*, and each assembly instruction contains a mnemonic and a sequence of operands as illustrated in Fig 4.

**2. Dynamic and Hybrid Analysis.** Dynamic analysis, or behavioural analysis, focuses on observing and studying a programme's behaviour as it executes within a simulated or controlled virtual environment (Murali, et. al., 2020). Dynamic analysis environment also uses emulators and hypervisors (Singh and Singh, 2018) to compare snapshots of the complete system state before and after a suspicious sample is executed. This analysis examines a range of activities, such as API Calls, Mutexes, File System Changes, Registry Changes, and Loaded DLLs (Guo, et. al., 2020). Some of the standard API calls and DLLs are listed in Table 2 and Table 3. The hybrid analysis technique combines both static and dynamic analysis intended to address the weaknesses of each methodology (Alsmadi and Alqudah, 2021). This type of analysis aims to identify the key sources of variation in a data set. It is useful for multiple data sources that overlap partially or completely, making it easier to interpret one study with the other. It reveals which variables are correlated and, therefore, may be related to each other; then, those variables can be used for subsequent analyses.

**3. Memory Analysis.** It provides a deeper understanding of malicious activities that only appear in a system's volatile memory, making it a crucial component of malware incident handling within SOCs (Arfeen, et. al., 2022). Memory analysis is especially

Table 3: DLLs used in Ransomware with Related Function Calls.

Name of DLL	Functions (API Call)
ADVAPI32.dll	CryptReleaseContext
CRYPT32.dll	CryptQueryObject
CRYPTNET.dll	CryptGetObjectUrl
CRYPTUI.dll	CryptUIDlgSelectCertificateFromStore

important when malicious actors use evasion techniques to leave as little evidence as possible on conventional storage media (Pavelea and Negrea, 2023). Finding the malware’s memory-resident components and learning about the group’s TTPs (Tactics, Techniques and Procedures) were made possible in large part by memory analysis.

### 4.3 Detection and Response Automation

The automation of detection and response capabilities using advanced technologies, such as artificial intelligence (AI) and machine learning (ML), orchestrates optimised threat detection and mitigation speed and accuracy. Such technologies prompt the containment of incidents and identification of the malware’s rapid lateral movement, demonstrating the effectiveness of automated responses in reducing the impact of large-scale attacks. In addition, tackling malware’s persistence and advanced behaviour, e.g., polymorphic malware (continuously modifies its code to avoid detection), makes automated methods essential for real-time threat identification (Kovács, 2022). Also, integrating Generative AI (GenAI) tools, like ChatGPT and Google Bard, into cybersecurity defence and offensive strategies underlines how they are used to launch attacks or proactively detect and address sophisticated threats. AI/ML-based technology strengthens and enhances the capabilities of SOC teams. However, SOC teams must evaluate and monitor the performance of this technology to ensure they remain effective in detecting and responding to malware incidents (Markeyvych and Dawson, 2023).

## 5 SOC CHALLENGES

This section identifies SOC challenges to increase incident response capabilities and reduce the risks related to malware incidents.

1. Documentation might be forgotten or left out in the hectic and high-stress environment of incident handling. The lack of standardised documentation techniques, time constraints, knowledge transfer

and retention, compliance, and legal considerations, and knowledge transfer and retention may provide challenges in documenting incidents.

2. Malware authors use a variety of strategies to obfuscate their code and conceal their presence, making it challenging to identify and link malware to particular individuals or campaigns. In the incident handling phase, polymorphic, advanced encryption, rootkits, and fileless techniques embedded inside genuine files are frequently used by sophisticated malware to avoid being discovered by conventional detection methods.
3. To exploit vulnerabilities and avoid detection, malicious actors constantly create brand-new, highly developed malware variants. Given that malware is dynamic, SOC teams must keep up with this rapid malware evolution and take proactive measures to foresee and address new threats and obstacles.
4. Incident response activities must be improved by adequate staffing, funding, and technological resources. Organisations should provide the SOC with the resources to handle this issue, including skilled staff, cutting-edge security resources, as well as adequate training to enable effective incident handling.
5. Successful malware incident response requires effective coordination and communication both within the SOC and with external parties. However, creating seamless collaboration can be difficult due to organisational barriers, a lack of standardised communication routes, or the participation of third-party vendors.
6. Malware incidents must be prioritised and triaged according to their potential importance and impact. It can be difficult to assess the urgency and severity of each occurrence since the earliest signs of compromise might not accurately reflect the full scope of the issue (Vielberth, et. al., 2020).

## 6 RESEARCH DIRECTIONS

Based on the identified challenges, the recommendations below are aimed at enhancing the handling of malware incidents within SOCs and improving their overall cybersecurity posture, tightening their incident response procedures, and better minimise the effects of malware incidents.

- (a) Automation. Effective incident analysis and detection tools such as automated and advanced

Table 4: Review of Literature on Malware Incident Handling Capabilities.

References	Data Collection	Detection	Static Analysis	Dynamic Analysis	Hybrid Analysis	Memory Analysis
(Shree, et. al., 2022)	×	×	✓	✓	×	✓
(Vielberth, et. al., 2020)	✓	×	×	×	×	×
(Pachhala, et. al., 2021)	×	×	✓	✓	×	×
(Muniz, et. al., 2015)	✓	×	×	×	×	×
(Hao et. al., 2022)	×	×	✓	×	×	×
(Wang and Zhu, 2017)	✓	×	×	×	×	×
(Sihwail, et. al., 2018)	×	×	✓	✓	×	×
(Hossain, et. al., 2021)	✓	×	×	×	×	×
(Sharma and Bharti, 2021)	×	✓	×	×	×	×
(Murthy, et. al., 2019)	×	✓	×	×	×	×
(Souppaya et al., 2013)	×	✓	×	×	×	×
(Okolica and Peterson, 2010)	×	×	×	×	×	✓
(Aslan and Samet, 2017)	×	×	✓	✓	×	×
(Pitolli, et. al., 2021)	×	✓	×	×	×	×
(Soni, et. al., 2022)	×	×	✓	✓	×	×
(Gandotra, et al., 2014)	×	×	✓	✓	×	×
(Gad, et. al., 2015)	✓	×	×	×	×	×
(Guo, et. al., 2020)	×	✓	×	×	×	×
(Murthy, et. al., 2019)	×	×	✓	✓	×	×
(Prähofer, et. al., 2012)	×	×	✓	✓	×	×
(Sihwail, et. al., 2018)	×	×	✓	✓	✓	×
(Wei, et al., 2019)	×	×	✓	✓	×	×
(Ali, et. al., 2020)	×	✓	×	×	×	×
(Murali, et. al., 2020)	×	×	×	✓	×	×
(Singh and Singh, 2018)	×	×	×	✓	×	×
(Guo, et. al., 2020)	×	×	×	✓	×	×
(Jindal, et. al., 2019)	×	×	×	✓	×	×
(Carrier, et. al., 2022)	×	×	×	×	×	✓
(Amer and Zelinka, 2020)	×	×	×	✓	×	×
(Choudhary and Vidyarthi, 2015)	×	×	×	✓	×	×
(Onwubiko and Ouazzane, 2020)	×	×	×	✓	×	×
(Alsmadi and Alqudah, 2021)	×	×	×	✓	✓	×
(Subedi, et al., 2018)	×	×	×	✓	×	×
(Ijaz, et. al., 2019)	×	×	×	✓	×	×
(Kara, 2022)	×	×	×	×	×	✓
(Or-Meir, et. al., 2019)	×	×	×	✓	×	×
(Chanajitt, et. al., 2021)	×	×	×	✓	×	×
(Aboaoja, et al., 2022)	×	✓	×	×	✓	×
(Hadiprakoso, et. al., 2020)	×	×	×	×	✓	×
(Arfeen, et. al., 2022)	×	×	×	×	×	✓
Total 100% (each column)	11.63%	18.60%	30.23%	53.49%	9.3%	11.63%

technologies, the management framework for automated triage, containment, and escalation for malware detection and analysis by SOC analysts (Hossain, et. al., 2021).

- (b) Incident Response Capabilities. Perpetual learning and training, threat intelligence to stay current with the latest malware trends and techniques, and regular drills for incident response to improve incident response capabilities (Ozer, M. et al., 2020).
- (c) Collaboration. Investigating the potential for collaboration between SOCs, other organisations, industry groups, and law enforcement agencies to share information and best practices on malware analysis, tracking and managing incidents (Daniel et al., 2023).
- (d) Root Cause Analysis. Investigating the potential for root cause analysis to understand the cause of an incident and take steps to prevent similar incidents in the future (Jaramillo, 2019).
- (e) Human Factors. A crucial component of human factors is the possible impact of handling a malware incident, which could result in morale decline, burnout, and higher turnover rates. SOC

staff members experience high stress levels due to the demanding nature of incident response and the ongoing evolution of cyber threats. Moreover, studies have indicated that insufficient training impedes SOC analysts' capacity to promptly and precisely address new threats (Daniel et al., 2023). It is essential to investigate the underlying causes of burnout, inadequate training, and teamwork and consider methods to address the human factor challenges.

- (f) Data Management in Malware Incident Handling. Analysing data governance entails evaluating how companies set up guidelines, protocols, and safeguards to guarantee the confidentiality, availability, and integrity of incident-related data. Improving data security procedures also entails protecting incident-related data from alteration or illegal access, identifying the underlying causes of problems with data quality, suggesting techniques for real-time validation and verification.
- (g) Scalability in Malware Analysis for Expanding Businesses. The scalability difficulties encountered when an organisation's growth exceeds the capacity of its malware analysis infrastructure, results in delayed threat detection and response.

Such challenges can take on new dimensions with the rise of cloud computing, best illustrated by the data breach cases (Khan et al., 2022).

- (h) Continuous Improvement and Post-Incidental Analysis. The analyses of the recent security breaches (Almulihi et al., 2022), reveal systemic weaknesses and inform strategies for preventing the recurrence of similar incidents and the significance of thorough post-incident analysis.

## 7 CONCLUSION

This paper presented the literature findings on challenges SOC analysts face and explored the complexities of the current malware events handling solutions and best practices used in SOC operations. It also called attention to the SOC architecture insights and operational requirements to empower SOC teams and security management professionals to tackle malware incidents and strengthen cybersecurity defences. It highlighted widely used malware analysis tools and techniques and discussed the research directions for enhancing and improving the overall cybersecurity posture. In summary, the areas that have not been adequately addressed by existing studies and therefore need further research are:

- An objective approach to incorporate automated triage process, advanced malware detection and analysis tools by SOC analyst, and novel management frameworks to address unified solutions addressing multiple collaborative work factors;
- An integration of the incident response processes with other workflows within the organisation to ensure a seamless and efficient response to cyber-attacks;
- A thorough planning of human factors capabilities assessment and an investment in developing a skilled and knowledgeable SOC team.
- A comprehensive data management strategy, including data governance, data quality, and data security, ensures that SOCs have accurate, complete, and timely data to support their incident response and analysis efforts.

## REFERENCES

- Aboaoja, et al (2022). Malware detection issues, challenges, and future directions: A survey. *Applied Sciences*, 12(17):8482.
- Ali, et. al. (2020). Identification of malware families for creating generic signatures: Using dynamic analysis and clustering methods. In *Inter. Symposium on Recent Advances in Electrical Engineering & Computer Sciences*, volume 5, pages 1–6. IEEE.
- Almulihi, A. H., Alassery, F., Khan, A. I., Shukla, S., Gupta, B. K., and Kumar, R. (2022). Analyzing the implications of healthcare data breaches through computational technique. *Intelligent Automation & Soft Computing*, 32(3).
- Alsmadi, T. and Alqudah, N. (2021). A survey on malware detection techniques. In *Inter. Conf. on Information Technology*, pages 371–376. IEEE.
- Amer, E. and Zelinka, I. (2020). A dynamic windows malware detection and prediction method based on contextual understanding of api call sequence. *Computers & Security*, 92:101760.
- Arfeen, et. al. (2022). Process based volatile memory forensics for ransomware detection. *Concurrency and Computation: Practice and Experience*, 34(4):e6672.
- Aslan, Ö. and Samet, R. (2017). Investigation of possibilities to detect malware using existing tools. In *IEEE/ACS 14th Inter. Conf. on Computer Systems and Applications*, pages 1277–1284. IEEE.
- Barrett, M. P. (2018). Framework for improving critical infrastructure cybersecurity. *National Institute of Standards and Technology, USA, Tech. Rep.*
- Carrier, et. al. (2022). Detecting obfuscated malware using memory feature engineering. In *ICISSP*, pages 177–188.
- Chanajitt, et. al. (2021). Combining static and dynamic analysis to improve machine learning-based malware classification. In *IEEE 8th Inter. Conf. on Data Science and Advanced Analytics*, pages 1–10. IEEE.
- Choudhary, S. and Vidyarthi, M. D. (2015). A simple method for detection of metamorphic malware using dynamic analysis and text mining. *Procedia Computer Science*, 54:265–270.
- Coscia, A., Dentamaro, V., Galantucci, S., Maci, A., and Pirlo, G. (2023). Yamme: a yara-byte-signatures metamorphic mutation engine. *IEEE Transactions on Information Forensics and Security*.
- Daniel, C., Mullarkey, M., and Agrawal, M. (2023). Rq labs: A cybersecurity workforce skills development framework. *Information Systems Frontiers*, 25(2):431–450.
- Gad, et. al. (2015). Monitoring traffic in computer networks with dynamic distributed remote packet capturing. In *IEEE Inter. Conf. on Communications*, pages 5759–5764. IEEE.
- Gandotra, et al. (2014). Malware analysis and classification: A survey. *Journal of Infor. Security*, 2014.
- Guo, et. al. (2020). File entropy signal analysis combined with wavelet decomposition for malware classification. *IEEE Access*, 8:158961–158971.
- Hadiprakoso, et. al. (2020). Hybrid-based malware analysis for effective and efficiency android malware detection. In *Inter. Conf. on Informatics, Multimedia, Cyber and Information System*, pages 8–12. IEEE.
- Hao et. al. (2022). Eii-mbs: Malware family classification via enhanced adversarial instruction behavior semantic learning. *Computers & Security*, 122:102905.

- Hossain, et. al. (2021). Automatic event categorizer for siem. In *Procs of the 31st Annual Inter. Conf. on Computer Science and Software Engineering*, pages 104–112.
- Ijaz, et. al. (2019). Static and dynamic malware analysis using machine learning. In *16th Inter. bhurban conference on applied sciences and technology*, pages 687–691. IEEE.
- Jaramillo, L. E. (2019). Malware threats analysis and mitigation techniques for compromised systems. *Jour. of Information Systems Engineering & Mangt.*, 4(1).
- Jindal, et. al. (2019). Neurlux: dynamic malware analysis without feature engineering. In *Procs of the 35th Annual Computer Security Applications Conference*, pages 444–455.
- Kara, I. (2022). Fileless malware threats: Recent advances, analysis approach through memory forensics and research challenges. *Expert Systems with Applications*, page 119133.
- Khan, S., Kabanov, I., Hua, Y., and Madnick, S. (2022). A systematic analysis of the capital one data breach: Critical lessons learned. *ACM Transactions on Privacy and Security*, 26(1):1–29.
- Kovács, A. (2022). Ransomware: a comprehensive study of the exponentially increasing cybersecurity threat. *Insights into Regional Development*, 4(2):96–104.
- Malwarebytes (2020). 2020 state of malware report.
- Markevych, M. and Dawson, M. (2023). A review of enhancing intrusion detection systems for cybersecurity using artificial intelligence (ai). In *Inter. conf. Knowledge-based Organization*, volume 29, pages 30–37.
- Mohanta, A. and Saldanha, A. (2020). *Malware Analysis and Detection Engineering: A Comprehensive Approach to Detect and Analyze Modern Malware*. Springer.
- Muniz, et. al. (2015). *Security Operations Center: Building, Operating, and Maintaining Your SOC*. Cisco Press.
- Murali, et. al. (2020). A malware variant resistant to traditional analysis techniques. In *Inter. Conf. on Emerging Trends in Information Technology and Engineering*, pages 1–7. IEEE.
- Murthy, et. al. (2019). Exploring the api calls for malware behavior detection using concordance and document frequency. *Inter. Jour. of Engineering and Advanced Technology*, 8(6):4991–4997.
- Okolica, J. and Peterson, G. L. (2010). Windows operating systems agnostic memory analysis. *Digital investigation*, 7:S48–S56.
- Onwubiko, C. and Ouazzane, K. (2019). Challenges towards building an effective cyber security operations centre. *International Journal On Cyber Situational Awareness*, Vol. 4(No.1):11–39.
- Onwubiko, C. and Ouazzane, K. (2020). Soter: A playbook for cybersecurity incident management. *IEEE Transactions on Engineering Management*.
- Or-Meir, et. al. (2019). Dynamic malware analysis in the modern era—a state of the art survey. *ACM Computing Surveys*, 52(5):1–48.
- Ozer, M. et al. (2020). Cloud incident response: Challenges and opportunities. In *Int. Conf. on Computational Science and Computational Intelligence*, pages 49–54. IEEE.
- Pachhala, et. al. (2021). A comprehensive survey on identification of malware types and malware classification using machine learning techniques. In *2nd Inter. Conf. on Smart Electronics and Communication*, pages 1207–1214. IEEE.
- Pavelea, A. and Negrea, P.-C. (2023). A comprehensive analysis of high-impact cybersecurity incidents: Case studies and implications.
- Pitolli, et. al. (2021). Malfamaware: automatic family identification and malware classification through online clustering. *Inter. Journal of Information Security*, 20(3):371–386.
- Prähofer, et. al. (2012). Opportunities and challenges of static code analysis of iec 61131-3 programs. In *Procs of 2012 IEEE 17th Inter. Conf. on Emerging Technologies & Factory Automation*, pages 1–8. IEEE.
- Sharma, S. and Bharti, S. (2021). Malware analysis using ensemble techniques: A machine learning approach. In *2021 Inter. Conf. on Artificial Intelligence and Machine Vision*, pages 1–5. IEEE.
- Shree, et. al. (2022). Memory forensic: Acquisition and analysis mechanism for operating systems. *Materials Today: Proceedings*, 51:254–260.
- Sihwail, et. al. (2018). A survey on malware analysis techniques: Static, dynamic, hybrid and memory analysis. *Int. J. Adv. Sci. Eng. Inf. Technol.*, 8(4-2):1662–1671.
- Singh, J. and Singh, J. (2018). Challenge of malware analysis: malware obfuscation techniques. *Inter. Journal of Information Security Science*, 7(3):100–110.
- Soni, et. al. (2022). Opcode and api based machine learning framework for malware classification. In *2nd Inter. Conf. on Intelligent Technologies*, pages 1–7. IEEE.
- Souppaya, M., Scarfone, K., et al. (2013). Guide to malware incident prevention and handling for desktops and laptops. *NIST Special Publication*, 800:83.
- Subedi, et al. (2018). Forensic analysis of ransomware families using static and dynamic analysis. In *IEEE Security and Privacy Workshops*, pages 180–185. IEEE.
- Vielberth, et. al. (2020). Security operations center: A systematic study and open challenges. *IEEE Access*, 8:227756–227779.
- VirusTotal (2022). Writing YARA rules. <https://yara.readthedocs.io/en/stable/writingrules.html>.
- Wang, Z. and Zhu, Y. (2017). A centralized hids framework for private cloud. In *18th IEEE/ACIS Inter. Conf. on Software Engineering, AI, Networking and Parallel/Distributed Computing*, pages 115–120. IEEE.
- Wazid, et al. (2019). Iomt malware detection approaches: analysis and research challenges. *IEEE Access*, 7:182459–182476.
- Wei, et al. (2019). Malware classification system based on machine learning. In *Chinese Control And Decision Conference*, pages 647–652. IEEE.