Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.052447

REVIEW

A Comprehensive Survey on Advanced Persistent Threat (APT) Detection Techniques

Singamaneni Krishnapriya* **and Sukhvinder Singh**

Department of Computer Science, School of Engineering and Technology, Pondicherry University, Kalapet, 605014, India

*Corresponding Author: Singamaneni Krishnapriya. Email: singamanenikrishnapriya@gmail.com

Received: 02 April 2024 Accepted: 27 June 2024 Published: 15 August 2024

ABSTRACT

The increase in number of people using the Internet leads to increased cyberattack opportunities. Advanced Persistent Threats, or APTs, are among the most dangerous targeted cyberattacks. APT attacks utilize various advanced tools and techniques for attacking targets with specific goals. Even countries with advanced technologies, like the US, Russia, the UK, and India, are susceptible to this targeted attack. APT is a sophisticated attack that involves multiple stages and specific strategies. Besides, TTP (Tools, Techniques, and Procedures) involved in the APT attack are commonly new and developed by an attacker to evade the security system. However, APTs are generally implemented in multiple stages. If one of the stages is detected, we may apply a defense mechanism for subsequent stages, leading to the entire APT attack failure. The detection at the early stage of APT and the prediction of the next step in the APT kill chain are ongoing challenges. This survey paper will provide knowledge about APT attacks and their essential steps. This follows the case study of known APT attacks, which will give clear information about the APT attack process—in later sections, highlighting the various detection methods defined by different researchers along with the limitations of the work. Data used in this article comes from the various annual reports published by security experts and blogs and information released by the enterprise networks targeted by the attack.

KEYWORDS

Advanced persistent threats; APT; cyber security; intrusion detection; cyber attacks

1 Introduction

Nowadays professional hackers are using sophisticated methods to conduct the attacks. Government and business associations are the main targets of these attacks [1]. Financial loss or disruption in network service are possible outcomes of these attacks. Cyberattacks are malevolent actions carried out via the Internet to steal sensitive data or obtain financial advantage. In addition, the attacker watches the target organization and tampers its network operations [2]. Keeping organizations and businesses safe from attack is one of their largest problems. They make constant efforts to defend their private data from intrusions [3]. The budget for security solutions increased in 2019 from \$114 billion to \$124 billion, and in 2020, it increased by 72%, according to a Gartner analysis [4]. The trend report (Micro) shows an increase in targeted attacks, where attackers focus on specific organizations with a clear goal and continue their efforts until they are successful. According to [5], the big four APT actors originate from China, Iran, North Korea, and Russia

Advanced Persistent Threats (APTs) are called "slow poisoning" cyber attacks mostly carried out between the nations/states over a long period. Generally, these types of attacks are conducted by highly skilled attackers. APTs were first identified in the year of 2006 [6] and became notable after operation aurora which was conducted against the Google in 2011 [7]. The APTs are defined as follows by NIST:

"An adversary with sophisticated levels of expertise and significant resources, allowing it through the use of various attack vectors (e.g., cyber, physical, and deception) to generate opportunities to achieve its objectives, which are typically to establish and extend its presence within the information technology infrastructure of organizations for purposes of continually exfiltrating information and to undermine or impede critical aspects of a mission, program, or organization, or place itself in a position to do so in the future; moreover, the advanced persistent threat pursues its objectives repeatedly over an extended period, adapting to a defender's efforts to resist it, and with determination to maintain the level of interaction needed to execute its objectives"

APT is the most significant risk to the governments & private organizations. According to the CISA, NCSC report on 8 April, 2020, Advanced Persistent Threat (APT) groups are using the COVID-19 issue to distribute malware by launching phishing websites and registering new domain names to trap many internet users [8]. Traditional security systems like intrusion detection and prevention systems (IDPs) have been introduced to identify network attacks. An intrusion is a series of actions conducted by a malicious individual that compromises a specific system [9]. Intrusion detection (ID) is a process of identifying malicious activities within the computer network by analyzing suspicious activities (Anomaly-based) or by matching the patterns (Signature-based).

The APT attackers are using unknown security holes to evade the current detection systems, which makes them difficult to detect. The attacker keeps changing their malware to avoid the current Intrusion detection systems. Most attackers develop their attack tools and malware to remain stealthy and persistent. According to the cyber-security reports [10], malware is significantly increasing every year, and most of the malware is distributed through Windows operating systems.

In 2018, Mandiant [11] surveyed the length of long-lasting attacks using dwell time (time interval between the attack entry and its detection). He observes that in the Asia-Pacific (ASPC) regions, the attacker can stay up to 498 days in the network by evading the current detection solutions depicted in Fig. 1. His survey results are depicted in Fig. 2. As evident recently, the APT 44 (Russia's Sandworm) attack was lost for four years, which led to the war in Ukraine [12].

The longer dwell time defines the attack sophistication. The survey also found that 44% of attacks are due to external sources. This highlights that the current detection systems cannot detect these types of stealthy attacks.

The rest of the manuscript is organized as follows: Section 2 describes the importance of proposed survey; Section 3 is background information which will help the readers to understand about APT attack and its process; Section 4 presents the case study; Section 5 presents the various detection methods and its comparison; Section 6 presents the preventive countermeasures against APT attack; Section 7 is Conclusion.

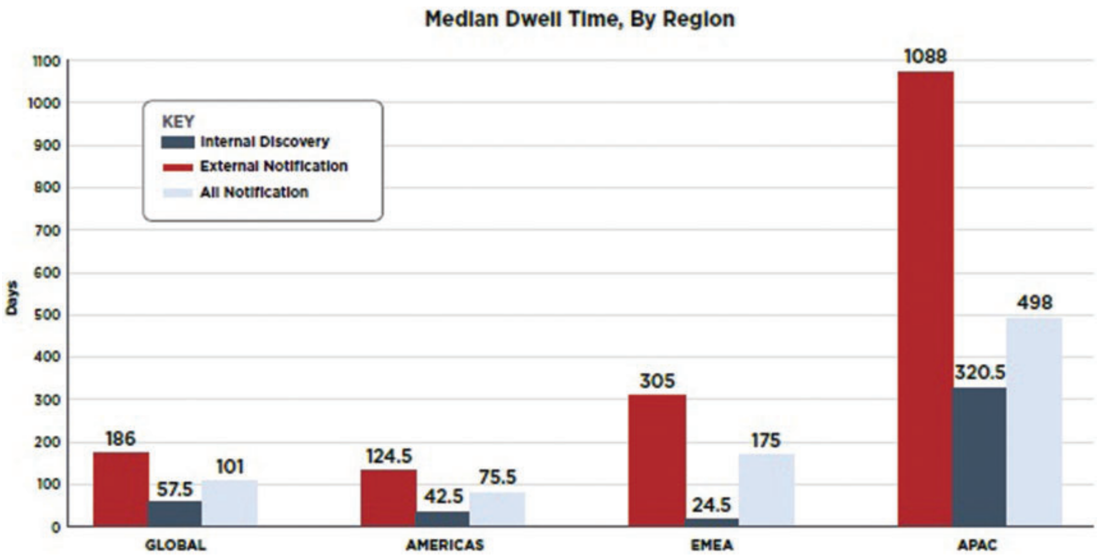

Figure 1: Dwell time by region

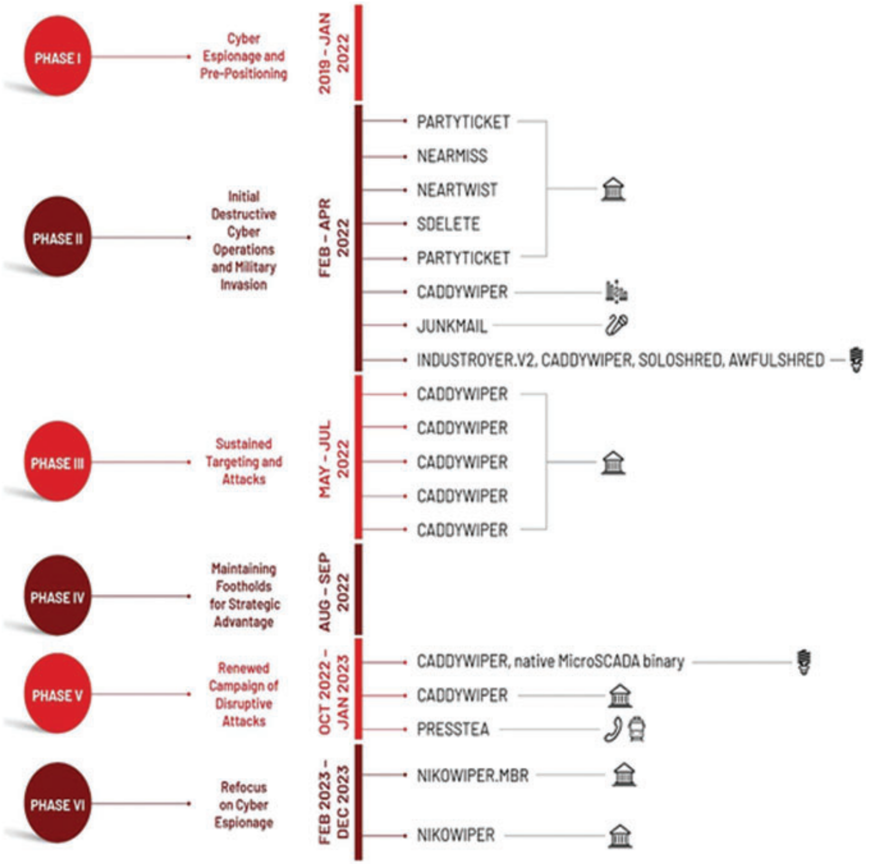
PHASE I
Cyber Espionage and Pre-Positioning
2019 – JAN 2022
PHASE II
Initial Destructive Cyber Operations and Military Invasion
FEB – APR 2022
PARTYTICKET
NEARMISS
NEARTWIST
SDELETE
PARTYTICKET
CADDYWIPER
JUNKMAIL
INDUSTROYER.V2, CADDYWIPER, SOLOSHRED, AWFULSHRED
PHASE III
Sustained Targeting and Attacks
MAY – JUL 2022
CADDYWIPER
CADDYWIPER
CADDYWIPER
CADDYWIPER
CADDYWIPER
PHASE IV
Maintaining Footholds for Strategic Advantage
AUG – SEP 2022
PHASE V
Renewed Campaign of Disruptive Attacks
OCT 2022 – JAN 2023
CADDYWIPER, native MicroSCADA binary
CADDYWIPER
PRESSTEA
PHASE VI
Refocus on Cyber Espionage
FEB 2023 – DEC 2023
NIKOWIPER.MBR
NIKOWIPER

Figure 2: Six phases of APT44

2 Motivation of Survey

This section examines the various recent surveys on APT detection methods that have been conducted. Since the first APT attack report, some works have been studied based on malware or attack techniques.

Some works are focused on proposed defensive methods [13,14] for detecting a possible APT attack. Nevertheless, the current information on APT-attack campaigns is scattered among practitioners, government sources, academic publications, and existing taxonomies, which tend to be limited in scope. Through this survey, we analyzed existing methods for detecting APT attacks. To ensure our survey's novelty and contribution, we compared the proposed study with existing surveys, as shown in Table 1.

Table 1: Comparison of existing surveys with the proposed survey

	[15]	[16]	[17]	[18]	[19]	Proposed survey
APT lifecycle	✓	✓	✓	✓		✓
Case-studies	✓	✓				✓
Mapping of attack stages to attack vectors		✓			✓	✓
Detection methods			✓	✓	✓	✓
Preventive countermeasures						✓

In [15], the authors analyzed several publicly available APT attack reports and concluded that spear-phishing and valid credentials are the most commonly used initial and lateral intrusion methods. In [16], the authors analysed several public reports of APT attacks and APT stages. Furthermore, different APT life-cycle models were examined. Finally, the five-stage life-cycle model was described, commonly used attack vectors were identified, and possible mitigation techniques were proposed.

The various machine learning algorithms and their usage recommendations in attack detection were presented. In [17], the author presented the detection methods using machine learning to attack detection. In [16], the authors presented a comprehensive review of User Behaviour Analytics (UBA) methodologies and their application to combat the APT attacks. In [20], the authors studied APT attacks more deeply than other works, reviewed APT attack case studies, and gave preventive countermeasures and existing detection methods for APT attacks. In [21], the focus is only on supply chain-based APT attacks. In [21], the authors presented the detection methods that are helpful in real-time detection by studying 26 papers.

However, the survey paper should provide a comprehend view of all limitation, which is available in discrete works. Also, each of these surveys is addressing the specific aspect of the APT attack. These limitations have given a motivation for this paper. Then, in this regard, this paper describes the following:

- Comprehensive discussion of APT attacks: We discussed several techniques used by APT attackers. As such, great emphasis on a detailed description of the APT-stages and potential attack techniques.
- APT detection methods: Reviewed the recent advances and methods employed for APT detection and their disadvantages. Next, comparison and summary tables are presented.

- Challenges and Future Directions: We highlight the challenges and research directions for the successful APT detection along with preventive measures.

3 Advanced Persistent Thretes (APTs)

An advanced persistent threat is a targeted attack that obtains illegal access to information and communication systems to sift confidential data or harm a business [11]. Since the release of Stuxnet [22], APT attacks have grown more deliberate and destructive, illustrating how simple it is to breach well-known systems while eluding many of the more advanced defense mechanisms meant to safeguard the computing environment. At the moment, many of these threats are unidentified. Once discovered, many of these threats—like APT10 [20] and APT41 [22]—reappear with altered capabilities to fulfill their intended purpose. These attacks resulted in significant financial, confidential information, and intellectual property losses. The APT attack has three characteristics [6,23] depicted in Fig. 3: (1) Advanced: Attackers use advanced tools and techniques during attack phases to the target; (2) Persistent: Attackers have strong determination towards their selected target. Attackers follow the slow process during the attack cycle; (3) Threat: The attacker can get access to information.

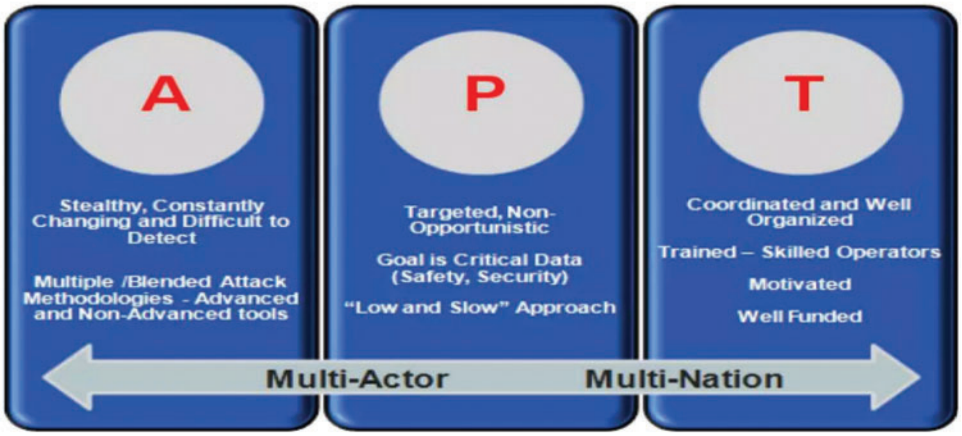
A
Stealthy, Constantly Changing and Difficult to Detect
Multiple /Blended Attack Methodologies - Advanced and Non-Advanced tools
P
Targeted, Non-Opportunistic
Goal is Critical Data (Safety, Security)
"Low and Slow" Approach
T
Coordinated and Well Organized
Trained – Skilled Operators
Motivated
Well Funded
Multi-Actor
Multi-Nation

Figure 3: Advanced persistent threat

3.1 Comparison of APT with Traditional Attack

APT attackers differ from other cybercriminals in their objectives and goals because they have specific targets. The reason behind the attack may be financial extortion, political manipulation, and industrial, military, economic, technical, and intellectual property espionage. The authors of [4] summarized the distinctions between traditional threats and APT attacks. The following characteristics have been considered: attacker, target, purpose, and approach shown in Table 2.

Table 2: Comparison of traditional attacks and APT

	Traditional attack	APT
Attacker	Mostly single person	Extremely resourceful, intelligent, driven, and well-organized group

(Continued)

Table 2 (continued)

	Traditional attack	APT
Target	Unspecified, most of the time individual-systems	Particular groups, state agencies, and private businesses
Purpose	Financial benefits, demonstrating abilities	Competitive advantages, strategic benefits
Approach	Single-run, “smash and grab”, short period	Continual attempts, remain low and slow, and eventually adapt to withstand defenses

The damage caused by these attacks is high due to sophisticated attack techniques and advanced attackers. Lemay’s survey [24] shows that 174 APT actors are identified globally.

The author presented a general summary of the attack groups and related open-source articles. The characteristics of APT attackers are:

- APT actors use unknown (Zero-day) vulnerabilities and develop their techniques and Tools for their attack campaign. Traditional attackers use known (existing) vulnerabilities and tools.
- They have strong determination towards their attack objective. They will spend considerable time and resources until they can reach the target.
- Based on the study of APT attacks, APT actors have extensive capabilities and are State sponsored for conducting the attack.
- APT actors will select the target before starting the attack. Generally, they will target the big organizations conducted between the nation-states. In contrast, regular attacks usually hope for quick wins but don’t think much more about the target (they will target the system known to them).
- Unlike traditional attackers, APT attackers have the same target over the years.

In [22], the authors describe the list of APT Attackers from the last ten years, including each group’s origin, target, the tools they use, and notable attack campaigns. Two cyber security research organizations–Crowdstrike and Mandiant (FireEye)-track and monitor the threat attackers. Mandiant numerically defines APT groups, and depending on the country, Crowdstrike titles APT groups by animals. For example, a China APT group was assigned “Panda” Iran to “Kitten” and a Russian group by “Bear”. Thirty-five nations have been suspected of funding cyber operations since 2005. Out of all eighty-eight operations that were conducted till 2020 and motive behind the attack may be data exfiltration, financial gain, espionage, etc. From the security statistics shown in Fig. 4, most APT groups originate from China.

3.2 APT Attack Process

An advanced persistent threat, also known as an APT, is an organized and sophisticated cyber-attack. The hacker enters a network and moves stealthily inside the network until the required information is gathered, or the objective is achieved. An APT attack is thoroughly prepared, intended to infiltrate a particular organization, and tailored to avoid security measures inside the target network environment. The APT attack has a specific target. Before starting the attack, the attacker gathers information about the target. The attacker builds or develops the tools and exploits the vulnerable point based on the collected data. After that, the attacker expands their attack from that point by sending commands or instructions from an external server (command and control). The APT attack process is explained with the help of a flowchart as shown in Fig. 5.

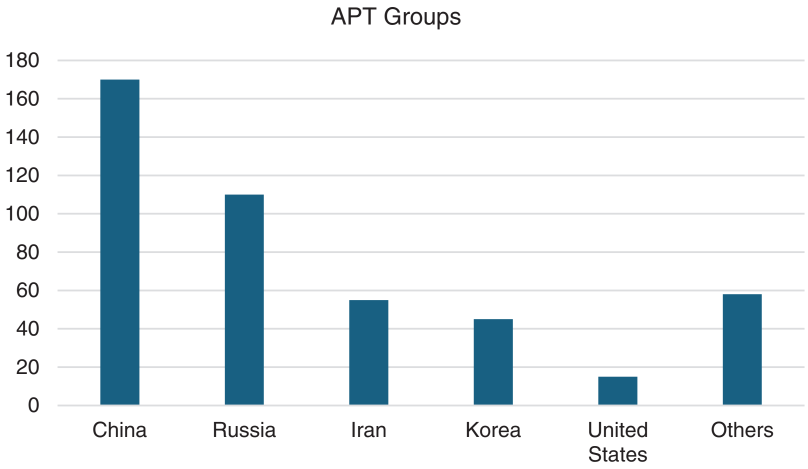

Figure 4: Origin of APT attackers

As mentioned earlier, APT is highly organized and persistent in reaching the target. Targeted attacks like APT are carried out in multiple stages. The overall attack processes are described using the "Life Cycle." The number of stages in the attack life cycle is not specific. In [25], Mandiant described the APT life-cycle model with eight stages; In [26], they described seven stages life-cycle model; In [13], authors described a six-stage attack life-cycle model; In [27], authors discussed 4 stage attack life-cycle model; In [28,29], authors discussed 5 stage model shown in Fig. 6.

The attacker decides the number of stages in the lifecycle based on the attack objective. It is essential to mention that the life cycles are analyzed to explain how the APT attack operates. However, each attacker can execute the steps in their order and use the TTPs adapted to reach the goal. These stages are not specific; depending upon the attack objective, the number of stages may vary. Addressing this, we have described the APT lifecycle model with five stages shown in Fig. 7.

These stages do not need to be in every APT attack lifecycle model; they could represent all APT attacks irrespective of the target goal. The typical APT steps are:

1. Information Gathering
2. Initial Intrusion
3. Command and Control Communication (C&C)
4. Lateral Movement
5. Attack Objective

For any APT attack, the first two stages are essential for the attackers to increase the attack success rate. The stages in the proposed lifecycle model are explained in a later section: Information Gathering and Initial Intrusion. The next three steps are appropriate based on the attack goal. The last sections explain these stages and possible methods attackers use.

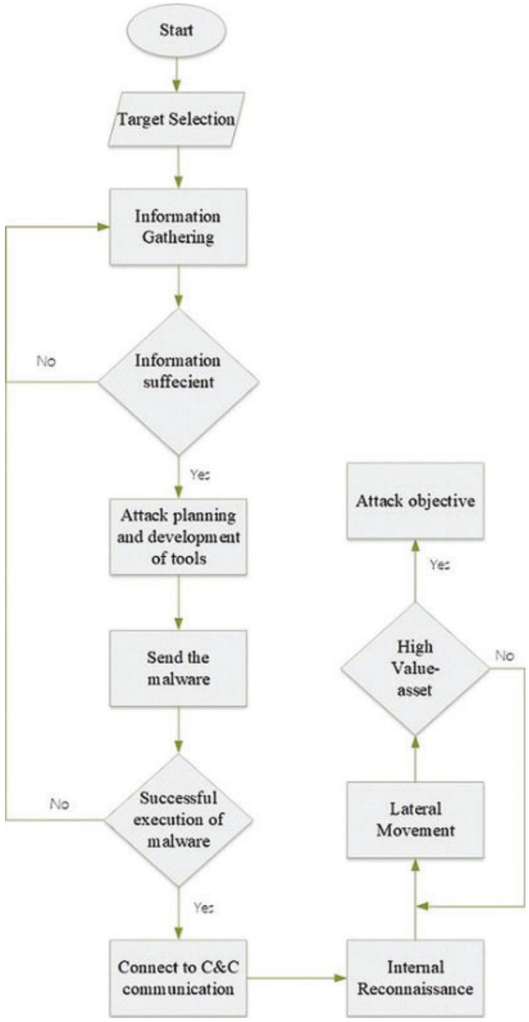

Figure 5: Origin of APT attackers

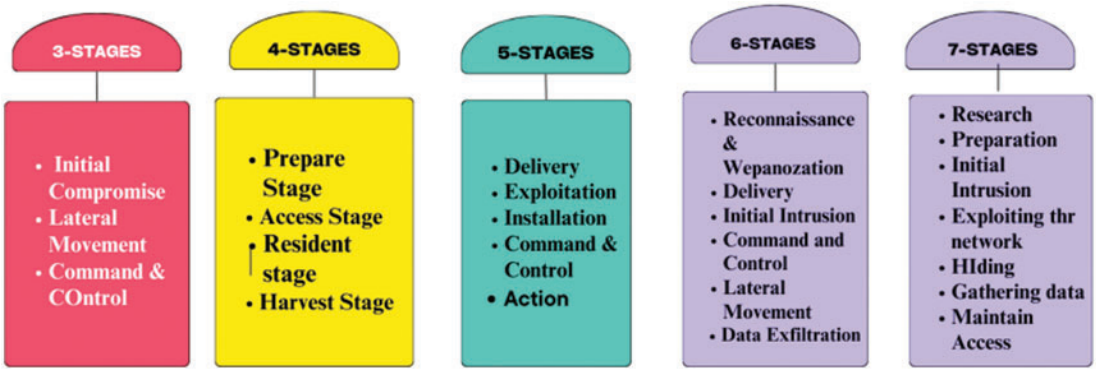

Figure 6: Different stages in APT lifecycle

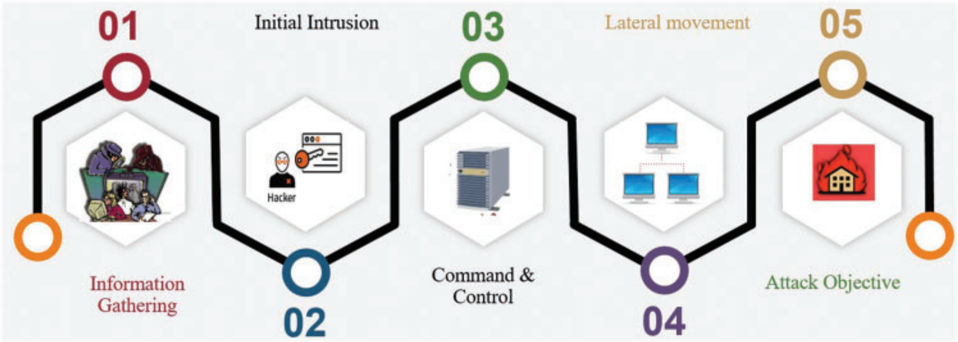

Figure 7: Typical APT attack steps

3.2.1 *Information Gathering*

It is also called Reconnaissance. It is the essential step in the attack process because attacker develop their tools and techniques based on the information gathered during this stage. If they know about the target, they are more successful in attack. In this stage, the attacker will extensively research the target and their assets to enhance their success rate. This can be done through inactive and passive network scanning Table 3.

Table 3: Information gathering

	Type of data	Technical, Non-technical
Information gathering	Data source	Online, Offline
	Method	Active scanning, Passive scanning

In active scanning, the attacker looks into the target through network traffic. In the passive scanning, they do not disturb the network functionality. Through scanning, the attacker will collect technical information (like Internet registry, Sub-net information (Registered and currently active), DNS information, Routing Information, Remote access Information, E-mail accounts information, Firewall, Antivirus, Wireless Information, Routers information, etc.) and non-technical information is (Business partners, News articles, Projects, Press notes, etc.) about the target network. The information gathered during this phase is handy to the attackers in initial intrusion and to move deeper inside the target. The attacker will also use public repositories for Domain information, vulnerability databases, press notes released, and supply chain management information [30]. By the end of this stage, the attacker will get sufficient information about the target for preparing an attack plan, developing attack tools, and finding the weakest (vulnerable) link in the target for the initial intrusion. Table 4 describes the information gathered and how it is helpful for the attacker to conduct an attack. The attacker will spend a considerable amount of time preparing the attack tools, creating a flexible environment to stay a long time inside the target network environment.

Fig. 8 represents the typical information gathering model consisting of different phases, which are essential to complete before the targeted attack starts.

Table 4: Gathered information and usage by attacker

Gathered information	Usage
Network infrastructure, Documentation including schematic and configuration files	Understand IDS configurations, fire-wall and where vulnerabilities are exist.
Organization chart	Determine which people to target for email and data theft, or as targets for spear-phishing campaigns.
Systems documentation	Identify where targeted systems existing within a victim network.
VPN configuration files	Determine which areas of the victim's network VPN users can access, then aim to steal VPN credentials.

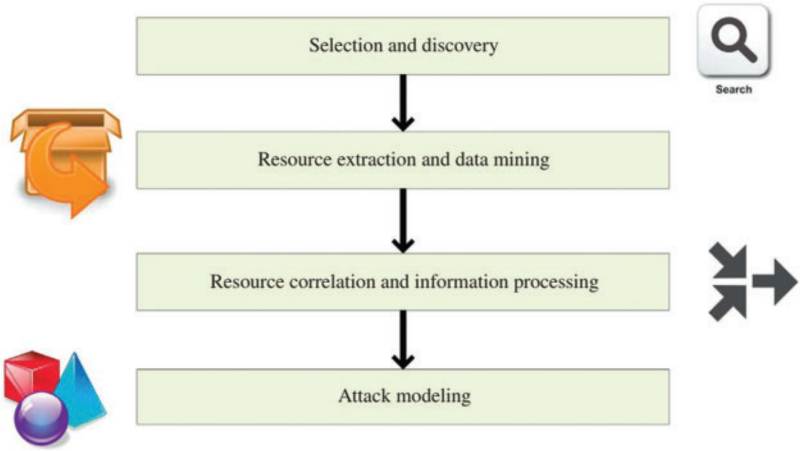
Selection and discovery
Search
Resource extraction and data mining
Resource correlation and information processing
Attack modeling

Figure 8: Intelligent gathering model

3.2.2 Initial Intrusion

It is also called point of Entry (POE). This step involves the attacker using various social engineering techniques, such as spear-phishing, water-hole attacks, and software vulnerabilities, to compromise the system or vulnerable point identified during the information-gathering stage through malware execution. The adversary will use the data acquired in the earlier phase to increase the probability of success rate. "Spear-phishing email" is the most widely used technique. In it, an attacker sends an email containing a file or URL that contains malicious code and points to a website that contains malicious code. The attacker will gain initial access (a foothold) to the target network Environment by successfully executing malware code. According to the Trend Macro report [31,32], most APT attacks use spear-phishing Emails for the initial intrusion. In this, the attacker will send an email containing a malicious code file (as an attachment) or URL (link) that directs to a website having malicious code, depicted in Fig. 9 [33].

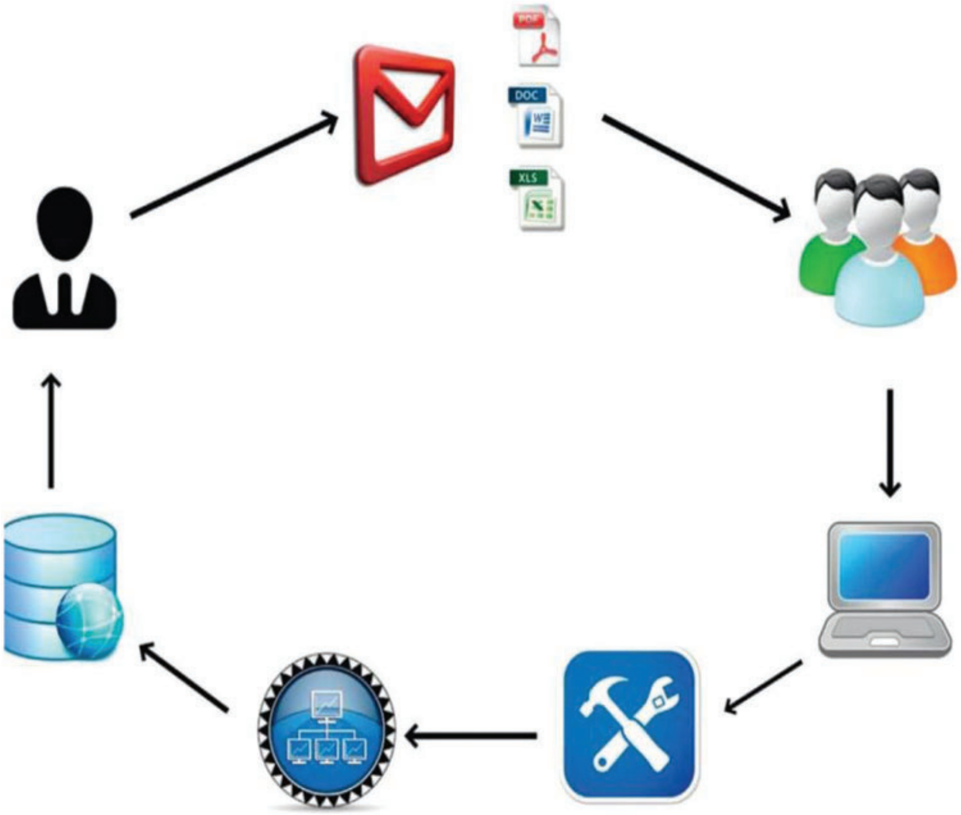

Figure 9: Spear phishing attack model used to launch targeted attacks

Along with these techniques, some APT groups use water-hole attack, removable media [34,35], and software used [36,37] used within a network and malicious code for the initial intrusion. Some APT attacks use known Vulnerabilities to exploit the system. The information about vulnerabilities is available in public databases like "the Common Vulnerability and Exposer List (CSV)" and "the Open Source Vulnerability Database (OSVDB)" [38], " the NIST National Vulnerability Database (NVD) [39]. Also, the vulnerability information is available on the dark web. According to the report [40], most APT campaigns use known vulnerabilities.

After successful malware delivery, the attacker will wait until it gets executed. Once the malware is executed, the attacker will find a way to enter the target organization's network. After that, the attacker will try to gain control over the compromised system for further attack progression.

3.2.3 Command and Control Communication

APT actors typically use commands and control (C&C or C2) channels to gain remote access to the target environment to execute malicious instructions or to exfiltrate the data [41]. In fact, during the installation phase, most backdoor malware is used to connect the victim's system to the attackers' C2 infrastructure [42]. Backdoors may share their IP address or domain name with the servers. Attackers have used various strategies to control compromised devices inside the target network environment remotely. Most APT attacks will use outbound connections to avoid the current detection techniques, as per the study [17,43] and our analysis. Since most organizations label HTTP-based communication acceptable, APT campaigns use this communication method. It is easy to identify the distinctive features of the remaining protocols [44,45].

3.2.4 Lateral Movement

Most APT groups try to move laterally through the network by infecting more devices until they reach the high-value targets. To transfer data between computers, they might use openly available tools such as the Windows credential gathering tool Mimikatz and the remote command-line execution tool WinExe [43]. The most often employed network reconnaissance and lateral movement strategies in the targeted attacks are [30] Information Reuse Attacks, Credentials Dumping, Pass-the-Hash Attack Model, File Sharing Services (Shared Access), and Batch Scripting, Command Execution and Scheduling.

The main goal of this phase is to expand their control by exploring the additional system toward the high-value target. The APT malware uses valid user credentials and standard operating system tools to appear in regular network traffic. The attacker relies on tools like “Windows-Credential-Editor(WCE) to extract the credentials. Once the attacker enters this stage, it is challenging to send out the network [46].

3.2.5 Attack Objective

The attack objective may include the following:

- Cyber espionage
- Sobotage
- Data Destruction
- Financial Theft
- Complete site takeover (DOS)
- Defacement (make the resources or services unavailable or disturb the functionality)
- Intellectual Property Rights theft

The damage caused by these attacks is high due to highly skilled attack actors. The statistical analysis of publicly available APT reports shows that information stealing is the primary objective of the maximum APT attacks [47,48]. Once the data servers are discovered, the attacker will transfer the data to the C&C infrastructure. To avoid detection, the attacker will compress the data before sending it to the server. The most commonly used technique is “staging servers” [29] for data aggregating. The attack objective is also not fixed. The attack objective may be stealing sensitive information, network service interruption, Data destruction, Data manipulation, Service stop, DOS attack, System shutdown, and Financial gain.

4 Case Study

4.1 Stuxnet

An APT that targeted the uranium enrichment plants’ industrial control systems (ICS), forcing the centrifuges to operate at dangerous speeds until they self-destructed, known as Stuxnet. Stuxnet aims to activate the payload in controllers at the uranium enrichment plant in Natanz, not in other situations. In June 2009, the first Stuxnet sample was found. It was incredibly complex and noteworthy because Stuxnet relied on four zero-day vulnerabilities, digital signature forgeries, and in-depth knowledge of the target environment or infrastructure [49].

Since the malware needs a thorough understanding of the uranium enrichment plant, there might have been a physical component to the survey. How the survey was carried out to create and implement successfully Stuxnet is still being determined. This could have started with an insider—someone employed by the plant, for example—who provided the appropriate person with access to crucial data. The reason behind the 2006 leak of Natanz's location and purpose by a dissident group is known, and it is possible that the Stuxnet developers used this information when developing the malware [50].

Although the exact source of the Stuxnet infection is still unknown, the malware was thought to be introduced into a computer using a removable drive. This is a likely infection method because Stuxnet can infect removable drives and do so in order to propagate itself. An insider who may have been compensated or coerced to carry this action could have done so. As an alternative, an attacker may have capitalized on people's innate curiosity by hiding many USB devices carrying Stuxnet in a well-traveled area, like a parking lot close to a target organization (Bursztein) [49].

Only some things about the lengthy and technical Stuxnet installation process are pertinent to or within the purview of this paper. We only describe the portions that can be used to derive features. The primary part of Stuxnet is the a. dll file that includes every exported function (referred to as "exports" in the literature) and resource that Stuxnet uses to carry out its operations (the purpose of Stuxnet is constant, but its behavior varies based on the environment it is in) (i.e., installed security products, the architecture of the victim system, etc.). This comes with two encrypted configuration blocks—dll data. The information Stuxnet collects about the victim machine is tracked by the configuration blocks, which it uses to function correctly. These three items are displayed in a wrapper program's "stub" section of a dropper component. It is this wrapper program that loads the main Stuxnet executable. DLL file and export calls. It also ensures that every export has access to all of the malware's required components by passing a pointer to its original stub section as a parameter to the export and doing the same for every export that comes after, usually how Stuxnet operates by injecting the whole program into one of the trusted processes that might be operating on the target computer.

The attacker exports the source once the DLL file is loaded for the first time. It checks whether the target computer is running in a compatible version of Windows along with security information and running processes to find the best infiltration point. Through collected information, if the attacker determines that Stuxnet is not running on a compatible version of Windows, the threat will be terminated, and nothing more will happen. If it discovers that it does not already have more privileges, it will use one of the two zero-day vulnerabilities to obtain them; if not, it will continue. After that, this export will inject the to call another export. dll file into the process, which the installation process will use as its primary export.

This new running export will confirm the accuracy of the configuration data that Stuxnet had previously collected. After this is established, it will verify that a given value in a particular registry key exactly matches 19790509, and at this point, it will stop all further execution. However, we observe that Stuxnet performed these exact checks on its target system, and we will revisit this idea in the future. This new running export will now check to see if the configuration data that Stuxnet obtained earlier is accurate. Once this is known, it will verify a particular value for a specific registry key, and if it exactly matches 19790509, it will stop all execution. This number is highly significant since it occurred on the same day. However, we highlight Stuxnet's actions in doing these same tests on its victim system and will return to this idea in a later section.

Stuxnet will then use the information in its stub section to generate three encrypted files, which it will subsequently write to disc. Verify that by decrypting a file it just wrote to disc, it then confirms that its current version is the same and continues if it is. Two files from its resources will be decoded and written into a disc. To escape detection, Stuxnet will modify the file creation times of these two files to coincide with the times of the other files in the system directory. Then, Stuxnet will add registry entries to the victim system to ensure these two files run whenever Windows boots up on the device. Two more exports are then started. New detachable drives will be infected, and the RPC server will be started for command and control activity. On the target computer, the other will contaminate Step 7 project files. Stuxnet will try command and control action after a brief wait.

A WinCC SIMATIC server was connected to specific Siemens PLCs during the Actions on Objectives phase of the Stuxnet kill chain. The system was then modified to run the centrifuges outside of safe conditions so they could self-destruct. Additionally, to prevent anyone from suspecting that something was amiss and allow the malware to continue damaging the system for as long as it needed to, the malware forced the system to stop alerting plant employees that the plant was operating in unsafe conditions. Figs. 10, 11 show the high-level kill chain model of Stuxnet [51].

Cyber Kill Chain (CKC) is another way to "decompose" the complex attack and identify the relevant characteristics, which was described in [43]. The author compared various APT-related taxonomies, analyzed the various APT campaigns, and proposed a new CKC-based taxonomy. Table 5 lists the techniques used during the life cycle [43] based on the Cyber Kill Chain (CKC).

5 Methodology

As mentioned in the introduction, the current work focused on discussing various APT detection methods. We used the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) approach, which involves identifying articles, scanning or selecting articles for relevancy, filtering, and finalizing the articles. PRISMA approach offers minimal items for studies on a particular subject. To be more precise, the initial steps under the current work have been as follows:

- The most well-known literature databases, including Scopus, ACM, IEEE Xplore, Science Direct, and Springer Nature were carefully examined to find relevant articles.
- The following keywords related to the subject of study were combined to query the aforementioned databases: "Advanced Persistent Threat" AND ("detection" OR "study" OR "machine learning" OR "supervised machine learning" OR "unsupervised machine learning" OR "graph-based analysis" OR " graph-based analysis") AND "security" AND ("intrusion detection systems" OR "IDS" OR "anomaly detection"). An additional search term, "APT" AND "Intrusion" AND "IoT vulnerabilities" OR "APT actors" OR "APT applicable schemes," was used to search the literature for Pertinent works by focusing on APT and its detection.

Second, a set of fundamental inclusion and exclusion criteria was used to combine the selection process results into a list of the papers that were ultimately chosen.

6 Detection Methods

6.1 Detection Based on Attack Step

APT attacks are targeted and operate in stealthy procedures; hence, detecting them in the early stages is difficult. During the attack process, there are three primary phases: Initial intrusion, Command & Communication, and Lateral Movement; during these, we have high chances for detection.

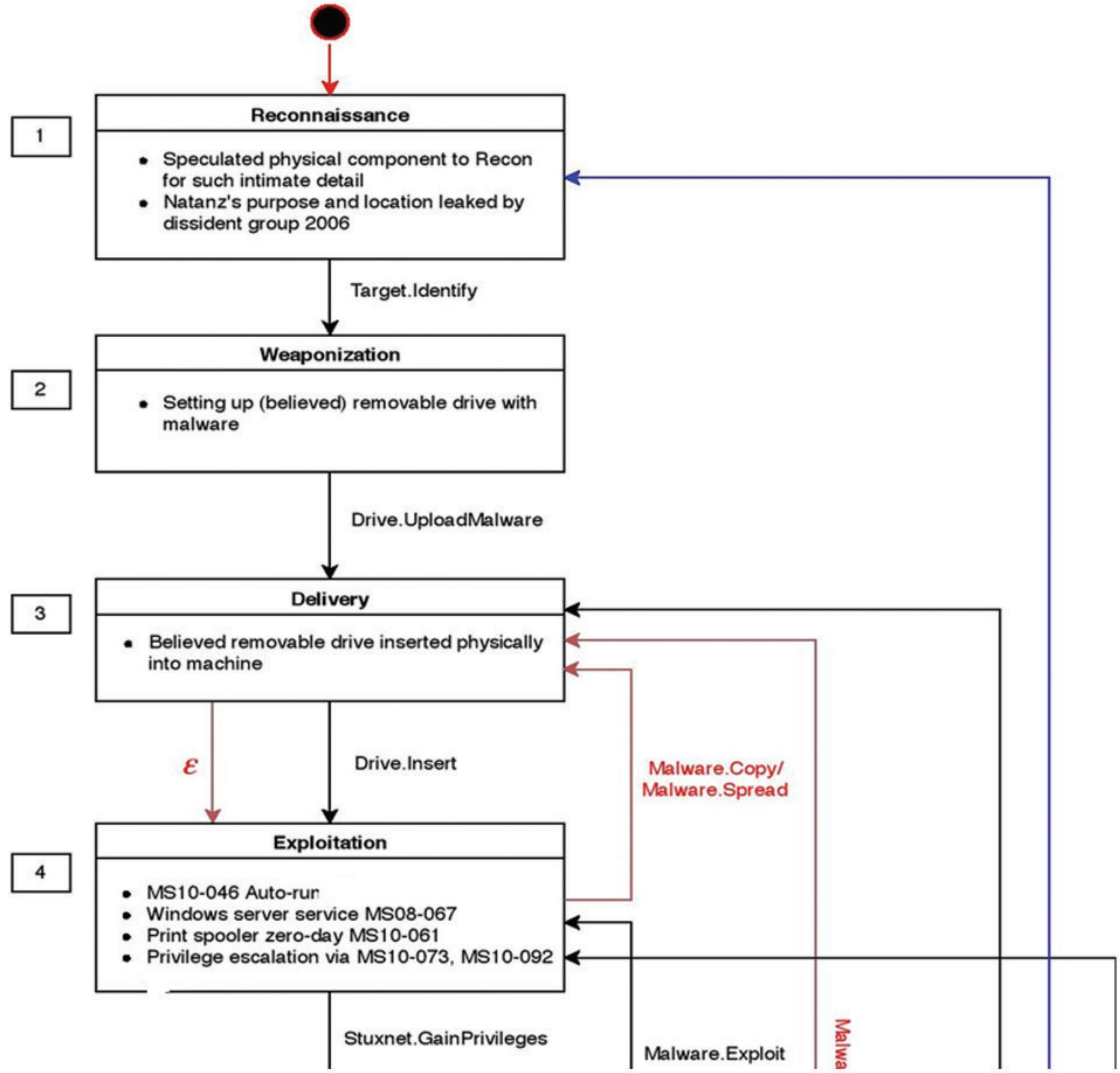
1
Reconnaissance
• Speculated physical component to Recon for such intimate detail
• Natanz's purpose and location leaked by dissident group 2006
Target.Identify
2
Weaponization
• Setting up (believed) removable drive with malware
Drive.UploadMalware
3
Delivery
• Believed removable drive inserted physically into machine
ε
Drive.Insert
Malware.Copy/
Malware.Spread
4
Exploitation
• MS10-046 Auto-run
• Windows server service MS08-067
• Print spooler zero-day MS10-061
• Privilege escalation via MS10-073, MS10-092
Stuxnet.GainPrivileges
Malware.Exploit
Malwa

Figure 10: Stuxnet diagram for detailing steps (1, 2, 3, 4) of kill chain

6.1.1 Detection of Initial Intrusion

As part of the APT attack lifecycle, the attacker establishes the entry point into the target network environment during the initial intrusion. The malware is installed on the susceptible host, creating a backdoor for additional attack attempts. The method that is frequently employed is spearphishing. Installing malware on the website that the affected employees are probably going to visit is another common way of infection. Identifying potentially harmful files or documents is crucial for early APT defense.

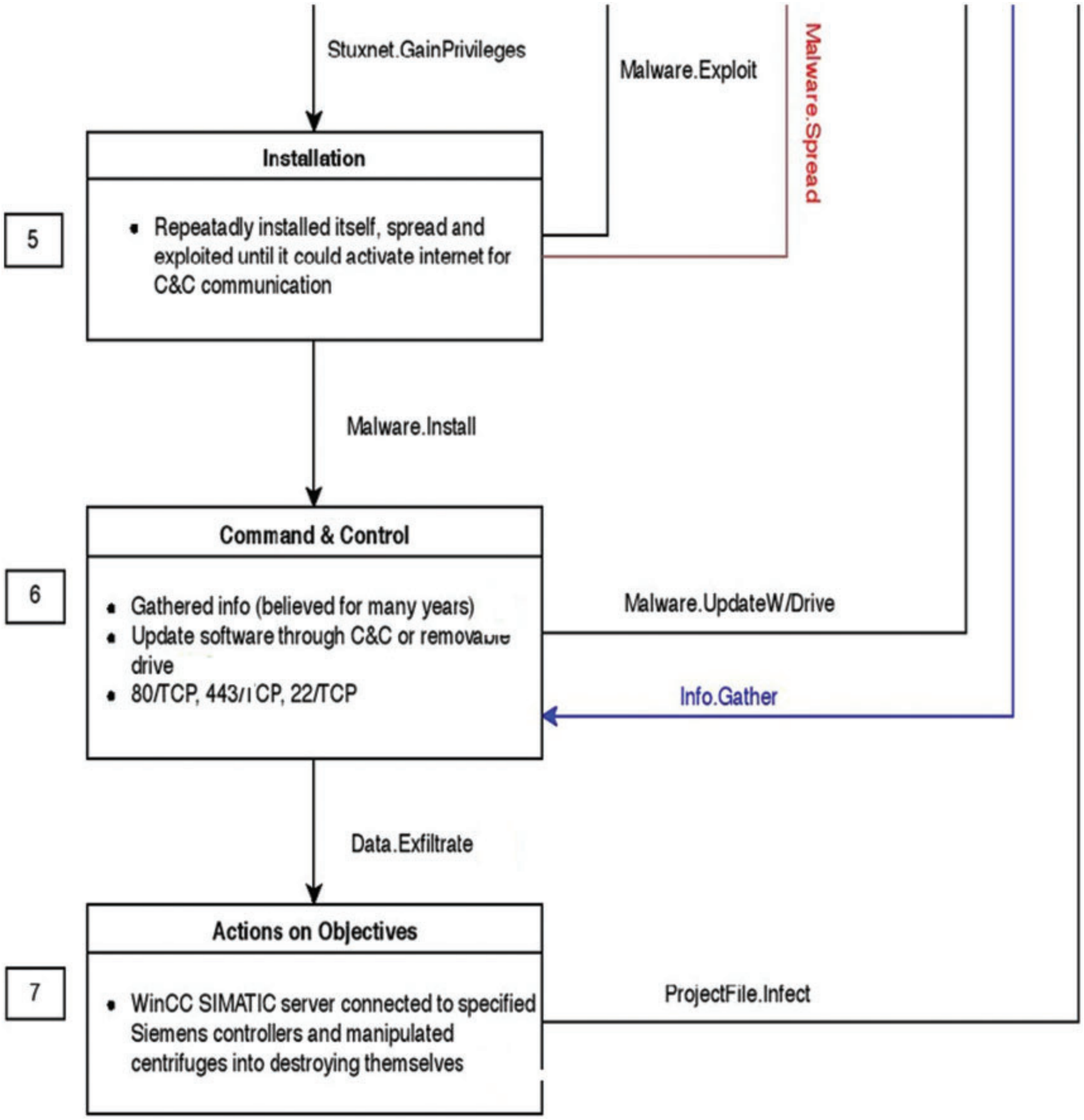
Stuxnet.GainPrivileges
Malware.Exploit
Malware.Spread
Installation
5
• Repeatadly installed itself, spread and exploited until it could activate internet for C&C communication
Malware.Install
Command & Control
6
• Gathered info (believed for many years)
• Update software through C&C or removable drive
• 80/TCP, 443/TCP, 22/TCP
Malware.UpdateW/Drive
Info.Gather
Data.Exfiltrate
Actions on Objectives
7
• WinCC SIMATIC server connected to specified Siemens controllers and manipulated centrifuges into destroying themselves
ProjectFile.Infect

Figure 11: Stuxnet diagram for detailing steps (5, 6, 7) of the kill chain

Table 5: APT features based on CKC

Attack phase	Features identified
Information gathering	Social engineering, Web based tools
Initial intrusion	Spear phishing: Attached file or link; Media which is removable; Water hole attack: through malicious DNS or software
Command & Control	Removable media; using protocols (HTTP, FTP, SMTP DNS, SMTP, POP3, SSH/TLS, Satelite)

(Continued)

Table 5 (continued)

Attack phase	Features identified
Lateral movement	Vulnerbilities in existing softwares; SQL injection; Driven by download; Bootkit; DLL slide loading or hijacking
Attack objective	Data destruction or exfiltration

In [52], the authors define advanced persistent social engineering (APSE) as social engineering in which a sophisticated intrusion continually penetrates the target's psychological aspect, causing the target to behave as the attacker intends readily. Authors proposed a framework that captures the change in human behavior.

In [53], a framework for detecting initial intrusion was proposed based on the analysis of OpenXML documents. The author used the IOM (Indicators of Malicious) for detecting malicious files. The proposed framework can see malicious documents automatically and is mutable. The author achieved better detection accuracy compared to existing methods. The limitation of the proposed framework is that it cannot process large files.

In [54], a module was presented to detect the distinguished exe files transferred over the network. The proposed module is based on comparing file name extension and MIME type of file, i.e., it will detect the files with .exe extension, but the content is not exe. The module processes the network traffic data of all points and filters the exe files transferred over the network. The filtration is done based on the content of the files. Finally, the content is verified, and an alert is raised upon detecting a distinguished exe file.

In [55], the authors presented a methodology for detecting the point-of-entry (POE) in APT attacks. The proposed algorithm is based on mathematical and computation analysis. The proposed algorithm searches for specific characters and words called tokens (like click here, free, replica, Viagra) in the Email. According to the result, spam mail will be distinguished from regular mail. All these tokens are defined to this detection algorithm before applying this.

In [56], the authors proposed two detection methods for detecting Spear-phishing emails. One is malicious domain detection, and the other is malicious file hash detection methods. These methods are blacklist-based methods for spear-phishing email email detection.

In [57], the authors introduced an active learning-based system for detecting PDF files for carrying the APT attack's malware code. The proposed system analyses the network traffic data and filters all PDF files over the network connection. The developed module filters the gathered files into the malicious and non-malicious base on white-list, repositories of antivirus signatures, and reputation systems. Finally, it will compare the remaining files' compatibility and add them to the blocklist or allowlist for future learning.

In [58], the authors investigated Duqu to detect the malware code used in the initial intrusion. Duqu is a piece of malware used in the APT attack against European companies. The main goal of this attack is data exfiltration. This work developed a toolkit for Duqu detection. This toolkit included six tools, which are grouped into three: file existence anomaly detection, register entries, and file properties detection and code injection analysis. Some of these tools are very general and can detect the APT even if there is a change in malware code.

In [59], the author introduced a detection framework for APT based on the analysis of directory logs. The proposed method will collect log data periodically and sequential context applied for behavior analysis. Based on the different behaviors, the probability Markov model is used to detect malicious behaviors.

6.1.2 Detection of Command and Control Communication

Command and control communication plays a vital role in an attack. The C&C communication is not just once but regularly multiple times. Generally, it is the first time after initial intrusion and later for data transfer. The attacker needs to maintain this communication till the end of an attack. This communication is also essential in other types of attacks like Botnets. Detection of Command and Control Communication helps to control all types of attacks, including APT. By the analysis of APT attacks, the following are some of the features identified by the researchers for detecting Command and Control Channel in APT attack detection:

- Most of the APT attack communication is HTTP-based [60].
- The Domains used for the C&C communication are independent [61].
- Frequent changes in the IP address of urls.
- Communication with unknown domains [62].
- Duration and frequency of communication.
- Low and slow communication [3].

The existing detection methods are based on analyzing HTTP features, DNS features, DNS names, and multiple traffic types inside the network [63].

Most recently, stealth attacks have deceitfully concealed malicious activity, which uses seemingly harmless applications to pose as regular connections to well-known online services. These techniques frequently avoid detection by signature-based techniques or traditional network monitoring because attackers often conceal Command and Control (C&C) servers within reputable cloud service providers, giving the appearance of average traffic anomalies. In [64], the authors presented Anteater, an application-level monitoring system. Anteater creates a thorough profile that describes the expected traffic patterns for every legitimate software's network traffic behavior. Anteater effectively identifies and intercepts the IP addresses linked to unusual program access by closely examining the network traffic configuration of a program. Tested on a dataset comprising more than 400 million real-world network traffic sessions, Anteater was deployed in an actual enterprise setting. Anteater's evaluation results show a high detection rate of malware injections, with less than a 0.1 percent false positive rate and an actual positive rate of 94.5%.

In [65], based on DNS traffic, the author presented a novel Deep-Learning-based detection technique for identifying malicious C&C communication. The DNS traffic data will be gathered, and pre-processing will be used to classify the data. They then assess the behavior of the processed data and, lastly, use Deep-Learning algorithms for the assessment. The primary drawback of the suggested detection method is that it will identify a regular domain name as malicious if an attacker uses one.

In [66], the authors introduced the method of monitoring and detecting APT attacks based on the unknown domain feature through the analysis of DNS logs and a variety of monitoring techniques. They achieved this by detecting the C&C channel using the Random Forest algorithm and 25 features.

In [67], the authors suggested a C&C domain detection system based on Domain-graphs. The suggested approach investigates the connection between IP addresses and domain names. It expands it to include mapping C&C domain names to detect malicious C&C. The primary constraint of the suggested work is that the author should have considered domain name relevance. The proposed method cannot identify the APT if the attacker's domain name is identical to the regular domain name.

In [68], the authors introduced a machine learning-based methodology for detecting malicious domains using an Extreme Learning Machine (ELM). Domain analysis is helpful to experts in detecting the command and control communication APT. An extreme learning machine is a modern neural network with a high learning rate and accuracy. The classification is based on the features extracted from multiple sources. These features are indicative of the association of domains and malicious activities. The features used for this detection could be more stable and adequate for the advanced attacker.

In [69], a new algorithm for detecting Advance Persistent Threats (APTs) was proposed based on a graph model of HTTP traffic. The compromised computer will connect to the command and control server at fixed or variable intervals, resulting in traces in HTTP traffic data. They build a graph for HTTP traffic and apply the graph pruning method to detect the APTs.

In [70], the authors proposed a new method for detecting APT malware and malicious C&C channels based on the DNS logs analysis. The proposed method computes a domain scoring matrix using Alexa ranking and Virus-Total's judgment result and identifies the regular domain. Finally, the anomaly detection algorithm (Global Abnormal Forest) is used for malicious C&C domains.

In [71], the authors proposed a method to examine HTTP-based network traffic to identify malicious C&C communication channels using Web-request Graphs. A work proposal has been developed based on the crucial finding that APT attacks will employ HTTP-based communication and that this type of communication will differ from ordinary communication traffic. They created a web request graph and filtered out malicious requests based on the dependencies between them. The author tested the suggested methodology with the help of nine APT attacks. The proposed work's drawback is its need to analyze vast network traffic volumes.

In [72], the authors introduced a new feature called Time Transforms for detecting APT attacks based on malicious payloads. The proposed method assumes that the compromised infrastructure and C&C server are in periodic communication. This is possible only when the attacker moves deeper inside the target network environment (Lateral Movement). The authors used machine learning algorithms to classify network traffic with malicious payloads. The major limitation of the proposed work is that observation of periodicity in APT is challenging.

A new monitoring system that can detect the domains that are maliciously controlled, based on bipartite graphs called " Segugio," was introduced [73]. The proposed system-generated bipartite graph for DNS logs. Next, it will extract the characteristics of every node in the graph. Finally, machine-learning algorithms are applied to categorize the domains controlled by malware. The limitation of the proposed system is it can detect malware-controlled domains. It is challenging to detect these domains if they have not been used by the compromised host previously.

In [74], the authors introduced the detection system for the C&C server by combining the Traditional intrusion detection methods and malicious DNS features. The author proposed a new security system for detecting the malware inside the network by analyzing DNS traffic and all data generated in the network. The proposed method reduces the amount of data we need to record and increases system sustainability. They extracted 14 different malicious features of DNS characteristics on network traffic and used these extracted features to train the Decision-Tree algorithm to detect APT C&C domains. Next, HTTP traffic features are used to build the intrusion-detection system.

Finally, the author sends three detection module results to the reputation engine to detect the infected IP address.

A novel search engine-based method for finding the C&C server address called "CAFSE" was introduced in [75]. CA-FSE is composed of five modules: The publish module (PM) is used to publish the C&C server IPs in several free blog diaries. The search engine assigns indexes to these diaries. Key Production Module(KPM): when an infected host uses this to produce the MD of data, the related keywords are transferred to SERPs (Search Engine Result Pages). The noise Item Filter Module (NIFM) was used to remove the other data except the C&C IP addresses. Finally, the Extraction and Conversion Module (ECM) will convert the obtained IP address into binary.

A scalable solution for detecting C&C called "Ctracer" was defined in [76], which can process a large amount of network traffic using Map-Reduce. The proposed solution undergoes three stages: DestHostIP takes web proxy logs as input and processes them with the help of a feature called "one malware communicates with multiple C&C" and groups the IPs. In the second stage, source and destination IPs are clustered based on HTTP features. Finally, the web reputation stage will check the grouped IPs using external Knowledge.

Most researchers use HTTP traffic-based detection methods since they don't need signatures [77], which may lead to a high false rate.

6.1.3 Detection of Lateral Movement

Lateral movement is the most prominent security threat in the network, and according to the study in [78], an average time of 107 days is taken to detect a suspicious packet.it is a critical step in the attack process where attackers try to expand the attack by penetrating the network environment. By successful detection of the lateral movement stage, we may stop the attacker before achieving the goal. Security experts in [79] described 20 lateral movement techniques currently employed by cyber attackers. According to an author [79], three general methods can be used to analyze lateral movement: Path analytics, graph analytics, and clustering.

In [5], an approach for measuring network vulnerability through graph analysis was proposed. The proposed method is used to detect attacks that use lateral movement and privilege escalation using the pass-the-hash technique. To detect an attack, they are using the graph metric, which measures how likely the node is reachable from an arbitrary node to make the network vulnerable. This metric is calculated during network security authorization, and it is used for detecting attacks such as pass-the-Hash (PTH).

In addition to increasing the attack surfaces for APT attacks, the new crown epidemic has led more companies to favor telecommuting for work. Following their initial intranet access, attackers will move horizontally to accomplish privilege escalation by utilizing server message block (SMB), remote file sharing (RDP), and other protocols. In [80], the authors proposed a method to detect lateral movement behavior in an intranet environment by designing a multidimensional detection framework based on the SMB protocol. This framework identifies the adversary's attack samples for lateral movement and combines active trapping and passive scanning with neural networks. By using actual malware samples, we confirm that the accuracy of neural network detection can reach approximately 90%. The author evaluates the efficacy of the active trapping technology in a simulated environment. The proposed method successfully identifies the lateral movement behavior in an intranet environment by utilizing the SMB protocol, as demonstrated by the experimental results.

In [81], the authors presented a comprehensive experiment to detect lateral movement. The author did the experiment on the publicly available APT datasets and evaluated the proposed tool with prominent results. The limitation of this work is that it needs to be applied to real-time network environments.

Furthermore, feature development and engineering approaches were employed to create a dataset that could be used to evaluate the performance of the core Neural Network (NN) supervised classification system. The NN model contained a multilayered hidden and convolutional NN blend, including TextCNN, LSTM, and FastText layers. The produced dataset included over 10,000 publically available harmful malware APT files. In [82], the authors applied various machine learning algorithms to detect the lateral movements using authentication logs. However, the author was left with a high false alarm rate due to poor data flow quality.

In [83], authors suggested LATTE, a brand-new graph-based security system for identifying malevolent lateral motion paths. The graph is intended to handle Windows security events that are produced by service ticket requests for Kerberos. Computers and user logins are the nodes in the graph, while computer-to-computer connections and user login events are the edges. The two primary modules of the suggested system are the forensic analysis module, which creates a graph-based path rate score to find additional compromised accounts or computers, and the compromised account or computer itself. The general detection module combines the path rate score with the remote file extension to filter out malicious paths. The general detection module is also employed to identify fresh attacks.

In [84], authors proposed a new detection method for lateral movement in APT attacks based on the traces left by the attacker at the host and network logs for detection. RDP (Remote Desktop Protocol) is a method for identifying the malicious host based on the traces in network logs. Finally, machine learning algorithms are used to classify the RDP session logs.

In [85], authors used a host-level process communication graph to determine the reasons for network connection and proposed a detection framework for lateral movement using distributed data-fusion dots. In order to detect lateral movement in the system, the connection elements are then aggregated into a system. In the multi-level fusion hierarchy, the authors used various clustering techniques to strike a balance between source utilization and fusion structure robustness. In order to identify any lateral movement, the overhead storage was assessed while the connection causality was being watched.

In [86], authors proposed lateral movement detection within a network of Bipartite anomaly scores. They used bipartite graphs to derive the bipartite anomaly scores. Although this method does not yield a compatible outcome, supervised classifiers with ancillary features have shown improvement. There are nine types of datasets in the authentication events record. The datasets are "time, source user, destination user, source computer, destination computer, authentication type, logon type, authentication orientation" and "success/failure." The researchers only focused on the first five data types in this method. The results might be inconsistent if all types are used.

In [87], a multi-stage transferable Markov model was proposed for detecting data exfiltration in APT attacks. The authors use the probability to predict the impact of the next APT attack action. To express the attack behavior, the author used the 3-tuple format. The proposed model is beneficial to security experts for the early detection of APT. The drawback of the proposed model is that the Author does not consider the dynamic nature of APT-AN (APT network).

6.1.4 Detection of Data Ex-Filtration

It is one of the attack objectives. In this, the target may lose their sensitive information. Sometimes, it may lead to another attack also.

The focus of recent cyberattacks has shifted from disrupting services or causing financial loss to secretly stealing private information. APTs, or advanced persistent threats, present a severe threat because of their sophisticated and dynamic attack mechanisms. The static and unadaptable nature of traditional deep learning techniques for APT detection renders them unfit for addressing the dynamic and evolving attack scenarios that are frequently observed in uncertain network traffic flows, like multi-stage APT attacks.

In [88], the authors suggested a framework called APT-DRL, based on the Deep Reinforcement Learning method for APT detection as a solution to stated problems. This method continually adjusts to new attack patterns by learning dynamically from interactions with the environment. Performance assessments show that APT-DRL can create new APT detection policies by efficiently learning from dynamic network interactions. As a result, APT-DRL outperforms Feed Forward Neural Network (FNN) models regarding learning speed and accuracy. FNN models are less flexible and adaptable than the suggested APT-DRL approach.

In [89], a detection method for data exfiltration was proposed based on DNS tunneling. The proposed method undergoes three phases: first, it collects the DNS log data over a long period; in the second phase, features extraction is based on the querying behavior of every domain. Finally, they apply the anomaly-based detection method to classify the malicious and normal domains. The major limitation of the proposed method is the author only focuses on detecting DNS under low-throughput conditions.

In [90], an APT detection method was proposed based on the events logs generated by Splunk and APT extracted features. By analyzing the data, the author defined some features most likely to be in every APT attack. Finally, the machine-learning tool kit detects data exfiltration based on the specified characteristics. The limitation of the proposed work is not detecting the special events generated in long-term APT attacks, detecting the events that may likely be a part of an APT attack.

6.1.5 Evaluation and Comparison

Detecting only one step is not sufficient for the APT attack. The output of these detection methods wants to correlate with other stages for detecting APT as a whole. Besides, we cannot predict the current status of attacks inside the network by using single-stage detection. Due to the dynamic nature of APT, we need an automatic detection system for the initial intrusion phase since it is the entry point of the attack. We need to analyse the data from multiple network points because the attacker may exist anywhere and enter from any network edge. Besides, the APT attack involves various compromised hosts, not one.

Few works are based on blacklisting and whitelisting, anomaly based [2,56,91]. But, APT malware can hide in the multiple layers of the network. For example, attackers keep changing their malicious URLs every time, so using blacklisting and whitelisting is insufficient to prevent these types of attacks. We need a correlation-based or detection method that can process complete network flow analysis methods for full-fledged APT attack detection.

The APT detection idea is to identify some unique features in APTs and track those features for detecting the life-cycle stages. The specified features are used for detecting the possible APT attack inside the network. For this, we need to collect the data from multiple points of the network and process it accordingly. Machine learning algorithms such as Decision trees, Neural networks, Deep learning, etc., help us process a large amount of data in less time. Some are introduced graph analysis based detection methods [48] and log based detection [92]. Graph analysis can process the high-level network for advanced attack detection. In this, we need to build the attack graph for the network. The attack graph is used for identifying the attack path of the attacker. The attack graph path analysis helps us in the detection of the most critical regions in the network. The following Table 6 summarizes the existing detection methods along with limitations.

Table 6: Comparison of stage based detection methods

Author	Stage	Data/Data-sets used	Method/Approach	Limitation
[2]	Initial intrusion	PDF files transferred over network connections	White lists, antivirus signature repository, reputation systems	The APT attacker can easily evade by using other techniques, rather than malicious PDFs.
[53]	Initial intrusion	OpenXML documents	IOM (Indicators of Malicious) score	Not suitable for large files.
[54]	Initial intrusion	Network traffic data	Based on the comparison of MIME type of the file and its file name extension	Agent based, able to detect only one step of APT attack.
[55]	Initial intrusion	E-mails transferred over network	Bayesian spam filtering is a mathematical concept used in spam or junk mail filters	There is no guarantee that the spam email must include with one of these tokens.
[56]	Initial intrusion	Network traffic data	Blacklist based detection	Able to detect only known samples.
[65]	Command & Control	DNS traffic	Random forest algorithm	Tested with only two datasets.
[61]	Command & Control	DNS traffic	RIPPER classification algorithm	Not able to detect the command & control communication when user is using web.
[66]	Command & Control	DNS traffic	Deep-learning algorithms	Not able to detect when malicious domain is using the normal domain name.
[68]	Command & Control	Multi type traffic	Machine-learning algorithms	Unstable and inadequate in case of advanced attacker.

(Continued)

Table 6 (continued)

Author	Stage	Data/Data-sets used	Method/Approach	Limitation
[69]	Command & Control	HTTP traffic	Graph-based detection	Very difficult in case of large amount network traffic analysis.
[71]	Command & Control	HTTP traffic	Graph-based detection	Not works with large network traffic.
[72]	Command & Control	Multi type traffic	Machine-learning algorithms	Observation of periodicity in APT is very difficult.
[73]	Command & Control	DNS traffic	Machine-learning algorithms	Difficult to detect these domains if these are not used by compromised host previously.
[74]	Command & Control	DNS traffic	Anomaly-detecting algorithms, signature-based algorithms	Not able to detect the malware relies on domain.
[76]	Command & Control	Multi type traffic	Map-reduce	Centered not on the internal network but on the network edge.
[83]	Lateral movement	Windows process log data	Graph-based detection	Consider only windows operating system events.
[84]	Lateral movement	Log data	Log data analysis, RDP protocol	Not detect the new attack patterns; security experts must manually design new ones.
[85]	Lateral movement	Host-level traffic	Analysis of host level process communication graph.	Fail to detect, if there are no behavioral changes in attack. Focused on the network edge rather than the internal network.
[86]	Lateral movement	Events record	Bipartite garaph	Author focused on first five data-types only. The results might be inconsistent if all types are used.
[87]	Lateral movement	Network log data	Markov model	Author not considered the dynamic nature of APT.

(Continued)

Table 6 (continued)

Author	Stage	Data/Data-sets used	Method/Approach	Limitation
[89]	Data exfiltration	DNS traffic	DNS log data analysis	Author only focuses on detecting DNS under low-throughput conditions.
[90]	Data exfiltration	LOG data	Events log data analysis	Not able to detect special events generated in long-term APT attacks.
[92]	Initial intrusion	Active directory log data	Anomaly-behaviour based detection	Able to detect only known anomalies.

6.2 Correlation Based Detection

For detecting long-term attacks like APT, we need correlation based detection echniques. APT is a combination of one or more attack techniques used across multiple stages. All stage techniques need to correlate for effective detection of an APT attack.

In [27], a security framework against APT was named OpenIoc. The proposed framework collects information about APT attacks using compromise (IOC) indicators, finds the techniques, tools, and procedures (TPs), and models the threat sequence based on IKC. Finally, similarity analysis is performed, i.e., the similarity between the threat sequence generated and the sequence of known APT is analyzed; if a match is found, it will raise an alert.

In [62], the authors proposed the "New York security situation awareness model (NSSA)" through knowledge graphs. Firstly, the vulnerability knowledge graph was constructed based on the vulnerabilities (which may be exploited by APT attackers) presented in the current network environment, and the targeted knowledge graph was combined to assess the current situation. However, it is only a comprehensive analysis.

The increasing sophistication of cyberattacks in the form of advanced persistent threats (APTs) has focused much attention on anticipating and countering APT attacks. Attack graphs, Hidden Markov Models, and Bayesian networks are examples of related studies, but they have four notable limitations: (i) non-standard attack modeling, (ii) lack of data-driven approaches, (iii) lack of real-world APT dataset and (iv) high system dependability. In [93], the authors proposed a system-independent data-driven approach called the Bayesian ATTandCK Network (BAN). In particular, BAN uses the MITRE ATT&CK® framework to model APT attackers using a Bayesian network that employs structure and parameter learning. The trained BAN aims to anticipate impending attack strategies and generate appropriate defenses. Furthermore, preprocess datasets use automatic and manual labeling to address the problem of inadequate data for APT prediction. Based on extensive evaluations, optimal parameters that BAN can use to handle APT attacks are demonstrated in experimental results.

Advanced persistent threats are becoming more sophisticated and frequent—Present-day detection systems generate alerts based on individual procedure analysis. Since APT attacks seldom consist of a single activity, it is necessary to correlate individual alerts to fully capture APT activity and give operators a better understanding of their situation. In [94], the authors used this to launch preemptive and focused countermeasures, enhancing security. The correlation engine presented by the author correlates alarms from standard rule-based systems with one another. They are using an APT scenario as an example. The author assessed the suggested solution and reviewed this methodology's benefits and drawbacks. In the face of sophisticated living-off-the-land attacks or even zero-day exploits, rule-based systems' limited informative value must be taken into account due to their quick and easy implementation, which is an add-on to SIEM.

Because nation-states and sophisticated corporations are increasingly interested in obtaining high-profile information, there has been an increase in advanced persistent threats (APT). Since APT attacks use standard benign tools and zero-day attacks, they are typically hard to detect. In addition, these attack campaigns are frequently extended to avoid detection. To identify unusual activity, in [95], the authors employed a method that gathers host node execution traces via a provenance graph. The author extracted the features from the provenance graph and used them to train an online adaptive metric learning system. Using a deep learning technique called online metric learning, a function to maximize the separation between dissimilar instances and minimize the separation between similar classes is learned. Finally, it demonstrated that the proposed method outperforms baseline models by increasing the true positive rate (TPR) by an average of 18.3% and improving detection accuracy by 11.3%. Furthermore, the proposed approach outperforms the performances of several state-of-the-art models in extensive attack datasets in binary and multi-class scenarios.

Advanced Persistent Threat (APT) identification and analysis is essential to modern network security. Provenance graphs, created from audit logs and widely used in the APT detection field, provide a multitude of contextual information that can be used to identify and analyze threats. Unfortunately, explanatory capabilities for detection results are often lacking in existing approaches, adding to the burden on security analysts. Analysts cannot quickly respond to threats when faced with coarse-grained detection results because they have to dig into audit logs or provenance graphs to identify attack entities and events.

In [96], the authors proposed a revolutionary attack detection method for APTs called attack intent-driven and sequence-based learning (AISL), which uses the provenance graph constructed against the various data. In [97], authors proposed a method for real-detection of APT attacks based on the causal relationship between the APT stages. Initially, the proposed method analyses the host-based traffic. It computes the host's infection score, which defines the likelihood of exposure to an APT attack. Based on the analysis of meta-alerts generated from security and non-security sensors, the author discovered the possible IKC of APT against the compromised host.

In [98], for attack detection, the authors suggest a sequence-based learning methodology (SLAII) that identifies attack intent by methodically integrating pertinent heterogeneous security data. The author creates a specific network event ontology and first looks into different attack detection data sources.

The ontology integrates heterogeneous data into a provenance network to guarantee data homogeneity. Second, we harness the security expertise of industry domain experts to understand the purpose of an attack, which allows us to find and exploit features highly connected with advanced persistent threats (APTs). The author tests SLAII in ten realistic environments using APT attacks. However, every APT attack has its techniques and customized tools. There is no guarantee that the proposed method will give promising results for every attack.

In [99], a detection method for APT was proposed based on the correlation of network events and operating system events using semantic relationships defined over the system entities. The proposed method detects malicious events based on the defined event correlations described in the security policies. The proposed method is not suitable for long-term attacks.

In [100], a novel intrusion detection system was proposed for APT attack detection and prevention. The proposed system undergoes two phases: the first phase is attacking scenario reconstruction; based on the events identified in the network, it will correlate or link the events related to the APT attack. The second phase is attack decoding; this phase uses the Hidden-Markov-Model (HMM) to determine the most likely sequence from given alerts. The proposed system can also predict the most probable step on behalf of the attack using each stage's probability.

In [101], they proposed an attack path modeling technique for cloud computing based on a Bayesian network. The author addressed the APT as a challenging and characterizing attack. Chain the conditional and marginal probabilities for characterizing the multiple attack paths from the attack source to the target node. They evaluated the likelihood of the APT attack path and proposed a Bayesian network-based optimal algorithm for finding the shortest way to target from multiple sources.

In [102], a new method for correlating APT alerts and logs named APTALCM was proposed. The proposed method undergoes two modules: Alert Instance Correlation Module (AICM), which will convert the alerts into ontology instances. The second module is the Log Instance Correlation Module (LICM), which correlates instances to the various APT steps—through this, the author achieved low-false positives.

Detecting the APT attacks in the early stages would minimize the damage and chance of preventing the attacker from achieving the attack goal. For early detection of an attack, In [103], authors proposed a new machine learning-based detection method for APT attacks called MLAPT. This system was able to detect the attack very quickly and accurately. MLAPT undergoes three modules: (1) Threat detection: this module will detect the different APT step techniques and raise an alert. (2) Early warning detection will link or correlate the alerts and identify early warning that may relate to APT; (3) Attack prediction, which predicts the probability of attack. The experimental evaluation of MLAPT achieved an early prediction accuracy of 84.8%. The main drawback of the proposed work is that the author considered only events generated from the network traffic, but in APT, a few stages execute the host level.

In [104], a new approach for detecting APT attacks based on the analysis of host-level data was proposed called HOLMES. The proposed method suggests the host audit data and generates a signal upon detection of activities of an ongoing APT attack. It used low-level data at the host level, like files and processes, etc., for alert correlation. The major limitation of the proposed work is that it needs to consider the system call information ineffective in attacking multiple entry points.

In [105], it proposed a novel APT detection method for SDN (Software Defined Networks). Based on IKC, the author described the APT attack in SDN as having four stages: The formation-investigation stage, scan stage, intrusion stage, and revenue stage. The proposed method uses pieces of evidence left by the attacker in every stage of attack for detection: (1) It will collect data from different data sources and apply different detection methods to analyze attack behavior. (2) It will calculate the degree of correlation between the collected attack behaviors and find the attack path. (3) Match the attack path with the attack tree to discover APT in SDN.

In [106], they creatively proposed a distributed framework architecture for detecting APT attacks named DFA-AD, based on multiple parallel classifiers. The proposed framework undergoes three phases: In the initial phase, the possible techniques used by an APT attack are identified by analyzing the network traffic data. Various machine learning classification algorithms are used to classify the events. The second phase is the event correlation module; it takes all the events generated in the first phase as input, correlates them, and raises an alert upon discovery of the APT attack. The third phase is the voting service, based on the correlation module's information, voting among the various detection methods and triggering an alert on an APT attack—the proposed achieved high accuracy and effectiveness.

In [107], a static APT detector based on correlation was introduced. The authors considered that the attacker undergoes five steps, viz., delivery, exploit, installation, command, and control, by detecting the individual techniques in every step, correlating them, and doing statistical analysis to detect APT.

In [108], authors have studied several works on anomaly-based methods and proposed a novel anomaly-based detection method. This work contrasts many other solutions that use a blocklist approach for detection. Their proposed work utilizes the logs produced by various systems and components in the ICT Networks as clauses for tracing the system events and dependencies among the events. Using the event correlation, the proposed solution extracts a model for the detection system, which is used to detect and distinguish meaningful logs through event classes containing implications between the events. This system is automatically generated to detect the realistic APT attack.

In [109], the APT attack and its characteristics are described, followed by the author's suggested model for describing APT-based attacks. They introduced a new model of attacks in the form of an attack pyramid. The top of the pyramid represents the attack, and the pyramid side defines the attack environment. The events generated in the communication connection are recorded through a security tool. Subsequently, all these events are linked based on correlation techniques. APT attacks are identified by analyzing integrated events concerning time and environmental conditions. The author used the MapReduce technique to tackle the problem raised in analyzing large volumes of data. The limitation of the proposed work is that the author considered only recorded events, and many irrelevant events are recorded in experimental analysis.

6.2.1 Evaluation and Comparison of Correlation Based Detection Methods

Table 7 shows the summary of some of the existing correlation-based detection techniques along with their limitations. Here, the correlation between potential APT attack life cycle techniques and detection methods is the basis for APT attack detection. Most methods use machine learning algorithms, For example [103,101]. However, machine learning methods have high hardware requirements, and paying attention to dealing with the overhead in computation is essential. Due to the publicly available assets on APT traffic shortage, researchers rely on synthetic data or construct their simulation datasets to evaluate their proposed methods. Most of the detection systems ignored the dynamic behavior of APT attacks.

Table 7: Comparison of correlation-based detection methods

Author	Data/Data-sets used	Techniques used	Approach	Details	Limitation
[27]	–	Indicators of compromise (IOC)	Mathematica frame-work	l Information flow analysis, IOC detection, Generating the IKC sequence	No real-time detection and not considered the data exfiltration
[100]	Synthetic dataset (6000 alerts)	Hidden Markov model (HMM)	Attack pyramid	–	Considered only the events generated by special resources, Ignored the attack-scenairo
[101]	Data for: Bayesian Net-work based weighted APT attack paths modeling in cloud computing	Bayesian Net-works	Attcak-Path	Attack path reconstruction	Comprehensive study only, left the detection as future-work
[103]	Matlab's Classi-fication Learner application is used to train the machine learning classification models.	DT,SVM, K-NN and Ensemble learning	MLAPT	Threat detection, Alert correlation, Attack prediction	Considered only the events generated by special resources. Incomplete modeling
[105]	Flow tables	Evidences left by the attacker in every stage	Attack tree	Data collection, Behaviour analysis, Correlation of events, Attack-path reconstruction	Dynamic Behaviour aolf-attack was ignored

(Continued)

Table 7 (continued)

Author	Data/Data-sets used	Techniques used	Approach	Details	Limitation
[106]	Semi-synthetic data-sets	SVM, dynamic Bayesian game model, regression tree and classification, and genetic programming.	DFA-AD	Network Traffic analysis, Events correlation, Voting service	No real-time detection, not considered the dynamic behaviour of attack
[108]	Semi-synthetic data sets	Whitelist	Behavior rule based detection	System event logs, Depen-dencies finding, Correlation	For complete APT attack we need to analysis large amount data over a period of time, but the proposed method considered the specific host network
[109]	Data collected from security systems inside the network	MapReduce	Attack-Pyramid	Threat Detection, Attack correlation	No real-time detec-tion and not considered the data exfiltration
[110]	IDS alerts	Links between the elementary attacks	Statistical analysis	Information-flow tracing, Alert correlation	Only tested two APT attacks

6.3 Detection Based on Information Flow and Data Audit

The APT attack can be detected by processing the information flow inside the target network environment. The collected data can be processed using game theory or flow graphs, which will help with security Professionals to understand the data flow.

Advanced Persistent Threats (APT) threaten critical industrial infrastructure by exploiting multiple zero-day vulnerabilities that possess the characteristics of burst, unknown, and cross-domain characteristics. The conventional wisdom typically establishes a security monitoring platform that connects remotely to the cloud-based threat intelligence center to thwart APT attacks. But in the real world, when few victim users are willing to share unfiltered attack samples out of privacy concerns, such a mentality is irrational. It makes it impossible to detect APT attacks rapidly without giving up more incentives.

In [8], the authors introduced the security solution for APT detection based on the honeypots. A honeypot is a security mechanism used to analyze and detect the attacker on the computer. The major limitation of the proposed work is that honeypots are passive. It will direct the attacker to the target without indicating security policies.

In [42], the authors provide a Deep Learning-based network forensics framework for digitally recognizing and tracing network threats while also offering a detailed overview of the network forensics procedure. The author used the capture network traffic and encryption to ensure the integrity and security of data. Following that, the feature filtering techniques were used to preserve critical traceability information, and the Deep Learning model was used with improved hyperparameter optimization approaches. Finally, a Multi-Layer Perceptual Deep Neural Network (MLP DNN) model with perceptual skills to detect strange events in the network. The author tested the framework's effectiveness with the UNSW-NB15 dataset. The results show that the suggested framework suits APT attack forensics scenarios. The proposed system excels at detecting and tracking network attack events compared to existing AI approaches.

The detection of traditional APT mainly targets a single step; the advantage is prominent in dealing with short spam attacks and in detecting long-term attacks. We need to keep track of different attack steps over a long period. In response to this problem, an efficient classification model for detecting APT was introduced [55]. This classification model uses the malware used by the APT attack for the evaluation. The proposed model undergoes four main phases. In the first one, they extract features like CPU usage, open ports, memory usage, and files in the system from the regular computer. Next, the malware code is injected into the regular system, and the feature extraction process is repeated. The third phase is the analysis phase—in this training, the machine learning algorithms with extracted features and finding the accurate ones. The last stage is the testing phase, which extracts the features from the target system and inputs these features to the proposed classification model. The model will raise an alert signal upon attack. The limitation of this work is that the rate or accuracy could be higher in fewer data samples and a long attack period.

In [110], the authors proposed a novel privacy-preserving few-shot traffic detection (PFTD) technique based on federated meta-learning (FML) to address this problem. PFTD approaches the APT detection task as an optimization process for model generalization, transferring acquired knowledge to identify unknown samples in the local area. In FML, client-side models transfer knowledge through two-phase updates over the query and support datasets, whereas the server-side model uses model aggregation to acquire global knowledge. These procedures gather relevant information to fend off APT attacks. Using a novel wisdom, we are able to achieve three benefits:

1. High personalization to cross-domain APT attacks.
2. Low latency detection for removing rules matching process.
3. High accuracy with a small number of attack samples.

Prolonged tests conducted on various benchmark datasets, such as CICIDS2017 and DAPT 2020, demonstrate the effectiveness of the suggested PFTD.

In [111], the authors address the fundamental ideas of Visual Analytics and puts forth a novel method for APT attack detection using anomaly and behavior-based analysis. The ultimate objective is to identify sophisticated cyber threats by combining their distinguishing traits to create a behavioral pattern that can be visually displayed for examination and analysis. This can be accomplished by combining highly detailed and dynamic visualization techniques with our Multi-Agent System for Advanced Persistent Threat Detection (MASFAD) framework.

The emergence of Advanced Persistent Threats (APTs) has made it more challenging to identify and understand computer system attacks. In [112], the authors proposed an Intrusion Detection System (IDS) that uses gradient-boosting algorithms and decision trees to efficiently detect APT activities at every stage of the APT life cycle. Furthermore, this model produces APT fingerprints by refining APT stages or attack paths that aid in early APT detection. Dataset APT (DAPT) 2020 is utilized to evaluate and validate this model. In most APT stages, the suggested model successfully classified APT activities with an accuracy greater than 97.63. Moreover, the model demonstrated efficacy in producing APT fingerprints.

The frequency of Advanced Persistent Threat (APT) attacks has increased due to the swift advancement of information technology. One area of intense concern for power system security research is the timely and accurate detection and response to these attacks. Currently used traceability graph-based detection techniques mainly rely on graph matching and label propagation algorithms. These techniques frequently call for manually created algorithmic rules and specific domain knowledge. However, as deep learning technology develops, less reliance on human intervention becomes more crucial.

In [113], the authors proposed an Intrusion Detection System (IDS) to effectively detect APT activities in each stage of the APT life cycle using decision trees and gradient-boosting algorithms. In addition, this model generates APT fingerprints by optimizing APT stages or attack paths that help the model with early APT detection. This model is evaluated and validated using Dataset APT (DAPT) 2020. The proposed model proved that effectively classified APT activities with more than 97.63 accuracy in most APT stages. Furthermore, this model proved effective in generating APT fingerprints.

In [114], they proposed methods to classify APT attacks by extracting the dynamic features. Extracted features are tested against the predestined prototypes for classification. The author applied the framework against two real-time attacks and yielded good results. However, there is no guarantee that all APT attacks have the same features.

In [115], the Generative Adversarial Network (GAN) and Long-Short-Term Memory (LSTM) were combined by the author to create a novel detection technique for APT attacks. Attack data and attack generation module determine this detection method. Using GAN simulation will generate many attack samples to optimize the discrimination model. The LSTM model ensures the timing and correlation in the APT attack sequence. Developing attack data using the generative model and optimizing the discriminant model improves the accuracy and decreases the false positive rate.

The APT malware can be present in the multiple layers of the network. So, continuous learning is needed to detect these types of attacks. The learning techniques, such as neural networks, perception, centroids, binary decision trees, deep learning. etc., must be used. In response to this problem, In [116], authors introduced a novel deep learning stack-based Approach for detecting APT. It is a theoretical approach. APT is considered a multi-stage and multimedia attack with a continuous strategic campaign. To detect these attacks, we must analyze entire network traffic data, especially raw data. We can catch certain behaviors and exceptions by combining different deep-learning methods.

In [117], a method based on machine learning and outlier detection was proposed. The proposed method uses input as event logs related to processes. The main focus of this work is detecting the attacks that require domain administrator privilege. Even in cases where compromised accounts are legitimate, abused processes can still be identified if they are not utilized in regular operations. The malicious process can be identified in an operational environment by using the outlier detection algorithm. The proposed detection method is efficient for mitigating the damage caused by APT attacks because it is only built-in Windows event logs and is easy to implement in the operational environment.

In [118], a theoretical framework was introduced to characterize information-based APT attacks. They proposed an entry model that studies how the attacker will select the optimal system in the network—and then introduced a dynamic model for internal reconnaissance, lateral movement, and strategic decision of targeted attack.

In [119], a new adaptive genetic algorithm was developed for optimizing the APT attack perdition. In this work, the authors used a BP neural network optimized by a genetic algorithm to predict the high-risk node in the network. Sequentially, the proposed algorithm traces the path of the APT attack.

In [120], a new intrusion detection system was introduced based on the patterns generated by active processes running on the system. The proposed system periodically collects information about every host's active processes within the network. The process tree will be constructed using the collected data, and the anomaly process will be detected depending on the parent-child relationship.

In [121], a new anomaly detection against APT attacks was proposed. The proposed detection system undergoes two stages: the first stage is the generation of behavior rules, and machine learning algorithms are used in this stage. The second stage is detecting abnormal behavior; MapReduce describes features based on large network data and compares these features with the behavior rules to determine if the host is abnormal.

Advanced persistent threats (APT) is a most sophisticated attack. Due to the attack process's dynamic nature, the traditional detection method fails to detect the APT. In [122], many approaches have been developed to monitor and analyze TCP/IP connections to extract the features that can be used by machine learning algorithms, focusing on reducing only false positives. However, current work has a limitation in reducing false positives and negatives. They proposed a new classification algorithm based on correlated fractal dimensions to reduce false positives and negatives. The algorithm highly accurately classifies the APT anomalous traffic using feature vectors obtained through the TCP/IP session information analysis.

In [123], they introduced a new framework for APT detection. The proposed framework has five modules: Network-traffic re-direction module, User-Agent, Reconstruction module, Dynamic analysis module, and Decision module. The network traffic modules collect the traffic generated at the various points of a network and redirect it to the reconstruction module. The user agent module gathers auxiliary information like mouse movements and process information. The redirected network traffic is reconstructed at the reconstruction module based on regular traffic generated at the host. The dynamic analysis module detects his malicious behavior based on the user agent's information and reconstruction module. The decision model makes the final decision according to pre-defined criteria.

In [124], authors considered a few characteristics (encrypted communication, long period, and imitating the expected behavior) for APT attack detection. The proposed method detects the most suspicious host that may be involved in APT attack activities using the ranking system. Therefore, the proposed approach is based on detecting abnormal behavior by network traffic analysis. After data analysis, ranks are assigned to suspicious nodes inside the network; due to this, expert analysis's workload also decreases. We can find the host possibly involved in data theft and other attack activities based on ranks. The proposed detection method will also work with encrypted communication. The proposed method requires flow collection and storage, Feature-Extraction, Feature-Normalization, and Spaciousness Ranking. The limitation of this approach is they ignored the attack attempts.

In [125], a comprehensive study of attack path modeling in the cloud environment was presented. The author characterized the attack paths based on the attack paths generated by the APT attacker for exploiting the network. Next, they calculated the conditional probability of attack-paths generated. Based on the value, they evaluated the likelihood of an APT attack in the generated attack path. The author used the optimization algorithm to find the shortest path from various sources and Bayesian networks to find the shortest APT attack path. However, detection is left as future work.

To detect a possible APT attack, In [126], authors proposed a system that analyses the information on the host side to detect the targeted attack. They used clustering techniques for grouping the systems according to their behavior concerning resources they have access to and requested (e.g., drive-by download, exploit kits). They correlate the information about the industry and location from which these systems operate to discover the attack. They implemented a novel system called SPuNge, which works in two phases: In the first phase, it will analyze the host traffic and identify malicious activities, and in the second phase, it will detect the system that performs a similar activity.

6.3.1 Evaluation of APT Detection Methods Based on Information Flow and Data Audit

Table 8 shows the summary of some of the existing detection techniques based on information flow and data audit along with their limitations. Regular monitoring and detection mechanisms and traditional detection and prevention methods are ineffective for detecting APT attacks due to their diversity, latency, and concealment. Therefore, new APT detection technology and techniques are needed. The proposed attack detection methods mainly focus on mainstream analysis and machine learning algorithms. However, machine learning methods require high hardware, so we need to plan how to deal with computation overhead. Generally, machine learning methods have three main modules: collecting the data, extracting features, and testing. Evaluating the APT detection methods requires data sets from real-time attack scenarios, and performance evaluation and comparison are much more complex than other computer science domains. The collected data cannot be used directly for the model evaluation. Therefore, data prepossessing and feature extraction are necessary. The extracted features of an APT detection solution do not need to be practiced in other solutions. Usually, the problem formalization influences this task, and determine the type of features that need to be extracted and selected.

For example, when processing the log data, the extracted features are unrelated to those extracted from malware behavior or network traffic. In [23], the authors listed some features associated with APT stages and related techniques. Social engineering-based detection techniques partially rely on social knowledge and publicly available data to identify early attacks. Attackers can use this information to trick others by making false claims and using various deceptive methods, making an attack much more likely. The information flow analysis-based detection technique looks for anomalies in the data flow. To circumvent detection, APT attackers frequently employ covert tactics to alter their network communications or blend in with regular network data streams. However, flow anomaly detection is limited by the abundance of data sources.

In [26], the authors describe the characteristics of APT traffic, the tools used for detecting network breaches, and the preventive measures against APT. As mentioned above, APT is a big challenge to the current intrusion detection prevention systems (IDPs) in real-time detection. Additionally, the APT attack was conducted over a long period. This makes the correlation of alerts a challenge to the current detection systems. In addition, if we miss one or more steps of an APT attack, it is easier to analyze the full APT attack. The existing correlation-based solutions need to improve in APT detection.

Technological growth with increased data, computational power, and the rapid development of innovative AI methods are rapid. According to the analysis [127], by 2025, 85% of enterprises and 70% of the population will be deployed to the cloud, and 12% of homes will be smart homes, creating a $100 million market by the year. Still, APT attacks are included, and with the help of AI technology, attackers will develop their attack tools and techniques. Then, detection and defense became more difficult.

Table 8: Comparison of data (Network tarffi, logs, etc.) audit based detection methods

Author	Algorithm	Method	Approach	Limitation
[116]	PCA, SVM, NB, DT, MLP	Early discovery of APT	Dataset pre-processing Dimensionality reduction Classifier	Uncomputed time complexity.
[118]	–	MATHEMATICAL MODEL	–	Not applicable to real-world data, not considered the data-exfiltration stage.
[120]	Parent-child relationship	Process-tree construction	Host-side network traffic analysis Active processes collection Process tree construction Anomaly process detection	Host-based solution.
[123]	-	Detectingmali-cious behaviors	Network traffic redirection User agent Reconstruction, Dynamic analysis	Suitable to detect passive attacks.
[124]	Ranking of host which is possibly involve in data exfiltration	Detection of attack signature	Detection of abnormal behaviours based on network traffic analysis	Attack attempts are ignores to simulate the normal behaviour, Generality is missing.
[128]	SVM	One-to-one network Intrusion detection for APT	Collection of network information, SVM parameters, Ant-colony algorithm Optimal network Intrusion detection	Not suitable for lingsequences, able to detect only certain type of behaviour anlamolies.

(Continued)

Table 8 (continued)

Author	Algorithm	Method	Approach	Limitation
[129]	Machine learning algorithms	detecting the attacks which need domain administrator privilege	–	Possibility of false detection, upon regular usage of valid tools and accounts by attacker.
[130]	K-NN and correlation fractal dimension	Fractal based anomaly detection using outlier detection	Combined Packet capture Feature vector extraction Noisy removal Classification	Malicious classification is very hard.
[131]	LR, GNB, DTM, RF, and LB	RDP-based LM detection	Pre-processing of dataset Defining metrics Apply ML algorithms Result comparision	In case of less number of data samples and long attack period, the rate or accuracy is low.

7 Countermeasures of APT for Prevention

Due to the advanced nature of APTs, only some solutions give adequate protection. The best practice is to use security countermeasures, resulting in multi-layered protection. Some existing solutions must be re-engineered to detect APT attacks, leading to additional research. For example, while genetic algorithms have proved the most suitable solution for malware detection, their applicability in a large dataset is the subject of further study.

Many organizations recognized the severity of APT attacks. The APT attacks have no rules. It reaches the goal using multiple attack vectors. The attackers have a specific target and use advanced tools to exploit the network's vulnerable holes in the best method possible. The APT attacker attackers will keep hitting the organization's systems silently until they eventually succeed. This is because people manage these networks, and we all know that employees are the most vulnerable point of the organization. Often, organizations cannot determine possible reasons for the attack after spending too much time on their security infrastructure.

We need to deal with them effectively since APT attacks are targeted and advanced, and we must follow a strategic and systematic approach. When an APT compromises an organization, it implies that the attack was very advanced. The likelihood of being discovered by attackers will be significantly decreased if you patch your network regularly and ensure your staff members are adequately trained. Specifically, we need to analyze the security solution inside the network and establish the countermeasures. The following are some countermeasures against the APT.

- Educating the employees about the need for security is the first step toward preventive measures. All the organization's employees should be educated to enhance the need for security.

- Restrict the employees to the organization domain to access the social engineering sites to prevent personal information sharing.
- Avoid revealing sensitive information during press releases and product catalogs.
- The APT attacker starts the attack by compromising the vulnerable point of the network. According to the study [44], most APT attacks are based on known exploits. Therefore, it is essential to apply patch dispatching as early as possible after the vulnerability report released report is released.
- Control the access rights of sensitive information and categorize the access rights according to the type of information.
- Security systems like IDPs, anti-virus, and anti-malware solutions must be installed and updated regularly.
- Ask the employees to use only authenticated software and tools.
- Update the current security solutions inside the network according to the APT attack behaviors. It is essential to manage the necessary control systems like ISO-2700, PIMS, and ISMS [132].
- Network separation helps prevent internet-based malware infections and data leakage.
- Develop and execute security policies to control the information that the employees could share.
- Firewalls on a network should be capable of identifying probes an attacker sends to look for vulnerabilities. Thus, if the firewall is equipped with a particular ruleset, it should perform a state-full inspection.
- Unused ports of the network must be closed. Only needed ports are kept open, as the intruder will try to enter through any open port.
- Implement the group policy security option, “additional restrictions for anonymous connections.”
- The best countermeasure against brute-force attacks is eliminating passwords and using a separate authentication form such as smart cards or biometrics.
- Restrict the employees to interactive logins and access to system programs that users do not require, such as cmd.exe, privilege use, auditing events such as account login, and system events.
- Do the regular back of all critical data, excluding the binaries, and perform a new installation from a trusted source.
- The network systems must be monitored at multiple points and levels.

Along with these countermeasures, authors [23] described the various monitoring, deception, and mitigation methods for APT defense.

8 Future Opportunities

Due to the dynamic nature of the APT attacks, cyber-security solutions are challenging, and defending against these advanced attacks is difficult. This section describes the various difficulties of protecting against APT attacks. The organization has a robust security system; the advanced attackers build their custom malware to bypass these security systems. APT attacks are long-period attacks with a combination of different techniques. To detect these types of attacks, we must identify and correlate individual events over a long time. Since APT attacks spend considerable time in each stage, some solutions exist for each attack, and only some still need to be explored. A spear-phishing attack is a commonly used method in the initial intrusion phase of the attack—the timely detection and removal of this type of email prevent the APT attack in the early stages. To detect file-less malware attacks, we need detection methods based on behavioural analysis techniques—detecting attacker movement inside the organization, which looks like legitimate system behaviour. Generally, hackers communicate with each other through forums called dark-web forums. These forums help hackers exchange their ideas, tools, and plans. Research in this area has the potential for a robust social influence [23].

Another critical challenge in the APT defense system is detecting data exfiltration in the organization with cloud services. It is simple to detect the data exfiltration in the organization without cloud services. A strong correlation is required to design a defense system in a cloud computing environment. The attack methods leave little pieces of evidence at each stage of an attack. For example,file-less malware is called an in-memory attack; these are not file-based but can trace the attack using these methods. Some APTs may go overlooked by one or more stages, so detection solutions proposed from the start to the end of the attack using advanced techniques are challenging.

9 Conclusion

This survey presented a comprehensive introduction to the APT and background knowledge about the APT attack process. We gathered information about how attackers and various techniques were used to conduct the APT. We analyzed the real-time APT attacks conducted to give a better understanding.

Also, we provided technical background on current API detection and mitigation approaches along with limitations and comparisons. Lastly, we highlight the shortcomings and restrictions found in the body of current research, clarifying the difficulties and suggesting areas for further investigation. We aim for this survey to be applicable to academia and industry to protect against the Internet community from APTs.

Acknowledgement: We would like to thank my research supervisor Sukhvinder Singh and my brother S. Chanti who has given their valuable inputs to publish my manuscript in the reputed journal.

Funding Statement: This study was not funded by anyone.

Author Contributions: The authors confirm their contribution to the paper as follows: design research architecture and analyze technological advancements: Singamaneni Krishnapriya, Sukhvinder Singh; data collection and analysis: Singamaneni Krishnapriya; enhance the scientific rigor and readability of the review: Sukhvinder Singh; draft manuscript preparation: Singamaneni Krishnapriya. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data and materials supporting the findings presented in this survey journal paper are readily available upon request.

Conflicts of Interest: The authors declare they have no conflicts of interest to report regarding the present study.

References

[1] G. Arka, "An overview article on 600% increase in cyber attack in 2021," *ResearchGate*, vol. 1, pp. 1–7, 2021. doi: 10.13140/RG.2.2.18205.52968.

[2] N. Nissim, A. Cohen, C. Glezer, and Y. Elovici, "Detection of malicious PDF files and directions for enhancements: A state-of-the art survey," *Comput. Secur.*, vol. 48, pp. 246–266, 2015. doi: 10.1016/j.cose.2014.10.014.

[3] D. Winder, "Persistent and evasive attacks uncovered," *Infosecurity*, vol. 8, no. 5, pp. 40–43, 2011. doi: 10.1016/S1754-4548(11)70069-9.

[4] J. Chen, C. Su, K. H. Yeh, and M. Yung, "Special issue on advanced persistent threat," *Futur. Gener. Comput. Syst.*, vol. 79, pp. 243–246, 2018. doi: 10.1016/j.future.2017.11.005.

[5] J. R. Johnson and E. A. Hogan, "A graph analytic metric for mitigating advanced persistent threat," in *IEEE ISI 2013–2013 IEEE Int. Conf. Intell. Secur. Inform. Big Data, Emergent Threat. Decis. Secur. Inform.*, Seattle, WA, USA, 2013, pp. 129–133. doi: 10.1109/ISI.2013.6578801.

[6] S. Singh, P. K. Sharma, S. Y. Moon, D. Moon, and J. H. Park, "A comprehensive study on APT attacks and countermeasures for future networks and communications: Challenges and solutions," *J. Supercomput.*, vol. 75, no. 8, pp. 4543–4574, 2019. doi: 10.1007/s11227-016-1850-4.

[7] W. Worrall, *The Biggest Hack in History-Operation Shady RAT.* Accessed: Jul. 17, 2024. [Online]. Available from: https://hacked.com/the-biggest-hack-in-history-operation-shady-rat/

[8] T. V. Roman Jasek Martin Kolarik, "APT detection system using honeypots," *Recent Adv. Autom. Control. Inf. Commun.*, vol. 2, pp. 23–29, 2013.

[9] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, pp. 384, 2019. doi: 10.1186/s42400-019-0038-7.

[10] A. Chidukwani, S. Zander, and P. Koutsakis, "A survey on the cyber security of small-to-medium businesses: Challenges, research focus and recommendations," *IEEE Access*, vol. 10, pp. 85701–85719, 2022. doi: 10.1109/ACCESS.2022.3197899.

[11] A. Waldman "Mandiant upgrades Sandworm to APT44 due to increasing threat," Accessed: Apr. 22, 2024. [Online]. Available: https://www.techtarget.com/searchsecurity/news/366581178/Mandiant-upgrades-Sandworm-to-APT44-due-to-increasing-threat

[12] P. Chen, L. Desmet, and C. Huygens, "A study on advanced persistent threats," in *Lecture Notes in Computer Science*, vol. 8735, pp. 63–72, 2014. doi: 10.1007/978-3-642-38709-8.

[13] C. Gan, J. Lin, D. W. Huang, Q. Zhu, and L. Tian, "Advanced persistent threats and their defense methods in industrial internet of things: A survey," *Mathematics*, vol. 11, no. 14, pp. 1–23, 2023. doi: 10.3390/math11143115.

[14] M. R. Murtaza, A. Siddiqi, M. A. Mugheri, and M. S. Kanwal Oad, "Advanced persistent threats defense techniques: A review," *Pakistan J. Comput. Inf. Syst.*, vol. 2, no. 2, pp. 53–65, 2017.

[15] P. L. Raogo Kabore, A. Kouassi, R. N'goran, O. Asseu, and Y. Kermarrec, "Review of anomaly detection systems in industrial control systems using deep feature learning approach," *Engineering*, vol. 13, no. 1, pp. 30–44, 2021. doi: 10.4236/eng.2021.131003.

[16] A. Al Mansur and T. Zaman, "User behavior analytics in advanced persistent threats: A comprehensive review of detection and mitigation strategies," in *ISAS 2023-7th Int. Symp. Innov. Approaches Smart Technol. Proc.*, 2023. doi: 10.1109/ISAS60782.2023.10391553.

[17] M. K. Daly, "The advanced persistent threat," in *23rd Large Install. Syst. Administartion Conf.*, Baltimore, MD, USA, 2009.

[18] S. Mönch and H. Roth, "Real-time APT detection technologies: A literature review," in *Proc. 2023 IEEE Int. Conf. Cyber Secur. Resilience, CSR 2023*, Venice, Italy, 2023, pp. 136–141. doi: 10.1109/CSR57506.2023.10224983.

[19] N. I. C. Mat, N. Jamil, Y. Yusoff, and M. L. M. Kiah, "A systematic literature review on advanced persistent threat behaviors and its detection strategy," *J. Cybersecur.*, vol. 10, no. 1, pp. 1–18, 2024. doi: 10.1093/cybsec/tyad023.

[20] PwC and BAE, "Operation cloud hopper," Accessed: Apr. 15, 2017. [Online]. Available: https://www.pwc.co.uk/cyber-security/pdf/pwc-uk-operation-cloud-hopper-report-april-2017.pdf

[21] Z. Tan, A. K. Marnerides, C. Anagnostopoulos, S. Puthiya, and J. Singer, "Affiliation not available advanced persistent threats based on supply chain vulnerabilities: Challenges, solutions and future directions," *TechRxiv*, vol. 11, pp. 15–31, Jan. 2024. doi: 10.36227/techrxiv.170594149.97651781/v1.

[22] Mandiant, "APT41 (Double Dragon): A Dual Espionage and Cyber Crime Operation," Accessed: Jul. 18, 2024. [Online]. Available: https://www.mandiant.com/resources/reports/apt41-double-dragon-dual-espionage-and-cyber-crime-operation

[23] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, “A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities,” *IEEE Commun. Surv. Tutorials.*, vol. 21, no. 2, pp. 1851–1877, 2019. doi: 10.1109/COMST.2019.2891891.

[24] A. Lemay, J. Calvet, F. Menet, and J. M. Fernandez, “Survey of publicly available reports on advanced persistent threat actors,” *Comput. Secur.*, vol. 72, pp. 26–59, 2018. doi: 10.1016/j.cose.2017.08.005.

[25] M. Tischer *et al.*, “Users really do plug in USB drives they find,” in *2016 IEEE Symp. Secur. Priv., SP 2016*, San Jose, CA, USA, 2016, pp. 306–319. doi: 10.1109/SP.2016.26.

[26] J. Vukalović and D. Delija, “Advanced persistent threats-detection and defense,” in *2015 38th Int. Conv. Inf. Commun. Technol. Electron. Microelectron., MIPRO 2015*, Opatija, Croatia, 2015, pp. 1324–1330. doi: 10.1109/MIPRO.2015.7160480.

[27] Q. Zhang, H. Li, and J. Hu, “A study on security framework against advanced persistent threat,” in *Proc. 2017 IEEE 7th Int. Conf. Electron. Inf. Emerg. Commun., ICEIEC 2017*, Macau, China, 2017, vol. 2, pp. 128–131. doi: 10.1109/ICEIEC.2017.8076527.

[28] S. Hussain, M. Bin Ahmad, and S. S. Uddin Ghouri, “Advance persistent threat—A systematic review of literature and meta-analysis of threat vectors,” *Adv Intell. Syst. Comput.*, vol. 1158, pp. 161–178, 2021. doi: 10.1007/978-981-15-4409-5.

[29] 5 Stages of an Advanced Persistent Threat Attack on Your Network. Accessed: Jun. 22, 2024, n.d.. [Online]. Available: https://www.whymeridian.com/blog/bid/399610/5-stages-of-an-advanced-persistent-threat-attack-on-your-network

[30] A. K. Sood and R. J. Enbody, “Targeted cyberattacks: A superset of advanced persistent threats,” *IEEE Secur. Priv.*, vol. 11, no. 1, pp. 54–61, 2013. doi: 10.1109/MSP.2012.90.

[31] Trend Micro, “Spear-phishing email: Most favored APT attack bait, trend mciro incorporated research paper,” *Res. Pap.*, vol. 1, pp. 1–8, 2012.

[32] F. Massacci and G. Di Tizio, “Are software updates useless against advanced persistent threats?” *Commun. ACM*, vol. 66, no. 1, pp. 31–33, 2022. doi: 10.1145/3571452.

[33] A. K. Sood, R. Enbody, A. K. Sood, and R. Enbody, *Chapter 2–Intelligence Gathering*, 1st ed. Elsevier Inc, Apr. 18, 2014.

[34] R. Benchea, C. Vatamanu, A. Maximciuc, and V. Lunacasu, “Bitdefender-APT28 under the scope: A journey into exfiltrating intelligence and government information,” 2015. Accessed: Apr. 01, 2024. [Online]. Available: https://www.bitdefender.com/blog/labs/apt28-under-the-scope-a-journey-into-exfiltrating-intelligence-and-government-information/

[35] J. Calvet, “Tidal Cyber-Sednit Espionage Group Attacking Air-Gapped Networks,” 2014. Accessed: May 19, 2024. [Online]. Available: https://app.tidalcyber.com/references/8673f7fc-5b23-432a-a2d8-700ece46bd0f

[36] A. Team, “The projectsauron APT,” 2016. Accessed: Apr. 01, 2024. [Online]. Available: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://media.kasperskycontenthub.com/wp-content/uploads/sites/43/2018/03/07190156/The-ProjectSauron-APT_Technical_Analysis_KL.pdf

[37] GReAT (Global Research & Analysis Team), “Turla renews its arsenal with Topinambour | Securelist,” Jul. 2019. Accessed: Jun. 07 2024. [Online]. Available: https://securelist.com/turla-renews-its-arsenal-with-topinambour/91687/

[38] J. McCarthy, Cybersecurity for Smart Inverters. doi: 10.6028/NIST.IR.8498.ipd.

[39] “National vulnerability database (NVD) | NIST,” Accessed: May 19, 2024. [Online]. Available: https://www.nist.gov/programs-projects/national-vulnerability-database-nvd

[40] Cybersecurity Ventures and Herjavec Group, “2019 official annual cybercrime report,” Feb. 2019. Accessed: May 19, 2024. [Online]. Available: https://library.cyentia.com/report/report_003357.html

[41] J. Gardiner, M. Cova, and S. Nagaraja, “Command & control: Understanding, denying and detecting,” 2014. Accessed: Apr. 01, 2024. [Online]. Available: http://arxiv.org/abs/1408.1136v1%5Cnpapers3://publication/uuid/4389EDB2-DA22-4672-8B1F-A0F60556CA73

[42] Y. Mei, W. Han, S. Li, and X. Wu, "A survey of advanced persistent threats attack and defense," in *Proc. 2021 IEEE 6th Int. Conf. Data Sci. Cyberspace, DSC 2021*, Shenzhen, China, 2021, pp. 608–613. doi: 10.1109/DSC53577.2021.00096.

[43] P. N. Bahrami, A. Dehghantanha, T. Dargahi, R. M. Parizi, K. K. R. Choo and H. H. S. Javadi, "Cyber kill chain-based taxonomy of advanced persistent threat actors: Analogy of tactics, techniques, and procedures," *J. Inf. Process. Syst.*, vol. 15, no. 4, pp. 865–889, 2019. doi: 10.3745/JIPS.03.0126.

[44] R. Coulter, J. Zhang, L. Pan, and Y. Xiang, "Domain adaptation for windows advanced persistent threat detection," *Comput. Secur.*, vol. 112, pp. 102496, Jan. 2022. doi: 10.1016/j.cose.2021.102496.

[45] Y. Wang, Q. Li, Z. Chen, P. Zhang, and G. Zhang, "A survey of exploitation techniques and defenses for program data attacks," *J. Netw. Comput. Appl.*, vol. 154, pp. 102534, 2020. doi: 10.1016/j.jnca.2020.102534.

[46] M. Ussath, D. Jaeger, F. Cheng, and C. Meinel, "Advanced persistent threats: Behind the scenes," in *2016 50th Annu. Conf. Inf. Syst. Sci., CISS 2016*, Princeton, NJ, USA, 2016, pp. 181–186. doi: 10.1109/CISS.2016.7460498.

[47] J. Al-Saraireh and A. Masarweh, "A novel approach for detecting advanced persistent threats," *Egypt Inform. J.*, vol. 23, no. 4, pp. 45–55, 2022. doi: 10.1016/j.eij.2022.06.005.

[48] M. Li, W. Huang, Y. Wang, W. Fan, and J. Li, "The study of APT attack stage model," in *2016 IEEE/ACIS 15th Int. Conf. Comput. Inf. Sci. ICIS 2016*, Okayama, Japan, 2016. doi: 10.1109/ICIS.2016.7550947.

[49] N. Virvilis and D. Gritzalis, "The big four—What we did wrong in advanced persistent threat detection?," in *2013 Int. Conf. Availability, Reliab. Secur., ARES 2013*, Regensburg, Germany, 2013, pp. 248–254. doi: 10.1109/ARES.2013.32.

[50] M. J. Assante and R. M. Lee, "The industrial control system cyber kill chain," 2015. Accessed: Apr. 01, 2024. [Online]. Available: www.lockheedmartin.com/content/dam/lockheed/data/corporate/documents/LM-White-Paper-Intel-Driven-Defense.pdf

[51] C. Atapour, I. Agrafiotis, and S. Creese, "Modeling advanced persistent threats to enhance anomaly detection techniques," *J. Wireless Mobile Netw., Ubiquitous Comput., Dependable Appl. (JoWUA)*, vol. 9, no. 4, pp. 71–102, 2018.

[52] K. Numada, S. Nozaki, T. Takaiwa, T. Ohki, and M. Nishigaki, "Perceiving human psychological consistency: Attack detection against advanced persistent social engineering," *Lecture Notes on Data Engineering and Communications Technologies*, vol. 193, pp. 152–162, 2024. doi: 10.1007/978-3-031-53555-0_15.

[53] H. M. Sun, C. E. Shen, and C. Y. Weng, "A flexible framework for malicious open XML document detection based on APT attacks," in *INFOCOM 2019-IEEE Conf. Comput. Commun. Work., INFOCOM WKSHPS 2019*, Paris, France, 2019, pp. 2005–2006. doi: 10.1109/INFCOMW.2019.8845281.

[54] I. Ghafir, V. Prenosil, M. Hammoudeh, F. J. Aparicio-Navarro, K. Rabie and A. Jabban, "Disguised executable files in spear-phishing emails: Detecting the point of entry in advanced persistent threat," in *Proc. 2nd Int. Conf. Future Netw. Distrib. Syst. (ICFNDS'18)*, New York, NY, USA, Association for Computing Machinery, 2018, vol. 44, pp. 1–5. doi: 10.1145/3231053.3231097.

[55] J. V. Chandra, N. Challa, and S. K. Pasupuleti, "A practical approach to E-mail spam filters to protect data from advanced persistent threat," in *Proc. IEEE Int. Conf. Circuit, Power Comput. Technol., ICCPCT 2016*, 2016, pp. 1–5. doi: 10.1109/ICCPCT.2016.7530239.

[56] I. Ghafir and V. Přenosil, "Advanced persistent threat and spear phishing emails," *Distance Learn. Simul. Commun.*, vol. 2015, pp. 34, 2015.

[57] N. Nissim *et al.*, "ALPD: Active learning framework for enhancing the detection of malicious PDF files," in *Proc. 2014 IEEE Jt. Intell. Secur. Inform. Conf., JISIC 2014*, The Hague, Netherlands, 2014, pp. 91–98. doi: 10.1109/JISIC.2014.23.

[58] B. Bencsáth, G. Pék, L. Buttyán, and M. Félegyházi, "Duqu: Analysis, detection, and lessons learned," in *ACM Eur. Workshop Syst. Secur. (EuroSec)*, 2012, vol. 2012.

[59] T. Schindler, "Anomaly detection in log data using graph databases and machine learning to defend advanced persistent threats," in *Lecture Notes in Informatics (LNI)*, vol. 275, pp. 2371–2378, 2017. doi: 10.18420/in2017_241.

[60] “DeepEnd Research: Library of malware traffic patterns,” Accessed: May 19, 2024. [Online]. Available: http://www.deependresearch.org/2013/04/library-of-malware-traffic-patterns.html

[61] X. Wang, K. Zheng, X. Niu, B. Wu, and C. Wu, “Detection of command and control in advanced persistent threat based on independent access,” in *2016 IEEE Int. Conf. Commun., ICC 2016*, Kuala Lumpur, Malaysia, 2016, pp. 1–6. doi: 10.1109/ICC.2016.7511197.

[62] D. X. Cho and H. H. Nam, “A method of monitoring and detecting APT attacks based on unknown domains,” *Procedia Comput. Sci.*, vol. 150, pp. 316–323, 2019. doi: 10.1016/j.procs.2019.02.058.

[63] R. Zhang, Y. Huo, J. Liu, and F. Weng, “Constructing APT attack scenarios based on intrusion kill chain and fuzzy clustering,” *Secur. Commun. Netw.*, vol. 2017, pp. 1–9, 2017. doi: 10.1155/2017/7536381.

[64] Y. Zhang, W. Liu, K. Kuok, and N. Cheong, “Anteater: Advanced persistent threat detection with program network traffic behavior,” *IEEE Access*, vol. 12, pp. 8536–8551, 2024. doi: 10.1109/ACCESS.2024.3349943.

[65] R. Zhang, W. Sun, J. Liu, J. Li, G. Lei and H. Guo, “Construction of two statistical anomaly features for small-sample APT attack traffic classification,” 2020. Accessed: Apr. 01, 2024. [Online]. Available: http://arxiv.org/abs/2010.13978

[66] G. Yan, Q. Li, D. Guo, and X. Meng, “Discovering suspicious APT behaviors by analyzing DNS activities,” *Sensors*, vol. 20, no. 3, pp. 1–17, 2020. doi: 10.3390/s20030731.

[67] Z. Ma and Q. Li, “A large-scale domain graph in information-centric IoT,” *IEEE Access*, vol. 7, pp. 13917–13926, 2019.

[68] Y. Shi, G. Chen, and J. Li, “Malicious domain name detection based on extreme machine learning,” *Neural Process. Lett.*, vol. 48, no. 3, pp. 1347–1357, 2018. doi: 10.1007/s11063-017-9666-7.

[69] T. Debatty, W. Mees, and T. Gilon, “Graph-based APT detection,” in *2018 Int. Conf. Mil. Commun. Inf. Syst., ICMCIS 2018*, Warsaw, Poland, 2018, pp. 1–8. doi: 10.1109/ICMCIS.2018.8398708.

[70] W. Niu, X. Zhang, G. Yang, J. Zhu, and Z. Ren, “Identifying APT malware domain based on mobile DNS logging,” *Math. Probl. Eng.*, vol. 2017, pp. 1–9, 2017. doi: 10.1155/2017/4916953.

[71] P. Lamprakis, R. Dargenio, D. Gugelmann, V. Lenders, M. Happe and L. Vanbever, “Unsupervised detection of APT C&C channels using web request graphs,” in *Lecture Notes in Computer Science*, vol. 10327, pp. 366–387, 2017. doi: 10.1007/978-3-319-60876-1_17.

[72] J. Lu, X. Zhang, J. F. Wang, and L. Y. Ying, “APT traffic detection based on time transform,” in *2016 Int. Conf. Intell. Transp. Big Data Smart City, ICITBS 2016*, Changsha, China, 2017, pp. 9–13. doi: 10.1109/ICITBS.2016.87.

[73] B. Rahbarinia, R. Perdisci, and M. Antonakakis, “Efficient and accurate behavior-based tracking of malware-control domains in large ISP networks,” *ACM Trans. Priv. Secur.*, vol. 19, no. 2, pp. 1–31, 2016. doi: 10.1145/2960409.

[74] G. Zhao, K. Xu, L. Xu, and B. Wu, “Detecting APT malware infections based on malicious DNS and traffic analysis,” *IEEE Access*, vol. 3, pp. 1132–1142, 2015. doi: 10.1109/ACCESS.2015.2458581.

[75] G. Wang, A. Zomaya, G. M. Perez, and K. Li, “Algorithms and architectures for parallel processing,” in *Lecture Notes in Computer Science*, vol. 9530, pp. 311–322, 2015. doi: 10.1007/978-3-319-27137-8.

[76] K. F. Hong, C. C. Chen, Y. T. Chiu, and K. Sen Chou, “Ctracer: Uncover C&C in advanced persistent threats based on scalable framework for enterprise log data,” in *2015 IEEE Int. Congr. Big Data, BigData Congr. 2015*, New York, NY, USA, 2015, pp. 551–558. doi: 10.1109/BigDataCongress.2015.86.

[77] S. Singh and S. K. V. Jayakumar, “A study on various attacks and detection methodologies in software defined networks,” *Wirel Pers. Commun.*, vol. 114, no. 1, pp. 675–697, 2020. doi: 10.1007/s11277-020-07387-y.

[78] A. A. A. Lah, R. A. Dziyauddin, and M. H. Azmi, “Proposed framework for network lateral movement detection based on user risk scoring in SIEM,” in *2018 2nd Int. Conf. Telemat. Futur. Gener. Netw., TAFGEN 2018*, Kuching, Malaysia, 2018, pp. 149–154. doi: 10.1109/TAFGEN.2018.8580484.

[79] B. Glithero, “Better threat detection and response with analytics for lateral movement,” Jan. 2017. Accessed: May 20. [Online]. Available: https://tanzu.vmware.com/content/blog/better-threat-detection-and-response-with-analytics-for-lateral-movement

[80] D. He, H. Gu, S. Zhu, S. Chan, and M. Guizani, "A comprehensive detection method for the lateral movement stage of APT attacks," *IEEE Internet Things J.*, vol. 11, no. 5, pp. 8440–8447, Mar. 2024. doi: 10.1109/JIOT.2023.3322412.

[81] J. Liu and J. Shi, "Leveraging token-based representation to detect lateral movement," in *2023 Asia-Pacific Conf. Image Process. Electron. Comput.*, Dalian, China, Apr. 2023, pp. 391–399. doi: 10.1109/IPEC57296.2023.00074.

[82] H. Bian, T. Bai, M. A. Salahuddin, N. Limam, A. A. Daya, and R. Boutaba, "Uncovering lateral movement using authentication logs," *IEEE Trans. Netw. Serv. Manage.*, vol. 18, no. 1, pp. 1049–1063, Mar. 2021. doi: 10.1109/TNSM.2021.3054356.

[83] Q. Liu *et al.*, "Latte: Large-scale lateral movement detection," in *IEEE Mil. Commun. Conf. MILCOM*, Los Angeles, CA, USA, 2019, pp. 952–959. doi: 10.1109/MILCOM.2018.8599748.

[84] T. Bai, H. Bian, A. A. Daya, M. A. Salahuddin, N. Limam and R. Boutaba, "A machine learning approach for RDP-based lateral movement detection," in *Conf. Local Comput. Networks, LCN*, Osnabrueck, Germany, 2019, pp. 242–245. doi: 10.1109/LCN44214.2019.8990853.

[85] A. Fawaz, A. Bohara, C. Cheh, and W. H. Sanders, "Lateral movement detection using distributed data fusion," in *Proc. IEEE Symp. Reliab. Distrib. Syst.*, Budapest, Hungary, 2016, pp. 21–30. doi: 10.1109/SRDS.2016.014.

[86] E. Goodman, J. Ingram, S. Martin, and D. Grunwald, "Using bipartite anomaly features for cyber security applications," in *2015 IEEE 14th Int. Conf. Mach. Learn. Appl., ICMLA 2015*, Miami, FL, USA, 2016, pp. 301–306. doi: 10.1109/ICMLA.2015.69.

[87] G. Ioannou, P. Louvieris, N. Clewley, and G. Powell, "A Markov multi-phase transferable belief model: An application for predicting data exfiltration APTs," in *Proc. 16th Int. Conf. Inf. Fusion, FUSION 2013*, Istanbul, Turkey, 2013, pp. 842–849.

[88] K. Saheed and S. Henna, "Deep reinforcement learning for advanced persistent threat detection in wireless networks," in *2023 31st Irish Conf. Artif. Intell. Cogn. Sci., AICS 2023*, Letterkenny, Ireland, 2023. doi: 10.1109/AICS60730.2023.10470498.

[89] A. Nadler, A. Aminov, and A. Shabtai, "Detection of malicious and low throughput data exfiltration over the DNS protocol," *Comput. Secur.*, vol. 80, pp. 36–53, 2019. doi: 10.1016/j.cose.2018.09.006.

[90] V. N. Harikrishnan and G. Kumar, "Advanced persistent threat analysis using splunk," *Int. J. Pure Appl. Math.*, vol. 118, no. 20, pp. 3761–3768, 2018.

[91] W. Niu, X. Zhang, G. Yang, R. Chen, and D. Wang, "Modeling attack process of advanced persistent threat using network evolution," *IEICE Trans. Inf. Syst.*, vol. 100, no. 10, pp. 2275–2286, 2017. doi: 10.1587/transinf.2016INP0007.

[92] C. -H. Hsieh, C. -M. Lai, C. -H. Mao, T. -C. Kao, and K. -C. Lee, "AD2: Anomaly detection on active directory log data for insider threat monitoring," in *2015 Int. Carnahan Conf. Secur. Technol. (ICCST)*, Taipei, Taiwan, 2015, pp. 287–292. doi: 10.1109/CCST.2015.7389698.

[93] Y. Kim, I. Lee, H. Kwon, K. Lee, and J. Yoon, "BAN: Predicting APT attack based on Bayesian network with MITRE ATT&CK framework," *IEEE Access*, vol. 11, pp. 91949–91968, 2023. doi: 10.1109/ACCESS.2023.3306593.

[94] S. D. Cakmakci, G. Gkoktsis, R. Buchta, K. O. Detken, F. Heine and C. Kleiner, "APT detection: An incremental correlation approach," in *Proc. IEEE Int. Conf. Intell. Data Acquis. Adv. Comput. Syst. Technol. Appl. IDAACS*, Dortmund, Germany, 2023, pp. 151–156. doi: 10.1109/IDAACS58523.2023.10348952.

[95] K. A. Akbar *et al.*, "Advanced persistent threat detection using data provenance and metric learning," *IEEE Trans. Dependable Secur. Comput.*, vol. 20, no. 5, pp. 3957–3969, Sep. 2023. doi: 10.1109/TDSC.2022.3221789.

[96] H. Yue, T. Li, D. Wu, R. Zhang, and Z. Yang, "Detecting APT attacks using an attack intent-driven and sequence-based learning approach," *Comput. Secur.*, vol. 140, pp. 103748, May 2024. doi: 10.1016/j.cose.2024.103748.

[97] M. Khosravi and B. T. Ladani, "Alerts correlation and causal analysis for APT based cyber attack detection," *IEEE Access*, vol. 8, pp. 162642–162656, 2020. doi: 10.1109/ACCESS.2020.3021499.

[98] H. Yue, T. Li, D. Wu, R. Zhang, and Z. Yang, "Detecting apt attacks using an extended sequence-based learning approach," *SSRN Electron. J.*, vol. 8, pp. 1–35, Oct. 2022. doi: 10.2139/ssrn.4238362.

[99] A. M. Lajevardi and M. Amini, "A semantic-based correlation approach for detecting hybrid and low-level APTs," *Futur Gener. Comput. Syst.*, vol. 96, pp. 64–88, 2019. doi: 10.1016/j.future.2019.01.056.

[100] I. Ghafir *et al.*, "Hidden Markov models and alert correlations for the prediction of advanced persistent threats," *IEEE Access*, vol. 7, pp. 99508–99520, 2019. doi: 10.1109/ACCESS.2019.2930200.

[101] A. Zimba, H. Chen, and Z. Wang, "Bayesian network based weighted APT attack paths modeling in cloud computing," *Futur Gener. Comput. Syst.*, vol. 96, pp. 525–537, 2019. doi: 10.1016/j.future.2019.02.045.

[102] X. Cheng, J. Zhang, and B. Chen, *Correlate the Advanced Persistent Threat Alerts and Logs for Cyber Situation Comprehension*. Singapore: Springer, 2019, vol. 1095. doi: 10.1007/978-981-15-0758-8_10.

[103] I. Ghafir *et al.*, "Detection of advanced persistent threat using machine-learning correlation analysis," *Futur. Gener. Comput. Syst.*, vol. 89, pp. 349–359, 2018. doi: 10.1016/j.future.2018.06.055.

[104] S. M. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. N. Venkatakrishnan, "HOLMES: Real-time APT detection through correlation of suspicious information flows," in *2019 IEEE Symp. Secur. Priv. (SP)*, San Francisco, CA, USA, 2019, pp. 1137–1152. doi: 10.1109/SP.2019.00026.

[105] S. Jia and Y. Xu, "The APT detection method in SDN," in *2017 3rd IEEE Int. Conf. Comput. Commun. (ICCC)*, Chengdu, China, 2017, pp. 1240–1245. doi: 10.1109/CompComm.2017.8322741.

[106] P. K. Sharma, S. Y. Moon, D. Moon, and J. H. Park, "DFA-AD: A distributed framework architecture for the detection of advanced persistent threats," *Cluster Comput.*, vol. 20, no. 1, pp. 597–609, 2017. doi: 10.1007/s10586-016-0716-0.

[107] J. Sexton, C. Storlie, and J. Neil, "Attack chain detection," *Stat Anal. Data Min*, vol. 8, no. 5–6, pp. 353–363, 2015. doi: 10.1002/sam.11296.

[108] I. Friedberg, F. Skopik, G. Settanni, and R. Fiedler, "Combating advanced persistent threats: From network event correlation to incident detection," *Comput. Secur.*, vol. 48, pp. 35–57, 2015. doi: 10.1016/j.cose.2014.09.006.

[109] P. Giura and W. Wang, "A context-based detection framework for advanced persistent threats," in *2012 Int. Conf. Cyber Secur.*, Alexandria, VA, USA, 2012, pp. 69–74. doi: 10.1109/CyberSecurity.2012.16.

[110] G. Brogi and V. V. T. Tong, "TerminAPTor: Highlighting advanced persistent threats through information flow tracking," in *2016 8th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Larnaca, Cyprus, 2016, pp. 1–5. doi: 10.1109/NTMS.2016.7792480.

[111] G. Nikolov and W. Mees, "Detection of previously unknown advanced persistent threats through visual analytics with the MASFAD framework," in *2023 Int. Conf. Military Commun. Inform. Syst. (ICMCIS)*, Skopje, North Macedonia, 2023, pp. 1–10. doi: 10.1109/ICMCIS59922.2023.10253465.

[112] S. Y. Yi, M. M. Singh, G. C. Sodhy, and T. Jabar, "Fingerprinting generation for advanced persistent threats (APT) detection using machine learning techniques," in *2023 13th Int. Conf. Inform. Technol. Asia (CITA)*, Kota Samarahan, Malaysia, 2023, pp. 31–36. doi: 10.1109/CITA58204.2023.10262639.

[113] Y. Hu, J. Wu, G. Li, J. Li, and J. Cheng, "Privacy-preserving few-shot traffic detection against advanced persistent threats via federated meta learning," *IEEE Trans. Netw. Sci. Eng.*, vol. 11, no. 3, pp. 2549–2560, 2023. doi: 10.1109/TNSE.2023.3304556.

[114] H. Bao, W. Wang, and F. Liu, "Towards open-set APT malware classification under few-shot setting," in *GLOBECOM 2023-2023 IEEE Glob. Commun. Conf.*, Kuala Lumpur, Malaysia, 2023, pp. 6844–6849. doi: 10.1109/GLOBECOM54140.2023.10437265.

[115] W. Ren *et al.*, "APT attack detection based on graph convolutional neural networks," *Int. J. Comput. Intell. Syst.*, vol. 16, no. 1, pp. 1–14, Dec. 2023. doi: 10.1007/s44196-023-00369-5.

[116] T. Bodström and T. Hämäläinen, "A novel deep learning stack for APT detection," *Appl. Sci.*, vol. 9, no. 6, pp. 1055, 2019. doi: 10.3390/app9061055.

[117] C. Do Xuan, "Detecting APT attacks based on network traffic using machine learning," *J. Web Eng.*, vol. 20, no. 1, pp. 171–190, 2021. doi: 10.13052/jwe1540-9589.2019.

[118] D. Yan, F. Liu, and K. Jia, "Modeling an information-based advanced persistent threat attack on the internal network," in *2019 IEEE Int. Conf. Commun. (ICC)*, Shanghai, China, 2019, pp. 1–7. doi: 10.1109/ICC.2019.8761077.

[119] T. Fu, Y. Lu, and W. Zhen, "APT attack situation assessment model based on optimized BP neural network," in *2019 IEEE 3rd Inform. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, Chengdu, China, 2019, pp. 2108–2111. doi: 10.1109/ITNEC.2019.8729178.

[120] Y. Tsuda, J. Nakazato, Y. Takagi, D. Inoue, K. Nakao and K. Terada, "A lightweight host-based intrusion detection based on process generation patterns," in *2018 13th Asia Joint Conf. Inform. Secur. (AsiaJCIS)*, Guilin, China, 2018, pp. 102–108. doi: 10.1109/AsiaJCIS.2018.00025.

[121] M. Lee, J. Choi, C. Choi, and P. Kim, "APT attack behavior pattern mining using the FP-growth algorithm," in *2017 14th IEEE Annu. Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, 2017, pp. 1–4. doi: 10.1109/CCNC.2017.8013435.

[122] Y. Ahmed, A. T. Asyhari, and M. A. Rahman, "A cyber kill chain approach for detecting advanced persistent threats," *Comput. Mater. Contin.*, vol. 67, no. 2, pp. 2497–2513, 2021. doi: 10.32604/cmc.2021.014223.

[123] Y. Su, M. Li, C. Tang, and R. Shen, "A framework of APT detection based on dynamic analysis," in *Proc. 2015 4th Natl. Conf. Electric., Electron. Comput. Eng.*, Xi'an, China, 2016, pp. 1047–1053. doi: 10.2991/nceece-15.2016.187.

[124] M. Marchetti, F. Pierazzi, M. Colajanni, and A. Guido, "Analysis of high volumes of network traffic for advanced persistent threat detection," *Comput. Netw.*, vol. 109, pp. 127–141, 2016. doi: 10.1016/j.comnet.2016.05.018.

[125] W. Zhao, P. Wang, and F. Zhang, "Extended petri net-based advanced persistent threat analysis model," in *Lecture Notes in Electrical Engineering*, vol. 277, pp. 1297–1305, 2014. doi: 10.1007/978-3-319-01766-2.

[126] M. Balduzzi, V. Ciangaglini, and R. McArdle, "Targeted attacks detection with SPuNge," in *2013 11th Annu. Conf. Privacy, Secur. Trust., PST 2013*, 2013, pp. 185–194. doi: 10.1109/PST.2013.6596053.

[127] D. Soldani, "5G and the future of security in ICT," in *2019 29th Int. Telecommun. Netw. Appl. Conf. (ITNAC)*, Auckland, New Zealand, 2019, pp. 1–8. doi: 10.1109/ITNAC46935.2019.9078011.

[128] H. B. Liu, T. B. Wu, J. Shen, and C. T. Shi, "Advanced persistent threat detection based on generative adversarial networks and long short-term memory," (in Chinese), *Comput. Sci.*, vol. 47, no. 1, pp. 281–286, 2020. doi: 10.11896/JSJKX.181102103.

[129] W. Matsuda, M. Fujimoto, and T. Mitsunaga, "Detecting APT attacks against active directory using machine leaning," in *2018 IEEE Conf. Appl., Inform. Netw. Secur. (AINS)*, Langkawi, Malaysia, 2018, pp. 60–65. doi: 10.1109/AINS.2018.8631486.

[130] S. Siddiqui, M. S. Khan, K. Ferens, and W. Kinsner, "Detecting advanced persistent threats using fractal dimension based machine learning classification," in *Proc. 2016 ACM Int. Work. Secur. Priv. Anal. Co-Located with CODASPY 2016*, New Orleans, LA, USA, 2016, pp. 64–69. doi: 10.1145/2875475.2875484.

[131] S. Chandran, P. Hrudya, and P. Poornachandran, "An efficient classification model for detecting advanced persistent threat," in *2015 Int. Conf. Adv. Comput., Commun. Inform. (ICACCI)*, Kochi, India, 2015, pp. 2001–2009. doi: 10.1109/ICACCI.2015.7275911.

[132] M. Edwards "What is ISO 27001, The information security (ISMS) standard," Accessed: May 20, 2024. [Online]. Available: https://www.isms.online/iso-27001/