Jordanian Journal of Computers and Information Technology (JJCIT), Vol. 09, No. 01, March 2023.

# AN EARLY DETECTION MODEL FOR KERBEROASTING ATTACKS AND DATASET LABELING

Remah Younisse[1], Mouhammd Alkasassbeh[1], Mohammad Almseidin[2] and Hamza Abdi[1]

*(Received: 25-Aug.-2022, Revised: 28-Oct.-2022, Accepted: 22-Nov.-2022)*


## *ABSTRACT*

*The wild nature of humans has become civilized and the weapons they use to attack each other are now digitized. Security over the Internet usually takes a defensive shape, aiming to fight against attacks created for malicious reasons. Invaders' actions over the Internet can take patterns by going through specific steps every time they attack. These patterns can be used to predict, mitigate and stop these attacks. This study proposes a method to label datasets related to multi-stage attacks according to attack stages rather than the attack type. These datasets can be used later in machine-learning models to build intelligent defensive models. On the other hand, we propose a method to predict and early kill attacks in an active directory environment, such as kerberoasting attacks. In this study, we have collected data related to a suggested kerberoasting attack scenario in pcap files. Every pcap file contains data related to a particular stage of the attack life-cycle and the extracted information from the pcap files was used to highlight the features and specific activities during every step. The information was used to draw an efficient defensive plan against the attack. Here, we propose a methodology to draw equivalent defensive plans for other similar attacks as the kerberoasting attack covered in this study.*



## *KEYWORDS*

*Kerberoasting attacks, Attack life-cycle, Dataset labeling, Early detection.*


## 1. INTRODUCTION

TData is the modern fuel to train and calibrate machine-learning (ML) and artificial-intelligence (AI) models. Generating the proper data for ML and AI models built for complicated problems requiring high computational powers is not easy. Labeling this data so that it would give the best results is also not easy.

Building intelligent intrusion-detection systems deploying ML and AI models to detect possible threats over computer networks is no longer a luxury. AI intrusion-detection systems ought to detect possible threats early, so that defense plans can occur early and a more secure network is gained.

Intrusions and attacks over the Internet can take many forms and patterns, each with specific distinguishing techniques and goals. Patterns were found in each attack, making it possible to model and categorize the stages and steps attacks can go through. Many models in this context were proposed, such as the MITRE attack model and the Lockheed Martin model. These models aim to put attack stages in a chain of steps, so that defenders can know at which stage the attack is and build defense policies based on the attack's detected stage. Knowing the attack stage can help kill the attack at this stage or mitigate the attack consequences.

ML and AI models should be built to advance the operation of intrusion-detection systems [1]. These models require a properly labeled dataset; the more significant the dataset, the better the model performance. The literature has a shallow amount of datasets in this context. Therefore, this study proposes a method to create and label datasets related to multi-stage attack life-cycle. We also show how the generated data can be used statistically to detect attacks in an active directory server environment. The detection method is built on the awareness of the attack life-cycle.

An AI intrusion-detection system that is aware of the attack stages, not only the attack occurrence, can be more helpful in providing the proper knowledge which a defender needs to act accordingly. This intelligent intrusion-detection system requires that we collect and label the proper datasets for training. A good defense plan can also be used and built based on the attack stages.

1. R. Younisse, M Alkasassbeh and H. Abdi are with Department of Computer Science, Princess Summaya University for Technology, Emails: r.baniyounisse@psut.edu.jo, m.alkasassbeh@psut.edu.jo and h.abdi@psut.edu.jo
2. M. Almseidin is with Department of Computer Science, Tafila Technical University, Email: alsaudi@ttu.edu.jo

Active directory (AD) is a centralized management system for Windows computers and accounts. Hence, attacking (AD) servers can take many strategies, all aiming to take access to other computers and accounts linked to the server [2]. Kerberos authentication is widely used in the AD environment, where the domain controller processes all authentications using authentication tickets known as TicketGranting Tickets (TGTs) and Service Tickets (STs).

The work presented in [3] has detailed the components, the operation and the attacks in an AD environment. The components of Kerberos were summarized into five components. The five components are: the transport layer, agents, encryption keys, tickets and privileged attribute service. The transport layer component uses port 88 for data exchange between agents *via* the TCP or UDP transport-layer. The agents are either client machines, service servers or Key Distribution Centers (KDCs). The KDC is a central server that works to authenticate users and distribute tickets between them, identifying users against services. Encryption keys and tickets can take many forms, each for a particular purpose during the user's authentication process and privileges granted.

A privileged Attribute Certificate (PAC) is an extension of Kerberos tickets that includes valuable information about the privileges of users. According to [3], the attacks in an AD environment can be classified into nine different forms. Every attack was detailed with mitigation and/or prevention techniques. The attack which we are interested in through this study is the "Kerberoasting" attack. This attack was defined as focusing on capturing a service ticket (TGS) from memory, then decrypting the hash of the offline service credential using a password-cracking tool. Kerberoasting takes advantage of service account delivery during the Kerberos authentication process. Users can request a service ticket from a Domain Controller (DC). The DC is the key distribution center (KDC) in the AD utilizing Kerberos. When clients request service tickets for given services from a DC, they use unique identifiers called Principal Service Names (PSNs). To get the Kerberos authentication, SPNs are required to be registered in AD with at least one service logon account [4].

Kerberos is the default and most widely used authentication service in Windows machines. Despite its strength in securely covering authentication and data communication over a non-secure channel, Kerberos has some weaknesses. It is a stateless protocol as TGS and AS servers do not have a memory of previously granted tickets. Password-guessing attacks can provide valuable information for attackers targeting a system that uses Kerberos for authentication. Another weakness is that all the device clocks trying to get authentication from the AS should be synchronized. Furthermore, finally, if an attacker gets access to the "KRBTGT" account, the attacker can take full access to the domain [3].

The Kerbroasting attack can be made in simple steps. First, the attacker should get a valid SPN name. Then, it requests a service ticket and part of the service ticket is encrypted with the target service account's password hash. Weak passwords can easily be de-hashed by the attacker and hence the account can be cracked.

In this study, we focus on the details of creating a "Kerberoasting" attack; during this step, a labeled dataset for the attack was generated. We also aim to study the attack's life-cycle and label the generated dataset based on the attack stages. A methodology to detect the attack at each stage was proposed and detailed.

A cyber kill chain presents a model for understanding the life-cycle of a cyber-attack and helps improve cybersecurity policies [5]. Lockheed Martin has developed a cyber-kill chain model which describes cyber-attack steps. The kill chain has distinct steps describing each cyberattack life-cycle stage. The kill-chain model helps understand how attacks are performed and helps set plans to deal with attacks, such as ransomware and security breaches. An attack is known to follow an ordered sequence of techniques; using the Lockheed model, these steps are reconnaissance, weaponization, delivery, exploitation, installation, command, control and actions [6]. During the reconnaissance stage, the attacker finds its victim and a proper entrance for the victim's systems. Based on the information collected during the first step, the weaponization step comes as a preparation step for the attack using applicable codes and tools. Delivery is the step when a malicious code or tool is planted in the victim's machine. The exploitation step follows, during which the weapon exploits vulnerabilities in the victim's machine, so that a backdoor channel is activated and access to the victim's machine is guaranteed. This is done during the installation step. The last two steps can be considered a stage when the catastrophe and the attacked machine have fallen.

This model was built so that cyber-attack stages can be mapped to each stage in the model. Defense policies are then built based on at which stage the attack is happening [7]-[8]. It is good to mention the MITRE attack [9]. It is a framework built to collect data and information about different cyber-attacks. It also suggests mitigation and protection techniques for the listed attacks.

The contribution of this study is the presentation of the details of creating a "Kerberoasting" attack. The attack traffic details were collected in pcap files using Wireshark and then processed and analyzed. The pcap files were used with CICFlowmeter to generate a labeled dataset for the attack. What distinguishes the generated dataset is that it is labeled according to the attack's stages rather than the attack's type. The attack life-cycle stages were then thoroughly studied and revised to extract the distinguishing features of each stage, so that the attack could be exposed at any stage. A methodology to detect the attack at each stage was proposed and detailed. The attractive aspect of the followed methodology in this study is that it can be mimicked through other studies with similar life-cycle stages to build efficient defensive plans.

This paper is organized as follows: In Section 2, we present some of the valuable works related to the work presented in this paper. In Section 3, the methodology followed to generate the attack, collect and label the dataset is presented. A method to detect the "Kerberoasting" attack is suggested and discussed in sub-section 3.3. Finally, in Section 4, the conclusion of the work is introduced.

## 2. RELATED WORK

In literature, many models were proposed to build defensive plans against specific attacks. Cyber-attacks can have different types and targets; hence, a general defensive plan which covers all of them is not practical, but taking them in separate cases would make the task more doable.

For example, the work presented in [10] covers the ransomware threats. The work uses Lockheed Martin's cyber kill-chain model as a road map, then every step during the study covers a particular stage of the ransomware attack life-cycle. Unlike other studies covering detection, prevention and mitigation plans against ransomware attacks in literature, the plan presented in [10] considers that the attack is not a single stage of action, but goes through different actions and stages and accordingly, their plan was built. The work presented here studies "Kerberoasting" attack with the same attitude as well.

Attacks in multimedia service environments and the existing cyber kill-chain models were analyzed in [11]. The study efficiently set proper prevention plans for these attacks and used the Lockheed Martin's model to build a defensive policy against internal and external attacks in multi-media service environments. Attacks in an active-directory environment have recently taken considerable attention due to their wide range of collateral damages in the AD environment; i.e., the works in [12][13][14].

The work presented in [2] collected a dataset from event logs of the domain controller, where clients' computers were recorded and divided into many categories. This division was carried out based on the nature of the event; the dataset was not labelled, as labeling the data for intrusion-detection systems is not an easy task. Hence, an unsupervised learning AI model was built for intrusion detection in an AD environment.

Attacks in an AD environment were also investigated and a machine-learning model for Kerberoasting-attack detection was built. The dataset used in this work is non-synthetic, collected from an AD environment of an entire organization with hundreds of daily users. The data was collected from "event 4679", which is the event generated whenever a key distribution center gets a Kerberos Ticket Granting Service (TGS) request only; so the dataset was not rich in size or features. SVM machine-learning models which used the collected dataset were built later for intrusion detection in the AD environment [4]. Table 1 includes a summary of related literature.

Studying the attack life-cycles has brought forth authoritative results in many cases. The attacks on IoT networks have become a significant concern in cybersecurity in the past few years. The study of [15] has shown that such attacks, for example, take a spiral rather than a sequential form. This information can help build detection and defense strategies against the attack. The study proves that a deep sight inside an attack life-cycle can take the defensive party into a better stronger place.

Attack life-cycle was taken into consideration in [16], but not Kerberos nor an AD attack; yet, the

methodology used in the study was inspiring, since an attack model was built based on the disclosed APT attack cases. The model built in [16] gives an overview of APT attack life-cycle and attack techniques and investigates the distinguishing features of the collected data at each stage. The art of telling an attack life-cycle through data collected in pacp files was mentioned in [17], which is also an inspiration that we used in this study. An enormous amount of data was collected in pcap files, then cleaned and looked for attack signatures in the network traffic that map directly to particular phases of the life-cycle. It was focused on specific parts of the life-cycle when searching for attack signatures.

Labeling the dataset based on the attack life-cycle rather than the attack type is a recent area of interest; it was mentioned in [6], generating two multi-stage attacks; a password-cracking and a DDoS attack; two labeled datasets were also created for the attacks. These datasets are expected to be helpful in building powerful intrusion-detection models.

In cyber forensics, the early knowledge of the attacks can help detect and investigate early. The attack life-cycle model of attacks provides clear directions and orientations about the case and about the next steps to be followed during the investigation [18]. According to [18], pcap files generated to record the packets traffic on a network can be of great benefit during digital forensics.

[19] proposed detection scenarios for monitoring the network for a the potential occurrence of a "Kerberoasting" attack. The methodologies were built based on log data and statistical methods without mentioning the attack life-cycle or the dataset used through the analysis. Not to mention some other useful tools which can digest statistics from pcap files, such as CIC flowmeter. These tools can generate a dataset that machine-learning models can use to train and predict intrusions. [20] also greatly analyzed the content of pcap files to detect attacks using DNS tunneling.

The main contribution of this work is to create a dataset for "Kerberoasting" attacks that is aware of the attack life-cycle. The dataset features' values vary from one stage to another during the attack life-cycle. Hence, training AI models with a dataset labeled according to the stage of the attack can give more detailed results, as it can provide conclusions of at what stage the attack is. Knowing the attack stage would improve the followed defense strategies based on the attack life-cycle. The work also presents the methodology followed to collect and label the dataset; hence, it can be followed to create similar datasets for "Kerberoasting" attacks and other multi-stage attacks to improve the detection models built to detect these attacks. These datasets would enhance the quality of the decisions taken and the defense plans against the attacks.

Table 1. Related work summery.

| Ref. | Used AI? | Attack | Contribution | Limitations |
|---|---|---|---|---|
| [4] | Yes, SVM models | Kerberoasting attack | Kerberoasting attack detection using data collected from Event log 4769. | The technique misses detecting some malicious events on the cost of decreasing false positive alarms. |
| [2] | Yes, unsupervised leaning was used | APT attacks against AD | The attack activity data recorded in the Event logs was used; a new method based on machine learning for detecting attacks was proposed. | False negative can occur if legitimate administrators use CLI tools regularly. |
| [16] | No | APT attacks | The attack life-cycle was modeled and explained. | - |
| [17] | No | Attacks in general were studied. | A method to study the attack life cycle through the use of pcap files was proposed | The dataset used was huge. |
| [19] | No | Kerberoasting attack | A method to study the attack life cycle of Kerberoasting attack through the use of logged data was proposed. | False positive depends on multiple factors. |
| [15] | Yes | Botnet attacks | A model representing the spiral life-cycle of the attack was used. | The high complexity of the proposed method. |

## 3. DATA COLLECTION AND ANALYSIS

In this section, we discuss how an attack on an active-directory server was generated. We also show

how the data was collected and labeled. The attack life-cycle is also discussed and detailed according to the Lockheed Martin's model attack life-cycle. We specifically discuss the famous "Kerberoasting" attack.

### 3.1 Attack Generation

The life-cycle of attacks in an active-directory AD environment was discussed in Mitre attack [9] and detailed in [21], where "Kerberoasting" is listed as a sub-technique of steal or forge Kerberos Tickets technique and falls under credential-access tactics. The attack which we have generated in this study was discussed and detailed in [22]. We used our prior knowledge about the attack to create it and collect the traffic data in pcap files to get a deeper understanding of the attack stages and identifying the features at each stage, which will help in diagnosing the attack at early or even late stages.

The attack stages are listed in Figure 1, along with the used tool being listed inside a cloud shape beside the stage; the attack starts by scanning open ports in the AD server. This step aims to investigate running services on the server and is used in many other attacks, detecting that a port-scanning process on a server can be considered an early alarm for potential attacks. In this step, we detected the Kerberos authentication service on port 88; from an attacker point of view, to perform an attack on Kerberos server, we need to know that port 88 at destination side is active and reachable. In the next step we performed another scanning process focusing on ports 139 and 445; this step aims to investigate publicly shared files and is considered a reconnaissance stage also. However, through this study, the second step gave no valuable conclusions.

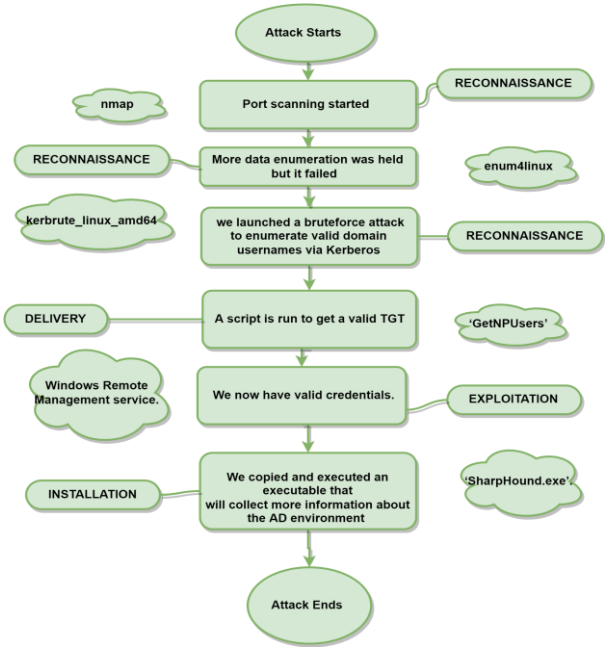


Figure 1. Flowchart of attack stages and tools.

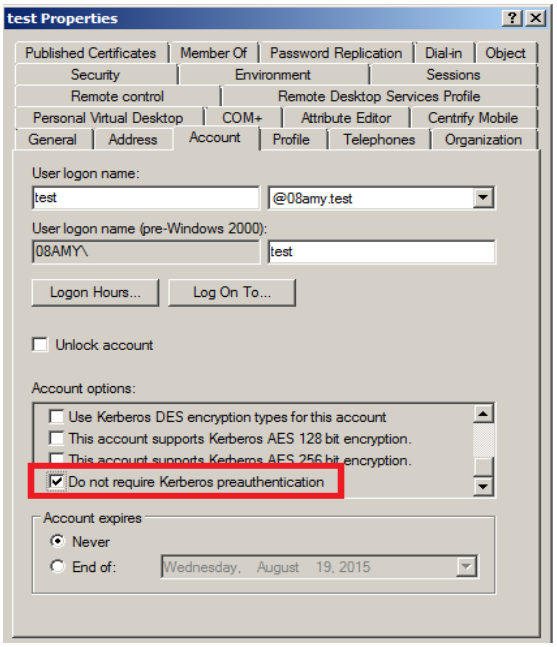


Figure 2. Attack stages and tools.

Next in step 3, knowing that Kerberos is running on the device from, we initiated a brute-force attack on the server using a publicly available dictionary in [23]. The dictionary contains around 100000 famous user names for Windows users; we grabbed a dictionary from the Internet and used it at this step. In step three, the brute-force attack aims to find legitimate user names registered on the Kerberos KD server. One of the valid user names was used in the next step to grant a valid ticket. At step 3 of the attack, we are still in the reconnaissance stage. We target user names whose owners have the property "Do not require Kerberos pre-authentication set (UF_DONT_REQUIRE_PREAUTH)"; this property can be activated by checking the option in the provided test-setting options as shown in Figure 2. These users have configured the Kerberos to consider that whenever they request a "TGT," the requester is the user name owner. Hence, when we requested a password to log into the server, the server responded with a hashed password which we dehashed offline and used to log into this account.

In the last step, we ran an executable file that collected more data about the AD environment. The

software we ran in this step is Sharphound.exe, which a data collector used to collect data from domain controllers and domain-joined Windows systems. The collected data can be information about security-group memberships or domain trusts, abusable rights on active directory objects or group policy links and some other information. The collected information at this step is usually used to facilitate the process of initiating more attacks and take control over the attacked systems.

In Table 2, we list every step with the software tools we used at each step. Also, in Table 2, we map every stage of the attack into a particular stage of the Lockheed Martin's attack stages. We have performed this attack to generate data about the "Kerberoasting" attack, representing an example of a multi-stage attack. We will explain in the following sub-sections how the collected data was used to create a labeled dataset with the label specifying the attack stage. The collected data was also analyzed to investigate for information that can be useful in intrusion detection and prevention strategies.

The attack in the AD environment is divided into stages, each representing a stage in the attack life-cycle model built by Lockheed Martin. Lockheed Martin's model was used in this study due to its simplicity and a direct reflection of multiple stages through the attack life-cycle. The stages and the mapping of Lockheed Martin's model are presented in Table 3.

Table 2. The tools used at each stage of the attack and mapping of Lockheed Martin's model.

| Step | Tool | Attack Stage |
|---|---|---|
| 1 | nmap | Reconnaissance |
| 2 | enum4linux | Reconnaissance |
| 3 | kerbrute | Reconnaissance |
| 4 | GetNPUsers | Delivery and then Exploitation |
| 5 | winrm and SharpBlood | Installation |

In this study, we have not only generated the attack, but also taken care of the attack stages. We knew that every stage of recorded traffic would carry different features and statistics. Hence, the collected data was transformed into CSV files using CIC Flowmeter software while we labeled the data according to the attack stage, as will be detailed in sub-section 3.2. The data exported into the CSV consists of 40 features plus the label column, representing the attack's stage. The listed features represent statistical information about the forward and backward traffic, timestamp and source and destination IP and source ports.

## 3.2 Data Collection and Labeling

In this sub-section, the methodology followed to collect the dataset is presented. The dataset was created with awareness of the stages the "Kerberoasting" attack will go through. Hence, the related network traffic for each stage was generated in an isolated pcap file using Wireshark. Then, the pcap files were fed into CIC flowmeter to generate the statistics or the features for every stage in a separate CSV file. The data in the CSV files was labeled according to the attack stage. Labeling the dataset according to the attack stage rather than labeling the dataset according to normal or anomaly packets makes the dataset more informative and aware of the stage at which the attack is. The analysis for each stage of the attack can be made with this dataset; it also can be used with AI and ML models which are more precise and specific about the severity of the attack. The presented method is not limited to "Kerberoasting" attack; it can be applied with other attacks to generate different datasets labeled according to the attack life-cycle.

To collect the data, an attack scenario was performed on a server running Kerberos on a virtual machine. In contrast, the traffic on the attacked virtual machine was collected and recorded in pcap files using Wireshark. The information saved in the pcap files was then explored using Wireshark. We also exported the data into CSV files, so that statistical analysis can be carried out on the data to extract a deeper insight into each stage of the attack. CIC flowmeter was used to record statistical information about the packets' flow during the attack stages.

We created the targeted attack from scratch; prior understanding of the attack was beneficial to understand the attack stages and record the network traffic during every stage. We created an actual crime scene, knowing that the crime scene is divided into multiple stages, each with certain attacker behaviors and actions. The attack scenario starts when the attacker uses a tool called nmap to identify

open ports and services. Based on the results returned by nmap, the attacker can notice that there are multiple ports open. The attacker then uses a tool called enum4linux to enumerate more data from the victim's machine. We can tell here that the output was not helpful, but it might be helpful in other scenarios. The attacker also notices that Kerberos was installed on the target's machine as well as other active directory services. The attacker launches a brute-force attack to enumerate valid domain user names *via* Kerberos using the following tool kerbrute_linux_amd64. After identifying a list of valid users, the attacker uses a script called 'GetNPUsers' that exploits a misconfiguration in the AD. This vulnerability is called 'ASREPRoasting'. After receiving a valid TGT, the attacker used a popular wordlist to try and crack the TGT. After successfully cracking the TGT, the attacker now has valid credentials. Using these credentials, the attacker tries to connect to the target machine using the Windows remote management service. After successfully logging in, the attacker copies and executes an executable that will collect more information about the AD environment called 'SharpHound.exe'. The returned file was copied back to the attacker's machine in order to import it into a tool called Bloodhound.

The same mentioned steps can be repeated to generate the attack. The resulting datasets can vary according to the settings on the attacked computer and the Kerberos valid user names. Nevertheless, in case that we guarantee that the Kerberos user name exists inside the used dictionary during the brute-force attack step and Kerberos pre-authentication is disabled on the attacked machine, we can tell that a similar, but not exact dataset will be generated. The attacker's computer can speed up or slow down the attack. The specifications of the network or the status which the attacker is using to access the victim's computer can result in slowing down or speeding up the attack time.

The steps, the tools and the comings of the scenario described here will be detailed and discussed later through the following sub-sections. The network traffic data of every stage of the attack was recorded into a separate pcap file. So, we have a pcap file containing information about the traffic at the reconnaissance, delivery and exploitation stages. Even stages can be divided into sub-stages; for example, the first sub-stage of the reconnaissance stage and the second sub-stage of the reconnaissance stage. This style of recording data about the attack stages can provide information not only about whether the attack is related to an anomaly or a regular packet, but also tell to what stage of the attack the packet can be related when we use the dataset with AI models for example. The literature is rich in labeled datasets for different types of cyber-attacks, while a shallow number of attack datasets are labeled according to the attack life-cycle stages.

The steps followed through this study, which are listed in Figure 1, summarize the attack steps and show that the attack covers four stages; reconnaissance, delivery, exploitation and installation. We also can realize that the reconnaissance step is divided into three steps. During data collection, we created the attack, followed the listed steps and recorded the traffic for each step in a pcap file. We also exported every pcap file into CIC flowmeter and created a labeled dataset for the attack, where the label is a keyword representing the stage to which the traffic belongs.

A labeled dataset containing information about the attack life-cycle can significantly benefit building machine-learning models that can predict the attack and the attack stage, so that prevention and mitigation plans can be drawn according to these stages. The steps we followed to create a dataset for an attack that is aware of the attack stages rather than the attack type can be followed for other attacks. More datasets can be created following the proposed method. This will help generate more rich datasets for intrusion-detection and prevention intelligent models.

During the data collection part of this study, we successfully generated and simulated the "Kerberoasting" attack. The attack was performed and recorded based on the understating of the attack stages. Every stage traffic was saved in a separate pcap file; unlike other datasets, which care about the type of the generated attack, we considered the attack stages the features of which differ for the same attack.

## 3.3 The Suggested Detection Method

The method used in this study to analyze the collected data was to investigate the packets at every stage in isolation. Finding distinguishing features in every stage and recording these features in the packet's content allow that rapid detection and prevention strategies can be set.

Attacks in the active directory, like most other attacks, start with the reconnaissance stage. As we can see in Table 3, dring the first step of the attack, 133225 packets were sent to the AD server in 3 minutes, which means that the flow was around 37 packets/sec. The fourth column in Table 3 shows the number of distinct destination ports collected during every attack stage. 65510 ports were scanned during the first three minutes of the attack. On average, every port was scanned twice during this stage. The count of the records and the number of destination ports were gathered by counting the tuples from the CSV files created for each stage of the attack. The time taken was concluded from the "timestamp" field listed in the CSV files. In table 3, the notes column is for the researchers' notes which were visually realized, since we have spent a reasonable time studying the distinguishing features and details for the generated CSV files.

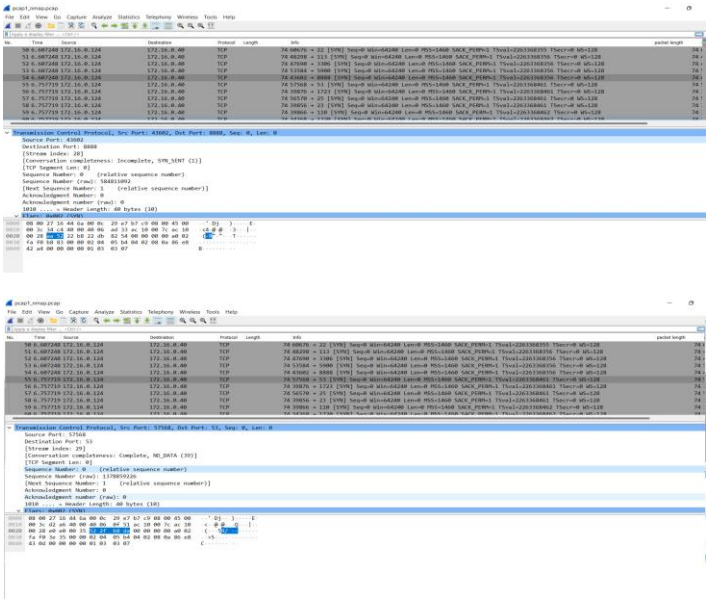

Figure 3. Packet content of two consecutive packets during the port-scanning stage.

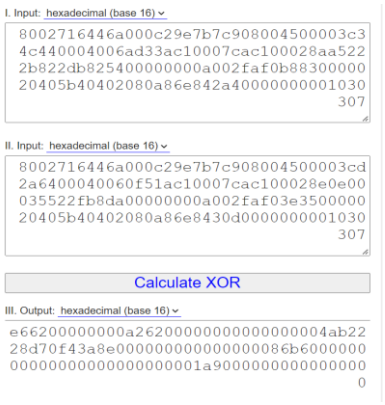


Figure 4. XOR operation between two consecutive packets.

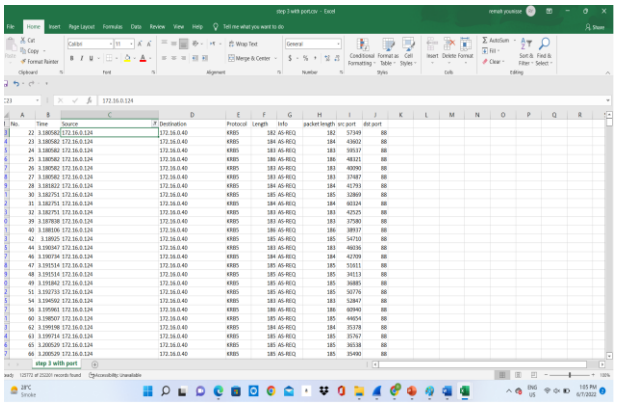

Figure 5. Data collected during brute-force attack on Kerberos.

This behavior of scanning more than 65000 ports in 3 minutes by the same IP address should be a strong indication of the reconnaissance stage of the attack life-cycle. This behavior was detected through our study in two places. First, when we analyzed the pcap files using Wireshark; Figure 3 illustrating the packet content showed high similarities between packets' contents, except for the fields; port numbers, the sequence number of the packet and timestamp value. Figure 4 shows the XORing result between two packets during this attack stage. One suggested method to detect the attack at this stage is performing an XOR operation between packet contents, excluding the fields mentioned earlier. Table 3 shows that the second stage of the attack took only 12 seconds and focused on scanning ports 139 and 445, where the shared data on Windows can be accessed. This step provides shallow information during the intrusion-detection process, but yet can be considered as an indication of scanning stage of a cyber-attack.

Table 3. Attack stages' traffic summery.

| Step | Time taken | Num. of Records | Num. of dst. ports | Notes |
|---|---|---|---|---|
| 1 | 3.0 min | 133225 | 65510 | Most packets are of the same length. |
| 2 | 12.0 sec | 555 | 31 | Mostly the scanned ports were 139 and 445. |
| 3 | 2.0 min | 252201 | 27890 | Port 88 was contacted 129534 times. |
| 4 | 4.0 sec | 113 | 19 | - |
| 5 | 3.0 min | 2235 | 19 | - |

The third stage shows dense traffic on the network, especially packets connecting to port 88, which is reserved for Kerberos. In Figure 5, it is clear that during the two minutes of running kerbrut, the packets were almost of the same length and the "DST port" value was 88, which is another indication of potential malicious purposes that can be used against the IP address "172.16.0.124". The extracted CSV files' "info" field contains information about the type of the exchanged messages.

No suspicious actions could be detected during stage 4, since the actions performed during this stage are a typical communication process handling a hashed password to the requester. We should re-mention here that in this stage, the attackers could get the password of users who used the "do not require Kerberos pre-authentication" option with their accounts' settings. The KDE sent an encrypted password and the attackers decrypted it offline. Finally, in stage 5, the information field contains the "SharpHound.exe" files which can be considered a keyword for potential attacks. Intrusion-detection systems should be built to consider such keywords to protect network systems.

## 4. CONCLUSION

In this work, we have presented the methodology to create a "Kerberoasting" attack and collect the traffic data using Wireshark. The collected data was based on the attack stage; so every managed pcap file contains information about a particular attack stage. Then, these pcap files were analyzed and studies were used to generate a labeled dataset for the attack. The labels are a reflection of the attack stage. Every stage of the attack was studied and analyzed, so that the unique characteristics of the attack stage are highlighted. The detection and defense plans against the "Kerberoasting" attack were based on these characteristics. At the early stages of the attack, we realized that the port-scanning process was evident as the destination port varied rapidly in a concise amount of time. Meanwhile, packets of the same size and data have been sent to the destination side with a different port number every time. At the later stage of the attack, a rapid amount of data was sent to port 88, trying to guess a valid user name. Through this work, we detailed how we could extract and use such information at every stage of the attack to build our defenses against the attack. We performed the attack and then studied its data and features to understand the threat in depth. The labeled dataset that we have produced can be used to build intelligent machine-learning models aware of the attack stages and life-cycle. The collected data was also exported into excel sheets and statistical analysis was carried out. In the future, we wish to use the collected data to build the intelligent intrusion-detection model that we have just mentioned.

## REFERENCES

[1] A. Yeboah-Ofori et al., "Cyber Threat Intelligence for Improving Cyber Supply Chain Security," Proc. of the IEEE Int. Conf. on Cyber Security and IoT (ICSIoT), pp. 28–33, Accra, Ghana, 2019.

[2] W. Matsuda, M. Fujimoto and T. Mitsunaga, "Detecting APT Attacks against Active Directory Using Machine Leaning," Proc. of the IEEE Conf. on Application, Information and Network Security (AINS), pp. 60–65, Langkawi, Malaysia, 2018.

[3] C. D. Motero et al., "On Attacking Kerberos Authentication Protocol in Windows Active Directory Services: A Practical Survey," IEEE Access, vol. 9, pp. 109289–109319, 2021.

[4] L. Kotlaba, S. Buchovecká and R. Lórencz, "Active Directory Kerberoasting Attack: Detection Using Machine Learning Techniques," Proc. of the 7th Int. Conf. on Information Systems Security and Privacy (ICISSP 2021), pp. 376-383, DOI: 10.5220/0010202803760383, 2020.

[5] M. Alkasassbeh et al., "Detecting Distributed Denial of Service Attacks Using Data Mining Techniques," Int. J. of Adv. Comp. Sci. and Appli., vol. 7, no. 1, 2016.

[6] M. Almseidin, J. Al-Sawwa and M. Alkasassbeh, "Generating a Benchmark Cyber Multi-step Attacks

Dataset for Intrusion Detection," J. of Intelligent & Fuzzy Systems, vol. 43, no. 3, pp. 3679-3694, 2022.

[7] M. Lehto, "APT Cyber-attack Modeling: Building a General Model," Proc. of the 17th Int. Conf. on Cyber Warfare and Security, vol. 17, DOI: 10.34190/iccws.17.1.36, 2022.

[8] M. Almseidin, J. Al-Sawwa and M. Alkasassbeh, "Anomaly-based Intrusion Detection System Using Fuzzy Logic," Proc. of the IEEE Int. Conf. on Inf. Tech. (ICIT), pp. 290–295, Amman, Jordan, 2021.

[9] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington and C. B. Thomas, "Mitre ATTACK: Design and Philosophy," Project No.: 10AOH08A-JC, The MITRE Corporation, 2018.

[10] T. Dargahi, A. Dehghantanha, P. N. Bahrami, M. Conti, G. Bianchi and L. Benedetto, "A Cyber-kill-chain Based Taxonomy of Crypto-ransomware Features," Journal of Computer Virology and Hacking Techniques, vol. 15, no. 4, pp. 277–305, 2019.

[11] H. Kim, H. Kwon and K. K. Kim, "Modified Cyber Kill Chain Model for Multimedia Service Environments," Multimedia Tools and Applications, vol. 78, no. 3, pp. 3153–3170, 2019.

[12] R. Badhwar, "Advanced Active Directory Attacks and Prevention," in Book: The CISO's Next Frontier, pp. 131–144, Springer, 2021.

[13] S. Muthuraj, M. Sethumadhavan, P. Amritha and R. Santhya, "Detection and Prevention of Attacks on Active Directory Using SIEM," Proc. of the Int. Conf. on Information and Communication Technology for Intelligent Systems (ICTIS 2020), vol. 196, pp. 533–541, 2020.

[14] T. Osmëni and M. Ali, "Exploration of the Attacking Web Vectors," Proc. of the IEEE Int. Conf. on Computing, Networking, Telecomm. & Eng. Sci. Appl. (CoNTESA), pp. 31–35, Tirana, Albania, 2021.

[15] A. Hassanzadeh and R. Burkett, "SAMIIT: Spiral Attack Model in IIoT Mapping Security Alerts to Attack Life Cycle Phases," Proc. of the 5th Int. Symposium for ICS & SCADA Cyber Security Research, pp. 11–20, DOI: 10.14236/ewic/ICS2018.2, 2018.

[16] M. Li, W. Huang, Y. Wang, W. Fan and J. Li, "The Study of APT Attack Stage Model," Proc. of the IEEE/ACIS 15th Int. Conf. on Computer and Inf. Sci. (ICIS), pp. 1–5, Okayama, Japan, 2016.

[17] J. D. Mireles, J.-H. Cho and S. Xu, "Extracting Attack Narratives from Traffic Datasets," Proc. of the IEEE Int. Conf. on Cyber Conflict (CyCon US), pp. 1–6, Washington, USA, 2016.

[18] A. Dimitriadis, N. Ivezic, B. Kulvatunyou and I. Mavridis, "D4i-digital Forensics Framework for Reviewing and Investigating Cyber Attacks," Array, vol. 5, p. 100015, 2020.

[19] L. Kotlaba, S. Buchovecká and R. Lórencz, "Active Directory Kerberoasting Attack: Monitoring and Detection Techniques," Proc. of the 6th Int. Conf. on Inf. Sys. Security and Privacy (ICISSP 2020), pp. 432–439, DOI: 10.5220/0008955004320439, 2020.

[20] M. Al-Kasassbeh and T. Khairallah, "Winning Tactics with DNS Tunnelling," Network Security, vol. 2019, no. 12, pp. 12–19, 2019.

[21] MITRE, "Active Directory," [Online], Available: https://attack.mitre.org/datasources/DS0026/, 2022.

[22] MITRE, "Use Alternate Authentication Material," [Online], Available: https://attack.mitre.org/techniques/T1558/, 2022.

[23] SecLists, "Common-credentials," [Online], Available: https://github.com/danielm iessler/SecLists/blob/master/Passwords/Common-Credentials/10-million-password-list-top-100000.txt, Accessed: Oct. 2022.

**ملخص البحث:**

يتخذ أمان الإنترنت عادةً شكلاً دفاعيّاً للتّصدّي للهجمات ذات الأغراض الخبيثة. ويمكن لأفعال المهاجمين على الإنترنت أن تتّخذ أنماطاً معينة وتأخذ خطواتٍ محدّدةً في كلّ مرّة ينفّذون فيها هجماتهم. ومن الممكن استخدام تلك الأنماط لتوقّع تلك الهجمات والتّصدّي لها وإيقافها. تقترح هذه الدّراسة طريقةً لوسم مجموعات البيانات فيما يتعلّق بالهجمات متعدّدة المراحل، تبعاً لمراحل تلك الهجمات بدلاً من نوع الهجوم. ويمكن استخدام مجموعات البيانات لاحقاً في نماذج تعلُّم الآلة لبناء أنظمةٍ دفاعيةٍ ذكيةٍ. ومن جهة أخرى، نقترح في هذه الدّراسة طريقةً لتوقّع الهجمات وقتلها مبكّراً في بيئة "الدّليل النّشط"، مثل الهجمات الّتي تهدف الى كَسْر كلمة السّرّ للوصول الى البيانات. في هذه الدّراسة، قمنا بجمع البيانات المتعلّقة بسيناريو مقترح لإحدى تلك الهجمات في ملفّات خاصّة، بحيث يحتوي كلٌّ من تلك الملفّات البيانات المتعلّقة بمرحلةٍ معينةٍ من مراحل "دوْرة حَياة" الهجْمة، وتمّ استخدام المعلومات المستخلصة من الملفّات لتسليط الضّوء على السِّمات والانشطة خلال كلّ خطوة. وتمّ استخدام تلك المعلومات في وضع خطّة دفاعية ضدّ الهجمة. كذلك نقترح منهجيةً لوضع خططٍ دفاعية لهجمات أخرى مشابهة غير هجمات كسْر كلمة السّرّ المستهْدَفة في هذه الدّراسة.