Detecting APT attacks against Active Directory using Machine Leaning

1st Wataru Matsuda
The University of Tokyo
Tokyo, Japan
coe@ml.sisoc.tokyo

2nd Mariko Fujimoto
The University of Tokyo
Tokyo, Japan
coe@ml.sisoc.tokyo

3nd Takuho Mitsunaga
The University of Tokyo
Tokyo, Japan
coe@ml.sisoc.tokyo

***Abstract*—In Advanced Persistent Threat (APT) attacks, attackers who can intrude into an organization network tend to stay inside the network or repeat intrusion multiple times until they are able to accomplish their goals. When Active Directory(AD), a centralization management system for Windows computers, is in place, attackers try to disguise themselves as users of legitimate Domain Administrator accounts, which is the highest privileged account of the AD environment. Activities on the Windows system are recorded in the built-in Windows activity logging system called the Event logs and is commonly used for investigation of attacks. However, if attackers leverage legitimate accounts or built-in Windows tools in order to avoid detection, it is quite difficult to detect attacks from Event logs since attackers' activities are recorded as activities of legitimate administrator accounts. Although there are various antivirus software, detecting such a sophisticated attack is often very difficult. In this research, we focus on processing attack activity data recorded in the Event logs, and propose a new method based on outlier detection and machine learning for detecting attacks that utilize legitimate accounts. We achieved a high precision rate even if legitimate Domain Administrator accounts are leveraged in attacks.**

***Index Terms*—machine learning, outlier detection, unsupervised learning, Active Directory, Event log, APT, Golden Ticket**

I. Introduction

Active Directory (AD) is a centralized management system for Windows computers and accounts. The fact that it uniformly manages an organization's resources has made it a common target of Advanced persistent threat(APT) attacks. There have been numerous reports on AD environments being exploited in APT attacks [1], and in many of the victims' organizations, it is common for the Domain Administrator privileges of the AD to be abused. The Domain Administrator account is an ideal target for attackers since it holds Administrator privileges to all the resources within the AD environment. Attackers who can get the Domain Administrator privilege likely create a backdoor that disguises itself as a legitimate account called "Golden Ticket" [2], in order to obtain long-term administrator privilege. Event logs are useful for attack detection and investigation because they record attackers' activities such as the abuse of accounts and processes etc. Unusual processes can be detected by analyzing event logs, and black-listing these abnormal processes is one of the solutions for a more secure system. In the case of black-listing however, security operators have to maintain the black lists or white lists often consisting of process names, directory name of executed processes, account information, etc. Black lists can also be a cause for increased false detection since attackers tend to abuse legitimate process. Along with white listing, these methods also tend to be time consuming should it be frequently maintained and updated. For these reasons, a new approach that does not depend on the black listing method is required for more accurate and efficient detection of attacks. In this research, we propose a new method for detecting attack activities from event logs using unsupervised machine learning. Identifing event logs that indicate an attack is difficult because the nature of the event logs greatly depends on the daily operation of each unique AD environment. Since unsupervised machine learning does not need label information (in this case whether the event logs indicate an attack or normal behavior), it is suitable for this case. In our method, only the event logs of the Domain Controller[1] during normal daily operations are needed as the dataset. Even if attackers take advantage of legitimate processes, our method can detect the attack activities by identifying processes that are not normally used in daily operations. Furthermore, our method can reduce operation costs because black lists and white lists are not needed.

II. Background

A. Summary of Active Directory

In this section, we explain the summary of the Kerberos authentication used mainly in an AD environment. In an AD environment, the Domain Controller uniformly processes all authentications, using authentication tickets called Ticket-Granting Tickets (TGT) and Service Tickets (ST).

- Ticket-Granting Tickets (TGT): A ticket that proves the authenticity of the user. The client requests for a TGT to the Domain Controller on its first authentication process, and the TGT is stored in the users' computer and repeatedly used until its expiration. The default expiration limit is ten hours since the ticket's creation.
- Service Ticket (ST): A ticket that authorizes the use of a service within the AD environment. Upon the use of a service, the user requests for a ST to the Domain

[1] An centralized authentication server of the AD environment

978-1-5386-6925-9/18/$31.00 ©2018 IEEE

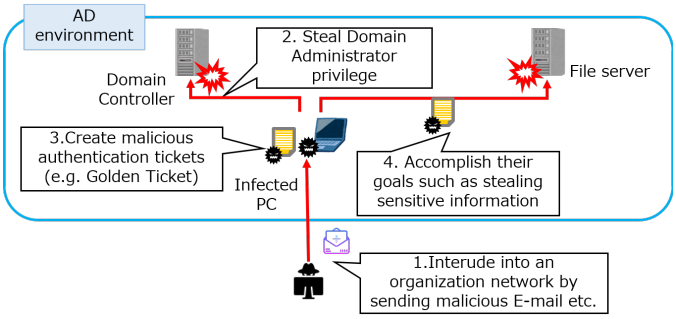

Fig. 1. Phases of attack against Active Directory

Controller and uses the ticket to prove its authenticity to the service server.

B. Attacks against Active Directory

This subsection describes typical attack methods against AD abusing the Kerberos authentication. An Example of attacks against AD are shown in Figure 1. There are several attack phases as follows.

1) Intrusion: an attacker infects a computer with malware by sending E-mail with malicious attach file or lead users to malicious web sites.
2) Steal Domain Administrator privilege: steal Domain Administrator privilege in some way such as leveraging vulnerabilities of AD
3) Create the Golden Ticket: create a Golden Ticket in order to gain long-term Administrator privileges
4) Attack with the Golden Ticket: expand infection or steal sensitive information using the Golden Ticket

Attackers who intrude into an organization network try to obtain Domain Administrator privileges in some way (e.g. privilege escalation, abusing credentials), and then attack the Domain Controller to create a Golden Ticket. Attackers who successfully create the Golden Ticket are able to disguise themselves as arbitrary Administrator accounts when intruding the system. A Golden Ticket is a TGT with a legitimate signature, which is created by attackers. Attackers tend to create a Golden Ticket that has access to any given Domain Administrator account and also has a significantly long term of validity such as ten years. The extended expiration limit of the Golden Ticket enables the attackers to continuously use it even after the password for the compromised account is changed. Furthermore, if attackers use the Golden Ticket with a legitimate account, it is often difficult to differentiate malicious attacks from normal authentications. Attackers not only use attack tools but also built-in Windows commands. Common commands that attackers tend to use are show in [3].

III. Previous Research

There are various research for detecting attacks against AD using machine learning, focusing on elements such as signatures and characteristics of attacks. In this section, we describe a summary of previous research.

A. Detection using authentication logs

Chih-Hung Hsieh et al. use unsupervised machine learning to analyze Event logs related to Kerberos authentication in chronological order to detect abnormal behavior of users [4]. Markus Goldstein et al. monitor abnormal user behavior by utilizing semi-supervised learning when analyzing Event logs related to authentication [5]. Both are monitoring methods that use the authentication logs, focusing on suspicious and abnormal authentications such as the change in frequency of authentications within a system or sudden requests from unexpected computers or accounts. In research [4], there is a problem in the detection accuracy as mentioned in the research paper: ”AD2 only can produce about 66% recall rate or accuracy. That may give us another conjecture that anomaly detection based on analyzing AD log may be limited by information which AD log can tell.”.

B. Detection using process logs

Michael Gough introduces the signature-based method for detecting processes with blacklist using Event logs related to process [6]. However, false negatives can occur if attackers had changed the file names of the tools since signature is based on the filename. Furthermore, false positive can occur if legitimate operators use commands which match one of the signatures for daily operation.

C. Detection through network traffic monitoring

Several methods are proposed for detecting attacks such as Golden Ticket through monitoring DC ' s network traffic [7], [8]. However, these features are not implemented in the Windows system, so it is necessary to install additional software or alter network structure in order to implement these methods.

D. Contribution of our proposed method

Our method has the following advantages.

- The method can detect attacks against AD with high accuracy even if attackers abuse the legitimate computers or accounts such as Golden Ticket attacks.
- The method can detect tools or commands abused by attackers which are not used for daily operation even if the file name is changed by attackers, or if commands/tools provided by Microsoft are leveraged.
- The method can minimize analyzing cost since it uses only Domain Controller's Event log for detection.
- The method can be implemented with only Domain Controller's log configuration, no need to install additional software and tools, alter existing network structure.

Table I shows features of the proposed method compared with previous research.

IV. Proposed Method

In this study, we propose a method for outlier detection with machine learning using the Domain Controller's Event logs related to processes. We focus on detecting attacks that require Domain Administrator privilege as described in phase

TABLE I
COMPARISON WITH PREVIOUS RESEARCH

Research[a]	Input data	Required settings	Method	Advantage	Disadvantage
[4], [5]	Authentication log	None	Machine learning	Does not need to alter any existing settings or configurations.	False negative can occur if legitimate accounts are compromised.
[6]	Process log	Audit Policy setting of Windows system	Signature based	Processes included in the signature can be detect certainly.	False negative can occur if attackers change the file name of processes used for attacks. False positive can occur if legitimate administrators launch processes which are included in the signature.
[7], [8]	DC's network traffic	Altering network structure	Monitoring suspicious Kerberos authentication	It can detect Golden Ticket attacks.	Difficult to implement in an operational environment in aspects of implementation cost.
Proposed method	Process log	Audit Policy setting of Windows system	Machine learning	It can detect abused processes if they are not used in daily operations even if legitimate accounts are compromised.	False negative can occur if legitimate administrators use CLI tools regularly.

[a]Reference to corresponding previous research

(2) to (4) in subsection II-B since attacks at these stages could potentially cause the severest of damages.

A. Dataset

Activities on a Windows system such as authentication requests and process execution are recorded on the computer as Event logs. Event logs are recorded on both the Domain Controller and each of the client computers, and is divided into several categories such as authentications and processes. Each category is assigned an unique Event ID. Our proposed method only uses the Domain Controller's Event logs related to process execution, as shown in Fig 2 and Table II. Event ID 4674 and 4688 are not recorded in the Event logs with the default settings, thus the Windows Audit Policy[2] should be enabled as shown in Table III.

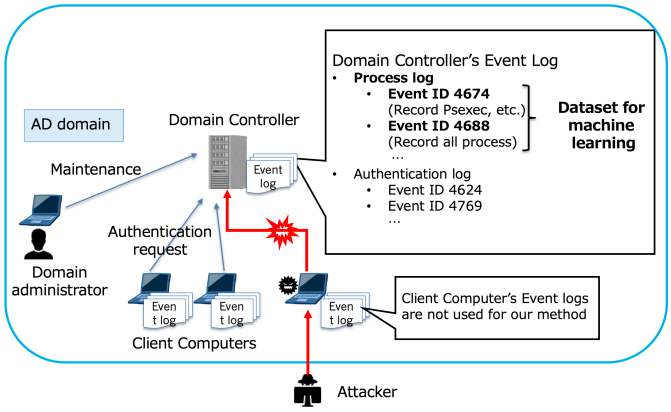

Fig. 2. Event ID used for input data

TABLE II
EVENT ID USED FOR INPUT DATA

Event ID	Description	Technical point
4674	An operation was attempted on a privileged object	Specific process information executed with special privileges(e.g. SeCreateGlobalPrivilege, SeSecurityPrivilege).
4688	A new process has been created	All processes information executed on the computer.

TABLE III
ADVANCED AUDIT POLICY CONFIGURATION

Event ID	Category	Audit Policy[a]
4674	Privilege Use	Audit Sensitive Privilege Use
4688	Detailed Tracking	Audit Process Creation

[a]Both "Success" and "Failure" should be enabled.

Event logs of both Event ID 4674 and 4688 are used as datasets because they are complementary to each other. Event ID 4688 records all processes while Event ID 4674 records only specific processes executed with special privileges[3]. Logs with Event ID 4674 are included because when attackers abuse a tool called "Psexec" in order to remotely access the Domain Controller, information about a temporary file (%SystemRoot%
PSEXESVC.exe) is recorded on the Domain Controller Event logs with the Event ID 4674. This information is useful for detecting the execution of Psexec becuase the temporary file name that is logged here is constant even if attackers

[2]It defines what types of events are written in the Security logs of Windows system.

[3]There are several privileges in a Windows system

TABLE IV
INFORMATION IN EACH EVENT LOG

Name	4674	4688
Account Name	○	○
IP Address	Use data in Event ID 4769	Use data in Event ID 4769
Service	○	-
Process Name	○	○
Object Name	○	-

subsequently change the file name.

This method uses information shown in Table IV extracted from each Event ID shown in Table II. ○ represents that the specific data is used as part of the dataset.

B. Detection algorithm

This subsection describes outlier detection algorithms that can detect abnormal processes rarely recorded in the usual operational environment.

1) Machine learning: Machine learning gives computer systems the ability to "learn" with data without being explicitly programmed. It's divided into supervised and unsupervised learning, and the proposed method uses unsupervised learning. Supervised learning requires the outputs to be already known and the data used for training to be labeled with correct answers. However, sometimes it may be difficult to give the labels of correct answers to a learning algorithm. For instance, in the case of attack detection, we have to analyze a real attacker's behaviors and specify whether each log is recorded by attacks or not. On the other hand unsupervised learning does not require labels of correct answers. That means we do not need to analyze attackers' behaviors. For these reasons unsupervised learning is widely used for anomaly or outlier detection. Especially in the AD environment, suspicious activities can be identified through comparing event logs with daily operation. For example, commands that are not used in daily operation indicate attacks. Identifing event logs that indicate an attack is difficult because the event logs greatly depend on daily operation. Therefore, unsupervised machine learning is suitable for detecting attacks against AD environments. There are several algorithms that are suitable for unsupervised learning as shown in Table V, so we will use these algorithms in our experiment and compare the detection rate.

2) Preprocessing for machine learning: This section describes the necessary preprocessing for machine learning. The Windows Audit Policy shown in Table III must be first configured, then the dataset must be extracted as follows.

1) Extract Event ID and information from Domain Controller's Event log(Security log) shown in subsection IV-A.
2) Remove noises from log.
3) Extracted source IP address information from Event ID 4769, and give it to Event ID 4688 and 4764.

Logs that show the following features should be removed because they can be identified as noise.

- Logs with blank values
- Logs frequently recoded regardless of user's operation. For instance, service.exe、lsass.exe recorded in Event ID 4674. However, if "PSEXESVC" is recorded in "Object Name" field, it should not be removed.

TABLE V
MACHINE LEARNING ALGORITHMS

Algorithm	Summary
One-Class SVM	An unsupervised algorithm that learns a decision function for novelty detection: classifying new data as similar or different to the training set. The training data is not polluted by outliers.
Isolation Forest	The IsolationForest 'isolates' observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature.
LOF	The LOF algorithm is an unsupervised outlier detection method which computes the local density deviation of a given data point with respect to its neighbors.

Source IP address information is required in order to identify the compromised computer, but Event ID 4674 and 4688 do not have source IP address information. Thus, we specify the Source IP address using Event ID 4769 recorded just before Event ID 4674, 4688 for each account. The reason why we use Event ID 4769 is that a ST is likely requested before process execution. A ST is also required before attackers use malicious authentication tickets such as the Golden Ticket. Through this step, our proposed method can identify the computer that runs each process.

V. EVALUATION

This section describes the evaluation to see if our proposed method can detect attacks against the AD. We compare the detection rate of the proposed method with previous research [4]. The reason why we select [4] for comparison is the detection rate is mentioned clearly.

A. Evaluation environment

Our evaluation environment is shown in Table VI. This environment is used in a production level training program we provide using actual machines and equipment. Users belong to an Active Directory domain with user rights (Domain Users). In this test environment, we use logs for test dataset shown in Table VII, which were collected during 14 days. A dataset ratio is the ratio of "the total number of logs" to "the number of dataset logs". A lower dataset ratio means that much smaller amount of data is needed for machine learning.

B. Contents of evaluation

We conduct an attack mocking a typical APT attack against the AD environment. We create a Golden Ticket for a legitimate Domain Administrator account and use it for the attack to see if the proposed method can detect attacks when a legitimate Domain Administrator account is compromised.

TABLE VI
EVALUATION ENVIRONMENT

Type	OS	Number of computers
DC	Windows Server 2008 R2	1
File Server	Windows Server 2008 R2	1
Client Computer	Windows 7 (x64)	39

TABLE VII
LOG DATA FOR EVALUATION

	Proposed method	Previous research
Total log size(MB)	231	22500
Total number of logs	161,476	12,310,519
Log size for dataset(MB)	25.3	5200
The number of logs for dataset	22,738	2,887,504
Dataset ratio[a]	0.11	0.23

[a](The number of dataset logs) / (The total number of logs)

1) Intrusion: We presuppose that the attacker has already intruded in a non privileged user's computer. This phase is excluded from detection using our method.
2) Steal Domain Administrator privilege : Access the DC with Domain Administrator privilege leveraging the vulnerability of MS14-068[4] [10].
3) Create the Golden Ticket: Mount the C drive of the Domain Controller using administrative shares [5], and place attack tools. Then Create the Golden Ticket for a legitimate Domain Administrator account.
4) Attack with the Golden Ticket: Expand infection to the Domain Administrator's computer, and mount the C drive of the file server using the Golden Ticket, then steal information in the shared folder.

Tools and commands used during each attack phase are shown in Table VIII. There is a possibility that attackers use other commands, however we use the least commands in order to evaluate whether the proposed method works if attackers use the least commands to avoid detection. We change the file name of a tool mimikatz[6] to see if the proposed method can detect attack tools even if the filename is changed.

In order to evaluate false detection rates, we also conduct mock administrative operations such as "add users", "reset user passwords", "check Event viewer", etc. We login to the Domain Controller with the compromised Domain Administrator account and perform operations using GUI tools.

[4]A vulnerability of AD which is often leveraged for APT attacks, attackers who have only Domain User privilege can get Domain Administrator privilege by leveraging it.

[5]Built-in hidden network shared resource in Windows system

[6]A tool for attacking Windows computers.

TABLE VIII
COMMANDS OR TOOLS USED FOR EACH ATTACK PHASE

Attack Phase	Command or Tool
2,3 and 4	mimikatz
2,3 and 4	klist
3 and 4	psexec
3 and 4	wmic
2	ipconfig
2	hostname
2	netstat
3	net
3	copy
3	schtasks

VI. EVALUATION RESULTS

This section describes the result of the evaluation.

A. Comparison with previous research

We compare with previous research and evaluate the detection rate. Precision is the ratio of correctly predicted positive observations to the total predicted as a positive class. Precision is given by the formula: Precision = TP / (TP + FP)

Recall is the ratio of correctly predicted positive observations to the all observations in actual positive class. Recall is given by the formula: Recall = TP / (TP + FN)

Accuracy is defined as the percentage of correctly classied positive and negative class. Accuracy is given by the formula: Accuracy = (TP + TN) / (TP + TN + FP + FN)

Table IX shows the detection rate of the proposed method compared with those of previous research [4]. We use "One-Class SVM" for machine learning algorithm which performed the highest detection rate described in subsection VI-B.

TABLE IX
COMPARISON WITH PREVIOUS RESEARCH

Method	Recall	Precision	Accuracy
4674 and 4688 [a]	1.0	0.81	0.95
4674 only [b]	1.0	1.0	1.0
4688 only [c]	1.0	0.97	1.0
Previous research	0.66	0.99	0.66

[a]Use both Event ID for dataset(3,127 logs).
[b]Use only Event ID 4674 for dataset(742 logs).
[c]Use only Event ID 4688 for dataset(2,385 logs).

As the result, our method achieves a low false negative rate compared with previous research.

B. Difference among algorithm

We evaluated the difference in detection rate among algorithms shown in Table V. We used Event ID 4674 as training data. We tested several variations of parameters for each algorithm and compared the highest detection rate. Table X shows the detection rate for each algorithm.

As the result, "One-Class SVM" yielded the highest detection rate.

TABLE X
RESULTS FOR EACH ALGORITHM

	Parameter	Recall	Precision	Accuracy
One-Class SVM	nu=0.1, kernel="rbf", gamma=0.01	1.0	1.0	1.0
LOF	n_neighbors = 100	0.05	0.07	0.74
Isolation Forest	random_state = rng	0.43	0.90	0.50

The number of total Event logs: 742

C. Effectivity of the preprocessing

We compared the difference in detection rate between logs with preprocessing and logs without preprocessing. Preprocessing is mentioned in subsection IV-B2.

- Logs with preprocessing: Proceed step (1), (2) and (3)
- Logs without preprocessing: Proceed only step (1)

We used logs with Event ID 4674 as the dataset and the One-Class SVM algorithm. The highest detection rate was compared through adjusting the parameters. Results are shown in Table XI.

TABLE XI
RESULT(EFFECTIVITY OF PREPROCESSING)

	Parameter	Recall	Precision	Accuracy
With preprocessing [a]	nu=0.1, kernel="rbf", gamma=0.01	1.0	1.0	1.0
Without preprocessing [b]	nu=0.001, kernel="rbf", gamma=0.01	1.0	0.06	0.06

[a]Total 742 logs.
[b]Total 9,132 logs.

As the result, preprocessing remarkably improves the detection rate.

D. Remarks on the results

Our method yields a high recall rate compared with previous research.

Regarding the algorithm used for machine learning, One-Class SVM is the most appropriate algorithm for our method.

In aspects of Event ID, when we train each Event ID separately, the detection rate was high compared with the case when we train both Event ID at the same time. We found out that Event ID 4674 achieved high detection rate, but commands or tools which can be detected using this event were limited. If attackers use any other commands besides these limited commands, false negative detection may occur. More commands can be detected by using logs with Event ID 4688. It is desirable to use logs of both Event ID 4674 and 4688 for detection, and train them separately in order to mitigate false negative.

In addition, preprocessing mentioned in subsection IV-B2 remarkably improves the detection rate, and makes it easier to identify the compromised computers.

E. Remarks on false detection

False negative occurs if both of the following conditions are met.

- Attackers disguises themselves as a legitimate Domain Administrator account.
- The compromised Domain Administrator account uses commands or tools regularly which the attacker also uses.

On the other hand, false positive occurs if legitimate Administrators use commands which are not used in the daily operation. For instance, we detected "ping" execution under environment which domain administrator does not use "ping". It is possible to find out whether the result is false positive or not through monitoring logs in a short period and raising alerts so that administrators can easily compare the alert with their real operations.

VII. CONCLUSION

It is often difficult to detect attacks against an AD since attackers tend to leverage legitimate accounts and processes. In this research, we proposed a method for detecting attacks with outlier detection using unsupervised learning, focusing on Event logs related to process creation. The method yields a high recall and precision rate even if a legitimate Domain Administrator account is leveraged, or a file name of an attack tool is changed. False detection could still occur in the case that domain administrators commonly use commands that attackers use, therefore there is room for future consideration. The proposed method is a highly practical detection method for mitigating damages of APT attacks because it only uses the built-in Windows Event logs and is relatively easy to implement in a running environment.

REFERENCES

[1] Shingo Abe,"Detecting Lateral Movement in APTs", JPCERT Coordination Center
[2] "Protection from Kerberos Golden Ticket", CERT-EU
[3] "Windows Commands Abused by Attackers", JPCERT Coordination Center
[4] Chih-Hung Hsieh, "AD2: Anomaly Detection on Active Directory Log Data for Insider Threat Monitoring", Institute of Information Industry Taipei
[5] Markus Goldstein, "Enhancing Security Event Management Systems with Unsupervised Anomaly Detection", German Research Center for Artificial Intelligence
[6] "Finding Advanced A*acks and Malware With Only 6 Windows EventID ' s", Splunk Inc.
[7] Idan Plotnik, "System, method and process for detecting advanced and APT attacks with the recoupling of Kerberos authentication and authorization"
[8] Darren B Schwartz, "Systems and methods for detecting and reacting to malicious activity in computer networks"
[9] "Detecting Lateral Movement through Tracking Event Logs", JPCERT Coordination Center
[10] "Vulnerability in Kerberos Could Allow Elevation of Privilege", Microsoft