# A Technique of Symptoms Analysis over Time for Detection of APT Attacks

Sang-Yong Choi[1], Sung-Jae Hwang[2], Yong-Min Kim[3], Dae-Jun Joo[4], Seung-Do Jeong[5], Bong-Nam Noh[6]
[1] *Interdisciplinary of Information Security, Chonnam National University/KAIST CSRC, Daejeon, Korea*
[2]*KAIST Graduate School of Information Security, Daejeon, Korea*
[3]*Dept. of Electronic Commerce, Chonnam National University, Yeosu, Korea*
[4] *KAIST Cyber Security Research Center, Daejeon, Korea* [5]*National Computing & Information Agency, Korea*
[6] *Dept.of Electronics Computer Engineering, Chonnam National University, Kwangju, Korea*
[1] *csyong95@kaist.ac.kr,* [3] *ymkim@jnu.ac.kr{Corresponding Author}*

## Abstract

*In general, the technologies of intrusion detections are based on anomaly detection and misuse detection, and modify these two detection methods to supplement the limitation of each other. As the attacking technique advances, however, existing methods have limitations in detecting attacks which utilize zero-day vulnerabilities. Due to these problems, a new attack called APT(Advanced Persistent Threat) attack which cannot be precisely detected with existing detection methods has been appeared. In this paper, we show the features of APT attack and the limitations of existing detection methods. Also we propose a new detection method to detect APT attack accurately.*

**Keywords:** Intrusion detection, APT, Symptoms Analysis

## 1. INTRODUCTION

Recently, the improvements of detection techniques make the attackers change the methods to bypass the detection systems. One of the core changes is combination of the various attacking technique. APT(Advanced Persistent Threat) is one of the typical techniques which have been mentioned a direct way of many large-scale attacks. There are various researches for detecting APT attack on a view of security control. However, there are limitations in existing techniques to precisely detecting the APT attack. The APT attack has two features; one is that it uses zero-day vulnerabilities. The other is that the attacks don't finish in one blow. Instead, it takes long time to achieve the goal of attack. These features make APT attack easy to bypass the current detection methods which based on signature.

In this paper, we introduce a new approach of detecting APT attack to overcome the limitations of existing techniques. It is called Symptoms Analysis over Time(SAoT), which uses two approaches to detect attacks. First, it is based on "Symptoms" which happens when attacks going. However, it is extremely difficult to distinguish the normal access from piecemeal symptom. Therefore, we consider second approach which is based on changes in symptoms over time.

## 2. Related Works

### 2.1. APT

Typically, APT attack uses latest zero-day vulnerabilities in order to bypass detection and blocking by antivirus programs and security systems. Moreover, it utilizes the media such as E-mail because it is difficult to detect through current security systems. ++We tested whether the current security systems are efficient in detecting the APT attack. The test was conducted to target the actual control center. The results show that there is a limitation in the current security systems to detect the APT attack [1]. Table 1 indicates whether the APT is detected during penetration.

**Table 1: Detection Results of APT Attack**

| Monitoring items | Detection Results |
|---|---|
| Encrypted traffic | X |
| Suspicious connections | Δ |
| Infected with Malware(PC, etc.) | X |
| Outbound traffic | X |
| E-mail containing Malware | X |

### 2.2 Methods of Intrusion Detection

Intrusion detection methods are generally divided into Misuse Detection and Anomaly Detection [2, 3]. Intrusion detection methods have been developed in various forms on the basis of these two methods [4, 5] such as Real-Time Intrusion Detection[6], Agent Based Network Intrusion Detection[7], and Feature reduction using a detection technique[8].

#### 2.2.1 State Transition Analysis

State Transition Analysis is the intrusion detection method[9] which generates scenario including the attacking processes. The scenario contains an initial state, *N* different transition states and compromised states which mean the attack has completed. State Transition Analysis method detects the attack through defined scenario which includes the essential nature that appears in transition state like process access and directory write access.

#### 2.2.2 NetSTAT

State Transition Analysis is considered as Host-based detection model, while NetSTAT model[10] is known as network-based detection model with the same structure. NetSTAT defines various scenarios based on network states, and utilizes them to detect the attack. In this model, the audit data is collected from multiple hosts and combined for the analysis.

#### 2.2.3 Limits of Conventional Method

State Transition Analysis and NetSTAT models are an improved methods compared to traditional signature-based

approach as they utilize the status of the host and network in order to detect the attack. However, there are a few limitations and it is inefficient to directly apply them for the detection of APT attack. One is that the State Transition Analysis Model only starts analyzing after the penetration. Moreover, the attack uses zero-day vulnerabilities and does not involve the user's behaviors which cannot be detected. In NetSTAT model, it may be possible to detect the attack before the actual penetration. On the other hand, the model does not consider duration of attack. Therefore, it has a limitation in detecting long-term attacks.

## 3. SAoT

In this paper, we propose a method that can detect APT attacks. This method improves existing detection methods. SAoT makes scenarios of attack procedures to use itself as a rule for attack detection. A difference between SAoT and existing models is that concept of time is included in these scenarios used in SAoT. Also, SAoT uses a key factor of existing models, the state of networks and the state of systems, to form these scenarios. The basic components of SAoT are follows.

### 3.1. Basic Elements of System

#### 3.1.1 Symptoms

A symptom is a phenomenon in systems and networks occurred by attacks. Symptoms are dependent on the structure of existing systems or networks. The symptoms occurred by attacks are different from those at norm state.

#### 3.1.2. Probe

Anomalies are gathered by Probe. Users cannot recognize symptoms before the attack, but can predict the occurrence of symptoms. Probe gathers everything which can be occurred, and processes it to save as meaningful data. In detail, Probe gathers various states of networks, information systems, and PC, and the threshold which determines whether the data is meaningful.

#### 3.1.3 Detection Rule

SAoT makes detection rules including the characteristics of APT to detect APT attack. In other words, the concept of time is included in detection rules. In detection rule, symptoms which occurred by attacks including zero-day attack are defined. It is not required to analyze dangerous vulnerabilities because the symptoms due to the vulnerabilities are similar. Detection rule is composed as matrix which consists of rows and columns. Columns are defined as events occurred at network routers, servers, and PCs in a timely manner. Rows are defined as symptom caused by each Nth trial of attack.

### 3.2 Procedure of Detection

SAoT has four steps to detect and alarm attacks.

1. Probe gathers symptoms at system- and network-level machines. Gathered data is normalized at canonicalization module inside the Probe. Then, normalized data with "*timestamp"* is stored at database.
2. An administrator inserts symptoms according to various scenarios as rules to detect attacks (Rule Database). Then, he/she defines the level as boundary-point which event should be occurred for each rule such as *Event_a*
3. In an analysis module, detected *Symptom_i* is compared with each rule which is stored at existing Rule Database. Then the most suitable rule is selected. If there is a symptom which matches the rule and higher boundary-point, the symptom is regarded as *Event_a*.
4. At detection module, each value of *Rule_i* under than *Step_id* of *Event_a* is compared to *TmSymptom_i* in Collect Database which collected last six-month. The conditions of comparison can be IP, time, port number, service, and resources according to the rule. If the result is higher than match-rate, alert occurs.

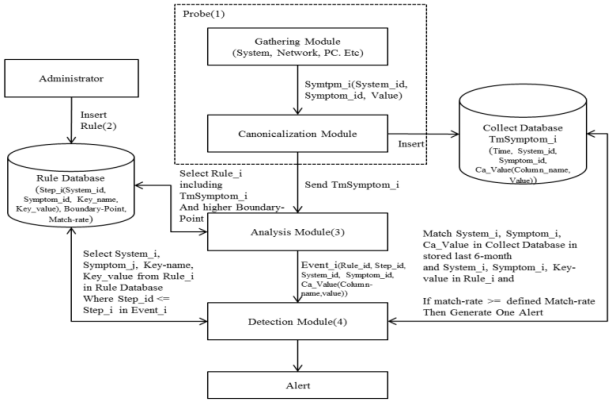


**Figure 1: Detection Process of SAoT**

These procedures are well illustrated in Figure 1.

### 3.3 Example

To show SAoT is a better detection method than existing methods, we exemplify the APT attack detecting procedure. APT attack procedure can be defined as Table 2.

**Table 2: APT Attack Process**

| | |
|---|---|
| A. | Spread malicious code on E-mail by using Zero-Day vulnerability |
| B. | Infected PC noticed to attacker(Automatic) |
| C. | Open backdoor to feasible the access |
| D. | Attacker seize the inside data by accessing infected PC |
| E. | Perform DDoS attack by using infected PC |

Symptoms on network when APT attack occurs can be defined as Table 3.

**Table 3: Network Symptoms**

| Step | Symptoms | Index |
|---|---|---|
| A | No specific symptoms | N/A |
| B | IN -> OUT specific IP session connection after transmitting SYN packet | N_1 |
| C | No specific symptoms | N/A |
| D | OUT->IN Well-known Port Connection | N_2 |
| E | IN->OUT bulk data transfer | N_3 |

Next, symptoms appears on the infected PC can be defined as Table 4.

**Table 4: PC Symptoms**

| Step | Symptoms | Index |
|---|---|---|
| A | No specific symptoms | N/A |
| B | IN->OUT specific IP session Establish after transmitting SYN packet | P_1 |
| C | Specific Internal port listen start. | P_2 |
| D | On P_2 state, from listen port attempts external IP connect or establish | P_3 |
| E | Increase establish count<br>Increase SYN sent count | P_4<br>P_5 |

On Table 5, rules are defined based on two symptoms. Symptom event "A" and "B" is hardly distinguishable from the normal connection behavior. Therefore, we could define that after event "C" is detection event boundary-point in example.

**Table 5: Rule of Detection (Rule_1)**

| System | A | B | C | D | E |
|---|---|---|---|---|---|
| Network | N/A | N_1 | N/A | N_2 | N_3 |
| PC | N/A | P_1 | P_2 | P_3 | P_4,P_5 |

In procedures of detection, there is no symptom defined at step “A”, so there is no *Event_a* defined. Then there are attacks, P_1 and N_1 is detected at step “B”. However event is not occurred because the threshold is set to “C”. Instead, P_2 is detected at step “C”. Obviously, *Event_a* is occurred because step “C” is the third step. It is regarded as an anomaly because specific internal port is in listen mode. Also, it is ambiguous to determine whether the event is normal. Once the event is compared to Collect Database at Detection Module, the event is regarded as abnormal if (1) N_1 and P_1 are detected in a same PC, (2) a session from internal to external for a specific IP is established, and (3) there is an internal specific port in listen mode. In that case, an alarm is occurred, and anomaly is detected based on the alarm. Therefore, if that *Event_a* is occurred and no symptom is detected in step “C”, it is trivial to confirm there is no attack.

## 4. CONCLUSIONS

In this paper, we propose SAoT, which is a new approach to detect APT attacks. This method has advantages of state-based approach such as State Transition Analysis and NetSTAT. Also this method reflects the attribution of APT to detect changes of symptoms in a timely manner. Using this method, we can detect anomalies in existing systems or networks although there are attacks using zero-day vulnerabilities. Moreover, we show the possibilities of detecting APT attacks, with some examples of APT attack scenarios. We expect that our method can be applied to existing ESMs or intrusion detection systems. However, it is required to use high-volume data processing techniques to normalize, store, and compare mass data in large-volume networks.

## ACKNOWLEDGEMENTS

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the CYBER SECURITY RESEARCH CENTER supervised by the NIPA(National IT Industry Promotion Agency), H0701-12-1001

## REFERENCES

[1] Sang-yong, Choi, “The reality of Security Monitoring and Improvement in terms of penetration” Network Security Workshop Korea, 2012. PP. 403-427

[2] Denning, D.E. , “An Intrusion-Detection Model”, Software Engineering, IEEE Transactions on, 1987, pp.222-232

[3] Rangadurai Karthick, R., Hattiwale, V.P., “Adaptive network intrusion detection system using a hybrid approach”, Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on, 2012, pp. 1-7

[4] Anita K. Jones, Robert S. Sielken, “Computer system intrusion detection: A survey” Computer Science Technical Report, 2000

[5] Kemmerer, R.A. , Vigna, G. , “Intrusion detection: a brief history and overview”, Computer(Journals), 2002, pp27-30

[6] Lunt, T.F., COMPCON Spring '89. Thirty-Fourth IEEE Computer Society International Conference: Intellectual Leverage, Digest of Papers., 1989, pp.348-353

[7] Ankita Agarwal, Sherish Johri, “Multi agent based approach for network intrusion detection using data mining concept", Journal of Global Research in Computer Science,2010, pp. 29-32

[8] Fiona Lowden Lawrence, Sanjay Kumar Sharma, “Network Intrusion detection by using Feature Reduction Technique”, International Journal of Advanced Research in Computer Science and Electronics Engineering, Vol.1, No.1(2012). pp. 27-32

[9] koral Ilgun, Rechard A. Kemmerer, “State transition analysis: a rule-based intrusion detection approach” Software Engineering, IEEE Transactions on, VOL..21, No. 3, March 1995, pp. 181-199

[10] Giovanni Vigna , Richard A. Kemmerer, “NetSTAT: A Network-based Intrusion Detection System”, Journal of Computer Security, 7(1), 1999