*Symmetry* **2014**, *6*, 997-1010; doi:10.3390/sym6040997

OPEN ACCESS

*symmetry*

**ISSN 2073-8994**

www.mdpi.com/journal/symmetry

*Article*

# MLDS: Multi-Layer Defense System for Preventing Advanced Persistent Threats

**Daesung Moon [1], Hyungjin Im [2], Jae Dong Lee [2] and Jong Hyuk Park [2,]***

[1] Network Security Research Team, Electronics and Telecommunications Research Institute, Daejeon 305-700, Korea; E-Mail: daesungm@gmail.com

[2] Department of Computer Science and Engineering and Department of Interdisciplinary Bio IT Materials, Seoul National University of Science and Technology, SeoulTech, 172 Gongreung 2-dong, Nowon-gu, Seoul 139-743, Korea; E-Mails: imhj9121@seoultech.ac.kr (H.I.); jdlee731@seoultech.ac.kr (J.D.L.)

* Author to whom correspondence should be addressed; E-Mail: jhpark1@seoultech.ac.kr; Tel.: +82-2-970-6702; Fax: +82-2-977-9441.

External Editors: Young-Sik Jeong, Laurence T. Yang, Stefanos Gritzalis

*Received: 1 October 2014; in revised form: 24 November 2014 / Accepted: 24 November 2014 / Published: 3 December 2014*

**Abstract:** Here we report on the issue of Advanced Persistent Threats (APT), which use malware for the purpose of leaking the data of large corporations and government agencies. APT attacks target systems continuously by utilizing intelligent and complex technologies. To overthrow the elaborate security network of target systems, it conducts an attack after undergoing a pre-reconnaissance phase. An APT attack causes financial loss, information leakage, *etc.* They can easily bypass the antivirus system of a target system. In this paper, we propose a Multi-Layer Defense System (MLDS) that can defend against APT. This system applies a reinforced defense system by collecting and analyzing log information and various information from devices, by installing the agent on the network appliance, server and end-user. It also discusses how to detect an APT attack when one cannot block the initial intrusion while continuing to conduct other activities. Thus, this system is able to minimize the possibility of initial intrusion and damages of the system by promptly responding through rapid detection of an attack when the target system is attacked.

**Keywords:** APT attack; defense in depth; multi-layer defense; intrusion detection

## 1. Introduction

The rise in the use of computers and the growth of the internet brought about cyber-crimes [1]. In an attempt to identify ways to prevent cyber-crimes, many studies have been conducted on security related systems. Meanwhile, cyber-attacks have become more sophisticated than ever. In response to the developments, the way the attack and defense between cyber-crimes and information security technologies occur have become increasingly complicated [2]. One of the most complex and advanced cyber-attacks in recent years is the Advanced Persistent Threat (APT), which attacks corporations and government agencies. An APT attack is one of the major cyber-attacks in addition to targeted attack. It is a module to use all the things known about the attack [3]. The some of the prominent cases for APT attack include Stuxnet, Duqu, Red October, Mask, *etc.*, and each of these attacks had a different target and purpose. An APT attack sets a target, unlike malwares such as Bot, Trojan and Worm, and conducts a sophisticated attack continuously. Those conventional attacks target unspecified individuals as relevant symptoms appear immediately with a single attack. On the other hand, APT attack avoids detection and leaks the information the attacker wants. In general, APT attacks aim to destruct industrial infrastructure and collect important corporate information. In addition, it finds and detects the vulnerable aspects of target companies and attempts an initial intrusion by analyzing technical information and personnel information with the aim of finding a target system along with the information of a target corporation, such as the corporation's objectives and antivirus capabilities based on a social engineering technique. As a result, it is difficult to defend initial intrusion with conventional antivirus systems [4]. Moreover, malicious software, which infiltrates into the target corporation, contains a variety of attack modules to achieve an objective. Thus, APT attack poses a huge risk and requires a new defense technique [5].

In this paper, we propose a Multi-Layer Defense System (MLDS) to monitor malware through collecting the log data, setting data and traffic data of various devices by installing network application, server and agent at an end-user in order to defend from continuous and elaborate APT attacks. We define the MLDS as a defense system which can prevent APT attack across multiple layers of TCP/IP. As a result, it can minimize the possibility of intrusion through malware and damages by promptly responding to a detected attack.

This paper consists of a total of 4 sections. In Section 2, Related Works, we discuss APT attack cases, attack phases and malware detection technologies. Section 3 discusses the proposed system and presents a service scenario. Finally, Section 4 presents concluding remarks and briefly discusses future works.

## 2. Related Works

In this section, related works are derived the essential elements of APT attack through a case analysis. In addition, we discuss the detection techniques for malware. By doing so, it presents step-by-step analysis for APT attacks.

### *2.1. Attack Cases*

APT attacks achieve their objective by using a variety of malware techniques. It uses an antivirus detection bypassing module, an information collection module, an administrator authority acquisition module, *etc.* Herein, a variety of zero-day vulnerabilities are used [6].

Stuxnet targeted Iran’s nuclear power plant and inserted a malicious program into the Programmable Logic Controller (PLC) of the reactor by infecting the Supervisory Control and Data Acquisition (SCADA) system. This destroyed the uranium enrichment program and interfered with the production of nuclear weapons. Since the component allows for 10 self-replications, it can spread very easily. The vulnerabilities targeted include Window shell LNK vulnerability, Window server service vulnerability, Window printer spooler vulnerability, and shared network service vulnerability. Moreover, Stuxnet made use of the vulnerability that one could transmit malware even to a system with a separate network by using USB, mobile storage device, in an attack. Stuxnet has not performed any particular activity for those other than the target systems and also used Rootkit in order to erase its existence.

Duqu has been known as a variant of Stuxnet. However, Stuxnet and Duqu have a different purpose. While Stuxnet destroys and breaks down a target system, Duqu aims to collect and acquire the information of a target system with the goal of leaking information. It attempted initial intrusion through spear phishing that would attach an infected Microsoft Office document, and conducted communication with C&C server by installing backdoor. Key-logging component collects such important data as password and any collected data would be used for acquiring an authority to access other systems in the network. In addition, Duqu deletes itself by using Rootkit 36 days after it was installed in order to remove any trace of intrusion.

Red October has targeted energy and nuclear-related facilities, as well as the aerospace industry of Eastern Europe and Central Asia. It invaded the target systems by sending infected Microsoft Office product files as an attachment. The attacker intruded on the other confidential systems in the same network based on the initially leaked information. Red October leaked the data of smart devices such as iPhone, Nokia and Windows Mobile from phones and portable disks. It brought about more damage by recovering and leaking the deleted files in the portable drivers. Moreover, it leaked even the setting data of network equipment of the target corporations such as routers and switches.

Mask contains Rootkit as APT attack that adapts to various environments and it is operated in Window 32bit, Window 64bit, Mac OS X and Linux environments. Moreover, it was used in the following smeared device OSs: Android and iPad/iPhone. Mask initially intruded the target system through spear phishing that used the vulnerabilities of Adobe Flash. The installed malware finds the antivirus in a target system and mimics the operation in order to bypass detection. Furthermore, it even changed the file names. Mask collected the network traffic data, keystrokes, Skype conversations, WiFi traffic analysis data, patch information from Nokia, and log data of screen captures.

### *2.2. Steps Followed by an APT Attack*

An APT attack uses a different method depending on the purpose. However, there are several common events [7]. In this section, we discuss the seven attack steps of APT and what each step implies.

**Step 1. Targeted System Recognition**: This step is to collect data on the target system before the attack. It is technical methods include examining ports allowing for an intrusion through port scanning. Its non-technical method is used as a tool for spear phishing and USB infection by acquiring the email address of a personnel with access to security information or a personnel with work assignments through social engineering.

**Step 2. Initial Intrusion**: This is the phase in which an actual APT attacks are conducted. That is, the target system comes under attack. The initial intrusion step attempts a spear phishing attack based on information collected from the targeted system recognition, which is the previous step. In particular, personnel managing a target system such as security personnel and network personnel would be the main target. An attacker transmits mainly websites or documents infected with malware to a target as an attachment and induces a target to click these websites or documents.

**Step 3. Backdoor Establishment**: An attacker creates a backdoor to access a system more easily later after the initial intrusion. An attacker makes sure that the C&C server created by himself and a target system will exchange information with each other directly. In addition, it allows an attacker to easily steal information by bypassing the surveillance network of antivirus products through exchanging the encrypted data that are disguised as normal data to the C&C server.

**Step 4. Internal Recognition**: In this step, an attacker attempts network scanning through a trusted computer. It discloses network structure mainly through the basic commands of Windows to make it hard to detect an attack. The information that can be obtained from the outside is limited; however, it is relatively convenient to obtain information from the infected inside. Thus, an attacker conducts reconnaissance through the internal network to which a target belongs. It obtains the network information of a target system through the trusted relationship.

**Step 5. Metastasis**: In this step, an attacker transmits malware to an end-point having obtained important information by using data obtained through internal reconnaissance. Alternatively, it infects all end-points using the same network. With this step as the last preparation, an attacker completes its preparation for leaking desired information.

**Step 6. Mission Complete**: This is the step in which an attacker successfully performs targeted actions. The purpose of an attacker is to destroy a system and leak secret information. An attacker uses such techniques as encryption, file compression, file splitting, *etc.* for data in order not to be detected by antivirus.

**Step 7. Hiding**: An APT attack does not reveal itself by hiding malicious software while there is no particular event for continuous attack. Some APT attacks contain Rootkit and acquire the root authority of a system. In addition, they hide the malware using Rootkit [8].

### *2.3. Malware Detection Techniques*

In this subsection, we discuss the techniques used to detect malware and the measures for each step of an APT attack [9,10].

Intrusion Detection System (IDS) is the system to detect malicious activities that take place in a computer. It is a system applied by a variety of intrusion detection technologies [11,12]. Thus, it is divided into host based intrusion detection system and network based intrusion detection system. The next represent the two types of IDS related APT actions.

- **HIDS (Host-Based Intrusion Detection Systems) Related APT Actions:** HIDS is the intrusion detection system for the host such as personal computer and server computer. HIDS analyzes the resources inside a computer such as log, file, folder, service and monitors and analyzes any trace of infection. In the case of an APT attack, it infects a system using an infected attachment; thus, the role of HIDS becomes very important. The detection technique of HIDS differs for each product; however, it saves the hash value of a file in order to confirm the presence of infection for a given file and checks the changes of a file periodically. In addition, it detects and analyzes any abnormal operation patterns by monitoring the system call or vector table provided by the operating system [13]. Most HIDS is implemented as a program in the form of an agent inside a computer [14].

- **NIDS (Network-Based Intrusion Detection Systems) Related APT Actions:** Unlike HIDS, it focuses on external interface. It monitors the presence of malicious activities that take place inside network through an abnormality of network traffic. NIDS detects service refusal attack, port scanning, packing sniping, *etc.* NIDS is required for all the steps excluding the hiding step of APT attack. However, APT attacks use complex malware that allows them to bypass the NIDS system of the target; thus, technologies to better defend from the attacks are continuously carried out.

An APT attack uses sophisticated and complex malware. To defend against an APT attack, it is imperative to use a variety of detection technologies. In the following section, we discuss the detection technologies used in IDS and NIDS. The next list describes three types of detection techniques.

- **Signature Detection:** One finds malware by distinguishing the existing signature data when a new file or traffic comes into a system as configuring the signature value such as code and pattern of malware, which was previously discovered by the detection method generated from the file that intruded the system as it was infected with malware. However, an APT attack has a different form from the existing malware since it uses advanced malware. Thus, the signature based detection has some limitation for malware detection. However, its rate for wrong detection is low; thus, other detection technologies are not used.
- **Virtual Sandbox Detection:** This is the technology to detect malware dynamically [15]. One determines the presence of malware from the information collected through the execution of a file. In general, Sandbox is known as an application emulator and it detects the presence of mali cious activities by executing the application in virtual space. It is used mainly for zero-day attack or transformed malware detection [16]. An APT attack intrudes a target system using zero-day vulnerability; thus, detection using virtual sandbox shall be mandatory for defending APT.
- **Machine Learning:** In recent years, the degree of complexity of malicious software has been on the rise. One of the most serious and heinous of the malicious software attacks can be said to be APT attacks. However, in fact, it is rare to see an APT attack that uses conventional malware. There are a number of variables even for those who attack the same vulnerability. As such, there have been many studies on the learning based detection technique to defend an attack of numerous variables and zero-day for vulnerabilities by learning malware dynamically [17–20]. The learning technique based algorithms include neural network, SVM (support vector machines-based), decision tree, Bayesian network, *etc.*

### *2.4. Analyses Information for Each Step of an APT Attack*

In this subsection, we present the main cases of an APT attack and the correlation with APT steps. In addition, it discusses the attack preference and the method for analysis, detection and prevention for each step. Table 1 shows whether one conducts APT attack steps for each case.

**Table 1.** Presence of attack steps for each case.

| **Type** | **Step of Attack** | | | | | | |
|---|---|---|---|---|---|---|---|
| | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
| Stuxnet | ● | ● | ● | ● | ● | ● | ● |
| Duqu | ● | ● | ● | ● | ○ | ● | ○ |
| Red Oct. | ● | ● | ● | ● | ○ | ● | ○ |
| Mask | ● | ● | ● | ● | ○ | ● | ● |

● Perform, ○ Nothing.

All of these cases conduct Step 4 and Step 5 from Attack 1. However, Step 5 and Step 7 show a different result for each APT case. In the case of Stuxnet, it shows a strong metastasis technique through USB, a network service, printer spooler, *etc.* and successfully destroys a system, which is the goal of an attack. However, Duqu, Red October and Mask do not have a specific metastasis technique. Moreover, Stuxnet conducts malware including RootKit. As a result, it is able to continue to attack the target system by hiding itself. In the case of Duqu, it does not attack continuously since it destroys and hides itself 36 days after it was installed. In addition, Mask concealed the existence of malware by mimicking the act of antivirus inside the target system.

We derived the detection method and the defense method as shown in Table 2 through the attack method for each APT attack step. The first row shows the steps of APT attack and the second row lists the attack methods used in each step. The third row lists the method and techniques to detect the attack methods for each step. The fourth row lists the techniques to defend against each step [21,22].

**Table 2.** Analysis for each step of APT attack.

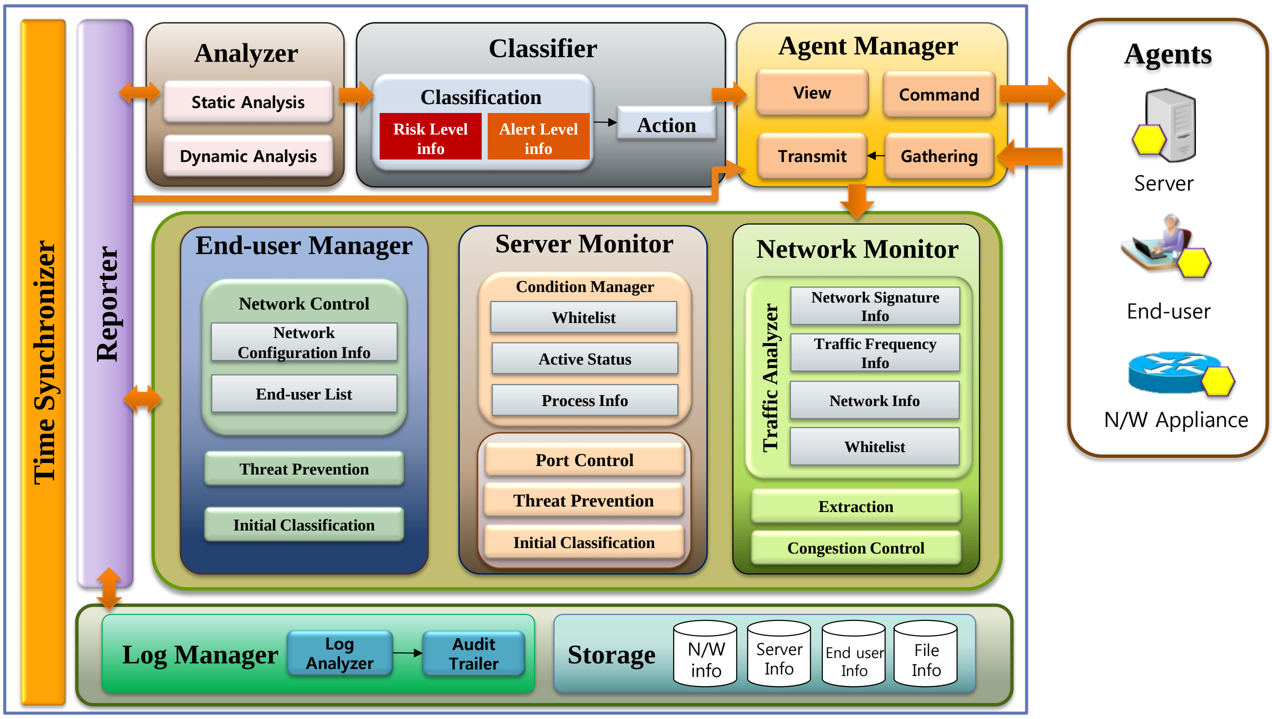
| **Step** | **Attack Method** | **Detection Method** | **Prevention Method** |
|---|---|---|---|
| 1 | Port Scan, Social Engineering | NIDS, Router logs, Web logs, Firewall logs | Firewall ACL, Security Education |
| 2 | Spear Phishing, Infection USB, Infected Website, Watering Hole | HIDS, NIDS, Mail Filter, Web Application Filter | NIPS, Application Firewall |
| 3 | Rootkit, C&C Server, | HIDS, NIDS, Antivirus | Firewall ACL, NIPS |
| 4 | Malware, Botnet | NIDS, HIDS | Network Segmentation |
| 5 | Malware, Botnet | HIDS, NIDS, Antivirus, Logging, Audit trail | NIPS |
| 6 | Malware, Botnet | NIDS, HIDS, Antivirus, Logging, Audit trail | Firewall ACL, Network Segmentation |
| 7 | Rootkit, Altering Log Records, Altering File Dates | Virus scanners, Traffic lensors | HIPS, NIPS |

## 3. MLDS

### *3.1. Architecture*

MLDS prevents malware infection for a target system by collecting and analyzing information from network appliance, server and end-user. The proposed MLDS consists of a total of 8 components (Classifier, Analyzer, Agent Manager, Server Monitor, End-user Manager, Network Monitor, Log Manager and Storage). Figure 1 shows the architecture of the MLDS.

**Figure 1.** Architecture of MLDS.



The *Time Synchronizer* module provides synchronization function for time. It provides reliability for log data by synchronizing time with components.

The *Reporter* module serves the communication between components. This module receives the data from various components such as the Analyzer, Log Manager, Storage, End-user Manager, Server Monitor and Network Monitor, then the information is passed to the other components.

The *Analyzer* component analyzes suspicious malware transferred by End-user Manager, Server Monitor and Network Monitor with aid of two modules. The Static Analysis module includes the signature and hash value of a file. The Dynamic Analysis module includes the sandbox, behavior based detection technique, *etc.* This component precisely analyzes suspicious files as malware translated by the End-user Manager, Network Monitor and Server Monitor. If the transferred file or traffic is malicious, analysis information of file will be sent to the Classifier.

The *Classifier* component is composed of two modules, the Classification module and the Action module. The Classification module has Risk Level Information and Alert Level Information. Risk Level Information means risk information that has already been detected as having malware. Alert Level information means warning information depending on location (server, end-user, or network) in which the malware attacks will be posted. Two kinds of the information help the Classification module to efficiently classify the malwares. The Classification module receives analysis information about the malwares from Analyzer component, then classifies the risk level and alert level based on behavior of the malwares. The Action module instructs the reaction and alerts the Agent Manager using the classified information.

The *Agent Manager* component includes four modules, such as View, Command, Gathering and Transmit. It manages the agents to run network appliances, servers and end-users that are present outside. The View module allows a system administrator to check the current state of the MLDS. The Command module transfers the command which is received through the Action module or the Reporter module and sent to agents. The Gathering module collects information coming from each agent. In addition, the Transmit module sends the gathered information to the End-user Manager, Server Monitor, and Network Monitor components.

The *End-user Manager* component is composed of Network Control, Threat Prevention, and the Initial Classification module. The Network Control module has a Network Configuration Information and a list of end users. If malware is detected by the end-user, Network Control module will exclude infected end-user on network. The Threat Prevention module automatically updates in order to maintain the latest version of the application in end-users. The Initial Classification module observes the running processes in the end-users. If a system configuration is changed or altered anomaly, this information will be sent to the Analyzer component.

The *Server Monitor* relies on following modules: Condition Manager, Port Control, Threat Prevention and Initial Classification. The Condition Manager module can check the state of a server through process info, active status and whitelist. Port Control module control the connectable port to the server. Initial Classification and Threat Prevention can conduct the same operation as module of the End-user Manager.

The *Network Monitor* consists of three modules. The Traffic Analyzer module can analyze traffic with information such as the network signature, traffic frequency, network information and whitelist. Network signature information stores the signature of malicious network traffics. Traffic frequency information means the pre-determined maximum amount of traffic for each piece of network equipment. In addition, the whitelist has information about approved traffic. Network information has the network configuration and Access Control List (ACL). The Network Monitor component can extract traffics deemed malicious file through the Extraction module and it can also conduct network congestion control through Network Congestion module.

The *Log Manager* contains the Audit Trail and Log Analyzer module. The Log Analyzer module receives the log, whose abnormal acts are detected by the Server Monitor, End-user Manager and Network Monitor Component. The Audit Trail module finds an end-user where malicious software is installed through analyzed log.

*Storage* is the module to save the information about network, server and end-user from network monitor, server monitor and end-user manager. It helps in analyzing the malware by transmitting the saved information to other modules when malicious software is detected later.

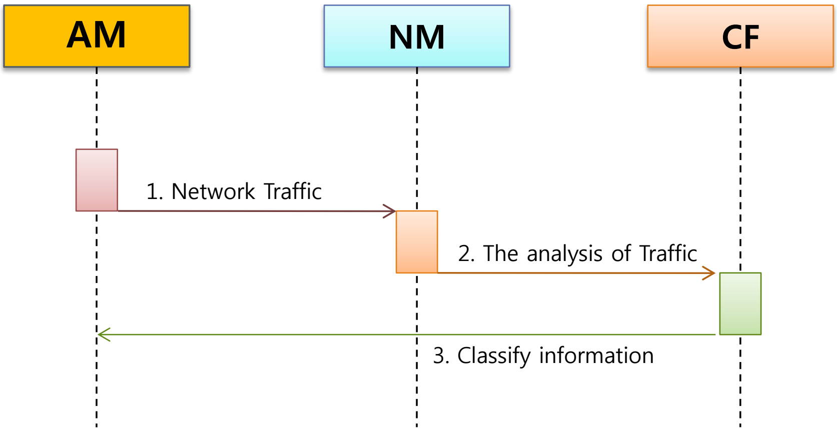
### *3.2. Service Scenario*

In this section, we discuss a service scenario for the proposed MLDS. The marks used in the service scenario proposed in this section are as shown in Table 3. In addition, Figure 2 represents the scenario for detecting APT attack as to 1, 3, 4, 5, 6 among the steps of APT attack mentioned in Section 2.2.

**Table 3.** Definition of acronyms.

| Term | Explanation |
|---|---|
| CF | Classifier |
| AM | Agent Manager |
| NM | Network Monitor |
| EM | End-user Manager |
| AZ | Analyzer |

**Figure 2.** Detection Scenario of Step 1, 3, 4, 5, 6.



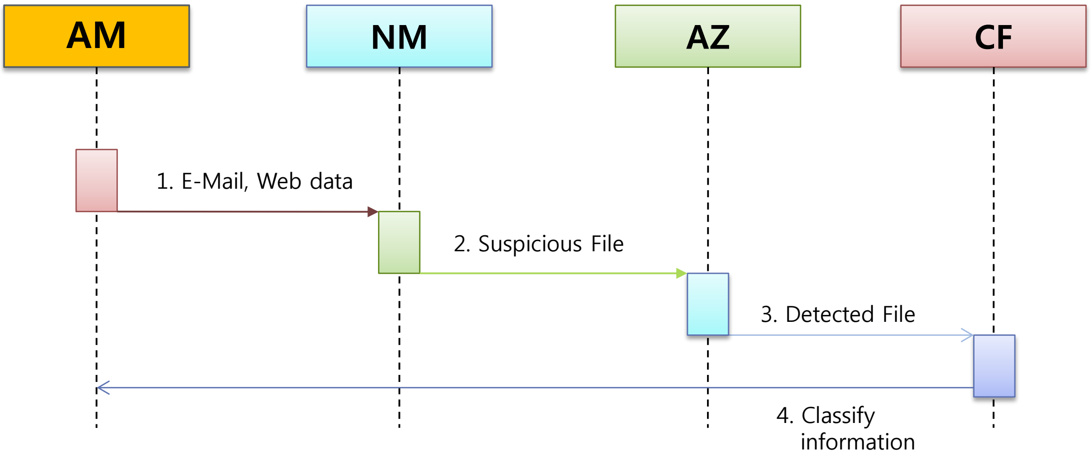
Step 1 (Detection Scenario of Step 1, 3, 4, 5, 6). See Figure 2.

1. AM → NM: sending an network traffic
   AM collects data from the network agent and transmits it to the network manager.
2. NM: analyzing the traffic
   NM analyzes traffic through various module such as network signature, traffic frequency and whitelisting.
3. NM → CF: sending the suspicious traffic
   When suspicious traffic is detected by the traffic analysis module of NM, it passes to the CF.
4. CF: classify suspicious
   CF classifies malicious traffics in accordance with risk level through Classify module.
5. CF → AM: classify information
   When certain traffic is classified as malicious by CF, AM is notified to protect the system from APT attacks.

Step 2 (Detection Scenario of Step 3). See Figure 3.

1. AM → NM: sending E-mail and Web Traffic
   Emails from each agent and network traffic data are delivered to AM as well.
2. NM → AZ: sending a suspicious file
   NM analyzes the traffic. When it detects any abnormality, the network stream extracts the file. It then transfers the file to AZ.
3. AZ: analyzing the file
   Basically, AZ determines if certain files are malicious by means of the static analysis module and then enhances the efficiency of detection by means of the dynamic analysis module.
4. AZ → CF: sending an detected file
   If a file is deemed as malware, it transmits the information to CF.
5. CF: classifying the file
   CF classifies the files in accordance with the detected risk level.
6. CF → AM: sending information
   When certain classified traffic is found dangerous, AM is notified to protect the system from APT attacks.

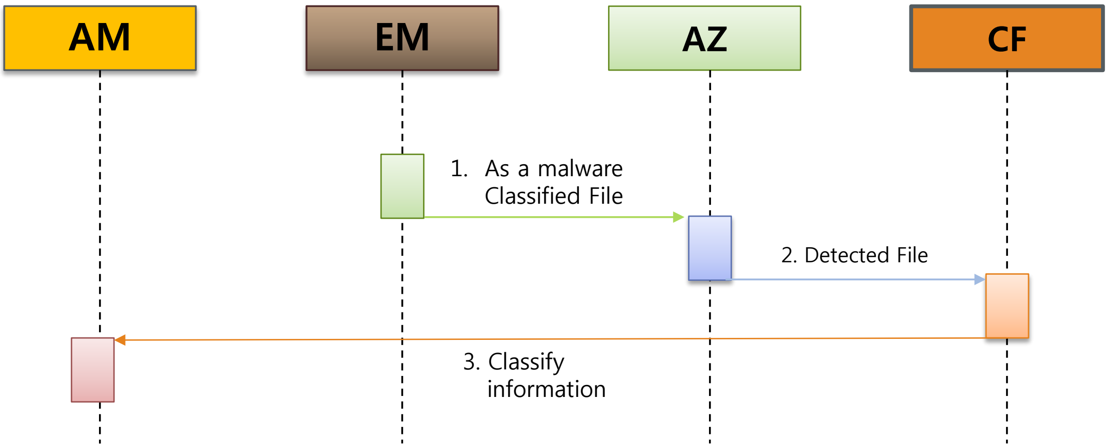
**Figure 3.** Detection Scenario of Step 2.



Step 3 (Detection Scenario of Step 3). See Figure 4.

1. EM: file analysis
   EM classifies files suspicious of malicious software based on the files transmitted by an end-user's agent.
2. EM → AZ: sending a classified file
   EM classifies the files saved by users primarily through the Initial Classifier.
3. AZ → CF: sending a detected file
   In the case of those files that are detected for abnormal act, it sends these files to AZ to analyze the presence of malicious files.
4. CF → AM: sending classified information
   CF classifies the risk level of analyzed files and sends this information to AM to synchronize with an end-user.

**Figure 4.** Detection Scenario of Step 7.



### *3.3. Case Studies*

In this subsection, we present how to defend an intrusion of APT attack using MLDS. Case 1 shows the prevention method for end-user from spear phishing at initial intrusion (Figure 5). Case 2 represents removal technique for end-user from malware infection and transition through USB (Figure 6).

**Figure 5.** End-User Infection through Spear Phishing.

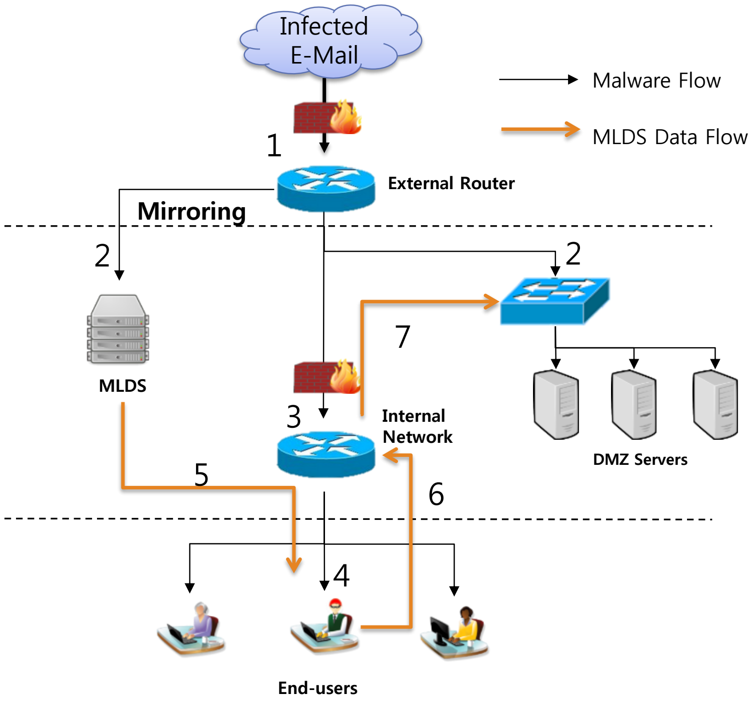
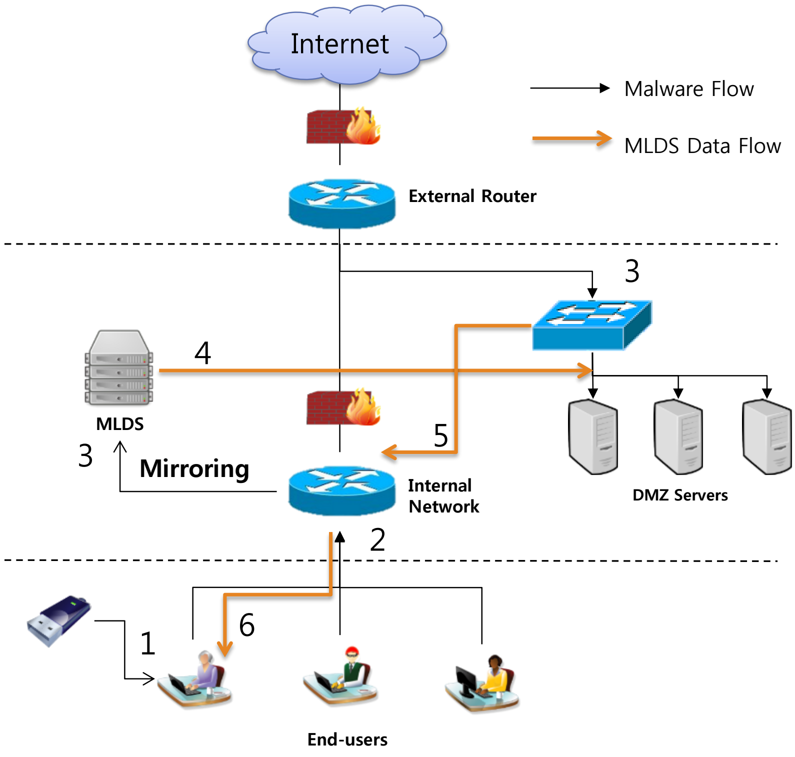


**Figure 6.** Prevention method from Infection and Metastasis through USB.



**Case 1. Prevention method for end-user form Spear phishing at initial intrusion**

At initial step indicates the flow of infected email to an external router. When an email reaches the inbox it is transmitted from the external router to the DMZ server and mirrored in MLDS simultaneously. The mirrored data starts in advance to conduct analysis in MLDS. At this point, when an end-user tries to read the email, the email will be transmitted to the internal router and then delivered to the end-user. Consequently, the end-user downloads the malware in attached files from email or by visiting the linked web pages. Each time the email is transferred, the log is delivered to the MLDS and continues to conduct the analysis. At this point, if malicious software detects malware file in the end-user, then the MLDS informs the presence of an infection to the end-user and shows the trace of malware through the log information. MLDS removes malware from all devices located in infected path for preventing additional infection through tracking down the infected path of the user. Figure 5 represents the defense scenario of MLDS as to the initial intrusion using spear phishing.

**Case 2. Prevention method from Infection and Metastasis through USB**

When an end-user is infected through USB, malware attempts to infect other end-users in the system. When malware identifies the presence of DMZ server through internal recon, it will infect the DMZ server. At this point, if malicious software is defected in the DMZ server, MLDS informs the DMZ server about the presence of an infection and analyzes the log information to identify the migration path of malicious software. It removes malicious software installed at each end-point based the migration path of malicious software. Figure 6 represents the defense scenario of MLDS for the infection and transition of an end-user through USB.

## 4. Conclusions

Various cyber threats have brought about numerous damages ranging from privacy information leakage to financial loss, to leakage of confidential corporate information. Of the cyber threats, APT attacks are particularly known for attacking continuously until they acquire long-time access authority or leak information by successfully intruding specific organizations or institutes. They many challenges for security, since they conduct an attack after sufficiently analyzing the vulnerabilities of a target system.

In this paper, we discussed the steps of attack through the cases of APT attack and proposed the need for an in-depth detection system. In this paper, we proposed the Multi-Layer Defense System (MLDS), which can conduct defense in depth by analyzing information of network, server, end-user, log, *etc.*, through installing agents at network appliance, server and end-user. As a result, MLDS detects APT attacks from various layers to enhance the performance. In addition, when the system is affected by APT attacks, MDLS minimizes the damage. In the future, it may be necessary to examine analysis algorithms used for file analysis and traffic analysis of the suggested system to detect malware accurately.

## Acknowledgments

This work was sponsored by the Korea Ministry of Science, ICT and Future Plan under Cyber targeted attack recognition and trace-back technology (SINBAPT) Project [13-921-06-001].

## Author Contributions

Daesung Moon: design of the total system; Hyungjin Im: mainly writing; Jae Dong Lee: research for the related works, analyzing and improving for the proposed system; Jong Hyuk Park: total supervision for the paper work, review and comments, *etc.*

## Conflicts of Interest

The authors declare no conflict of interest.

## References

1. Julian, J.-J.; Nepal, S. A survey of emerging threats in cybersecurity. *J. Comput. Syst. Sci.* **2014**, *80*, 973–993.
2. Jingle, I.D.J.; Rajsingh, E.B. ColShield: An effective and collaborative protection shield for the detection and prevention of collaborative flooding of DDoS attacks in wireless mesh networks. *Hum.-Centric Comput. Inf. Sci.* **2014**, *8*, doi:10.1186/s13673-014-0008-8.
3. Feng, L.; Liao, X.; Han, Q.; Li, H. Dynamical analysis and control strategies on malware propagation model. *Appl. Math. Model.* **2013**, *37*, 8225–8236.
4. Hoang, T.; Nguyen, T.; Luong, C.; Do, S.; Choi, D. Adaptive cross-device gait recognition using a mobile accelerometer. *J. Inf. Process. Syst.* **2013**, *9*, 333–348.
5. Misra, A.K.; Verma, M.; Sharma, A. Capturing the interplay between malware and anti-malware in a computer network. *Appl. Math. Comput.* **2014**, *229*, 340–349.
6. Xenakis, C.; Ntantogian, C. An advanced persistent threat in 3G networks: Attacking the home network from roaming networks. *Comput. Secur.* **2014**, *40*, 84–94.
7. Mustafa, T. Malicious data leak prevention and purposeful evasion attacks: An approach to Advanced Persistent Threat (APT) management. In Proceedings of the Saudi International Electronics, Communications and Photonics Conference (SIECPC), Riyadh, Saudi Arabia, 27–30 April 2013; pp. 1–5.
8. Lu, H.; Wang, X.; Zhao, B.; Wang, F.; Su, J. ENDMal: An anti-obfuscation and collaborative malware detection system using syscall sequences. *Math. Comput. Model.* **2013**, *58*, 1140–1154.
9. Sheen, S.; Anitha, R.; Sirisha, P. Malware detection by prunng of parallel ensembles using harmony Search. *Pattern Recognit. Lett.* **2013**, *34*, 1140–1154.
10. Modi, C.; Patel, D.; Borisaniya, B.; Patel, H.; Patel, A.; Rajarajan, M. A survey of intrusion detection techniques in Cloud. *J. Netw. Comput. Appl.* **2013**, *36*, 42–57.
11. Liu, G.; Wang, X. Homomorphic subspace MAC scheme for secure network coding. *ETRI J.* **2013**, *35*, 173–176.
12. Li, X.; Wang, X.; Xu, X.; Jin, L. A distributed implementation algorithm for physical layer security based on untrusted relay cooperation and artificial noise. *ETRI J.* **2014**, *36*, 183–186.
13. Santos, I.; Brezo, F.; Ugarte-Pedrero, X.; Bringas, P.G. Opcode sequences as representation of executables for data-mining-based unknown malware detection. *Inf. Sci.* **2013**, *231*, 64–82.
14. Qin, Y.; Tong, W.; Liu, J.; Zhu, Z. SmSD:A smart secure deletion scheme for SSDs. *J. Converg.* **2013**, *4*, 30–35.

15. Younghee, P.; Reeves, D.S.; Stamp, M. Deriving common malware behavior through graph clustering. *Comput. Secur.* **2012**, *39*, 419–430.
16. Yong, Q.; He, J.; Yang, Y.; Ji, L. Analyzing malware by abstracting the frequent itemsets in API call sequences. In Proceedings of the 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Victoria, Australia, 16–18 July 2013; pp. 265–270.
17. Abuzaid, A.M.; Saudi, M.M.; Taib, B.M.; Zul Hilmi, A. An efficient trojan horse classification (ETC), IJCSI. *Int. J. Comput. Sci. Issues* **2013**, *10*, 96–103.
18. Nissim, N.; Moskovitch, R.; Rokach, L.; Elovici, Y. Novel active learning methods for enhanced PC malware detection in windows OS. *Expert Syst. Appl.* **2014**, *41*, 5843–5857.
19. Malkawi, M.; Murad, O. Artificial neuro fuzzy logic system for detecting human emotions. *Hum.-Centric Comput. Inf. Sci.* **2013**, doi:10.1186/2192-1962-3-3.
20. Verma, O.P.; Jain, V.; Gumber, R. Simple fuzzy rule based edge detection. *J. Inf. Process. Syst.* **2013**, *9*, 575–591.
21. Rasheed, H. Data and infrastructure security auditing in cloud computing environments. *Int. J. Inf. Manag.* **2014**, *34*, 364–368.
22. Jouini, M.; Rabai, L.B.A.; Aissa, A.B. Classification of security threats in information systems. In Proceedings of the 5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014) Procedia Computer Science, Hasselt, Belgium, 2–5 June 2014; pp. 489–496.