PAPER Special Section on Security, Privacy and Anonymity in Computation, Communication and Storage Systems

Modeling Attack Process of Advanced Persistent Threat Using Network Evolution**

Weina NIU[†], ***Nonmember***, **Xiaosong ZHANG**[†*a)], ***Member***, **Guowu YANG**[†], **Ruidong CHEN**[†], ***and*** **Dong WANG**[†], ***Nonmembers***

SUMMARY Advanced Persistent Threat (APT) is one of the most serious network attacks that occurred in cyberspace due to sophisticated techniques and deep concealment. Modeling APT attack process can facilitate APT analysis, detection, and prediction. However, current techniques focus on modeling known attacks, which neither reflect APT attack dynamically nor take human factors into considerations. In order to overcome this limitation, we propose a Targeted Complex Attack Network (TCAN) model for APT attack process based on dynamic attack graph and network evolution. Compared with current models, our model addresses human factors by conducting a two-layer network structure. Meanwhile, we present a stochastic model based on states change in the target network to specify nodes involved in the procedure of this APT. Besides, our model adopts time domain to expand the traditional attack graph into dynamic attack network. Our model is featured by flexibility, which is proven through changing the related parameters. In addition, we propose dynamic evolution rules based on complex network theory and characteristics of the actual attack scenarios. Finally, we elaborate a procedure to add nodes by a matrix operation. The simulation results show that our model can model the process of attack effectively.

key words: *attack process modeling, APT, TCAN, complex network theory*

1. Introduction

Network attacks have evolved from viruses and worms to advanced attacks, multiple attacks, coordinated attacks and customized attacks nowadays [1]. Since Advanced Persistent Threat (APT) [2]–[4] coning as a new concept by the US Air Force in 2006 [5], it is flourishing as a security marketing buzzword in network security. According to statistics, APT attackers hit Google's Gmail service up to thirty days in 2009 [6]. The private key server of RSA was compromised in 2011 [7]. The source code of Kappa and Comodo were stolen in 2012 [8]. Banks and three television stations in Korea encountered organized attacks with the typical characteristics of APT in 2013 [9]. In the early days, APT was a specialized term in the military. There are many different opinions in defining APT. Currently, the US National Institute of Standards and Technology [10] acknowledge that APT attack is launched by high-skilled and well-funded attackers. Such attack comprises multiple attack vectors to exfiltrate information or sabotage the infrastructures.

In the recent years, APT attack detection and analysis attract great attention in the academic field and industry due to highly customized samples and deep concealment [11]–[13]. Several countries regard defense against APT attacks as an essential part of the national network security. The United States Department of Defense demonstrably pointed out that the detection and defense of APT attacks are the most fundamental part of the whole risk management links. Explaining, detecting and predicting APT attacks are indispensable to model the procedure of APT attacks. Network attack modeling has been studied for several years. Many models have been proposed so far, such as attack tree [14], attack graph [15] and attack net [16]. Leaves or branches of attack trees are linked to AND or OR gates, which indicates that the attack methods like Attack pyramid describes an attack path going across different environments of the organization. Attack graphs capture changes over time, which combines vulnerabilities correlation. Meanwhile, attack networks express state changes during attack procedure (generally refers to the change of attack phase). However, these methods neither present dynamic change of attack nodes nor consider human factors, which is essential in actual APT attack.

The main goal of this paper is to identify the hosts that definitely involved in the attack process. To overcome current limitations, we propose a network-evolution-based approach to model the attack process of APT attacks. In our model, the time domain and human factors are taken into considerations.

In our model, nodes and edges of traditional attack graph are redefined. Communication-contact network [17] is introduced to indicate the influence of social engineering, since social engineering can be exploited in each APT case. By analyzing the network formation, our approach can simulate popular APT cases. The main contributions of this paper are summarized as follows:

(1) We define stochastic model to characterize node status changes in the target network, which indicates nodes participating in the attack process.

(2) We propose a Targeted Complex Attack Network

Manuscript received December 13, 2016.
Manuscript revised May 24, 2017.
Manuscript publicized July 21, 2017.

[†]The authors are with School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, China.
[*]Presently, the author is with Center for Cyber Security, University of Electronic Science and Technology of China, Chengdu, China.

[**]This paper was presented at the 9th International Conference on Security, Privacy and Anonymity in Computation, Communication and Storage (SpaCCS 2016).

a) E-mail: johnsonzxs@uestc.edu.cn

DOI: 10.1587/transinf.2016INP0007

(TCAN) model to visualize the attack process of APT based on dynamic attack graph and use trust relationship to conduct an attack.

(3) We implement simulation under network environment of several thousand hosts to demonstrate the feasibility of the proposed approach.

To our knowledge, this paper is the first model focusing on attack network to help security analysis and understand APT attack mechanism. Few results exhibit attack process from the angle of network attack formation. This work considers suspicious hosts involved in APT-related activities by social engineering and network penetration. Moreover, our model demonstrates the attack situation dynamically based on network evolution. In the simulated experiment, scale-free network is used to express network structure.

The remaining of this paper is organized as follows: Sect. 2 gives an overview of the related work; Sect. 3 describes the APT campaigns; Sect. 4 represents how the TCAN is generated; Sect. 5 shows the preliminary experimental results; discussion and conclusion are summarized in Sect. 6.

2. Related Work

In recent years, we have witnessed a number of network attack models. However, there are few on modeling of APT attacks [18]–[22]. These works mainly depend on five sophisticated models of network attack:

1) Attack tree model

The attack pyramid model on attack tree [18] introduced the concept of domain. The ultimate attack goal is placed at the top of the pyramid. Each node represents an event recorded in the monitored environment, including three types: candidate event, suspicious event and attack event. The attack pyramid has six levels corresponding to six stages of the attack process at a coarse-grained level. Moreover, each level contains multiple planes indicating the environment where attack evolves: user, network, application, physical environment, and etc. This model is used to provide guidelines for detecting APT using attributes and time. However, attack pyramid model is hard to represent attack states.

2) Attack net model

Zhao et al. [19] proposed EPANM model by extending the structure of classical Petri net. EPANM extracted attack object O from state space set S and divided transition node set into environment condition set, vulnerability utilization set, penetration expansion set, attack intention set, denoted as C, V, E, I, respectively. Thus, this model is expressed by a six-tuple {O, C, V, E, I, S}. They did scene hierarchical analysis first and then associated scenes with extended Petri net next. Finally, they generated formal expression using state change as $O \times C \times V \times E \times I \times S \longrightarrow S$. This model combines attack scene, attack process, and state space. However, EPANM model has poor adaptability because it limits attacks process to eight states: information collection, vulnerability obtaining, vulnerability handling, intranet entrance, authorization getting, expansion scanning, threat processing, and behavior concealing. Thus, this model cannot reveal attack process dynamically.

3) Attack kill chain model

APT attack is a multi-staged attack with a similar mechanism with APT attack. The attack kill chain model [20], [21] can describe the phases of an APT attack based on the concept of intrusion kill chain [23]. Chen et al. [20] divided APT attack into six stages: reconnaissance and weaponization, delivery, initial intrusion, command and control, lateral movement, and data exfiltration. But this model lacks the description of the state change.

4) Markov model

The phases of exfiltration APT maps into a cyber kill chain. The impact on the target network from the next attack action can be measured using a probability. Thus, a novel Markov Multi-Phase Transferable Belief Model (MM-TBM) is proposed. It adopts a tree-structure to visually express the phases of an exfiltration APT [21]. There are five levels: insertion, exploitation, command and control, exfiltration, especially. The attacker's behaviors are expressed by a set T={N, B, P}, where N indicates nodes sets of five levels; N_{ij} indicates the i_{th} node in level i; B indicates branches sets; P indicates the prior beliefs for the branches of B. This model is to hypothesize assessment and conflicts management as well as guide the network administrator to detect APT attack early. However, this model cannot represent a state change in the procedure of APT attacks.

5) Game model

Fang et al. [22] used a 5-tuple (C, T, ⟨P1, P2⟩, ⟨$S1, S2$⟩, ⟨U1, U2⟩) to express their model. C denotes a set of nodes in the network; T represents a set of node status; ⟨P1, P2⟩ represents the set of policies for participants; ⟨$S1, S2$⟩ represents the set of policies for participants; ⟨U1, U2⟩ represents reward function. They first classified attack and defense strategies. Then they predicted the optimal attack path of an attacker and the best-response strategies for a defender by quantifying rewards. However, this model did not consider human factors.

3. APT Campaigns

In this section, we introduce the characteristics, various phases, detection methods of APT.

3.1 APT Characteristics

The specific attack target, sophisticated attack tactics, evolutionary attack steps, long-term latent penetration, repeated attack attempts are the most relevant characteristics of APT [24]. Here, we propose new characteristics of APT, like circuitous attack, herd immunity, and percolation phenomena, by analyzing recently five years exposed APT cases.

Circuitous attack: Attackers adopt circuitous routes to perform operations in order to hide the attack source instead of the shortest path.

Herd immunity: Nodes cannot be attacked successfully because of insufficient attack loads. For instance, attack sample is suitable for Windows and Linux platform but does not work in a programmable logic controller (PLC) program. Therefore, these devices in PLC cannot be exploited by attackers.

Percolation phenomena: Node in the attack network and its associated edges are deleted due to link breaking, information error etc.

3.2 APT Phases

APT attackers acquire various attack phases to infiltrate into the target network and to maintain long-term access to the environment. Although different attacks on the specific implementation process are not the same, the attack phases are common. The APT attack lifecycle can be divided into five phases: (1) reconnaissance, (2) delivery, (3) initial intrusion, (4) operation, (5) attack benefit, based on intrusion kill chain (IKC) model [23] through the analysis of the actual APT attacks.

Attackers often use big data analysis technology [25] and social engineering [26] to mine related information in the first stage. And then, attackers deliver malware to the target environment by phishing emails, malicious websites, removable media, etc. In the third stage, attackers were authorized to access the target network and exploit vulnerabilities in the target network. Next, attackers maintain the long-term access to the target by installing the remote access Trojan [27] or the back door in the fifth stage. Last, attackers collect sensitive information and transport them to the particular position, or attackers destroy infrastructures of the target system.

3.3 APT Detection Methods

At present, APT attack detection concentrates on determining whether attack events are advanced, multiple, and customized. Mainstream detection methods can be divided into four categories: malicious code detection [28], host application protection [29], network intrusion detection [30], and big data analysis [31], [32]. However, the former three methods have serious false negatives, because they have only covered a certain stage of APT attack. The big data analysis based on the idea of network forensics has become an effective method to detect APT attack.

4. Targeted Complex Attack Network

In this section, we give a detailed description of our network-evolution-based modeling approach [33]. At first, some preliminaries are shown, and a stochastic model is introduced, some necessary definitions are defined next; then the specific process of node adding is illustrated; the last part of this section describes the derivation of our model.

4.1 Preliminaries

The attack graph model proposed by Phillips and Swiler [34] is used to describe all the attack paths. Nodes of attack graph represent the attack states. Edges represent the attack behaviors. The premise behind invading paths acquisition is to generate attack graph. However, traditional analysis methods based on attack graph lack adaptive abilities in real scenarios. Meanwhile, the attack graph evolves continuously with the progress of the attack. Thus, we extend attack graph by introducing time domain to describe the dynamic characteristics of the attack.

When an APT attack emerges, physical contact through network topology and human interaction coupled with each other. The human interaction can promote the network attack process. First, the intrusion track combining with two kinds of spreading dynamics is of great importance for predicting and controlling APT. We introduce a multiplex network with two layers to represent the formation of attack network. A communication network and a contact network represent trust relationship and network topology, respectively. APT attack conducts through the physical contact layer, or by means of communication layer. Second, in the target network, an improved susceptible-infected-recovered (SIR) model [35] is proposed to describe the attack situation. In this model, each node in the target network is in one of the five states: susceptible, provide-information, attack-successful, active-propagation and removed. Third, we introduce temporary transitional state and connection-successful state to indicate which nodes can be used in the current attack step. Any susceptible node, which has a link with active-propagation nodes or has a trust relationship with their users, is added to the stage of transitional.

The status change of nodes in the target network is shown in Fig. 1.

In the beginning, we assume that all the devices in the network environment are susceptible, and each device has a user. The state of a device changes from susceptible to transitional in the following two situations: (1) this device has a connection to the active-propagation device, (2) the user of this device trusts the user of active-propagation device, that is to say, there is a trust relationship among the users of device changing status and device in active-propagation. A device changes stages from transitional to connect-successful,

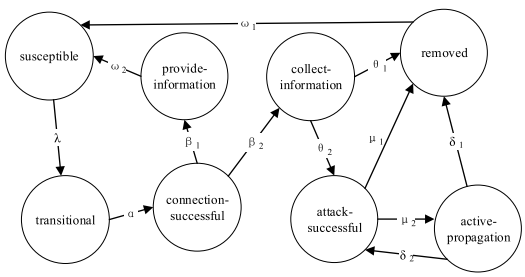


Fig. 1 Statues change of nodes.

when this device in the selection region. After connecting with attackers successfully, this node will join into two statuses: provide-information and attack-successful. When a compromised node has been chosen as the stepping stone, the node is added to state active-propagation. Only if this device is used as a stepping stone in the next attack step, it will infect other nodes. Moreover, the state transition from active-propagation to removed, attack-successful to removed in our model represent that the compromised computer is detected and fixed. Removed nodes will disconnect with the current active-propagation node in the target network. Nodes in provide-information state or nodes in removed state are transformed into susceptible when these nodes connect to the current active-propagation node.

We employ stochastic model to characterize the dynamic behavior of the target network. Let S(t), PI(t), A(t), R(t) be the number of hosts at attack step t in state susceptible, provide-information, attack-successful, removed state, respectively. Let N be the total number of hosts in the target network. After a node connects to the attackers successfully, it is in one of these status: provide-information, attack-successful, removed. One more assumption is that the number of nodes remains unchanged. As a result, the model we employ is as follows:

$$\begin{cases} S(t+1) = \omega_1 R(t) \oplus \omega_2 PI(t) \oplus (1-\lambda\alpha) S(t) \\ PI(t+1) = \lambda\alpha\beta_1 S(t) \oplus (1-\omega_2) PI(t) \\ A(t+1) = (1-\mu_2)\lambda\alpha\beta_2\theta_2 S(t) \oplus \delta_2 P(t) \oplus (1-\mu_1) A(t) \\ P(t+1) = \lambda\alpha\beta_2\theta_2\mu_2 S(t) \\ R(t+1) = \lambda\alpha\beta_2\theta_1 S(t) \oplus \mu_1 A(t) \oplus \delta_1 P(t) \\ S(t+1) \oplus PI(t+1) \oplus A(t+1) \oplus P(t+1) \oplus R(t+1) = N \end{cases} \tag{1}$$

$\oplus$ denotes the AND operation of two sets. The result of $C = A \oplus B$ is the sum of the number of A and B minus the number of intersection of these two sets. λ denotes infection rate, which is decided by the topology and trust relationship of active-propagation machines. α denotes state transition rate from transitional to connection-successful. β_1, β_2 denote the state transition rate from connection-successful to provide-information and collect-successful, respectively. θ_1, θ_2 denote state transition rate from collect-information to removed and attack-successful, respectively. μ_1, μ_2 denote state transition rate from attack-successful to removed and active-propagation. σ_1, σ_2 denote state transition rate from active-propagation to removed and attack-successful. ω_1, ω_2 denote state transition rate from removed to susceptible, from provide-information to susceptible, respectively.

Definition 1 A node is expressed by a three-tuple N=(description, host, status), which is used to indicate a device in the network.

Description is used to describe factors that affected nodes into connect-successful; represents node id, that is specified as an IP address; represents the node status, which is in one of the five states: susceptible, provide-information, attack-successful, active-propagation and removed.

Definition 2 An attack edge is expressed by a three-tuple $e = (n_i, n_j, R)$.

R represents relationships exploited by this attack behavior, which is subjected to $R = R^{Topology} \bigcup R^{Trust}$, indicated as T1, T2, where $R^{Topology}$ represents the topology relationship. If the value of R is R^{Trust}, this attack is going by social engineering. Thus, we take human factors into considerations by using trust relationship to conduct an attack in the actual attack process.

Definition 3 The current attack situation is described by a complex attack network $CAN = (N, E)$ from the formalization perspective, where N represents the set of node involved in the procedure of attack, E represents the set of attack edges.

4.2 Node Adding Progress

There are two layers in our multiplex network: communication network labeled A, contact network labeled B. Network A and network B express trust relationship and topology relationship, respectively. Supposing that N indicates their node number. Thus, two random networks can be generated according to the given degree distribution. There is no self-loops or repeated links in these two networks. Meanwhile, there is no correlation between the generated double-layer network. Each node in layer A is matched with that of layer B one-to-one. Thus, links in double layers are scarcely overlapped when the network is very large and sparse. The theoretical framework of the asymmetric interacting spreading processes in this paper can be easily generalized to the multiple networks with inter-layer degree correlations and overlapping links.

Integer N indicates the number of nodes and users in the target network. In the following example, N equals 4. A topology relationship between devices and a trust relationship between users consisting of n nodes can be represented by square matrix T1 and T2, they called the topology matrix and the trust matrix [36]. Each element t_{ij} (in the i-th row and the j-th column) in the topology matrix is a constant 1 or 0 to indicate whether there is a directed link from node i to node j or not. Similarity, element t_{ij} in the trust matrix is 1 to indicate user j trust user i. Then, an attacker can exploit trust relationship to compromise the node i.

The topology matrix and trust matrix of Fig. 2 are:

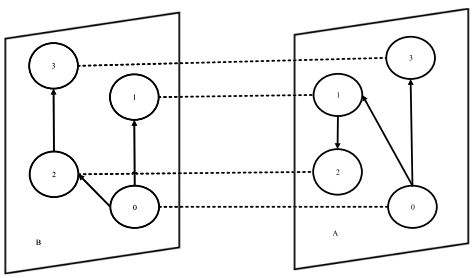


Fig. 2 A topology and trust relationship as an example.

$$T1 = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad T2 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

In the papre, we use a row vector S with length n to represent these states of all the n nodes in the target network. And each element s_i (the i-th element) of the state vector takes four bits to indicate the state of node i: 0000 represents susceptible; 1000 represents transitional; 1100 represents provide-information; 1110 represents attack-successful; 1111 represents active-propagation; 0110 represents removed. The state transitions of devices from susceptible to transitional are related to the topology relationship of the target network and trust relationship of users in the target network. In our example, we suppose that the node 0 is active-propagation, other nodes are susceptible, then the initial node state is expressed as S=[1111 0000 0000 0000]. The lower bit of node state is 1 or 0 indicating whether the current node is able to infect other nodes or not. In the same way, the higher bit indicates whether Active-propagation node intrudes the current node. S_t will stand for the state vector at step t.

4.2.1 Matrix Operations

1. Lowest bit, highest bit, and middle bits

The operation L(T), H(T), and M(H) indicate the lowest bit, the highest bit and middle bits of vector T, respectively. The result is also a vector of the same length with vector T. For example if the vector T is [1111 0000 1000 0110], then L(A) is [1 0 0 0] and H(A) is [1 0 1 0], M(A) is [11 00 00 11].

2. OR

We use the sign '+' to denote the OR operation of two vectors. Thus, each element ci of C=A+B indicates that the result of the OR operation of the i-th of vector A and the i-th of vector B. That is, $c_i = a_i + b_i$, i=1, 2, 3, ..., n. For instance, if A is [1 0 0 0] and B is [0 1 1 0], then the result of C is [1 1 1 0].

3. AND

We use the sign '$\oplus$' to represent the AND operation of two vectors. Thus, each element ci of $C = A \oplus B$ indicates that the result of the AND operation of the i-th of vector A and the i-th of vector B. That is, $c_i = a_i \oplus b_i$, i=1, 2, 3, ..., n. For example, if A is [1 0 0 0] and B is [0 1 1 0], then the result of C is [0 0 0 0].

4. Matrix multiplication

We use the sign '$\times$' to represent the matrix multiplication operation of a vector and a matrix. Thus, each element ci of $C = A \times B$ can be computed by Eq. (2).

$$c_i = \sum_{i=1}^{n} a_k b_{ki}, \tag{2}$$

where A is a row vector with n columns; B is a square matrix; $a_k b_{ki}$ indicates that the result of the AND operation of the i-th of vector A and the element in the k-th row and the i-th column of vector B; $\sum$ indicates the OR operation of all the results of those AND operation. For instance, A is [1 0 0 0] and B is the topology matrix of Fig. 2, then the result of C is [0 1 1 0].

5. Concatenation

We use the sign '$\bigotimes$' to represent the concatenation operation of two vectors with the same length. Thus, each element c_i of $C = A \bigotimes B$ can be expressed as $a_i b_i$. That is, the higher part and the lower part of ci is the i-th element of A and the i-th element of B, respectively. For example, A is [1 1 1 0], B is [1 0 0 0], then the result of A$\bigotimes$B is [11 10 10 00].

4.2.2 The Formulation of First State Transition

Here, we only consider the state transition from susceptible to transitional. We use a temporary vector S^a_{t+1} of length n to denote the nodes will be directly infected by the targeted complex attack at attack step t+1. The i-th element of S^a_{t+1} is 1 or 0 indicating whether the node i is infected or not at attack step t+1. Since only susceptible nodes with topology connection with active-propagation nodes, or whose users has a trust relationship with users of active-propagation nodes may become transitional nodes. Thus, S^a_{t+1} follows Eq. (3).

$$S^a_{t+1} = L(S_t) \times T1 + L(S_t) \times T2, \tag{3}$$

where S_t indicates the state vector at attack step t; $L(S_t)$ indicates the lowest bit of each element in vector S_t; T1, T2 are the topology matrix and trust matrix, respectively. Considering the example in Fig. 2 with initial state vector S_0 = [1111000000000000], the result of S^a_1 is [0111] indicating that node 1, node 2 and node 3 can be intruded at attack step 1.

Finally, the state vector at attack step t+1 is expressed as Eq. (4).

$$S^a_{t+1} = (H(S_t) + S^a_{t+1}) \bigotimes M(S_t) \bigotimes L(S_t), \tag{4}$$

The highest bit of S_{t+1} indicates whether targeted complex attack network intrudes the current node at attack step t+1; and the lowest bit of S_{t+1} indicates whether the current node is able to attack other nodes or not. We use Fig. 2 as an example, S_1 is [1111100010001000].

4.2.3 The Node Selection

Traditional BA model [37] takes no comprehensive affecting factors into considerations, when new nodes are ready to join the network. Moreover, there are no node-deletion and herd-immunity in the traditional BA model. Through the previous analysis, we establish TCAN by adopting improved BA model. There are n factors that affected nodes adding into the directed complex attack network, which is expressed as $\{f_1, f_2, \ldots, f_n\}$.

In our model, destruction level (DL), persistence strength (PS), and exposure degree (ED) as references of

dot-attraction which is used to choose nodes point to this new node. Suppose that attributes of the active-propagation node is $\{f_1^0, f_2^0, \ldots, f_n^0\}$, and there are n nodes in transitional whose attributes are $\{f_1^1, f_2^1, \ldots, f_n^1\}, \ldots, \{f_1^i, f_2^i, \ldots, f_n^i\}, \ldots, \{f_1^n, f_2^n, \ldots, f_n^n\}$. We calculate the distance between the current node and active-propagation node using $\sqrt{|f_1^i - f_1^0|^2 + \cdots + |f_n^i - f_n^0|^2}$, where i=1, …, n. Then, an attacker selects the nearest M nodes with transitional according to the distance from the current active-propagation node.

Integer n indicates the number of devices in the target network. In the following example, n equals 4. A topology relationship between devices and a trust relationship between users consisting of n nodes are shown in Fig. 2.

The current active-propagation node is 0, and the target node is 2. The attribute of node 0 is $\{f_1^0, f_2^0, \ldots, f_n^0\}$. According to the topology relationship of the target network, node 1 and node 2 change from susceptible state to transitional state, whose attributes are $\{f_1^1, f_2^1, \ldots, f_n^1\}$, $\{f_1^2, f_2^2, \ldots, f_n^2\}$. According to the trust relationship of the target network, node 1 and node 3 are transferred to transitional state. And attributes of node 1, node 3 in the trust network are $\{f_1^{1'}, f_2^{1'}, \ldots, f_n^{1'}\}$, $\{f_1^{3'}, f_2^{3'}, \ldots, f_n^{3'}\}$. Then, an attacker calculates the distances between the current susceptible nodes with the active-propagation node. Distances are calculated according to $\sqrt{|f_1^1 - f_1^0|^2 + . + |f_n^1 - f_n^0|^2}$, $\sqrt{|f_1^2 - f_1^0|^2 + . + |f_n^2 - f_n^0|^2}$, $\sqrt{|f_1^{1'} - f_1^0|^2 + . + |f_n^{1'} - f_n^0|^2}$, $\sqrt{|f_1^{3'} - f_1^0|^2 + . + |f_n^{3'} - f_n^0|^2}$. Assuming that there is one node selected, thus, the nearest node is transferred into connect-successful. Here, node 1 is selected. Then, node 2 is attacked by trust relationship.

4.3 Model Derivation

According to the definition of a dynamic network [35], the dynamic attack network can be regarded as the attack subgraph sequences over the series of consecutive time steps. Figure 3 shows that attack network generated from initial attack states: CAN_0, CAN_1, CAN_2, CAN_3 represent that attacker has no connection at t_0. Then he accesses to B with user privilege at t_1 using a weak password on FTP service running on B. Next he gets root privilege of D at t_2. Finally, he gets the user privilege of C at t_3. Thus, the time domain of evolutionary attack graph is $[t_0, t_3]$. Thus, the dynamic attack network can be expressed as $\{\langle CAN_0, t_0\rangle, \langle CAN_1, t_1\rangle, \langle CAN_2, t_2\rangle, \langle CAN_3, t_3\rangle\}$.

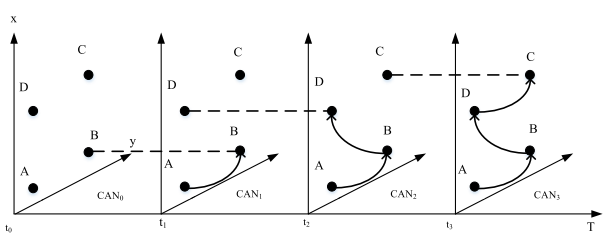


Fig. 3 A simple attack process as an example.

According to definition 3, we define attack network at time t as $CAN_t = (N_t, E_t)$.

$N_t = \{n_i | i = 1, 2, \ldots, m_t\}$ represents the set of all nodes involved in the attack until time t, where n_i represents a node, indicated as "◯". $E_t = \{e_j | j = 1, 2, \ldots, s_t\}$, where e_j represents an attack edge, indicated as "⟶".

Although the procedure of APT attack has a certain stability of time and space, with APT attack continues, there are new attack behaviors occurring. Thus, new nodes and edges need to join the attack network. If previous attacks failed, nodes and edges existed in the attack network need to be removed. However, most time attackers collect information about the target. This phenomenon is consistent with individual human mobility patterns. In other words, although most attack behaviors are placed soon after a previous attack behavior, occasionally there are long periods without any attack activity.

According to network evolution theory [33], APT attack process subjects the following steps:

1. Adding: Node and edges have characteristics of dynamic growth with the attack progress;

2. Removing: A node may become a failure node, once the previous attack behavior fails. For example, service as the prime attack target is shut down etc. before the attack succeeds.

Thus, the CAN at time t+1 can be expressed as $CAN_{t+1} = (N_{t+1}, E_{t+1})$, which meets the following conditions:

$$(1) N_{t+1} = (N_t \cup \{n_a^{t+1}\} - \{n_d^{t+1}\}),$$
$$(2) E_{t+1} = (E_t \cup \{e_a^{t+1}\} - \{e_d^{t+1}\}.$$

$\{n_a^{t+1}\}$ represents the new adding nodes. $\{n_d^{t+1}\}$ represents nodes set which needs to be removed from the attack network. $\{e_a^{t+1}\}$ represents the attack edges joining into the attack process. $\{e_d^{t+1}\}$ represents the attack edges removed from attack process once the target node is removed.

In our model, each device joining the targeted complex attack network must go through series of stage transitions, one is susceptible→transitional→connection-successful→provide-information, another is susceptible→transitional→connection-successful→collect-information→attack-successful. Thus, new nodes in provide-information, attack-successful, active-propagation at the current attack step are added into the targeted complex attack network. When the target node reaches the attack-successful state, this network is completed.

Based on the analysis hereinabove, the attack network generation process is described as shown in Algorithm 1.

Initialization: The original attack situation is expressed as CAN_0, thus, $TCAN = CAN_0$. SN and NN is a struct type, which includes a parent node and a list of children node. SN represents the starting node, NN points to the current node. N is the attack step. n is the number of component nodes in the current complex attack network.

There are two types of attack behaviors, one is informa-

Algorithm 1 TCAN generation algorithm

1: Input: CAN_0 represents initial attack relationships of attacker, number of attack nodes in the current network is n, information about new attack behavior and feedbacks of pervious attack behaviors are represented by DAG, ultimate target node is T.
2: Output: $TCAN = \{\langle CAN_i, t_i\rangle\}$.
3: Initialize: $SN, NN, TCAN = TCAN_0, N = 0$.
4: **while** $(NN \neq T)$
5: **for** (attack behaviors in DAG)
6: $N++$
7: **if** (previous attack failed)
8: delete NN.children(fail) and edges
9: **for**(each target node c_i in current attack)
10: **if**(previous is not used to collection information)
11: $NN = NN.children(success)$
12: **end if**
13: NN point to c_i
14: **end for**
15: **end if**
16: $TCAN = TCAN + \{\langle CAN_N, t_N\rangle\}$
17: $n = numberofCAN_N$
18: **end for**
19: **end while**
20: Output $TCAN$

Table 1 The contrast analysis between models.

M	Pha	A-d	S	H-f	F-e	Pre	N-t
TCAN	✓	✓	✓	✓	✓	✓	✓
Attack Pyramid [18]	✓	×	×	✓	×	✓	×
EPNAM [19]	✓	×	✓	✓	✓	×	×
IKC [20]	✓	×	×	×	×	×	×
MM-TBM [21]	✓	×	×	×	×	✓	×
OPAG [22]	×	✓	×	✓	✓	✓	×

M: Model; Pha: Phase; A-d: Attacked device;
S: State; H-f: Human factor; F-e: Formal expression;
Pre: Prediction; N-t: Non-traceability

tion collection, and the other is a real attack using network intrusion technologies or intelligence techniques. Here, we suppose there is no state change of the attacker at the time of collecting information. Thus, starting node of attack behavior after information collection is the same as the previous attack. We can get feedback on previous attack behavior according to the new DAG, which is used to describe current attack intention.

5. Experiments

5.1 Comparative Analysis of TCAN and Other Models

From the considered factors' point of view, we compare our proposed TCAN model with other newly emerging APT models, which is shown in Table 1.

TCAN model directly studies the characteristics of APT formation, development, convergence and exit, taking full account of the phase, devices and human factors in the attack process. Furthermore, TCAN model makes use of the formal expression to illustrate attack in order to select attack paths and erase traces and thus makes attack detection products cannot trace the attack source. It would appear (Table 1) that APT models which are proposed in recent years do not consider non-traceable except for TCAN model. In addition to OPAG model [22], they do not represent the device node involved in the attack process. However, OPAG model [22] does not indicate phases of APT attack. Attack Pyramid [18] is only as a conceptual model of attack detection, which not only does not highlight devices but also does not show nodes' state changes. Attack Pyramid [18] represents attack process graphically and discovers attack path through correlation analysis, which lakes of formal description and provides the basis for tracing and tracking. EPNAM model [19] extends the Petri net model. However, its scalability is poor because node states are limited to 8. EPNAM model [19] does not consider the predictability. IKC model [20] only divides the APT attack process into different phases in order to take phased defensive measures. MM-TBM model [21] combines the BP algorithm with the DS theory to solve the problem about evidence conflict. MM-TBM model [21] takes phase into consideration in the modeling process and is capable of predicting attack in the next step can by combining with the evidence obtained from the existing safety equipment. OPAG model [22] based on game theory gives the best attack strategy and the best defensive measures of each attack step, but does not reflect the state changes and considers concealment. In conclusion, our proposed TCAN model considers more factors than other five models. Thus, TCAN model has better expansibility and expression in the process of characterizing APT attack, so as to facilitate APT detection through large data fusion technology in the later.

5.2 Simulated Experiment

5.2.1 Simulated Attack

In this section, we test the feasibility of our model by attempting to simulate a complex network attack for data exfiltrations on a real network. The network is a large network consisting of about 10K hosts, where Marchetti et al. [30] verify that their approach is able to detect the most suspicious hosts that may be involved in the APT-related attack. The purpose of the experiment is to verify whether the proposed model is able to steal information by integrating human factor with the vulnerability associated with devices in the target network. The topology of the experimental environment is shown in Fig. 4. 10.10.10.129 is a Web server accessed through Tomcat server by users on the Internet. There is a remote code execution vulnerability in Struts2 used on this server. m programmers are divided into two groups: one group with C, the other with Java, and n tester, their operating systems are Windows. We only list three computers to simplify the network structure: one is a developer with C, one is a developer with JAVA, and the third is a tester. And the development server can be accessed by programmers. Moreover, there are two servers in this network: development server and test server, whose IP address are

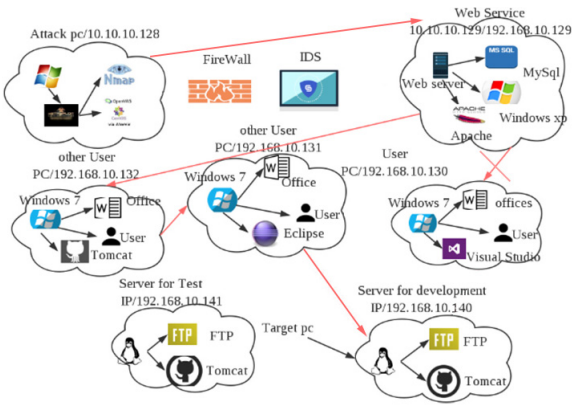


Fig. 4 The topology of simulated environment.

192.168.1.140, and 192.168.1.141, respectively. The development server is used to store all the source codes, important project plans, and procedural documentation. All the products need to be deployed on the test server.

The simulated attack process is as follows:

Firstly, attacker whose IP address is 10.10.10.128 accesses the Web server and investigates the configurations information.

Secondly, the attacker finds a remote command execution vulnerability of structs 2 used by this server. Then, he exploits this vulnerability to get web shell. Next, he gets the server permissions in order to invade the entire server by using local privilege escalation vulnerability.

Thirdly, this attacker listens on the web server to get username and password about the user in 192.168.10.130 accessed to the development server. The attacker accesses the server in 192.168.10.140 using this username and password.

Fourthly, attacker fails to attack a host of 192.168.10.130, because this employee has left from this company for personal reasons. Then, he tries to forge a malicious link on the site. After a time, a user of 192.168.10.132 clicks this link and controlled by the attacker.

Fifthly, the attacker finds the user of 192.168.10.131 trust the user of 192.168.10.132. Then attacker sends an email with Trojan to the user of 192.168.10.131. The host of 192.168.10.131 is controlled by the attacker. Fifth, the attacker finds the user of 192.168.10.131 trust the user of 192.168.10.132. Then attacker sends an email with Trojan to the user of 192.168.10.131. The host of 192.168.10.131 is controlled by the attacker.

Sixthly, the attacker can access the development server and steal important information.

The TCAN evolutionary process of this simulated attack is shown graphically in Fig. 5.

In the figure above, A stands for the attacker, B website server accessed by users through the Internet, C host 192.168.10.130, D host 192.138.10.132, E host 192.168.10.131, F development server 192.168.10.140, which is the target node in our simulated attack. T1, T2 indicate topology relationship and trust relationship. PI, CI,

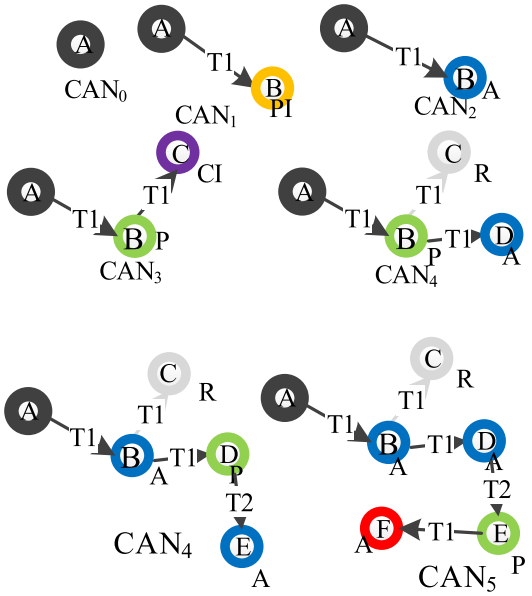


Fig. 5 A TCAN evolution process.

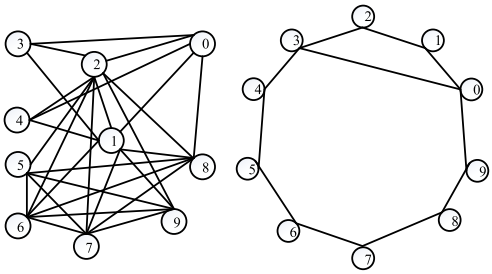
A, P, R represent provide-information, collect-information, attack-successful, active-propagation, removed state, respectively. Meanwhile, they are described using circle with different colors, like black cycle denotes attacker, yellow cycle node in provide-information state, blue cycle node in attack-successful, green cycle node in active-propagation state, the red cycle is target node.

5.2.2 Experiment Analysis

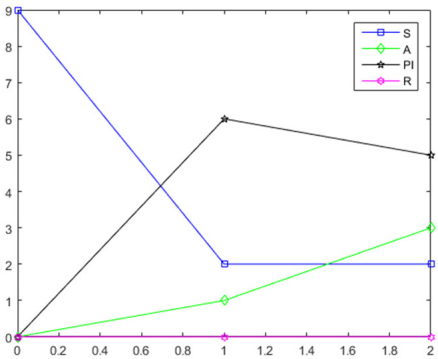
To validate our model's applicability to different attack scenarios, we choose the scale-free network and small world network as the topology and trust relationship, respectively. We run our proposed TCAN model on a desktop with Intel Core i7 at 3.6 GHz and 16GB of RAM memory. Without loss of generality, experimental results reported below are the average of 100 repeated experiments. The measures of interest include the number of nodes in the attack-successful state, provide-information state, and removed state at the end of the attack, attack steps acquired by this attack. To jumpstart the APT attack progress, the initial number of active propagation nodes is set to one. In other words, the statue of attacker is active-propagation.

The first simulation is the baseline experiment. The number of topology size is set to 10. The topology and trust relationship of the target network is shown in Fig. 6 (a). Node 5 is in active-propagation state. Node 9 is the target node. The input variables are assigned the value shown in Table 2 based on our proposed model. The parameter of probability provide-information is set to 0.8. The parameters of removal rate and failure rate are set to 0.1.

Status of nodes in the target network at every attack step is shown in Fig. 6 (b). Figure 7 presents the detailed attack path, where T1 is the topology relationship, and T2 is the trust relationship.



(a) Topology and trust relationship of the simulated experiment



(b) The number of different states node

Fig. 6 The baseline experiment.

Table 2 Variable values for baseline experiment.

Variable	Value
β_1	0.8
β_2	0.2
θ_1	0.1
θ_2	0.9
μ_1	0.1

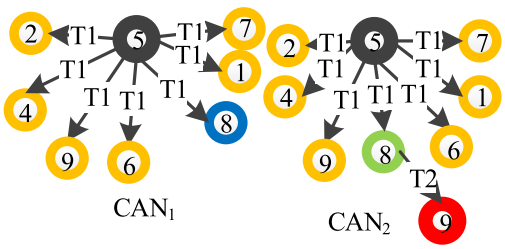


Fig. 7 TCAN evolutionary process of baseline experiment.

In this simulated experiment, attacker collects information about target network from public sources, indicated as node 1, 2, 4, 6, 7, 9. Then attacker breaks into the node 8 through network infiltration method. Finally, the attacker controls the target node using social technologies. This attack pattern is similar to Operation Aurora [38]. Operation Aurora was a serious cyber attack caused by APT in 2009. In this attack, the user names and passwords of sensitive users accessed to google server were stolen. Its consequences led to the theft of important email information about these sensitive users.

The following analysis is implemented using the proposed TCAN on the background of Google Aurora.

1: The attacker collects information about staff in target network from open source.

2. A certain employee E is targeted. Then, the attacker collects information about this certain employee E from social network websites like Facebook, Twitter, and LinkedIn. Next, specific friend F who likes photograph is selected. Followed is that the attacker breaks into the host of friend F through network infiltration method.

3: The attacker pretends to be the friend E and sends an instant message to this employee in order to invite him to enjoy the latest photos. But the URL points to a web page loading shellcode and Javascript, which is managed by a Web server forged by an attacker. The employee E clicks the link to enter the malicious website forged by the attacker, which can cause the overflow of IE browser with this specific employee in Google. The host of this specific employee executes FTP download program locally. The host of this specific employee downloads more programs to execute, such as Trojan. Then, the connection is established between target host and attacker host through SSL Tunnel.

Table 3 The attack steps to different topology size.

Topology size	Attack steps
1000	21
2000	116
3000	85
4000	29
5000	217

Table 4 The attack steps to different topological complexity.

Topological complexity	Attack steps
0.02	56
0.04	217
0.06	246
0.08	25
0.1	42

5.3 Comparison with Different Parameters

5.3.1 Different Topology Sizes and Complexity

In this section, we discuss our model's flexibility to a different topology. The experiment result is shown in Tables 3 and 4. The attack steps increase with the expanding of topology at first. Then, it begins to decrease, which is mainly because of complex topology or close distance from starting node to the end node. Table 4 shows that attack steps decrease when the topology network reaches a certain complexity. However, the attack steps increase the probability of edge addition at 0.1. That is because the attributes of node give even more weight to select attack paths.

5.3.2 Different Probability Provide-Information

Table 5 and Fig. 8 show the sensitivity of APT attack progress to different probability provide-information. It is

Table 5 The attack steps to different probability of provide-information.

Provide-information	Attack steps
0.5	8
0.6	29
0.7	108
0.8	217
0.9	27

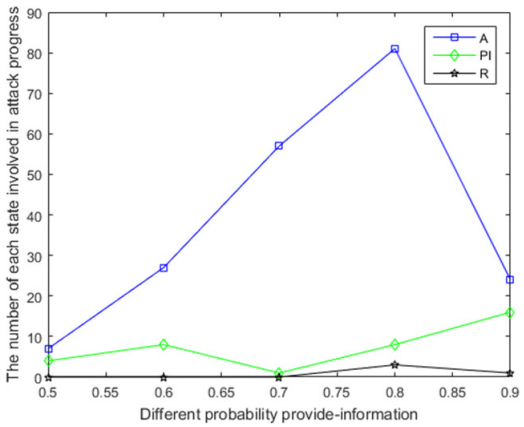


Fig. 8 The number of each state involved in the attack progress to different probability provide-information.

Table 6 The attack steps to different removal rate.

Removal rate	Attack steps
0.01	23
0.02	43
0.03	41
0.04	58
0.05	68
0.06	117
0.07	186
0.08	224
0.09	239
0.1	295

obvious that the larger probability provide-information results in larger attack steps. The number of nodes involved in the attack progress increases with the probability provide-information as well. It can be seen that a serious fall in attack steps when the probability of provide-information increases to a certain size. Thus, we can make a conclusion that the more information about the target node collected by attackers, the faster, more cost-effective to reach their attack goal.

5.3.3 Different Removal Rate

In this section, we target our model's sensitivity to different removal rate. This type of experiment result is shown in Table 6 and Fig. 9, which is useful for guiding the development of anti-malware products. If it is much easier to detect attacks, and the attack steps raise high enough, then the network security analyst may be more likely to detect this attack.

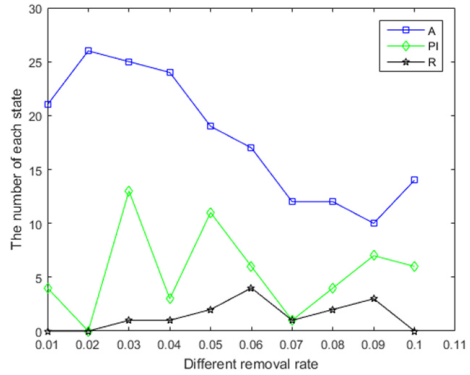


Fig. 9 The number of each state at the end of this attack to different removal rate.

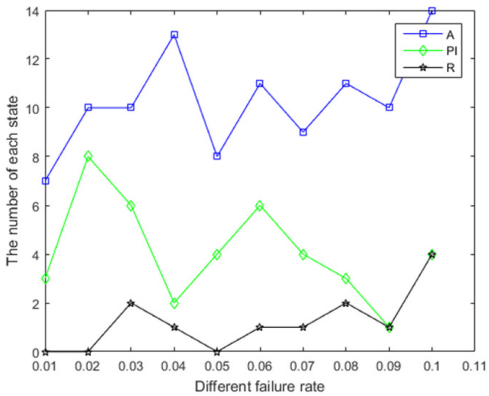


Fig. 10 The number of each state at the end of this attack to different failure rate.

Table 7 The attack steps to different failure rate.

Failure rate	Attack steps
0.01	8
0.02	26
0.03	121
0.04	174
0.05	192
0.06	221
0.07	249
0.08	273
0.09	302
0.1	400

5.3.4 Different Failure Rate

In the end of this section, we study the effect of failure rate on APT attack progress. It is obvious from Table 7 and Fig. 10 that different failure rates have their impacts on the attack progress. The anti-malware systems deployed in the target network defend network attacks at a fast-enough rate, then this APT attack can actually increase attack steps.

6. Discussion and Conclusion

In this paper, we introduce a network-evolution-based at-

tack network generating mechanism to express APT attack. Here, a double-layer network is introduced to elucidate the effects of human interaction in layer A, topology link in layer B. The reinforcement of information captured by an attacker, the capacity of attack load have an effect on the probability of host joining into the attack network. Moreover, we found that there are many factors influencing attack choice. For example, an attacker chooses the sub-optimal attack path to escape detection. Thus, there are five kinds of nodes, including initial nodes, failure nodes, information nodes, immune nodes, and target nodes in our model. Failure nodes explain percolation phenomena in actual APT attacks. We cannot reconstruct the entire attack path when the failure nodes existing in this attack process. Immune nodes exist outside the attacking net in isolation, which reveals the herd immunity phenomena. There are no edges pointing to other nodes of information nodes, which illustrates cumulative advantage existing in the derivation of TCAN. At the same time, there is at least one node called target node, which does not have outing edges of the TCAN model. The TCAN model building completes after the target node added to the net.

The challenges of studying the attack procedure combining intricate interplay between network penetration and social engineering are generating interesting science. In this work, we consider human factors and reinforcement effect of cumulative information about host node, and then study their impact on the attack network dynamics. Meanwhile, network structure and state transition probability have an effect on the TCAN generation. The difference between our model and other models lies in the time domain. In our work, each attack step can be expressed visually. In addition, we take human factors into considerations at every attack step by introducing multiple networks. It is represented by a communication contact double-layer network in order to conduct an attack by means of a trust relationship and physical connection. Meanwhile, node status change is expressed using improved SIR model. Our model is expressive and flexible. The experimental results in this paper validate our viewpoints. Dynamic network studies of network attacks are still in their initial stages. Studying the properties of TCAN model from the angle of the network is a meaningful future work.

Acknowledgments

This work was supported by National Natural Science Foundation of China under grants No.61572115.

References

[1] P. Chen, L. Desmet, and C. Huygens, "A study on advanced persistent threats," IFIP International Conference on Communications and Multimedia Security, pp.63–72, Springer, 2014.

[2] F. Li, A. Lai, and D. Ddl, "Evidence of advanced persistent threat: A case study of malware for political espionage," 2011 6th International Conference on Malicious and Unwanted Software (MALWARE), pp.102–109, IEEE, 2011.

[3] I. Jeun, Y. Lee, and D. Won, "A practical study on advanced persistent threats," Computer Applications for Security, Control and System Engineering, pp.144–152, Springer, 2012.

[4] M. Ask, P. Bondarenko, J.E. Rekdal, A. Nordbø, P. Bloemerus, and D. Piatkivskyi, "Advanced persistent threat (APT) beyond the hype," Project Report in IMT4582 Network Security at Gjøvik University College, Springer, 2013.

[5] M. Cloppert, "Security intelligence: Introduction (pt 1)," SANS Digital Forensics and Incident Response Blog, 2009.

[6] D. Alperovitch et al., Revealed: Operation shady RAT, McAfee, 2011.

[7] A.W. Coviello, "Open letter to RSA customers," RSA [database online], 2011.

[8] C. Raiu, "Cyber-threat evolution: The past year," Computer Fraud & Security, vol.2012, no.3, pp.5–8, 2012.

[9] K. Pipyros, L. Mitrou, D. Gritzalis, and T. Apostolopoulos, "A cyber attack evaluation methodology," Proc. 13th European Conference on Cyber Warfare and Security, pp.264–270, 2014.

[10] C. Furlani, Managing information security risk: Organization, mission, and information system view, NIST Special Publication 800-39, 2011.

[11] P. Hu, H. Li, H. Fu, D. Cansever, and P. Mohapatra, "Dynamic defense strategy against advanced persistent threat with insiders," 2015 IEEE Conference on Computer Communications (INFOCOM), pp.747–755, IEEE, 2015.

[12] J. Vukalović and D. Delija, "Advanced persistent threats — Detection and defense," 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp.1324–1330, IEEE, 2015.

[13] O.S. Adebayo and N. AbdulAziz, "An intelligence based model for the prevention of advanced cyber-attacks," 2014 The 5th International Conference on Information and Communication Technology for The Muslim World (ICT4M), pp.1–5, IEEE, 2014.

[14] B. Schneier, "Attack trees," Dr. Dobb's Journal, vol.24, no.12, pp.21–29, 1999.

[15] S. Jajodia, S. Noel, and B. O'Berry, "Topological analysis of network attack vulnerability," Managing Cyber Threats, pp.247–266, Springer, 2005.

[16] J.P. McDermott, "Attack net penetration testing," Proc. 2000 Workshop on New Security Paradigms, pp.15–21, ACM, 2001.

[17] Q.-H. Liu, W. Wang, M. Tang, and H.-F. Zhang, "Impacts of complex behavioral responses on asymmetric interacting spreading dynamics in multiplex networks," Scientific Reports, vol.6, Article Number 25617, 2016.

[18] P. Giura and W. Wang, "A context-based detection framework for advanced persistent threats," 2012 International Conference on Cyber Security (CyberSecurity), pp.69–74, IEEE, 2012.

[19] W. Zhao, P. Wang, and F. Zhang, "Extended Petri net-based advanced persistent threat analysis model," Computer Engineering and Networking, pp.1297–1305, Springer, 2014.

[20] P. Bhatt, E.T. Yano, and P. Gustavsson, "Towards a framework to detect multi-stage advanced persistent threats attacks," 2014 IEEE 8th International Symposium on Service Oriented System Engineering (SOSE), pp.390–395, IEEE, 2014.

[21] G. Ioannou, P. Louvieris, N. Clewley, and G. Powell, "A Markov multi-phase transferable belief model: An application for predicting data exfiltration APTs," 2013 16th International Conference on Information Fusion (FUSION), pp.842–849, IEEE, 2013.

[22] X. Fang, L. Zhai, Z. Jia, and W. Bai, "A game model for predicting the attack path of APT," 2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing (DASC), pp.491–495, IEEE, 2014.

[23] E.M. Hutchins, M.J. Cloppert, and R.M. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," Leading Issues in Information Warfare & Security Research, vol.1, p.80, 2011.

[24] M. Ussath, D. Jaeger, F. Cheng, and C. Meinel, "Advanced persistent

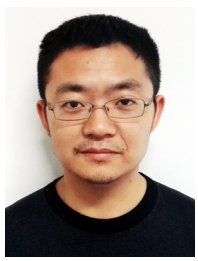
threats: Behind the scenes," 2016 Annual Conference on Information Science and Systems (CISS), pp.181–186, IEEE, 2016.
[25] M. Chen, S. Mao, and Y. Liu, "Big data: A survey," Mobile Networks and Applications, vol.19, no.2, pp.171–209, 2014.
[26] S. Granger, "Social engineering fundamentals, part I: Hacker tactics," Security Focus, vol.18, Dec. 2001.
[27] A. Beuhring and K. Salous, "Beyond blacklisting: Cyberdefense in the era of advanced persistent threats," IEEE Security Privacy, vol.12, no.5, pp.90–93, 2014.
[28] N. Villeneuve and J. Bennett, "Detecting APT activity with network traffic analysis," Trend Micro Incorporated Research Paper, 2012.
[29] A. Sharma and S.K. Sahay, "An effective approach for classification of advanced malware with high accuracy," International Journal of Security and Its Applications, vol.10, no.4, pp.249–266, 2016.
[30] M. Marchetti, F. Pierazzi, M. Colajanni, and A. Guido, "Analysis of high volumes of network traffic for advanced persistent threat detection," Computer Networks, vol.109, Part 2, pp.127–141, 2016.
[31] J. Kim, T. Lee, H.-G. Kim, and H. Park, "Detection of advanced persistent threat by analyzing the big data log," Advanced Science and Technology Letters, vol.29, pp.30–36, 2013.
[32] F. Barceló-Rico, A.I. Esparcia-Alcázar, and A. Villalón-Huerta, "Semi-supervised classification system for the detection of advanced persistent threats," Recent Advances in Computational Intelligence in Defense and Security, pp.225–248, Springer, 2016.
[33] B. Skyrms and R. Pemantle, "A dynamic model of social network formation," Adaptive Networks, pp.231–251, Springer, 2009.
[34] C. Phillips and L.P. Swiler, "A graph-based system for network-vulnerability analysis," Proc. 1998 Workshop on New Security Paradigms, pp.71–79, ACM, 1998.
[35] R.M. May and A.L. Lloyd, "Infection dynamics on scale-free networks," Phys. Rev. E, vol.64, no.6, 066112, 2001.
[36] T. Chen, X.-S.. Zhang, H.-Y. Li, D. Wang, and Y. Wu, "Propagation modeling of active P2P worms based on ternary matrix," Journal of Network and Computer Applications, vol.36, no.5, pp.1387–1394, 2013.
[37] P. Holme, B.J. Kim, C.N. Yoon, and S.K. Han, "Attack vulnerability of complex networks," Phys. Rev. E, vol.65, no.5, 056109, 2002.
[38] G. Kurtz, "Operation aurora hit google, others," Published online at http://siblog.mcafee.com/cto/operation-%E2, vol.80, 2010.

Weina Niu received her bachelor degree in Software Engineering from Shenyang Normal University in 2011 and now is a Ph.D. candidate in School of Computer Science and Engineering, University of Electronic Science and Technology of China. Her research interest includes network attack detection and dynamic program analysis.

Xiaosong Zhang was born in Sichuan Province, China, on June 15, 1968. He received his M.S. and Ph.D. degrees from University of Electronic Science and Technology of China, Chengdu, China, in 1999 and 2011, respectively. Since 2011 he has been a professor in information security at University of Electronic Science and Technology of China. His research interest includes cryptograph, dynamic program analysis and information security.

Guowu Yang received his B.S. degree from Wuhan University of Technology in 1999, and received his Ph.D. degree from Portland State University in 2005. Since 2006 he has been a professor in Computer Software and Theory at University of Electronic Science and Technology of China. His research interest includes quantum cryptography, formal verification and machine learning.

Ruidong Chen received his bachelor and M.S. degree from University of Electronic Science and Technology of China in 2008 and 2012, and now is a Ph.D. candidate in School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interest includes network attack detection and machine learning.



Dong Wang received his bachelor and M.S. degree from University of Electronic Science and Technology of China in 2008 and 2011, and now is a Ph.D. candidate in School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interest includes traffic analysis and information security.