Multimodal Graph Analysis of Cyber Attacks

MULTIMODAL GRAPH ANALYSIS OF CYBER ATTACKS

Nirnimesh Ghose, Loukas Lazos and Jerzy Rozenblit

Department of Electrical and Computer Engineering
University of Arizona, Tucson, AZ
{nghose, llazos, jerzyr}@email.arizona.edu

Ronald Breiger

School of Sociology
University of Arizona, Tucson, AZ
breiger@email.arizona.edu

ABSTRACT

The limited information on the cyberattacks available in the unclassified regime, hardens standardizing the analysis. We address the problem of modeling and analyzing cyberattacks using a multimodal graph approach. We formulate the stages, actors, and outcomes of cyberattacks as a multimodal graph. Multimodal graph nodes include cyberattack victims, adversaries, autonomous systems, and the observed cyber events. In multimodal graphs, single-modality graphs are interconnected according to their interaction. We apply community and centrality analysis on the graph to obtain in-depth insights into the attack. In community analysis, we cluster those nodes that exhibit "strong" inter-modal ties. We further use centrality to rank the nodes according to their importance. Classifying nodes according to centrality provides the progression of the attack from the attacker to the targeted nodes. We apply our methods to two popular case studies, namely *GhostNet* and *Putter Panda* and demonstrate a clear distinction in the attack stages.

Keywords: Cyber-attacks, multimodal graph, community analysis, centrality analysis.

1 INTRODUCTION

The number of cyber-attacks launched on a daily basis against a variety of targets has exploded in recent years (Choo 2011, Hutchings 2014). The goals of such attacks are multifold and range from monetary gains, identity theft and ransomware to exfiltration of secrets and spying (Sanger and Myers 2018), damaging the Internet and other critical infrastructures (Zhu, Joseph, and Sastry 2011, Woolf 2016), and revealing state secrets. Despite the awareness on cyber-security issues, most entities–users, corporations, civic organizations, governments–remain vulnerable to existing and future threats. Protecting against existing threats and, most importantly, predicting future ones depends on the understanding that security experts obtain on the methods used to deploy and propagate cyber-attacks and the interactions between the involved entities.

The main method to analyze cyber-attacks, especially those occurring at scale, is to follow a point-based approach where attack evidence is manually scrutinized to discover and recreate attack attributes. Usually, this process is completed several months after the attack has occurred when traces, actors, and other means of the attack are no longer evident. Moreover, current state-of-the-art cyber-analysis methods have limited predictive and attribution capabilities (Xie, Li, Ou, Liu, and Levy 2010, Michael, Wingfield, and Wijesekera 2003). The majority is reactive in nature and relies on evidence produced by post-event electronic trace analysis (Choo 2011, Hutchings 2014). As a result, cyber-attacks can at best be detected while they occur and, in most cases, are discovered well after the attack has terminated.

However, detecting and analyzing attacks is not sufficient. To understand why a cyber-attack has occurred, and most importantly, when it will occur and by whom it may be perpetrated, we need to determine what the identities, values, incentives, and communication modes of the involved individuals are. This paper

SpringSim-ANSS, 2019 April 29-May 2, Tucson, AZ, USA; ©2019 Society for Modeling & Simulation International (SCS)

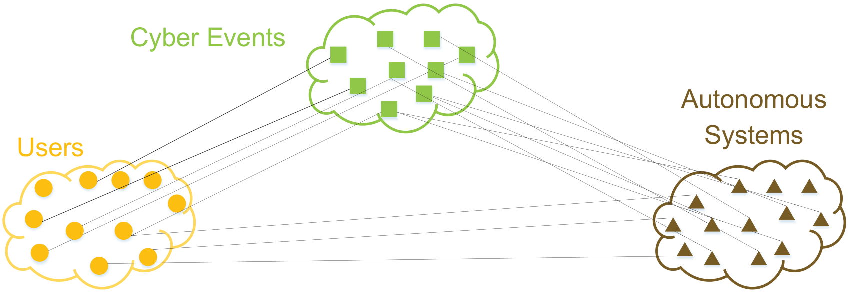


Figure 1: A tripartite graph representing the actors, actions, and means of a cyber attack.

aims at developing systematic methods for classifying adversarial groups based on distinct features of their cyber footprint. To accomplish this goal, we (a) develop comprehensive models of cyber-attack features using feature-extraction techniques on diverse data sources (electronic traces, IP/AS connectivity maps, geolocation, social engineering attack logs, malware databases), (b) classify adversarial groups based on their feature similarities, and (c) enhance the group classification using analytic techniques from social network science. We model cyber attacks as multimodal graphs based on cyber activities, as shown in Fig. 1. Cyber activities were divided into three modalities; users (adversaries and targets), cyber events, and autonomous systems (ASes). This representation associates diverse modalities of information in a unified graph. A graph edge could represent a physical connection between two network nodes, a logical connection between two adversarial entities, or the association of an adversarial entity with an event.

To test the validity of our approach, we analyzed two case studies. The first was a case of malware-based cyber espionage against the Tibetan community, known as *GhostNet* (Deibert, Rohozinski, Manchanda, Villeneuve, and Walton 2009). The evidence collected between June 2008 and March 2009 suggested that the suspicious malware had infected over 1,200 computers, of which 30% was considered as high-value diplomatic and political targets. The purpose of the malware was to exfiltrate sensitive information related to diplomatic and other issues of the Tibetan community at large. Our analysis was based on an evidence-based report published by the Citizen Lab and the SecDev group (Deibert, Rohozinski, Manchanda, Villeneuve, and Walton 2009). The second attack was a case of malware-based cyber espionage targeting the Government, Defense, Research, and Technology sectors in the United States. This attack, which has become known as the *Putter Panda* (Team 2014), started in 2007 and continued until 2012. The purpose of the malware was to exfiltrate sensitive documents from the targeted groups. Our analysis was based on the extensive evidence-based report published by the Crowdstrike Global Intelligence Team (Team 2014).

To analyze each case, we first model each attack as a multimodal graph. We perform community analysis and Bonacich centrality analysis (Bonacich 1987) on the graph. In particular, we apply the algorithm proposed by Melamed *et al.* (Melamed, Breiger, and West 2013) to discover node clusters with “strong” inter-modal ties. Moreover, we rank the nodes in the graph according to their importance using the centrality measure (Bonacich 1987). By classifying nodes according to importance, we deduce the progression of the attack from the attacker nodes to the victims. We also use the negative Bonacich centrality measure (Bonacich 1987) to recognize nodes which control the attack. Using the knowledge gained from the use cases, we make generalized deductions for similar attack types, which can help predict and prevent future cyber-events.

The paper is organized as follows. In Section 2, we describe the different modalities that we use to model cyber-attacks and the algorithms for community and centrality analysis. In Section 3, we give a brief summary of the use cases that we analyzed (GhostNet and Putter Panda) and build the corresponding multi-mode graphs, perform the community discovery and centrality analysis on the graphs and make deductions from these findings. In Section 4, we review prior works related to graph analysis of cyber attacks and also review the application of community and centrality on cyber attack analysis. We conclude in Section 5.

2 MODELING CYBER-ATTACKS AS MULTIMODAL GRAPHS

In this section, we model cyber attacks using multimodal graphs. We then discover the attack features by applying community discovery and computing node centrality.

2.1 Creating the Cyber-Attack Graph

To represent the cyber attack as a graph, we organize the data extracted from the attack reports into the three modalities. shown in Figure 1. These are the actors or users, the recorded cyber events, and the autonomous systems (ASes) over which attacks flow. Although additional modalities can be used to further distill attack attributes, we selected these three modalities due to their generality. After classifying the report data to one of the three modes, we build the adjacency matrix for a tripartite graph where the edges represent the relations between the nodes. First, we discuss the three modalities in detail.

User modality U: The user modality represents any identified nodes that initiated an attack vector and their respective targets. Adversarial users may include command and control servers, compromised hosts that infect other hosts, reflection hosts, nodes that are recruited by Botnets, and any other unique entity that participates in the attack. These entities are differentiated by their reported Internet Protocol (IP) addresses. The target nodes include any end node that is affected by the attack and may include servers, routers, and end-user machines. Like the adversarial users, target users have uniquely attributed to IP addresses.

Cyber Events modality E: Graph nodes in the cyber event modality represent any visible actions occurring between the users in the cyber domain. These include social engineering (phishing) events use to gradually gain access to more important users and organizations, port scanning actions for identifying vulnerabilities, malware infection, data exfiltration, etc.

Autonomous Systems modality AS: Much of the evidence available in the attack reports is tied to specific IP addresses that were discovered during forensic analysis. However, including raw IP addresses in the graph representation fails to exploit the hierarchical structure of the IP addressing system. To capture the possible correlation between the IP addresses, we attribute them to the Autonomous Systems (ASes). We use the CAIDA IPv4/IPv6 Prefix-to-Autonomous System (AS) mappings (Caida 2018) derived from RouteViews data (of Oregon 2018) to map IP addresses to AS numbers and use them as nodes in the graph representation.

We employ an adjacency matrix A to describe the multimodal cyber attack graph. The element $a_{ij} \in \mathbf{A}$ describing the graph between a set $\mathbf{X}$ and a set $\mathbf{Y}$ is equal to one if an edge exists between the i^{th} element of $\mathbf{X}$ and j^{th} element of $\mathbf{Y}$. Otherwise, $a_{ij} = 0$. Let U be the set of users with cardinality $|U| = u$, E be the set of events with cardinality $|E| = e$, and AS be the set of autonomous systems with cardinality $|AS| = s$. We generate three adjacency matrices to capture the intra-modal relationship between the three modalities. Adjacency matrix $\mathbf{A_{U,E}}$ with dimension $u \times e$ shows the relationships between cyber events and users, $\mathbf{A_{U,AS}}$ with dimension $u \times s$ shows the relationships between ASes and users, and $\mathbf{A_{E,AS}}$ with dimension $e \times s$ shows the relationships between cyber events and ASes. Note that the adjacency matrix representation ignores the intra-nodal relationships between nodes within the same set. This is because most forensic analysis report primarily focuses on the inter-nodal relationships.

2.2 Graph Analysis

We now briefly discuss the algorithms implemented for community discovery and centrality analysis. The main motivation for adopting the particular methods was their applicability to multimodal graphs.

2.2.1 Community Discovery

Communities within graphs refer to densely – connected subsets of vertices. Such communities exhibit common properties whose interpretation depends on the data modeled by the graph. In our context, a community refers to a collection of vertices that are strongly correlated and span multiple modalities. Each community may signify different attributes of the attack such as a set of victims targeted by the same adversarial users, the exploitation of common vulnerabilities to compromise the targets, or the use of distinct autonomous systems to perpetrate the attacks. Our goal here is to apply community discovery on the multimodal graph to characterize the attributes of attacks. Moreover, we aim at revealing possible hidden correlations between modalities that are not visible if the different graph modalities were examined independently.

Discovering cohesive subgroups, cliques, modules, or communities in networks has a rich history in the computer and social sciences (Fiedler 1973, Newman and Girvan 2004, Jackson 2010, Fortunato 2010). However, most prior works focus on discovering communities within graphs of the same modality. Melamed *et al.* (Melamed, Breiger, and West 2013) presented a novel extension to the Newman's spectral partitioning approach of modularity maximization (Newman 2006) and applied it to multimodal graphs. We chose the method proposed by Melamed *et al.* (Melamed, Breiger, and West 2013) to discover communities because it is directly applicable to the graph we generate and the community partitioning is optimal as it maximizes modularity. The steps for community discovering are summarized as follows:

1. Generate the *the observed adjacency matrix* $\mathbf{Z_o}$ using the three adjacency matrices $\mathbf{A_{U,E}}, \mathbf{A_{U,AS}}$, and $\mathbf{A_{E,AS}}$:
$$\mathbf{Z_o} = \begin{bmatrix} 0 & \mathbf{A_{U,E}} & \mathbf{A_{U,AS}} \\ \mathbf{A^T_{U,E}} & 0 & \mathbf{A_{E,AS}} \\ \mathbf{A^T_{U,AS}} & \mathbf{A^T_{E,AS}} & 0 \end{bmatrix}_{n\times n},$$
where $n = (u+a+s)$ denotes the total number of nodes in the graph.
2. Compute *the expected matrix* $\mathbf{Z_p}$ of size $n \times n$ where each element p_{ij} is calculated as:
$$p_{ij} = \frac{k_i \times k_j}{2\times m}, \ \forall\, i, j = 1, \ldots, n.$$
Here, k_x denotes the degree of node x in the observed adjacency matrix $\mathbf{Z_o}$, and $m = \Sigma_n k_i/2$ denotes the total number of edges in $\mathbf{Z_o}$.
3. Compute matrix $\mathbf{Z_b}$ as:
$$\mathbf{Z_b} = \mathbf{Z_o} - \mathbf{Z_p}.$$
4. Compute the eigenvectors of $\mathbf{Z_b}$.
5. Use the eigenvectors to split the graph into c communities. For splitting into two communities ($c = 2$), all nodes with positive value in the first or leading eigenvector or the first column in the eigenvector matrix belong to the first community, and all nodes with negative in the leading eigenvector belong to the second community. Subsequent splits of the nodes into more than two communities ($c > 2$) are identified similarly by looking at the signs of the eigenvector associated with the second leading eigenvalue, and so on. For example, for splitting the graph into three communities ($c = 3$), all nodes with positive value in the leading eigenvector belong to the first community, for the second community, all nodes having a negative value for the first eigenvector and positive value for the second eigenvector or the second column of the eigenvector matrix are selected. Finally, all nodes with negative value in both eigenvectors are placed in the third community.
6. Compute the *community matrix* $\mathbf{S}$ with dimensions $n \times c$, where an element $s_{ij} = 1$ if the i^{th} node belong to the j^{th} community.
7. Compute the community modularity as

$$Q = \frac{\text{trace}(\mathbf{S}^T \times \mathbf{Z_b} \times \mathbf{S})}{4 \times m}.$$

8. Repeat steps 5 – 7 for different values of c to maximize the value of Q. The number of communities is upper bounded by one plus the number of positive eigenvalues of $\mathbf{Z_b}$ (Newman 2006).

The optimal community splitting is archived when the community modularity value is maximized. By studying splitting of the communities we learn about the interaction between various nodes. To further discover the importance of individual nodes in the attack, we study node centrality.

2.2.2 Centrality

In graph theory, centrality has been used as a measure for identifying the most influential nodes in a graph, by ranking all nodes according to some criterion of importance. We adopt both positive and negative centrality in our context to rank the nodes of the graph. Positive centrality is the rank assigned to a node i according to the number of distinct shortest paths from i to every other node normalized overall shortest paths. In our context, we use positive centrality to discover densely-connected nodes that could play a pivotal role in the attack. These can be key victim nodes which were used to propagate the attack at high rates, or command and control servers that controlled a large number of hosts.

Moreover, negative centrality has been used to rank nodes according to their role in interconnecting graph cliques. It is defined as the rank assigned to a node i according to the probability that a random path starting at i reaches any other node in the graph normalized overall random walks. If the graph has well-separated communities, the values of the negative centrality are similar for nodes in the same clique whereas nodes connecting cliques exhibit higher centrality values. Bonacich used negative centrality to measure the controlling power of a node (Bonacich 1987). He defined a node to be in a controlling position if it connects two cliques of a graph. In our context, we use negative centrality to identify nodes that “bridged” key communities or modalities. The steps for computing the node centrality are summarized as follows:

1. Compute the observed adjacency matrix $\mathbf{Z_o}$.
2. Compute the positive and negative centrality according to the following equation

$$\mathbf{C} = [\alpha(I - \beta\mathbf{Z_o})^{-1}\mathbf{Z_o}O]_{n\times 1},$$

where O is a $n \times 1$ column matrix of ones, I is the identity matrix of size $n \times n$, α is a normalization factor, and β is used to control the length of the paths considered in the ranking. For the positive centrality calculation, β is set to the inverse of the largest eigenvalue of $\mathbf{Z_o}$, whereas *beta* is set to the inverse of the smallest negative eigenvalue when the negative centrality is computed.

In matrix $\mathbf{C}$, the first u rows denote the centrality of the users set U, the following e rows denote the centrality of the cyber events E, and the last s rows give the centrality of the ASes AS. The value of β governs the path lengths considered in the centrality computation. When β is close to zero, only direct connections are used to assess the centrality. Larger values correspond to larger radii of concern. By tuning β we can make centrality to correspond to a local or a global connectivity measure. For our application, we are interested in recognizing nodes who are locally central or central in a community. Hence, we compute the centrality for β close to zero.

3 CYBER ATTACK CASE STUDIES

To test the effectiveness of a multimodal graph approach on cyber-attack analysis, we consider two well-known case studies for which public data is available. Specifically, we analyzed the detailed unclassified reports for the GhostNet (Deibert, Rohozinski, Manchanda, Villeneuve, and Walton 2009) and Putter Panda (Team 2014) real-world attacks.

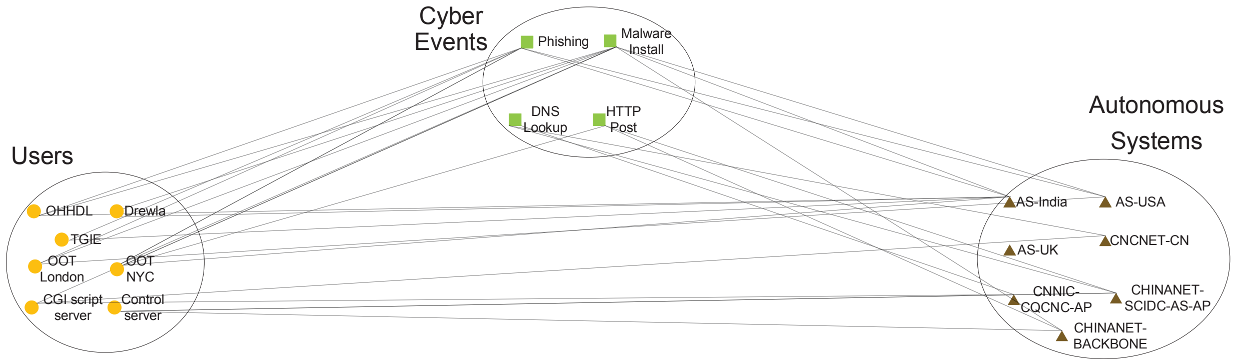


Figure 2: GhostNet attack graph. Circle represent nodes in the the user sets, squares represent nodes in the cyber event set, and triangles represent nodes in the AS set.

3.1 The GhostNet Attack

From June 2008 to March 2009, the Citizen Lab and the SecDev group conducted an extensive investigation of a cyber-espionage case against the Tibetan community around the globe, known as Ghostnet (Deibert, Rohozinski, Manchanda, Villeneuve, and Walton 2009). In Ghostnet, malware was encapsulated in legitimate email messages and email attachments for the purpose of exfiltrating sensitive data. Many of the IP addresses of the discovered adversarial nodes were geolocated to Chinese-owned ASes. Earlier, these attacks were believed to be restricted to the Tibetan community, but similar attack traits were traced to other users who supported the Tibetan community.

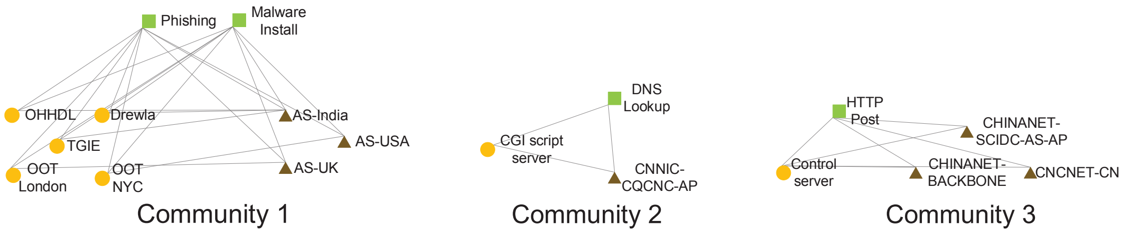
Generating the Adjacency Matrix: First, we describe how we used the report to define the node sets for Ghostnet and then generate the adjacency matrix of the tripartite graph representation of the attack.

User set U: The set of users consisted of all the targets discovered in the GhostNet investigation. Moreover, the set included all adversarial hosts that were discovered, which can be categorized to two types: 1) common gateway interface (CGI) script servers which controlled the web-based interfaces for sending and receiving data to/from the malware and 2) C2 servers which received the exfiltrated data and sent commands to the malware as well.

Cyber Events set E: The GhostNet attack was initiated from a social engineering event in which first, novice users were targeted through phishing emails to gain access to information about important users in the organization. The social engineering consisted of a phishing event followed by the malware install. Once the malware was installed, it exfiltrated any information it deemed valuable according to a set of predefined keywords. To exfiltrate data, the malware first performed a DNS lookup and used an HTTP post to send data to the remote C2 servers. According to this analysis, E consists of four events: phishing, malware install, DNS lookup, and HTTP post.

Autonomous Systems set AS: The AS set consisted of all the distinct ASes that were discovered based on the IP addresses in the user set. GhostNet used different website domains registered under the same name as C2 servers. The IP addresses of these websites belong to ASes in China. The reports concealed the IP addresses of the victim hosts, to preserve host privacy. We used geolocation information to assign users to unique ASes. In particular, we used AS-India, AS-USA, and AS-UK for any host located in India, USA, and UK, respectively. The identified adversarial ASes were 1) CHINANET-BACKBONE, 2) CNNIC-CQCNC-AP, 3) SENNB-AS-KR, 4) CNCNET-CN, and 5) CHINANET-SCIDC-AS-AP.

From the GhostNet report, we generated the tripartite graph shown in Figure 2. The edges of the graph were placed according to the cyber events stated in the report. For instance, an infected host connected to the



Community 1

Community 2

Community 3

Figure 3: GhostNet communities. Circles represent the user modality, squares represents the cyber events modality, and triangles represent the AS modality.

phishing and malware install cyber events. Moreover, if exfiltration took place, is connected to the DNS lookup and HTTP post events. The receiving C2 server also connected to those events.

3.1.1 Community Discovery in Ghostnet

We applied the community discovery algorithm described in Section 2.2.1 on the tripartite graph of Figure 2. The modularity measure was maximized at $Q_{\max} = 0.452$ when the graph was split into three communities. The three discovered communities were:

Community 1: The first community included all the victim hosts, namely Office of His Holiness the Dalai Lama (OHHDL), Tibetan Government-in-exile (TGIE), Drewla, Office of Tibet (OOT) London, and OOT NYC, the ASes through which the victim hosts were reached and the cyber events that initially took place at the victims.

Community 2: The second community included the CGI Script server which performed the DNS Lookup cyber event to collect all the data that is uploaded by the malware.

Community 3: Finally, the third community included the control server that performed the HTTP Post and all the ASes that hosted server instances.

From the community discovery, we can see a clear classification of the attack stages and the involved entities from each modality and at each stage. We can infer the actions that took place, the targeted hosts, and the locations of the victims. Moreover, the community discovery successfully isolated the actions of the adversarial nodes.

3.1.2 Centrality of GhostNet

To analyze the GhostNet attack from the centrality standpoint, we employed the method described in Section 2.2.2. For computing the positive centrality, we set $\alpha = 1$, and $\beta = 0.212$. That is, we took into account all possible paths in the graph as a normalization factor, but concentrated on the impact of relatively short paths. The results show that most nodes have a centrality equal to 0. The cyber events related to phishing and the malware infection of the victim hosts exhibited the highest centrality in the graph, indicating their crucial role in the entire operation, whereas the victim hosts were the least central. This is intuitive since the victim hosts are the endpoints of the attack.

We further analyzed the negative centrality by setting $\beta = -0.285$ which is the inverse of the smallest eigenvalue. The highest negative centrality was exhibited by the C2 servers whereas the lowest negative centrality was exhibited by the victim hosts. Again this is intuitive because the C2 servers were acting as the connecting entity between the victims and the cyber events.

3.2 The Putter Panda Attack

A global intelligence team named CrowdStrike had been tracking cyber spying activity originating from Shanghai, China until 2012. The activities were traced to the People's Liberation Army Third General Staff Department (GCD), 12th Bureau Military Unit Cover Designator (MUCD). CrowdStrike named this adversary group as "Putter Panda". Putter Panda targeted high-profile government, defense, research, and technology sectors in the United States. More specifically, they targeted US Defense and European satellite and aerospace industries including the Toulouse Space Center in Europe and telecommunication companies in Europe and Japan. The attack exploited vulnerabilities in the Adobe Reader and Microsoft Office platforms to deploy custom malware embedded in email attachments. First, the attacker sent phishing emails to the victim with attachments that were opened in Microsoft Office or Adobe Reader. The vulnerability of Microsoft Office and Adobe Reader was exploited by the attacker in attaching Malware as C++ code which ran in the background to install malware on the victim hosts when the attachment was opened. This malware connected to and from Command and Control server through HTTP Get and Post to get instructions and uploaded ex-filtrated data respectively.

Generating the Adjacency Matrix: First, we describe how we generated the adjacency matrix of the attack from the forensics report.

Users set U: The user set for the Putter Panda attack consisted of the victim hosts who, according to the report, were the US government sector, the US defense sector, the US research sector, the US technology sector, European aerospace companies, the Toulouse Space Centre, and telecommunication companies in Japan. No further details about the victim identities were included in the report, so these entities were included as part of the user set. Moreover, all adversarial hosts were traced to Shanghai, China. The report named these hosts as a C2 server, Chen Ping aka CPYY, httpchen, Comment Panda, Vixen Panda, and pngdowner.

Cyber Events set E: Similar to the GhostNet attack, the Putter Panda attack was bootstrapped using social engineering, followed by malware infection and extraction of valuable data. The phishing and malware install events were represented as individual nodes in E. Moreover, many events were included that represented the malware operation. The attack deployed several types of RAT malware including 4H RAT, 3PARA RAT, pngdowner and httpclient malware. All events associated with each malware and described in the report were included in E. In total, this set contained 39 cyber events.

Autonomous Systems set AS: The Putter Panda report names several domains that hosted C2 servers. These domains were traced to IP addresses which were further traced to ASes that owned the corresponding IP space. Similar to the GhostNet, the ASes of the victim users of Putter Panda were hidden to preserve the privacy of the victims. Hence, their ASes was mapped to their geolocation as AS-USA, AS-Europe, and AS-Japan. The ASes of adversarial nodes were mapped to real AS numbers (attacker IPs were known) and included fifteen ASes in the USA, five in the Taiwan, four in the Korean peninsula, two in the Netherlands, and one each in Denmark, Kazakhstan, Fiji, Thailand, China, Japan, Russia, Great Britain, and Indonesia.

From the three sets and the CrowdStrike report, we generated the tripartite graph and the corresponding adjacency matrix that represents the graph. We do not show the graph result due to its visual complexity. The edges of the graph were placed according to the cyber events stated in the report.

3.2.1 Community Discovery in Putter Panda

We applied the community discovery algorithm described in Section 2.2.1. The modularity measure was maximized at $Q_{\max} = 0.361$ when the graph was split into three communities. The three discovered communities were:

Community 1: The first community consisted of victim hosts and cyber events occurring on those hosts related to the malware infection (phishing plus malware install). The community further included those ASes who served the victim hosts. Interestingly, the community also included the AS were the C2 server of the httpclient malware resided. We consider this to be an aftermath of the relatively low connectivity of this AS (it did not fit well with any community).

Community 2: The second community consisted of adversarial users, events, and ASes shared between the C2 servers of 3PARA RAT, pngdowner and httpclient.

Community 3: The third community consisted of users, events, and ASes associated with the C2 server of 4H RAT.

The most critical observation of the community discovery is the separation of the 4H RAT malware from all others. The main reason for this separation is the distributed nature of the C2 operation for the 4H malware type. The C2 servers of 4H RAT were spread over a large number of ASes (more than all other malware types combined). The community discovery allowed us to identify this unique pattern of 4H RAT.

3.2.2 Centrality of Putter Panda

To analyze the graph centrality, we ranked every graph node according to both positive and negative centrality. To compute the positive centrality, we set $\beta = 0.198$ based on the inverse of the largest eigenvalue of the adjacency matrix. The largest positive centrality was recorded for the cyber events related to the malware infection, followed by the victim hosts and the corresponding ASes that serve them. This ranking reveals that the most important phase of the attack is the infection stage during which the victim host render control to the C2 servers.

To compute the negative centrality, we calculated $\beta = -0.115$ and ranked all nodes of the graph. The highest-ranked nodes were cyber events related to C2 server operations related to data exfiltration. This is because this was the most common operation among all malware types and uniquely connected victim hosts with C2 servers and their respective ASes. The events were in a controlling position during the attack as they acted as a broker for transferring commands from the C2 servers to the victims and extracting any information that was deemed valuable.

4 RELATED WORKS

In this section, we review work related to the analysis of cyber attacks. We contrast the analysis presented in this work with the existing works in the same area. Further, we discuss various application community discovering and clustering algorithms and centrality application in the area of analysis of cyber attacks.

4.1 Analysis of Cyber Attacks using Multimodal Graphs

The graphical analysis had been used extensively in the past for the analysis of cyber attacks (Kaynar 2016). In the past, researchers have used Bayesian network analysis for cyber attacks (Poolsappasit, Dewri, and Ray 2012, Pecchia, Sharma, Kalbarczyk, Cotroneo, and Iyer 2011, Wu, Yin, and Guo 2012). The main idea is that with the use of background knowledge of attackers and attack mechanisms, Bayesian analysis recognizes potential attack paths in a system and develops algorithms to compute an optimal subset of attack paths. The researchers have also performed the Hidden Markov Model (HMM) analysis of cyber attacks (Liu and Liu 2016, Wang, Zhang, and Kadobayashi 2013). In these works, authors first map the behavior of the adversary to an attack graph model, which is then mapped problematically using HMM, and then the Viterbi algorithm was used to calculate the maximum probability state transition sequence. Authors have also performed a Schmitt-analysis in linking social events to cyber attacks (Caton 2013). However, the case studies are of assumed ground truth rather than considering a real-world attack as we do in this paper.

4.2 Community Discovering and Clustering Algorithms

Clustering algorithms had been used in past to detect cyber attacks such as intrusion detection (Shittu, Healing, Ghanea-Hercock, Bloomfield, and Rajarajan 2015, Casas, Mazel, and Owezarski 2012), anomaly detection (Akoglu, Tong, and Koutra 2015), and many more. The main idea behind the intrusion detection using a clustering algorithm is based on the main idea of clustering the attack nodes in the graph together. Event correlation proposed by Shittu *et al.*(Shittu, Healing, Ghanea-Hercock, Bloomfield, and Rajarajan 2015) use the post-correlation methods to cluster the correlated attacks together. Akoglu et al. (Akoglu, Tong, and Koutra 2015) have summarized various applications of clustering approach to detect anomaly detection when the attacker is tampering sensitive data. In the work on network intrusion detection (Casas, Mazel, and Owezarski 2012), have proposed a technique which uses Sub-Space Clustering and Multiple Evidence Accumulation for automatic network intrusion detection. These techniques target to cluster and detect the attacker. Whereas in our work, we propose to detect clusters or communities which shows both adversaries and attacked nodes.

4.3 Centrality Algorithms

Closeness centrality measure was implemented by the authors (Srivastava, Morris, Ernster, Vellaithurai, Pan, and Adhikari 2013) to detect an attack in a smart grid setting, where some of the attack data is missing. Chen *et al.*(Chen, Choudhury, and Hero 2016) proposed an intrusion detection system based on multi-centrality graph analysis of principal component analysis. In addition, researchers (Du and Yang 2011) have used the basic idea of degree centrality to show the relationship between attack sources, and collaborative attack patterns. Researchers have also used centrality metric derived from complex network theory and social network analysis to identify the malicious attacks on the electric power transmission grid. Wang *et al.*(Wang, Scaglione, and Thomas 2010) have derives the importance measure from the centrality of a graph to detect vulnerable nodes in a smart grid. Researchers (Genge, Kiss, and Haller 2015) have also used the centrality measure to detect anomalies for detecting the cyber attack. Contrasting these works in this paper we use centrality measure to recognize the important nodes which were attacked as well as used negative centrality to detect nodes which facilitate the attack by transmitting commands and results.

5 CONCLUSION

We addressed the problem of modeling and analyzing cyber attacks using a multimodal graph approach. We showed how cyber events, actors, victims, and networks that connect them can be jointly modeled using a multimodal graph, where several graphs of various modalities are combined to represent the attributes of the attack. We used community discovery analysis to identify coalitions of nodes of different modalities that exhibited strong correlation. We further used centrality to rank the nodes of the graph according to their importance in the attack. We applied our methods to two popular case studies, namely *GhostNet* and *Putter Panda*. Our findings demonstrated a clear distinction between the victim hosts and the ASes that serve them, the command and control servers that regulated the attack, and the data exfiltration process. Moreover, the centrality analysis revealed the key nodes that facilitated the attack, a role that was commonly assumed by command and control servers. For future work, we propose to embed the derived properties as ground truth to generate synthetic data. We will study the data to derive new techniques for detecting attacks in real-time.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful comments. This research was supported in part by the NSF under grants CNS-1347075 and CNS-1409172. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the NSF.

References

Akoglu, L., H. Tong, and D. Koutra. 2015. “Graph based anomaly detection and description: a survey”. *Data mining and knowledge discovery* vol. 29 (3), pp. 626–688.

Bonacich, P. 1987. “Power and centrality: A family of measures”. *American journal of sociology* vol. 92 (5), pp. 1170–1182.

Caida, NetGeo 2018. “Routeviews Prefix to AS mappings Dataset (pfx2as) for IPv4 and IPv6”. http://www.caida.org/data/routing/routeviews-prefix2as.xml.

Casas, P., J. Mazel, and P. Owezarski. 2012. “Unsupervised network intrusion detection systems: Detecting the unknown without knowledge”. *Computer Communications* vol. 35 (7), pp. 772–783.

Caton, J. L. 2013. “Exploring the prudent limits of automated cyber attack”. In *Proc. of international conference on Cyber conflict (CyCon)*, pp. 1–16. IEEE.

Chen, P.-Y., S. Choudhury, and A. O. Hero. 2016. “Multi-centrality graph spectral decompositions and their application to cyber intrusion detection”. In *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4553–4557. IEEE.

Choo, K.-K. R. 2011. “The cyber threat landscape: Challenges and future research directions”. *Computers & Security* vol. 30 (8), pp. 719–731.

Deibert, R. J., R. Rohozinski, A. Manchanda, N. Villeneuve, and G. Walton. 2009. “Tracking GhostNet: Investigating a cyber espionage network”. *Information Warfare Monitor*, pp. 6.

Du, H., and S. J. Yang. 2011. “Discovering collaborative cyber attack patterns using social network analysis”. In *Proc. of International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*, pp. 129–136.

Fiedler, M. 1973. “Algebraic connectivity of graphs”. *Czechoslovak mathematical journal* vol. 23 (2), pp. 298–305.

Fortunato, S. 2010. “Community detection in graphs”. *Physics reports* vol. 486 (3-5), pp. 75–174.

Genge, B., I. Kiss, and P. Haller. 2015. “A system dynamics approach for assessing the impact of cyber attacks on critical infrastructures”. *International Journal of Critical Infrastructure Protection* vol. 10, pp. 3–17.

Hutchings, A. 2014. “Crime from the keyboard: organised cybercrime, co-offending, initiation and knowledge transmission”. *Crime, Law and Social Change* vol. 62 (1), pp. 1–20.

Jackson, M. O. 2010. *Social and economic networks*. Princeton university press.

Kaynar, K. 2016. “A taxonomy for attack graph generation and usage in network security”. *Journal of Information Security and Applications* vol. 29, pp. 27–56.

Liu, S.-c., and Y. Liu. 2016. “Network security risk assessment method based on HMM and attack graph model”. In *2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)*, pp. 517–522. IEEE.

Melamed, D., R. L. Breiger, and A. J. West. 2013. “Community structure in multi-mode networks: Applying an eigenspectrum approach”. *Official Journal of the International Network for Social Network Analysts*.

Michael, J. B., T. C. Wingfield, and D. Wijesekera. 2003. “Measured responses to cyber attacks using Schmitt analysis: a case study of attack scenarios for a software-intensive system”. In *Proc. of COMPSAC 27th Annual International Computer Software and Applications Conference*, pp. 622–626. IEEE.

Newman, M. E. 2006. “Modularity and community structure in networks”. *Proc. of the national academy of sciences* vol. 103 (23), pp. 8577–8582.

Newman, M. E., and M. Girvan. 2004. “Finding and evaluating community structure in networks”. *Physical review E* vol. 69 (2), pp. 026113.

University of Oregon 2018. “Route Views Project”. http://www.routeviews.org/routeviews/.

Pecchia, A., A. Sharma, Z. Kalbarczyk, D. Cotroneo, and R. K. Iyer. 2011. “Identifying compromised users in shared computing infrastructures: A data-driven bayesian network approach”. In *Proc. of IEEE International Symposium on Reliable Distributed Systems*, pp. 127–136. IEEE.

Poolsappasit, N., R. Dewri, and I. Ray. 2012. "Dynamic security risk management using bayesian attack graphs". *IEEE Transactions on Dependable and Secure Computing* vol. 9 (1), pp. 61–74.

Sanger, David E. and Myers, Steven Lee 2018. "After a Hiatus, China Accelerates Cyberspying Efforts to Obtain U.S. Technology". https://www.nytimes.com/2018/11/29/us/politics/china-trump-cyberespionage.html.

Shittu, R., A. Healing, R. Ghanea-Hercock, R. Bloomfield, and M. Rajarajan. 2015. "Intrusion alert prioritisation and attack detection using post-correlation analysis". *Computers & Security* vol. 50, pp. 1–15.

Srivastava, A., T. Morris, T. Ernster, C. Vellaithurai, S. Pan, and U. Adhikari. 2013. "Modeling cyber-physical vulnerability of the smart grid with incomplete information". *IEEE Transactions on Smart Grid* vol. 4 (1), pp. 235–244.

Team, Crowdstrike Global Intelligence 2014. "CrowdStrike Intelligence Report: Putter Panda".

Wang, S., Z. Zhang, and Y. Kadobayashi. 2013. "Exploring attack graph for cost-benefit security hardening: A probabilistic approach". *Computers & security* vol. 32, pp. 158–169.

Wang, Z., A. Scaglione, and R. J. Thomas. 2010. "Electrical centrality measures for electric power grid vulnerability analysis". In *Proc. of IEEE Conference on Decision and Control (CDC)*, pp. 5792–5797. IEEE.

Woolf, Nicky 2016. "DDoS attack that disrupted internet was largest of its kind in history, experts say". https://www.theguardian.com/technology/2016/oct/26/ddos-attack-dyn-mirai-botnet.

Wu, J., L. Yin, and Y. Guo. 2012. "Cyber attacks prediction model based on Bayesian network". In *Proc. of IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, pp. 730–731. IEEE.

Xie, P., J. H. Li, X. Ou, P. Liu, and R. Levy. 2010. "Using Bayesian networks for cyber security analysis". In *Proc. of IEEE/IFIP international conference onDependable Systems and Networks (DSN)*, pp. 211–220. IEEE.

Zhu, B., A. Joseph, and S. Sastry. 2011. "A taxonomy of cyber attacks on SCADA systems". In *2011 IEEE International Conferences on Internet of Things, and Cyber, Physical and Social Computing*, pp. 380–388. IEEE.

AUTHOR BIOGRAPHIES

NIRNIMESH GHOSE is a Ph.D. candidate in the Electrical and Computer Engineering Department at the University of Arizona. His research interests include wireless network security, physical layer authentication, aviation security, and interaction between cybersecurity and social networks. His email address is nghose@email.arizona.edu.

LOUKAS LAZOS is an Associate Professor of Electrical and Computer Engineering at the University of Arizona. He received his Ph.D. in Electrical Engineering from the University of Washington in 2006. In 2007, he was the co-director of the Network Security Lab at the University of Washington. His main research interests are in the areas of wireless security, user privacy, wireless communications, cloud security, and network visualization. His email address is llazos@email.arizona.edu.

JERZY ROZENBLIT is a University Distinguished Professor, Raymond J. Oglethorpe Endowed Chair in the Electrical and Computer Engineering Department, with a joint appointment as Professor of Surgery in the College of Medicine at the University of Arizona. During his tenure at the University of Arizona, he established the Model-Based Design Laboratory with major projects in design and analysis of complex, computer-based systems, hardware/software codesign, and simulation modeling. He presently serves as Director of the Life-Critical Computing Systems Initiative, a research enterprise intended to improve the reliability and safety of technology in healthcare and life-critical applications. His email address is jerzyr@email.arizona.edu.

RONALD BREIGER is a Regents' Professor and professor of sociology at the University of Arizona. He works in the areas of social network theory and methods, networks and culture, and measurement issues in cultural and institutional analysis. With John Mohr and Robin Wagner-Pacifici, he imagines and formulates approaches for automated analysis of national security texts. With Meltem Odabaş and Thomas Holt, he is studying the economic sociology of networks in online stolen data markets. With David Melamed and Eric Schoon, he continues work on a dual network recasting of regression modeling that places variables and cases within a common space. His email address is breiger@email.arizona.edu.