2018 1st International Conference on Data Intelligence and Security

Community Finding of Malware and Exploit Vendors on Darkweb Marketplaces

Ericsson Marin, Mohammed Almukaynizi, Eric Nunes and Paulo Shakarian
Arizona State University
Tempe, Arizona
Email: {esmarin, malmukay, enunes1, shak}@asu.edu

***Abstract*—Many people involved in malicious cyber activity rely on online environments to improve their hacking skills and capabilities, among which, darkweb marketplaces are one of the most prevalent. Vendors advertise and sell their wares worldwide on those markets, generating communities of like-minded individuals focused on sub fields of hacking. As there is no direct communication between vendors in these environments, identifying the communities formed by them becomes challenging; especially with the absence of ground truth knowledge to validate the results. In this paper, we develop a method based on Machine Learning and Social Network Analysis (SNA) to identify and validate communities of malware and exploit vendors, using product offerings in 20 different marketplaces on the darkweb. To validate the viability of our approach, we cross-validate the community assignments of common individuals selling their products on two mutually exclusive sets of marketplaces, demonstrating how the multiplexity of social ties can be used to detect and validate communities of malware and exploit vendors.**

I. Introduction

Recently, darkweb sites have become the main venue for online purchasing of malicious hacking products and services by cyber criminals. An example that illustrates this fact is given by Nunes et. al in [1]. An exploit targeting Microsoft Windows operation system was for sale on a darkweb market in March 2015. The vulnerability was disclosed by Microsoft a month earlier, with no publicly available exploit at that time. Four months after the availability of the exploit, FireEye[1] reported that the Dyre banking Trojan, designed to target organizations to steal credit card information, used the exploit.

In this context, consider the importance of finding communities of vendors with similar hacking expertise for surveillance purposes. Many vendors possibly linked to the Dyre Banking Trojan, could be automatically identified if at least one of them had been already confirmed as offering the exploit online. In more complex scenarios, those communities might correspond to sets of individuals dealing with similar products or services in multiples sub fields of hacking simultaneously, such as carding, phishing and keyloggers. Therefore, identifying malicious hacking vendors' communities may reveal patterns about the structure, organization, operation, and information flow of their corresponding networks, helping intelligence agencies target critical communities for removal or surveillance [2]

In this paper, we explore a new method based on social network analysis and machine learning techniques to identify and validate communities of malware and exploit vendors on darkweb marketplaces. We collect information about hacking-related product offerings in 20 different markets, from where we produce a similarity matrix of the vendors. To create this matrix, we leverage unsupervised learning to cluster the vendors' products into 34 hacking categories according to [3].

Then, we quantify the similarity between vendors analyzing the number of product categories shared between them and also the number of products they have in each product category. Finally, as a way to address the lack of ground truth (the existing communities), we split the marketplaces into two disjoint sets, in order to detect the community overlapping between them. We believe the multiplexity of social ties [4], which makes individuals interact in multiple domains, can help us to validate a considerable part of the mined communities.

This paper makes the following main contributions: 1) We cluster around 40,000 hacking-related products gathered in [1] using 34 product categories specified in [3]; 2) We calculate the similarity of malicious hacking vendors using four different metrics, based on their shared product categories and their corresponding number of products, to infer the implicit connections between them; 3) We perform community finding to detect the communities of hacking-related vendors in two disjoint sets of marketplaces, validating our results by checking the overlapping between them. We found the ARI score achieves 0.445 using our method, while randomly assigning individuals to communities yields an ARI of -0.006.

II. Dataset of Darkweb Marketplaces

In this work, we collect data provided by a commercial version of the system described in [1], from where we select 20 popular English hacking-related marketplaces on the darkweb [2]. Table I shows the size of our original dataset, including number of markets, products and vendors.

TABLE I
SCRAPED DATA FROM DARKWEB MARKETPLACES.

Original		Filtered	
Marketplaces	20	Marketplaces	20
Products (Total)	74,200	Products (Total)	40,610
Products (Distinct)	51,902	Products (Distinct)	27,581
Vendors	7,055	Vendors	390

[1]A major cybersecurity firm.

[2]Collection of websites that exist on encrypted networks of deepweb. It is a region that is intentionally and securely hidden from view, being not found by traditional search engines and not visited by traditional browsers [5].

978-1-5386-5762-1/18/$31.00 ©2018 IEEE
DOI 10.1109/ICDIS.2018.00019

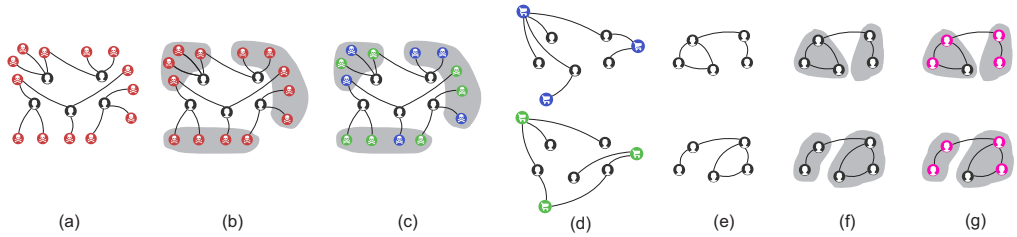

Fig. 1. System overview. (a) Creating the bipartite network of Vendors X Products, (b) Clustering the products in product categories, (c) Splitting the marketplaces into two disjoint sets (green and blue), (d) Creating a bipartite network of Vendors X Product Categories for each set of markets, (e) Projecting the bipartite networks (Vendors X Products Categories) to monopartite networks (Vendors X Vendors) for each set of markets, (f) Finding the communities of vendors in each set of markets, (g) Calculating the community overlapping between the two set of markets.

III. Methodology

This section explains the steps of our approach designed to address the community finding of malicious hacking-related vendors. Figure 1 illustrates the system overview.

A. Creating the Bipartite Network of Vendors X Products

The first step of our approach consists of collecting the malicious hacking products offered by each vendor on the marketplaces to generate a bipartite network of vendors and their corresponding products. Therefore, the nodes of this bipartite graph are formed by vendors and products, while the edges are created only between these two type of nodes. We apply string-match over the vendors' screen-name and products' names to uniquely identify them in this paper [3].

As our community finding method uses duplicated vendors in different set of marketplaces to validate the results, we filter our dataset considering only vendors present in at least two markets, and naturally, their corresponding products. Table I shows the size of the filtered dataset. Using this filtered dataset, we generate a bipartite graph considering the vendors and distinct products as two disjoint sets of nodes, and the total number of products as the connecting edges.

B. Clustering the Products in Product Categories

As we collect data from different sites, there is inconsistency as to how products are categorized on each site - if such non-trivial categorization even exists for a given site. Furthermore, there is a clear absence of a standardized method for vendors to register their products. As a consequence, the majority of the products are unique when compared with simple matching or regular expression technique (we observe that around 70% of the distinct products belong to single vendors).

In order to mitigate this inconsistency, we cluster the products in 34 hacking categories according to [3]. The idea is to make the vendors share more information, assigning similar products to the same product category. This strategy allows us to generate a more precise matrix of vendors similarity, using their shared product categories and the corresponding products. Following the approach in [3], we apply character n-grams in range 3 to 6 over the product names, to engineer features that represent products as vectors. Then, we value all features using TF-IDF, after eliminating stopping words and executing steaming. Finally, we run K-Means using cosine similarity as the distance function (spherical K-Means [6]) in the entire dataset (27,581 distinct products) - to produce the 34 product categories detailed in [3]. The top 5 ones with respect to the number of products are: Netflix-related, Viruses/Counter AntiVirus, VPN, Keyloggers and Linux-related, with 3786, 3216, 3064, 2662, 1875 products respectively.

[3] We leave other methods of similarity-based comparison to future work.

C. Splitting the Marketplaces into Two Disjoint Sets

In this work, we split our dataset into two disjoint partitions of marketplaces. We use an optimization process to produce a division that maximizes the redundancy of vendors, so that we can verify if the duplicated individuals form similar communities in both partitions. Table II presents the results of the data split algorithm, showing that we found 329 duplicated vendors in our two disjoint sets of markets.

TABLE II
Results of the Split of Marketplaces.

	Set_1	Set_2
Number of Markets	10	10
Number of Products	13,486	27,124
Number of Vendors	345	374
Number of Duplicated Vendors in Both Sets of Markets	329	

D. Bipartite Networks of Vendors X Product Categories

At this point, we are able to connect the vendors to their product categories in a bipartite graph, allowing us to check which product categories are shared between them. Figure 2 illustrates this process with a subset of the graph within Set_1.

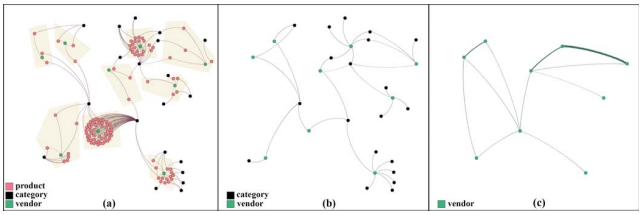

Fig. 2. Sample of the network of vendors, products and product categories (a). Projection of the network of (a) in a network of vendors and product categories (b). Projection of the network of (b) in a network of vendors (c).

In panel (a), we see the vendors connected to their products, which in turn are connected to their product categories. In panel (b), we plot the same graph without the products to better visualize the shared product categories between the vendors. Vendors who were previously disconnected from others (note the 9 disconnected components highlighted in panel (a), since the vendors only own exclusive products) are now connected

TABLE III
SIMILARITY METRICS.

Metric	Formula
Jaccard [8]	$J(V_i, V_j) = \frac{M_{11}}{M_{01}+M_{10}+M_{11}}$, where V_i and V_j are two binary vectors corresponding to the assignment of existing product categories for vendors i and j, M_{11} represents the number of product categories where V_i and V_j both have a value of 1, M_{01} represents the number of product categories where the product category of V_i is 0 and the product category of V_j is 1, and M_{10} represents the number of product categories where the product category of V_i is 1 and the product category of V_j is 0.
Cosine [8]	$Cos(V_i, V_j) = \frac{V_i \bullet V_j}{\|V_i\|\|V_j\|}$, where V_i and V_j are two non binary vectors corresponding to the assignment of the total number of products within each existing product categories that belong to vendors i and j.
Correlation [8]	$Corr(V_i, V_j) = \frac{cov(V_i, V_j)}{\sigma_{(V_i)} * \sigma_{(V_j)}}$, where V_i and V_j are two binary vectors corresponding to the total number of products within each existing product categories that belong to vendors i and j, $cov(V_i, V_j)$ is the covariance of V_i and V_j, and $\sigma_{(V_i)}$ is the standard deviation of V_i.
Tanimoto [8]	$T(V_i, V_j) = \frac{V_i \bullet V_j}{\|V_i\|^2 \|V_j\|^2 - V_i \bullet V_j}$, where V_i and V_j are two non binary vectors corresponding to the assignment of the total number of products within each existing product categories that belong to vendors i and j.

using the shared product categories in panel (b). We observe the majority of vendors ($\approx 63\%$) are assigned to more than one product category in both sets of markets. This increases the probability of creating new connections in the graph, although vendors assigned to only one product category are also in most cases sharing it with other individuals.

E. Projecting Bipartite Networks (Vendors X Product Categories) to Monopartite Networks (Vendors X Vendors)

Our challenge here is to project the bipartite graphs (Vendors X Product Categories) in monopartite graphs (Vendors X Vendors). This step is crucial for this paper, since the algorithms we use to find the communities of vendors are designed to work with networks with only one type of node, and not to work with multimodal networks [7].

To accomplish this task, we create a similarity matrix between vendors using two pieces of information: their product categories and their corresponding products. The former information basically creates a binary matrix connecting vendors and product categories, where "1" means the vendor has a least one product in the corresponding product category, and "0" otherwise. The later information adds magnitude to the product categories owned by vendors, including the number of products vendors have within each product category. We use both information to create a similarity matrix between vendors, considering an edge between two specific individuals if the corresponding similarity (weight) is greater than a given threshold δ. To calculate this weight, we use four different similarity metrics according to Table III.

The idea of calculating those similarity metrics is to use them to weight the edges of our graphs, since those weights should represent the level of similarity between vendors. We rely on the assumption that vendors with a high similarity based on their product offerings will form a community of interests in the real world. Figure 2 illustrates this projection process of a bipartite network of vendors and product categories - panel (b) - to a monopartite network of vendors - panel (c). Now, it is possible see in panel (c) that vendors are directed connected to each others in a weighed graph.

F. Finding the Communities of Vendors

After producing the network of vendors, we search for their potential communities in both sets of marketplaces. For this task, we use the Louvain heuristic method of community detection [7], which optimizes the modularity objective function [9] [4] to uncover a non-overlapping community network structure. Figure 3 shows the produced communities and inform the modularity Q found in Set_1 (0.579) and in the Set_2 (0.514) using Jaccard similarity metric. These values indicate that both networks present a considerable clustering property.

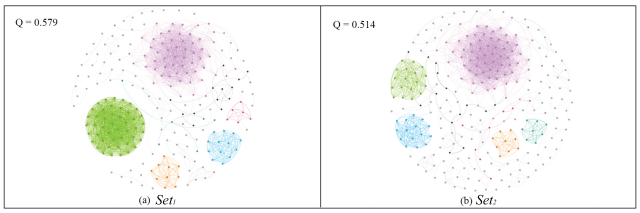

Fig. 3. Communities found for $\delta = 0.51$ in Set_1 (a) and Set_2 (b) using Jaccard similarity metric.

G. Calculating the Vendors Community Overlapping

Finally, we move to the final step of our work: the validation of the found communities. We accomplish this task checking the vendors community overlapping in both sets of marketplaces. A high agreement here would mean a strong similarity between the vendors and consequently a strong likelihood of they belong to the same community in the real world.

In order to calculate this level of agreement between both sets, we use the Adjusted Rand Index (ARI) proposed in [10], which produces a score between [-1,1] accordingly. Jointly, we prune the generated networks varying the threshold δ in [0,0.99] (considering the step as 0.01), to verify how the ARI changes correspondingly. We also analyze the trade-off between the threshold δ and the total number of vendors possible to be identified. Figure 4 shows the results for our networks of vendors created using the four similarity metrics.

As verified in Figure 4, the highest value identified for ARI (0.445) is produced when we use Jaccard similarity to create the network of vendors, and when we set the value of δ as 0.51. These results show that only the product categories are relevant to create the similarity matrix of vendors (binary matrix), and their magnitude (number of products in each category) should be avoided. Here, the number of communities identified in

[4]Modularity measures how much the community structure found is distant from randomly generated ones [9], returning a scalar value in [-1,1].

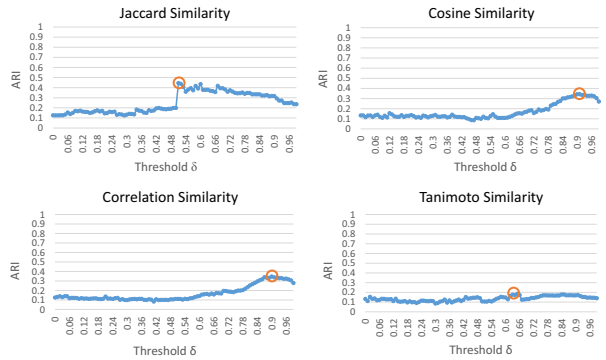

Fig. 4. Curve of ARI produced for our networks when we vary δ.

Set_1 and Set_2 is 37 and 48 respectively, and the number of vendors correctly assigned in both sets to the same community is 169. This number represents 51.3% of the possible number of vendors that could be identified in both sets of markets.

After identifying the communities and the corresponding vendors in both sets, we make a final examination of the distribution of vendors per communities. Our intention is to check if the creation of those agreements between both sets could not be easily done at random. In order to accomplish that, we get the number of vendors present in each community and apply the same distribution to a randomly community assignment method, carrying out this experiment for both sets. Our results show a value of -0.006 calculated for ARI, which demonstrates a non-randomness property of our method.

IV. Related Work

To the best of our knowledge, this is the first applied study where social network information of malware and exploit vendors is derived from darkweb malicious hacking marketplaces. Previous studies examined characteristics of malicious hacking forums, aspects of non-malicious hacking markets, or the products for sale in a hacking market.

In [11], two topic-based social networks were created, one from the topic creator and another from the repliers perspective. The authors tried to identify group of topics as well as group of key-members who created them over a single forum. Yang et. al [12] used the same dataset, but they tried to form clusters of users based on their messages timestamps. Then, they compared user activeness to discover the focused theme of discussion. Anwar et. al [2] treated each post as entity with its own related information, using a collection of 58 Surface Web forums. They tried to cluster those posts using agglomerative clustering, based on similarities between each pair of entities. Unlike our study, the relationships in darkweb forums can be easily observed from the post/reply activities of forum users, while in marketplaces there is no explicit communications amongst vendors. Then, inferring those relationships is a significant contribution of this paper.

Our previous work on marketplaces [13], [1], [3] focused on: 1) a game theoretic analysis of a small subset of the data in this paper; 2) classify products as malicious hacking-related; 3) categorize malicious hacking-related products for sale; and did not attempt to find social structures, such as communities.

V. Conclusions

In this work, we mine communities of malware and exploit vendors on 20 darkweb marketplaces, connecting different vendors based on their product offerings. The multiplexity of social ties allowed us to find these hidden communities using two disjoint sets of markets and then successfully validate the results. We use a combination of product clustering and similarity functions to connect vendors and community finding algorithms to identify their communities. Finally, we analyze the overlapping of the communities identified in our both sets of markets, observing a reasonable value for this metric. Our method is one further step towards comprehending implicit social networks formed on darkweb, helping intelligent agencies to track suspicious hacking-related organizations.

Acknowledgement: Some of the authors were supported by the Office of Naval Research (ONR) and the National Council for Scientific and Technological Development (CNPq-Brazil). Paulo Shakarian is supported by the Office of the Director of National Intelligence (ODNI) and the Intelligence Advanced Research Projects Activity (IARPA) via the Air Force Research Laboratory (AFRL) contract number FA8750-16-C-0112. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of ODNI, IARPA, AFRL, or the U.S. Government.

References

[1] E. Nunes, A. Diab, A. Gunn, E. Marin, V. Mishra, V. Paliath, J. Robertson, J. Shakarian, A. Thart, and P. Shakarian, "Darknet and deepnet mining for proactive cybersecurity threat intelligence," in *Proceeding of ISI 2016*. IEEE, 2016, pp. 7–12.

[2] T. Anwar and M. Abulaish, "Identifying cliques in dark web forums - an agglomerative clustering approach," in *IEE ISI*, June 2012, pp. 171–173.

[3] E. Marin, A. Diab, and P. Shakarian, "Product offerings in malicious hacker markets," in *2016 IEEE Conference on Intelligence and Security Informatics (ISI)*, Sept 2016, pp. 187–189.

[4] D. Watts, *Six Degrees: The Science of a Connected Age*. W. W. Norton, 2004.

[5] J. Robertson, A. Diab, E. Marin, E. Nunes, V. Paliath, J. Shakarian, and P. Shakarian, *Darkweb Cyber Threat Intelligence Mining*. Cambridge University Press, 2017.

[6] K. Hornik, M. Kober, I. Feinerer, and C. Buchta, "Spherical k-means clustering," *Journal of Statistical Software*, vol. 50, Sep 2012.

[7] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[8] P. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*, ser. Always learning. Pearson Addison Wesley, 2006.

[9] M. E. J. Newman, "Modularity and community structure in networks," *Proceedings of the National Academy of Sciences*, vol. 103, no. 23, pp. 8577–8582, 2006.

[10] L. Hubert and P. Arabie, "Comparing partitions," *Journal of Classification*, vol. 2, no. 1, pp. 193–218, Dec. 1985.

[11] G. L'Huillier, H. Alvarez, S. A. Ríos, and F. Aguilera, "Topic-based social network analysis for virtual communities of interests in the dark web," *SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 66–73, Mar. 2011.

[12] C. C. Yang, X. Tang, and X. Gong, "Identifying dark web clusters with temporal coherence analysis," in *IEE ISI 2011*, July 2011, pp. 167–172.

[13] J. Robertson, V. Paliath, J. Shakarian, A. Thart, and P. Shakarian, "Data driven game theoretic cyber threat mitigation," in *Proceedings of the 13th AAAI*, ser. AAAI'16. AAAI Press, 2016, pp. 4041–4046.