Inferring Discussion Topics about Exploitation of Vulnerabilities from Underground Hacking Forums

Felipe Moreno-Vera*
*Federal University of Rio de Janeiro (UFRJ),

***Abstract*—The increasing sophistication of cyber threats necessitates proactive measures to identify vulnerabilities and potential exploits. Underground hacking forums serve as breeding grounds for the exchange of hacking techniques and discussions related to exploitation. In this research, we propose an innovative approach using topic modeling to analyze and uncover key themes in vulnerabilities discussed within these forums. The objective of our study is to develop a machine learning-based model that can automatically detect and classify vulnerability-related discussions in underground hacking forums. By monitoring and analyzing the content of these forums, we aim to identify emerging vulnerabilities, exploit techniques, and potential threat actors. To achieve this, we collect a large-scale dataset consisting of posts and threads from multiple underground forums. We preprocess and clean the data to ensure accuracy and reliability. Leveraging topic modeling techniques, specifically Latent Dirichlet Allocation (LDA), we uncover latent topics and their associated keywords within the dataset. This enables us to identify recurring themes and prevalent discussions related to vulnerabilities, exploits, and potential targets.**

***Index Terms*—Topic Modeling, Vulnerabilities, Exploits, Underground Hacking Forums, Latent Dirichlet Allocation, Cybersecurity.**

I. Introduction

The exploitation of vulnerabilities in the wild poses significant risks to the security and integrity of computer systems, networks, and sensitive data. With the constant evolution of cyber threats, it is crucial to identify and address vulnerabilities promptly to mitigate potential damages. Understanding how these vulnerabilities are exploited in real-world scenarios is essential for developing effective defense mechanisms and proactive security measures [3], [6].

In recent years, there has been an increasing focus on monitoring underground hacking forums as a valuable source of intelligence regarding vulnerabilities and exploits. These forums serve as platforms for cybercriminals and hackers to exchange knowledge, discuss techniques, and share information on newly discovered vulnerabilities. By monitoring and analyzing these discussions, security researchers and practitioners can gain insights into emerging trends, exploit techniques, and potential targets in the wild [2], [21].

Understanding the exploitation of vulnerabilities in the wild through monitoring underground hacking forums can provide valuable insights for improving security practices. Early detection and proactive measures based on real-world intelligence can significantly enhance the effectiveness of vulnerability management and incident response strategies. This research aims to contribute to this growing body of knowledge by exploring the exploitation of vulnerabilities in the wild, focusing on the analysis of underground hacking forums. By leveraging machine learning and topic modeling techniques, we seek to uncover hidden patterns, emerging trends, and potential targets, providing valuable insights to strengthen cybersecurity practices and defend against evolving threats.

Paper structure. The remainder of this paper is structured as follows. In Section II, we discuss related work, Section III presents our methodology. Section IV reports and discusses our results, and Section V concludes.

II. Related Work and Background

In what follows, we discuss related work and background pertaining to the main themes of our work.

A. Exploitation of vulnerabilities in the wild

The exploitation of vulnerabilities in the wild is an active area of research and a growing concern in the field of cybersecurity. As attackers continuously target vulnerabilities to compromise systems and gain unauthorized access, researchers have focused on understanding these exploits and developing effective defense mechanisms. In this section, we provide a brief overview of key research contributions related to the exploitation of vulnerabilities in the wild.

Automated vulnerability scanners, such as Nessus and OpenVAS, have been widely used to detect vulnerabilities by scanning systems and applications for known weaknesses. Additionally, manual analysis techniques, including reverse engineering and fuzzing, have been employed to uncover new and unknown vulnerabilities [13], [22].

Collaboration and information sharing play a crucial role in combating vulnerabilities. Platforms and databases, such as the Common Vulnerabilities and Exposures (CVE) system and the National Vulnerability Database (NVD), provide standardized information about vulnerabilities, including their severity and available patches. Furthermore, coordinated disclosure practices, such as responsible disclosure and bug bounty programs, facilitate the reporting and fixing of vulnerabilities [7], [8]

B. Topic modeling

Topic modeling has been extensively researched in the field of natural language processing (NLP) and machine learning. Researchers have proposed various methods and techniques to extract latent topics from textual data, leading to advancements in understanding and organizing large document collections.

In this section, we provide a brief overview of key research contributions in the field of topic modeling.

Latent Dirichlet Allocation (LDA) [5] introduces a generative probabilistic model for topic modeling. LDA assumes that each document is a mixture of a few topics, with each topic represented as a distribution over words. Another approach is Non-negative Matrix Factorization (NMF) [11]. This method factorizes the document-term matrix into two non-negative matrices representing the document-topic and topic-word distributions, showing good performance in topic extraction.

Probabilistic Latent Semantic Analysis (pLSA) [9] extends the latent semantic analysis by modeling documents as a mixture of latent topics using an expectation-maximization algorithm. Other approaches, such as Hierarchical Dirichlet Process (HDP) [23], are a Bayesian nonparametric extension of LDA. Dynamic Topic Models (DTM) [4] capture the temporal evolution of topics in a document collection extending LDA by introducing time slices and modeling topic transitions over time.

In this work, we use topic modeling for the exploitation of vulnerabilities in underground hacking forums. We apply Latent Dirichlet Allocation (LDA) to a dataset collected from these forums to uncover hidden topics and discussions. The goal is to identify key themes related to exploit techniques, vulnerabilities, and potential targets, providing valuable insights into the landscape of vulnerability exploitation.

TABLE I
STATISTICS ABOUT CONSIDERED FORUMS. FORUMS ARE DIVIDED INTO BOARDS, AND BOARDS ARE DIVIDED INTO THREADS. EACH THREAD CONTAINS A LIST OF POSTS. RANKED BY THE NUMBER OF THREADS.

Forum	#Users	#Boards	#Threads	#Posts
Hackforums	630,331	177	3,966,270	41,571,269
MPGH	478,120	715	763,231	9,363,422
Antichat	79,769	60	242,064	2,449,404
Offensive Community	11,800	58	119,228	161,492
DREADditevelidot	44,631	382	74,098	294,596
RaidForums	29,038	73	33,240	214,856
Runion	16,719	19	16,792	240,632
Safe Sky Hacks	7,433	44	12,956	27,018
The-Hub	8,243	62	11,274	88,753
Torum	3,813	11	4,328	28,485
Kernelmode Forum	1,644	11	3,438	25,825
Germany Ruvvy	2,206	42	2,845	20,185
Garage4hackers	880	31	2,096	7,697
Greysec	728	25	1,630	9,228
Stresser Forum	777	16	702	7,069
Envoy Forum	362	76	454	2,163
Total	1,316,494	1,802	5,254,646	54,512,094

III. METHODOLOGY

In this section, we explain our methodology, present the dataset, and how to perform pre-processing.

A. Datasets

1) CrimeBB: Cambridge Cybercrime Centre makes available sixteen underground forums through CrimeBB. CrimeBB comprises hierarchical data based on websites (see Figure 2. Table I presents general statistics in CrimeBB. In CrimeBB, we have 1,316,494 users interacting on 16 websites, 1,802 boards, 5,254,646 discussion threads, and 54,512,094 posts. After analyzing our dataset, we found about 650 null values in thread titles, 7,203 posts without content, and 7,769 null usernames.

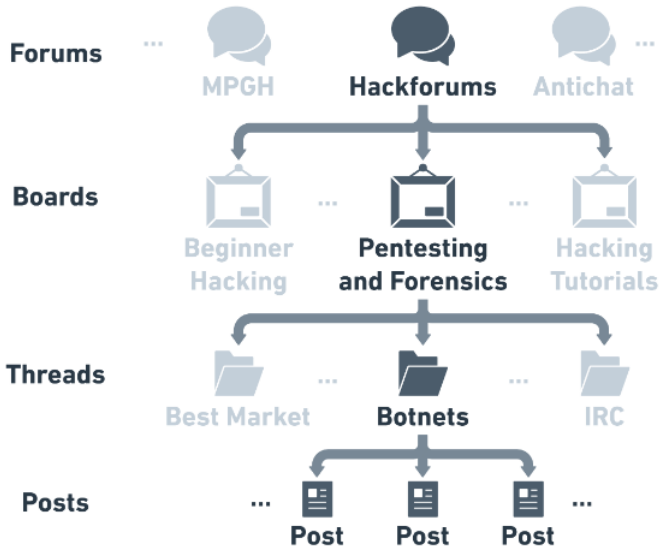

Fig. 1. CrimeBB dataset, showing the hierarchical composition of websites, boards, threads, and posts.

2) NVD data: The National Vulnerability Database (NVD) is a comprehensive repository of information about software vulnerabilities and security issues. It is maintained by the National Institute of Standards and Technology (NIST) in the United States. The NVD dataset provides detailed information about known vulnerabilities in various software products, including operating systems, applications, libraries, and hardware.

B. Text Preprocessing

We will divide our preprocessing into two steps: (i) nlp preprocessing, (ii) language evaluation, and (iii) feature extraction.

1) NLP pre-processing: We implement a text preprocessor that helps identify and keep relevant words to consider. We implement a library to preprocess text and evaluate language in order to facilitate our dataset preparation. We must be careful to select which word should be filtered. To do this, we filter the following characters:

- **Stopwords:** These are the words in any language which does not add much meaning to a sentence. Some samples of stopwords are pronouns, adverbs, articles, etc.
- **Punctuations:** These are symbols that you add to a text to show the divisions between different parts of it, such as Periods, commas, semicolons, question marks, apostrophes, and parentheses.
- **Special Characters:** These Are the symbols used in writing, typing, etc., that represent something other than a letter (outside the 26 letters used in US English) or number, such as §, à, é, î, œ, ü, ñ, etc.

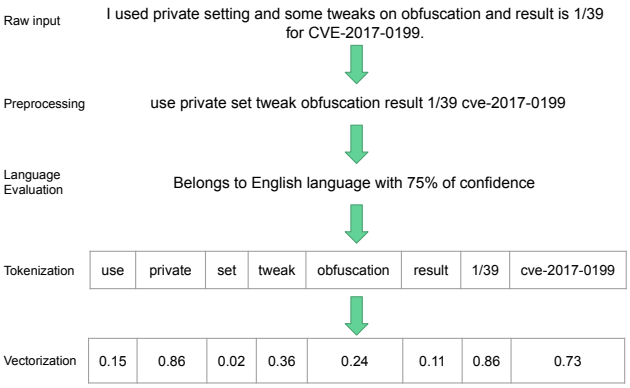

Fig. 2. Text preprocessing pipeline, we show all steps from raw input until vectorization. Note that we only keep and lemmatize words to their basic root form but not all words. This post was taken from the HackForums website.

- **Emojis:** These are a form of pictorial language used to express an idea, also called "digital images". These symbols denote an emotion or an action. It can be an image or textual composed symbol such as ":)", ":C", ":-)", etc.

After filtering out these characters from the raw input, we proceed to identify the language to which the input text belongs. Subsequently, we choose to convert all words using lemmatization instead of stemming. This choice is justified because, in text classification, lemmatization facilitates the creation of high-quality, contextually accurate, and semantically meaningful feature representations. This, in turn, contributes to improved classification accuracy, reduced noise, and enhanced generalization across various types of text data.

2) Language evaluation: We define the **Indicator Language Function** (ILF) denoted by $\mathbb{1}_{ilf}$. In Equation 1 we define our ILF, taking two parameters the word w and the language L to eval:

$$\mathbb{1}_{ILF}(w, L) = \begin{cases} 1, & \text{if} w \in L \\ 0, & \text{Otherwise} \end{cases} \tag{1}$$

Moreover, we define the **Language Ratio Function** (LRF) for a set of words and languages. This function allows us to identify and calculate what percentage of words within a phrase, paragraph, or text, in general, belongs to a specific language L. In Equation 2 we define our LRF, taking two parameters the text t with n words and the language L to eval:

$$\text{Ratio}_{LRF}(t, L) = \frac{1}{\text{n}} \times \sum_{\substack{i=1 \\ \text{w}_i \in t}}^{n} \mathbb{1}_{ILF}(w_i, L) \tag{2}$$

Finally, the **Determine Language Function** (DLF) is defined using the previously calculated Language Recognition Factor (LRF). This function determines the most probable language to which the input text belongs. In Equation 3, we define our DLF, taking a text t as a parameter. This text is evaluated in a set of languages $\mathbb{L}$ (English and Russian language by default):

$$\text{Language}_{DLF}(t) = \max_{\forall L \in \mathbb{L}} \text{Ratio}_{LRF}(t, L) \tag{3}$$

This step helps us to identify the main language within a thread (we only focus on the English language. Otherwise, we filter it). That's how We identify and filter threads from the Antichat forum. After this step, we proceed to perform text embedding, converting text into vectors.

3) Tokenization & Feature Extraction: In this step, we perform the feature extraction from textual information. After pre-processing all textual information and performing the tokenization of words, we will use text encoding-based methods such as Bag-Of-Words (BoW) [10] and Term Frequency - Inverse Document Frequency (TF-IDF) [20].

C. Threads Processing

In order to perform a Topic Modeling, we need to organize our data. In this step, we perform two main tasks: (i) Filter threads, and (ii) Label the topic of threads.

1) Filtering Threads: First, we employ a filtering process to identify all posts referencing at least one Common Vulnerabilities and Exposures (CVE) code. We concatenate and merge all posts with their corresponding parent thread. Then, by using case-insensitive regular expression `cve-[0-9]{4}-[0-9]{4,}` (slightly more specific than `cve(-id)?(?i)` used in [1] and [18]) we search for posts referring to vulnerabilities by their CVE identifiers. In Figure 3, we present this process, showing that we ignore thread that does not cite any CVE reference.

2) Labeling the Target topics: We defined 4 main discussion topics to label the thread content. We use the following code book to label the threads manually:

- **PoC:** (1) contain keywords such as PoC, tutorial, guide (given the appropriate context of producing tools in a lab or controlled environment); (2) provide a tutorial description about how to build a PoC or (3) discuss vulnerabilities without signs of using exploits in the wild.
- **Weaponization:** (1) contain keywords such as vulnerability and exploit (given the appropriate context of weaponization); (2) discuss the availability of fully functional or highly mature exploits, providing references or source code.
- **Exploitation:** (1) mention a well-known hacker group; (2) contain references to cryptocurrencies and keywords such as bitcoin, exploitation, and attack (given the appropriate context of attacks in the wild); (3) discuss approaches to make exploits fully undetectable; or (4) involve markets of exploits.
- **Other:** Discussion about anything but the topics above.

Note: From this labeling instead of using them as labels, we decide to set them as topics.

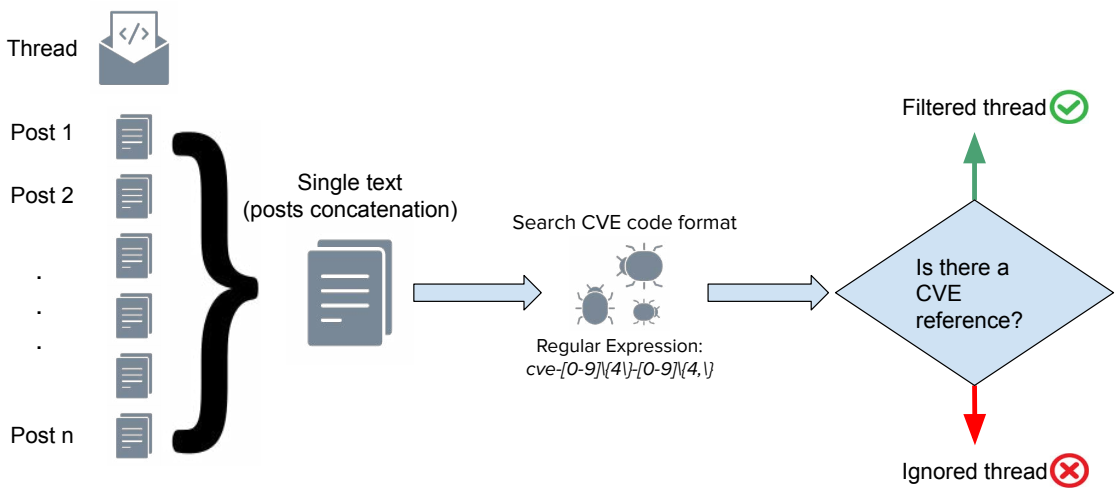

Fig. 3. Post concatenation by thread: If at least one post cites a CVE code, we take all others posts from the same thread as one text sample. Otherwise, the complete thread is ignored and excluded from the dataset. This is why we don't use all labeled threads.

IV. Discussions & Results

In this section, we discuss our analysis and the topic modeling.

A. Datasets

1) CrimeBB: Across all CrimeBB forums, we found 4,098 posts citing 1,498 unique CVEs under 1,700 discussion threads within 149 boards. To analyze the content of threads and posts, we utilize data sourced from CrimeBB. We employ a filtering process where we specifically search for citations of CVE codes in their complete format rather than just isolated words. We found that

2) NVD: We utilize information sourced from the National Vulnerability Database (NVD) to ascertain the characteristics of the vulnerabilities under examination. These attributes encompass factors like severity, quantified through the Common Vulnerability Scoring System (CVSS) as depicted in Figure 4. Notably, more than 60 % of the cited CVEs are designated with a high severity level in version 2. However, in version 3.1, we encountered difficulty in determining the present severity level of our CVE codes.

B. Text Preprocessing

1) NLP pre-processing: We developed an NLP library to reproduce the text preprocessing presented in this document. The code is available at https://github.com/fmorenovr/nlpToolkit/. There, we implement an easy way to analyze languages and allow the definition of additional characters to be filtered. Besides, we found three main languages in CrimeBB: 13 websites with 4,992,945 threads discussed in English, 1 website with 242,064 threads discussed in Russian, and 2 websites with 19,637 threads discussed in Deutsch. We only focus on English websites. From this, we perform the feature extraction.

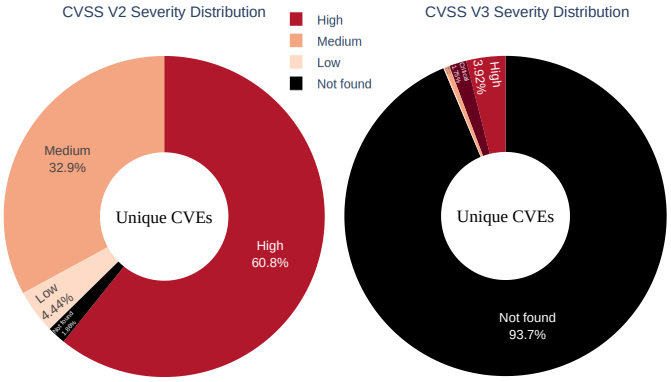

Fig. 4. Common Vulnerability Scoring System (CVSS) severity level, we compare the version 2 and 3.1 of CVSS scores. We note that about 908 969 CVE codes are not found in the CVSS 3.1 version.

2) Tokenization & Feature Extraction: We perform the tokenization in a different way from previous works [12], [14]–[18]; in this methodology, we will use the corpus of the language. This allows us to construct the dictionary and assign an ID to each token to identify groups of related words. We perform and join both Bag-Of-Words and TF-IDF to obtain the language corpus as features.

C. Threads Processing

After filtering threads that cite at least one CVE code (see Figure 3), we find about 1,677 threads that explicitly cite 1,068 unique CVEs from 14 websites. Besides doing an intersection of the filtered threads and the topic-labeled threads, we got 1,067 threads with a corresponding topic.

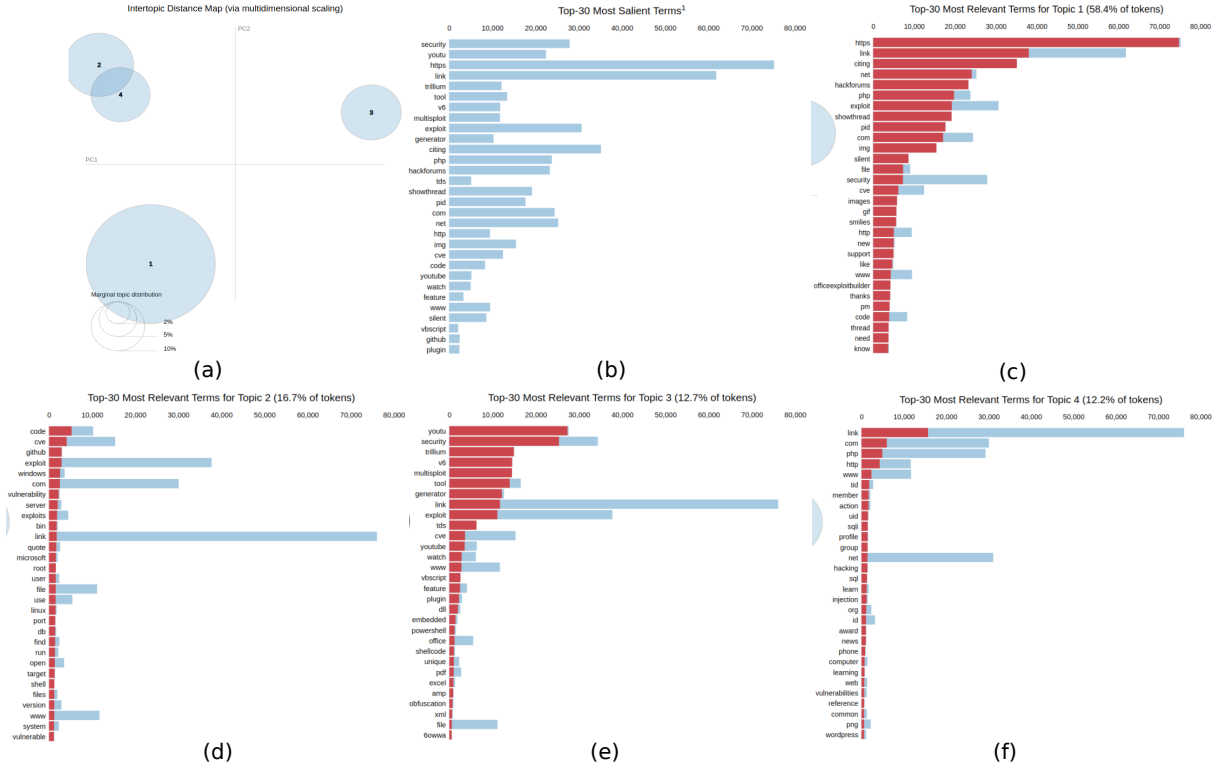

Fig. 5. Topic group projection by principal components. (a) The radius of each group determines the marginal topic distribution, (b) the top 30 most salient terms, (c) the top 30 most relevant terms for topic *PoC*, (d) the top 30 most relevant terms for topic *Weaponization*, (e) the top 30 most relevant terms for topic *Exploitation*, and (f) the top 30 most relevant terms for topic *Others*.

D. Topic modeling

Topic modeling is conducted on the corpus generated in the preceding stages, employing the Latent Dirichlet Allocation (LDA) algorithm. The objective is to deduce the thematic focus of the discussion by identifying pertinent terms associated with each topic. The training of the LDA model is facilitated using the Gensim library [19]. As previously indicated, our analysis encompasses four distinct topics: Proof of Concept (PoC), weaponization, exploitation, and other themes, labeled with corresponding IDs 1, 2, 3, and 4, respectively.

In Figure 5 (a) we show the Topic group projection by principal components, we note that topics *Weaponization* and *exploitation* have an intersection. This is an interesting result due to the proximity between a vulnerability from weaponization to exploitation. In Figure 5 (b) we show the 30 top words in all topics, we note that words such as ”https”, ”link”, ”citing”, and ”exploit” are the most relevant.

In Figure 5 (c) using only the 58.4 % of tokens the most relevant for the *PoC* topic are ”https”, ”link”, ”php”, etc. We note that in general, those words are relevant for all topics except for ”code” and ”security”. In Figure 5 (d) we show the relevant words for the topic *weaponization* using only the 16.7 % of tokens are the words ”code”, ”cve”, ”github”, etc.

In Figure 5 (e) we show the relevant words for the topic *exploitation* using only the 12.7 % of tokens are the words ”security”, ”trillium” (the user who cite the highest quantity of CVE codes [18]), ”multisploit”, ”tool”, ”exploit”, etc. Finally, In Figure 5 (f) we show the relevant words for the topic *others* using only the 12.2 % of tokens are the words ”link”, ”com”, ”php”, ”member”, ”profile”, ”learn”, etc.

We note that for each topic we have some relevant words that let us understand the main discussion. In the intersection between the topics *Weaponization* and *exploitation*, we have some relevant words such as ”code”, ”cve”, ”exploit”, ”link”, and ”vulnerability”. Besides, in the topic *PoC* the most relevant word is ”https”, ”link”, ”citing”, ”net”, etc. This happens due to the nature of the discussion, sharing links, tutorials, code, etc. From this, we observe the topic modeling could infer the discussion theme within a thread. Furthermore, we know that in each thread can be discussed several themes but only one topic.

V. Conclusion

In conclusion, applying topic modeling techniques to the study of exploitation in the wild offers valuable insights and benefits in the field of cybersecurity. By analyzing textual data related to vulnerabilities, exploits, and real-world attack scenarios, topic modeling contributes to a deeper understanding of the exploitation landscape. Topic modeling aids in understanding exploit trends: By extracting latent topics from data sources such as vulnerabilities, exploit forums, or security incident reports. This understanding helps security professionals, and researchers stay informed about emerging threats and prioritize their defense strategies accordingly.

REFERENCES

[1] Allodi, L.: Economic factors of vulnerability trade and exploitation. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security. pp. 1483–1499 (2017)

[2] Anderson, R., et al.: Measuring the changing cost of cybercrime. The 2019 Workshop on the Economics of Information Security (2019)

[3] Basheer, R., Alkhatib, B.: Threats from the dark: a review over dark web investigation research for cyber threat intelligence. Journal of Computer Networks and Communications **2021**, 1–21 (2021)

[4] Blei, D.M., Lafferty, J.D.: Dynamic topic models. Proceedings of the 23rd international conference on Machine learning (2006)

[5] Blei, D.M., Ng, A., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2001)

[6] Campobasso, M., Allodi, L.: Threat/crawl: a trainable, highly-reusable, and extensible automated method and tool to crawl criminal underground forums. In: APWG eCrime 2022 (2022), arXiv:2212.03641

[7] Chen, D.D., Woo, M., Brumley, D., Egele, M.: Towards automated dynamic analysis for linux-based embedded firmware. In: Network and Distributed System Security Symposium (2016)

[8] Edkrantz, M., Truvé, S., Said, A.: Predicting vulnerability exploits in the wild. 2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing pp. 513–514 (2015)

[9] Hofmann, T.: Probabilistic latent semantic analysis. In: Conference on Uncertainty in Artificial Intelligence (1999)

[10] Juluru, K., Shih, H.H., Murthy, K.N.K., Elnajjar, P.: Bag-of-words technique in natural language processing: A primer for radiologists. Radiographics : a review publication of the Radiological Society of North America, Inc p. 210025 (2021), https://api.semanticscholar.org/CorpusID:237009513

[11] Lee, D.D., Seung, H.S.: Learning the parts of objects by non-negative matrix factorization. Nature **401**, 788–791 (1999)

[12] León-Vera, L., Moreno-Vera, F.: Car monitoring system in apartments' garages by small autonomous car using deep learning. In: Annual International Symposium on Information Management and Big Data. pp. 174–181. Springer, Springer International Publishing (2018)

[13] Liang, H., Pei, X., Jia, X., Shen, W., Zhang, J.: Fuzzing: State of the art. IEEE Transactions on Reliability **67**, 1199–1218 (2018)

[14] Moreno-Motta, J., Moreno-Vera, F., Moreno, F.A.: Morarch: A software architecture for interoperability to improve the communication in the edge layer of a smart iot ecosystem. In: Smart Trends in Computing and Communications, pp. 185–195. Springer (2022)

[15] Moreno-Vera, F.: Performing deep recurrent double q-learning for atari games. In: 2019 IEEE Latin American Conference on Computational Intelligence (LA-CCI). pp. 1–4 (2019). https://doi.org/10.1109/LA-CCI47412.2019.9036763

[16] Moreno-Vera, F., León-Vera, L., Moreno-Motta, J., Guizado-Vasquez, J., Vera-Panez, M.: Comparison of the learning curve and adaptive behavior from kids to adults using computational thinking with block-programming to technology enhanced learning. In: 2019 IEEE World Conference on Engineering Education (EDUNINE). pp. 1–5 (2019). https://doi.org/10.1109/EDUNINE.2019.8875764

[17] Moreno-Vera, F.: Understanding safety based on urban perception. In: International Conference on Intelligent Computing. pp. 54–64. Springer (2021)

[18] Moreno-Vera, F., Nogueira, M., Figueiredo, C., Menasché, D.S., Bicudo, M., Woiwood, A., Lovat, E., Kocheturov, A., de Aguiar, L.P.: Cream skimming the underground: Identifying relevant information points from online forums. In: 2023 IEEE International Conference on Cyber Security and Resilience (CSR). pp. 66–71 (2023). https://doi.org/10.1109/CSR57506.2023.10224941

[19] Rehurek, R., Sojka, P.: Gensim–python framework for vector space modelling. NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic **3**(2) (2011)

[20] Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. Inf. Process. Manag. **24**, 513–523 (1988), https://api.semanticscholar.org/CorpusID:7725217

[21] Susan Morrow, T.C.: The future of cybercrime & security (2019)

[22] Sutton, M.S., Greene, A.R., Amini, P.F.: Fuzzing: Brute force vulnerability discovery (2007)

[23] Teh, Y.W., Jordan, M.I., Beal, M.J., Blei, D.M.: Hierarchical dirichlet processes. Journal of the American Statistical Association **101**, 1566 – 1581 (2006)