Internet-Wide Scanners Classification using Gaussian Mixture and Hidden Markov Models

Giulia De Santis*, Abdelkader Lahmadi†, Jérôme François*, Olivier Festor*†
*INRIA - Nancy Grand-Est, Villers-les-Nancy, France †LORIA, University of Lorraine, Vandoeuvre-les-Nancy, France

Email: {giulia.de-santis, abdelkader.lahmadi, jerome.francois, olivier.festor}@inria.fr

***Abstract*—Internet-wide scanners are heavily used for malicious activities. This work models, from the scanned system point of view, spatial and temporal movements of network scanning activities, related to the difference of successive scanned IP addresses and timestamps, respectively. Based on real logs of incoming IP packets collected from a darknet, Hidden Markov Models (HMMs) are used to assess what scanning technique is operating. The proposed methodology, using only one of the aforementioned features of the scanning technique, is able to fingerprint what network scanner originated the perceived darknet traffic.**

***Index Terms*—Network scanning, ZMap, Shodan, Darknet, Gaussian distribution models, Hidden Markov Models**

I. Introduction

Network Scanning Activities (NSAs) are "reconnaissance techniques able to determine open ports and services" [1]. They are used by attackers as first steps of exploitation attempts, to take control over the host [2], exploit vulnerabilities [3] and gather information about the system that will be targeted by an Advanced Persistent Threat [4]. To ensure efficient defense, threats need to be mitigated during the scanning activities [2]. Models describing scanning techniques can help experts to assess if and what scanning technique is faced by the network.

In [5], authors used Hidden Markov Models (HMMs) to model the scanning intensity of Shodan [6] and ZMap [7] using logs collected by a darknet. Their work is here extended, covering spatial and temporal movements of a single execution of the scanner using the same modeling tools. These models are then used to efficiently identify what scanning tool is used by attackers. We use, as in [5], real logs of probing packets of the scanners collected by a darknet. Since it has no active hosts, all incoming traffic is undesired. Collected datasets cover long intervals of time, containing logs of multiple executions of the scanning technique. We split then each dataset into samples containing logs of a single execution by considering time lapses between two consecutive scanned IP addresses: when it is an outlier, an execution ends and the following begins. Once these samples are available, they are used to learn mixture distribution models, and their respective HMMs. Mixture models create clusters of logs contained in each sample and assign to each cluster a Gaussian distribution. HMMs, whose states are the clusters provided by the mixture model, are built and then applied to test samples to assess what scanning technique has been used to generate the collected logs.

To summarize, we firstly build Gaussian mixture models and HMMs of scanning activities by considering differences of destination IP addresses and timestamps of their successive probing packets perceived by a darknet. Secondly, we use the obtained HMMs to classify probing packets according to the network scanner that generated them.

The remainder of the paper is structured as follows. Section II reviews scanning methods, existing work on detection of NSAs and models. Section III details how to build and validate HMMs of the considered scanner. Section IV provides an excursus on the obtained experimental results and section V concludes the work and outputs future work.

II. Related work

NMap[1], ZMap [7], Shodan[2] [6] and Masscan [8] are among the most known and used network scanners. Their usage varies around the world: ZMap is mainly used in western countries whereas Masscan and Unicorn[3] are popular in South East Asia [2]. Also the services they commonly target vary: one third of probes of ZMap are directed to port 22, whereas NMap mostly scans port 23 and Masscan mainly scans uncommon non-privileged ports [2].

Approaches to detection of NSAs are split into two main categories: single-source and distributed approaches [9]. The former relate to single-source NSAs, that consist of a host collecting information about one (one-to-one) or multiple (one-to-many) hosts. The latter relates to detection of distributed NSAs, which exploit various hosts to gather information about one (many-to-one) or multiple (many-to-many) hosts. Since in this work we fingerprint both ZMap [7] and Shodan [6], which are distributed scanners, we focus on the latter.

Coordinated NSAs, i.e., multiple single source scans such that, all together, cover a large portion of the targeted space and the overlap between portions of the space scanned by each scan is as small as possible, are detected by Gates [10]. This approach firstly detects single-source scans, and then merge them into coordinated ones if they act in an orchestrated manner. The tool used during a scanning campaign is identified by Ghiëtte et al. [2], who focus on analyzing scanned ports and

[1]https://nmap.org/
[2]https://www.shodan.io/
[3]http://sectools.org/tool/unicornscan/

use inspections of source code of scanners to find characteristic patterns in scan traffic, applied then to identify the used scanning tool. This approach works then only when the source code of the scanner is available, which is not always the case (as for Shodan [6]). We solve this issue, sharing the same goal (i.e., to detect the tool used by coordinated NSAs), by applying machine learning techniques: more in details, it relies on HMMs.

HMMs are a statistical tool [11] initially applied in speech recognition [12]. They have also been used to detect multi-stage network attacks [13], and compared with neural networks decision tree algorithm. HMMs can be applied to model complex Internet attacks, consisting of several steps that may need an extended period of time to occur, and enhance current intrusion detection methods since the latter can only identify individual stages of complex and elaborated attacks, whereas HMMs can describe correlation and order of steps forming a complex attack, and need fewer training examples to provide good descriptions of the attack, if compared to decision tree algorithms and neural networks. HMMs have been also used to describe malicious traffic and attacks targeting SSH port 22 [14], with the purpose of modeling behaviors of attackers. Our work aims to model first steps of complex attacks. The models we build can then be used by intrusion detection systems and also SIEMs (Security Information and Event Management) to improve their detection capabilities and to prevent complex and targeted attacks before they succeed.

III. Classification Methodology

Our goal is to assess what scanner originated logs gathered by a scanned network. Figure 1 provides an overview of the processing steps of the methodology to achieve this goal.

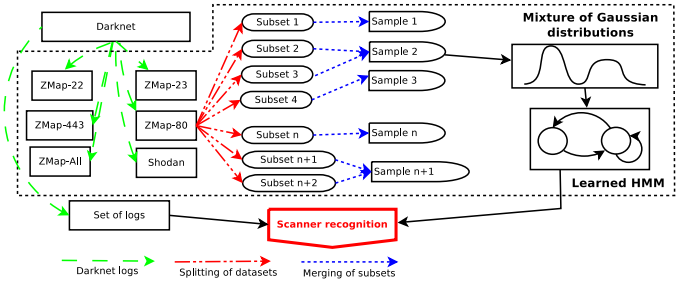

Fig. 1. *Processing steps of the proposed methodology*

Our approach consists in learning models of logs generated by each scanning tool, and applying these models to identify those tools in real traces. We leverage HMMs because they are able to easily handle observations from various unknown groups. We also aimed to answer the question "Is the scanning technique behaving differently according to the scanned port?".

A. From logs to samples

To accomplish these aforementioned tasks, we used real network logs of ZMap and Shodan, respectively. After ordering them in an increasing chronological order, we computed differences of 2 consecutive destination IP addresses and differences of timestamps. Then, common ports below 1024 between the two datasets are identified, and datasets split into (sub-)datasets, one for each considered common port and scanner.

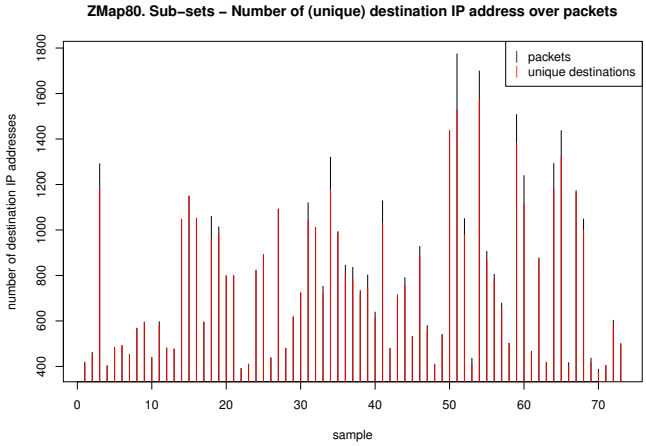

Fig. 2. *Number of unique IP addresses over number of packets of a ZMap execution targeting port 80*

Finally, for each (sub-)dataset, we extracted a "learning" dataset and analyzed its time gaps: when an outlier appears, an execution of the considered scanning technique has terminated and a new one starts. This leads to split the (sub-)dataset into sub-sets. Consecutive sub-sets are then merged into *samples* S_i when their mutual Jaccard index is ≤ 0.1 (i.e., the overlapping between the two is small). This merging makes us sure that each sample S_i contains logs of a single execution of the scanning technique. Data corresponding to time difference outliers in each sample are removed. Each sample is then labeled by its scanner type and targeted port, "scanner-port". The result is a set of labeled samples scanner-port-S_i (i.e., zmap-22-S_i).

Figure 2 shows the number of packets (black) and of unique scanned IP addresses (red) contained in each sample generated by ZMap scanning port 80. Small size samples are discarded by considering only the ones whose length is greater than the average length of the obtained samples.

B. Learning models of scanning techniques

This work models spatial and temporal movements of scanners, i.e., differences of destination IP addresses and of timestamps, with mixtures of Gaussian distributions and HMMs. We used Gaussian distributions for both observations because time-gaps are continuous and discrete distributions based on large samples can be easily approximated by the Gaussian distribution thanks to the Central Limit Theorem [15].

Since one single Gaussian distribution is not sufficient to model logs in samples (see Figure 3), mixture distribution models are required, which cluster logs and provide a probability to each cluster. Transition probabilities between clusters are

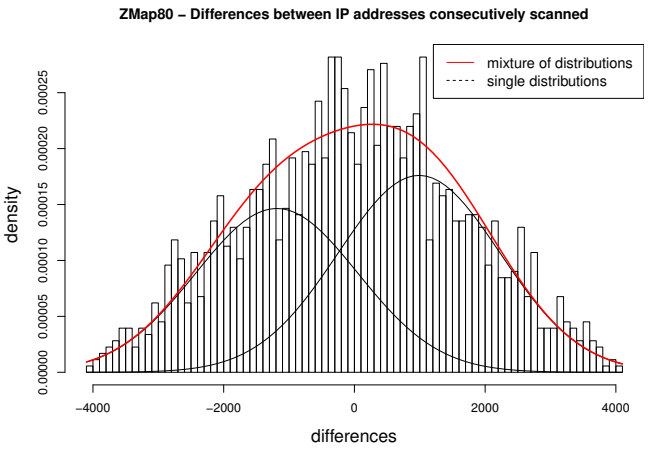

Fig. 3. *Mixture of Gaussian distribution of differences of IP addresses for a ZMap execution targeting the port 80*

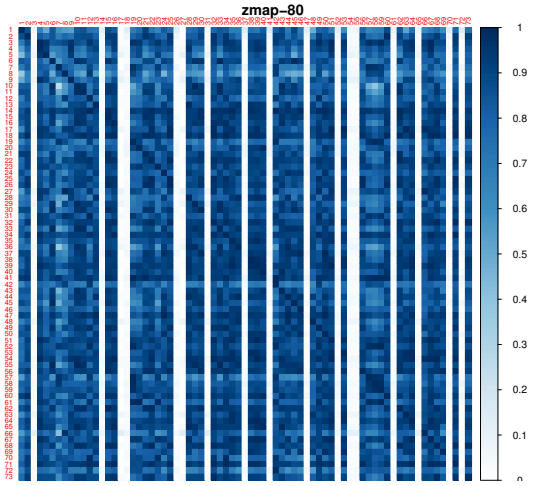

Fig. 4. *Normalized log-likelihood matrix of learned HMMs models and 73 samples for a ZMap scanning activity over port 80*

provided by HMMs, whose states are the m clusters provided by the mixture distributions model.

1) Mixture distribution models: Samples contain logs from the scanned network. Since logs are in unobserved groups, each with its own Gaussian distribution, and the selection of one group is independent from the previous choice, independent Gaussian mixtures distribution models are needed. They consist of m groups or clusters, each with its own Gaussian distribution $p_1, p_2, \ldots, p_m$, and a *mixing distribution* $\delta = (\delta_1, \delta_2, \ldots, \delta_m)$ which selects one of the clusters. The selection of the group is established by a random variable performing the mixing. The distribution $p_i, i = 1, 2 \ldots, m$, being active when the observation was done is unknown.

2) Hidden Markov Models (HMMs): Probabilities to move between clusters are provided by HMMs, that consist of an unobserved parameter process $\{C_t : t = 1, 2, \ldots, m\}$ satisfying the Markov Property: $\Pr(C_t|C_{t-1}, \ldots, C_1) = \Pr(C_t|C_{t-1})$, and a state-dependent process $\{X_t : t = 1, 2, \ldots\}$ such that its distribution depends only on the current state of C_t: $\Pr(X_t|\boldsymbol{X}^{(t-1)}, \boldsymbol{C}^{(t)}) = \Pr(X_t|C_t)$ [5]. C_t establishes the cluster of the observation. It is possible to go from a state to another with *transition probabilities*, associated with each pair of states [13]. The state remains unknown: only the observation is visible.

An HMM is generated from logs of each learning sample. Once all the HMMs are available, the question "What is the set of good candidate models for the considered observation?" needs to be answered. For this, each sample S_i was fitted to the HMM built on sample S_j, HMM$_j$ and its log-likelihood computed. Figure 4 shows the normalized matrix of all the obtained log-likelihoods where rows represent samples and columns their associated HMMs. A normalized log-likelihood value close to 1 (i.e., color close to blue in Fig. 4) states for a good model for the considered sample. The obtained HMMs are then grouped using *DBSCAN* [16] which, requiring only two parameters, ϵ and the minimum number of points in each cluster, gathers pairs of points whose distance is less or equal to the chosen ϵ and if one of them is surrounded by a sufficient number of other points. Then, once groups are formed, a model is selected from each one to be the searched *candidate model* for the considered cluster. Indeed, all models within a single group can be considered as equivalent to describe the underlying samples, but only a single one is required.

The result is a list of *candidate models* M_i for each combination "scanner-port" (i.e., zmap-23-M_i).

C. Fingerprinting of scanning techniques

All the *candidate models* are validated on test samples. For each of them, it is known the originating scanner and the targeted port: the corresponding couple "scanner-port" is the *true label* of the considered test sample. Each of them is fitted to each *candidate model*, and the log-likelihood computed: the model with the highest one best describes logs of the test sample, and its label "scanner-port" is stored as the *predicted label* of the test sample. Finally, we investigated if the proposed method is able to identify the scanner that originated the test sample, and eventually the scanned port, i.e., if true label and predicted label are equal.

IV. Experimental results

In our experiments, we used one dataset for ZMap and one Shodan, both containing logs of real IP traffic perceived by a darknet. Table I shows inner characteristics of the two scanners: ZMap being horizontal and fast when scanning one single port [7], and Shodan being more massive and targeting various ports [6].

Each dataset is split according to destination ports. Only (sub-)datasets related to the 17 ports in common between the two scanners and lower than 1024 are taken into account. So, in total, 34 (sub-)datasets have been analyzed. For each of them, disjoint *learning* and *test* datasets are selected, each with a randomly selected initial log and a number of temporally consecutive logs equal to 10% and 5%, respectively, of the

TABLE I
DETAILS OF SCANNING DATASETS FOR ZMAP AND SHODAN

Dataset	ZMap	Shodan
# Sources	253	13
# Destinations	4096	4096
# Ports	28	244
Duration (days)	533.69	12
# packets	7992496	708160

length of the dataset. Learning datasets are used to learn HMMs, whereas test ones to validate them. Each learning or test dataset is split into samples, each corresponding to one single execution of the considered scanning technique, according to outliers of time gaps between consecutive received packets (see Sec. III-A). Then, differences of destination IP addresses and of timestamps are modeled with mixtures of Gaussian distributions and HMMs. The latter are then clustered using the DBSCAN [16] method, and a model of each cluster is selected randomly to be the *candidate model* for the group. The result is a list of *candidate models* for each pair "scanner-port", that are then validated. Accuracy of the classification is computed by counting how many times predicted labels are equal to true labels. In other words, when looking at Figures 5, 6, 7 and 8, rows of the matrices are reserved for *true labels* and columns for *predicted labels*. Elements of the each row count how many times a sample, whose *true label* is the one of the corresponding row summed, has been labeled with each *predicted label* (one for each column). Let us look at Fig. 5, and more in detail at the row labeled "zmap-443": 1 test sample generated by ZMap scanning port 443 has been labeled with the predicted label "zmap-995", 9 with label "zmap-22", 3 with "zmap-21", etc. Only 8 have been correctly labeled with the predicted label "zmap-443". Row "zmap-443" of Figures 5 and 7, respectively, also shows that the number of test samples generated by ZMap targeting port 443 are more than the others. Accuracy is then computed by summing elements of the diagonals and by dividing the result by the number of all samples (i.e, the sum of all elements of the matrices).

A. Differences of scanned IP addresses

This section shows results obtained applying the aforementioned methodology to model spatial movements of scanners, i.e., differences of consecutive targeted IP addresses. 91,3% of the candidate HMMs have 2 different states: logs are clustered into 2 groups, corresponding to backward or forward spatial movements over destination IP addresses, respectively.

Figures 5 and 6 show how many test samples have been correctly and wrongly labeled when fingerprinting scanning activities. The presented method correctly detects what scanning technique generated the test samples (see Fig. 6), but is not able to detect what port is targeted (see Fig. 5).

Accuracy of the selected candidate models is tested as detailed in Section III-C. When assessing both the scanning technique generating the sample and the targeted port, accuracy is really low (0.06). But, when focusing only on the detection of the scanning technique, it reaches 0.952. More in

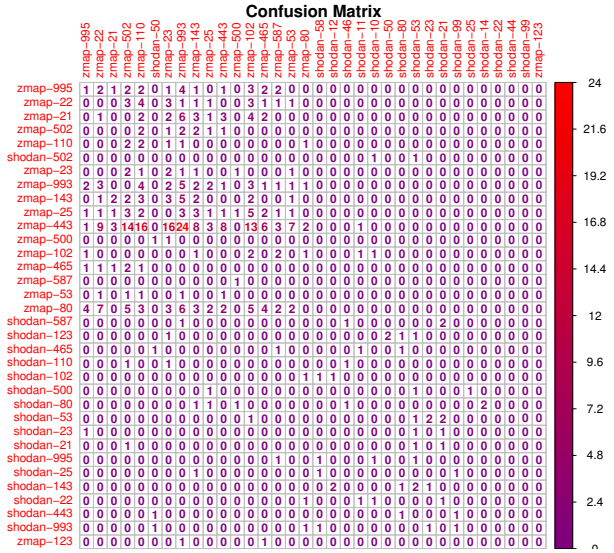

Fig. 5. *Confusion matrix of the classification of samples using differences of IP addresses per scanner-port.*

	zmap	**shodan**
zmap	368	4
shodan	17	50

Fig. 6. Confusion matrix of Zmap and Shodan classification using differences of IP addresses per scanner.

detail, we had 21 samples wrongly classified: 17 generated by Shodan labeled as being originated by ZMap, and 4 assigned to Shodan while truly generated by Zmap. It is possible, with an accuracy of 95%, to determine what scanner originated the test logs, and that spatial movements of the scanning activity do not vary with the scanned port and differ only between different scanners.

B. Time gaps

Differences of timestamps of logs (i.e., temporal movements) have been modeled following the methodology detailed in Section III. Selected candidate HMMs for differences of timestamps have between 1 and 7 states, with 92.2% having up to 4 states, and have been learned from the same learning samples used to learn HMMs for differences of IP addresses. A selection of *candidate models* for each combination "scanner-port", that are then tested on test samples, is then obtained.

Accuracy of models has been computed, both to detect only the scanning technique generating logs of test samples, and the scanned port together. In the first case, accuracy is 98%. We are thus able to model and fingerprint ZMap and Shodan. But, accuracy for the pair "scanner-port" is low (16%). ZMap and Shodan thus don't change their behavior when targeting a particular port. Figures 7 and 8 show how many samples have been correctly/wrongly labeled. Only 10 test samples have been wrongly assigned to the scanner Shodan while they are truly generated by ZMap (see Fig. 8), and all the samples generated by Shodan have been correctly classified.

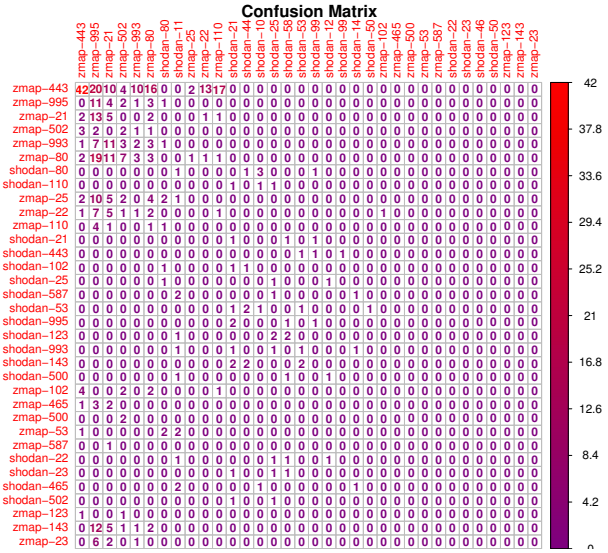

Fig. 7. *Confusion matrix of the classification of samples using differences of timestamps per scanner-port*

	zmap	**shodan**
zmap	362	10
shodan	0	67

Fig. 8. *Confusion matrix of ZMap and Shodan classification using differences of timestamps per scanner*

V. Conclusions and Future Work

Having accurate models of scanning tools is essential for early detection of larger and more impacting attacks. To build such models, we investigated spatial (differences of consecutive destination IP addresses) and temporal (differences of consecutive timestamps between two consecutive received packets) movements of scanners. To build log clusters we used mixtures of Gaussian distributions, and to establish transition probabilities between clusters HMMs.

We considered two predominant Internet-wide scanners, ZMap and Shodan, and 17 ports. For each pair "scanner-port", we extracted a bunch of learning samples, each modeled with an HMM. Resulting models are then clustered and a selection of *candidate models* outputted. All the selected models for all pairs "scanner-port" have been validated on *test samples*. Both for spatial and temporal movements, selected HMMs can be well used to identify what scanning technique originated logs in samples with an accuracy greater than 95%, but aren't able to detect the scanned port. We conclude that the behavior of scanners is not related to the scanned port and that attackers use the same configurations when scanning various ports.

This work can be enhanced by building HMMs able to model both kinds of movements, and can be used to assess if and what scanning technique is currently faced by the system. More future work will consist also in testing various NSAs detection methods on the same dataset, and in including the models produced here for early warning of cyber attacks and advanced persistent threats, since scanning activities are often used in their reconnaissance phase [4].

Acknowledgement

This work was partially funded by HuMa, a project funded by Bpifrance and Region Lorraine under the FUI 19 framework. It is also supported by the High Security Lab hosted at Inria Nancy Grand Est (http://lhs.inria.fr).

References

[1] S. Panjwani, S. Tan, K. M. Jarrin, and M. Cukier, "An Experimental Evaluation to Determine if Port Scans are Precursors to an Attack," in *International Conference on Dependable Systems and Networks, 2005*. IEEE, 2005, pp. 602–611.

[2] V. Ghiëtte, N. Blenn, and C. Doerr, "Remote Identification of Port Scan Toolchains," in *8th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2016, pp. 1–5.

[3] Z. Durumeric, M. Bailey, and J. A. Halderman, "An Internet-Wide View of Internet-Wide Scanning," in *23rd USENIX Security Symposium*, 2014, pp. 65–78.

[4] P. Chen, L. Desmet, and C. Huygens, "A Study on Advanced Persistent Threats," in *Communications and Multimedia Security*, ser. Lecture Notes in Computer Science, B. De Decker and A. Zúquete, Eds. Springer Berlin Heidelberg, 2014, vol. 8735, pp. 63–72. [Online]. Available: http://dx.doi.org/10.1007/978-3-662-44885-4_5

[5] G. De Santis, A. Lahmadi, J. Francois, and O. Festor, "Modeling of IP Scanning Activities with Hidden Markov Models: Darknet Case Study," in *8th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. IEEE, 2016, pp. 1–5.

[6] R. Bodenheim, J. Butts, S. Dunlap, and B. Mullins, "Evaluation of the Ability of the Shodan Search Engine to Identify Internet-Facing Industrial Control Devices," *International Journal of Critical Infrastructure Protection*, vol. 7, no. 2, pp. 114–123, 2014.

[7] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-Wide Scanning and its Security Applications." in *Usenix Security*, 2013.

[8] R. D. Graham, "MASSCAN: Mass IP Port Scanner," *URL: https://github. com/robertdavidgraham/masscan*, 2014.

[9] M. H. Bhuyan, D. Bhattacharyya, and J. K. Kalita, "Surveying Port Scans and their Detection Methodologies," *The Computer Journal*, vol. 54, no. 10, pp. 1565–1581, 2011.

[10] C. Gates, "Coordinated Scan Detection," in *NDSS*, 2009.

[11] L. Rabiner and B. Juang, "An Introduction to Hidden Markov Models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[12] B. H. Juang and L. R. Rabiner, "Hidden Markov Models for Speech Recognition," *Technometrics*, vol. 33, no. 3, pp. 251–272, 1991.

[13] D. Ourston, S. Matzner, W. Stump, and B. Hopkins, "Applications of Hidden Markov Models to Detecting Multi-Stage Network Attacks," in *Proceedings of the 36th Annual Hawaii International Conference on System Sciences*. IEEE, 2003, pp. 10–19.

[14] A. Sperotto, R. Sadre, P.-T. de Boer, and A. Pras, "Hidden Markov Model Modeling of SSH Brute-Force Attacks," in *International Workshop on Distributed Systems: Operations and Management*. Springer, 2009, pp. 164–176.

[15] A. C. Berry, "The Accuracy of the Gaussian Approximation to the Sum of Independent Variates," *Transactions of the American Mathematical Society*, vol. 49, no. 1, pp. 122–136, 1941.

[16] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.