[1] This is a preprint version of the paper. The final version was published at the Journal of Reliable Intelligent Environments. The final version can be found in https://doi.org/10.1007/s40860-020-00120-3

Big Data in Cybersecurity

A Survey of Applications and Future Trends

Mohammed M. Alani

Received: date / Accepted: date

Abstract With over 4.57 billion people using the Internet in 2020, the amount of data being generated has exceeded 2.5 quintillion bytes per day. This rapid increase in the generation of data has pushed the applications of big data to new heights; one of which is cybersecurity.
The paper aims to introduce a thorough survey on the use of big data analytics in building, improving, or defying cybersecurity systems. This paper surveys state-of-the-art research in different areas of applications of big data in cybersecurity. The paper categorizes applications into areas of intrusion and anomaly detection, spamming and spoofing detection, malware and ransomware detection, code security, cloud security, along with another category surveying other directions of research in big data and cybersecurity. The paper concludes with pointing to possible future directions in research on big data applications in cybersecurity.

Keywords big data · security · cybersecurity · big data analytics · security analytics

1 Introduction

With over 4.57 billion users connected to the Internet in 2020, according to [37], the amounts of data and metadata being generated has reached astonishing numbers. Each day, people and devices connected to the Internet generate over 2.5 quantillion bytes of data[87]. This massive amount of data comes from different sources on the Internet with social media users taking the lead in terms of amounts of data being generated. Instagram users upload an average of 95 million photos and videos per day. Facebook users publish 510,000

Senior Member of the ACM
Toronto, Canada
E-mail: m@alani.me

comments and 293,000 status updates every hour. Other communication platforms also generate massive amounts of data. With an average of 156 million email messages sent every minute and Skype users making 154,200 calls every minute, not only social media users are the cause of the surge of data generation [87].
Classical techniques in data processing are no longer capable of handling these large amounts of data. Hence, a proper definition of big data would be the "extremely large data sets that cannot be processed with conventional data processing techniques" [77].
In this paper, we present a thorough survey of applications of big data in various areas of cybersecurity. The paper starts with brief introductions to the main subjects of the paper; big data and cybersecurity. The second section of the paper introduces a quick review of previous similar work followed by a section explaining the methodology used in selecting papers to be included in the review. The next six sections dive into various applications of big data in cybersecurity. Each of these sections is further divided into two subsections to present a brief introduction to the area, and then discuss the research published in that area. The eleventh section presents discussions based on the findings of the review of literature, while the last section presents the conclusions and expected directions in future work. Appendix A presents tables summarizing all papers included in the review.

1.1 Big Data and Big Data Analytics

As legacy database systems are not capable of handling the enormous volume and velocity of big data, new, or developed versions of, data processing algorithms are needed [88]. Figure 1 shows the multiple stages of big data processing.
Big data comes in many shapes and sizes and almost always not ready to be

Fig. 1 Big Data Processing Stages

directly used in analytics. Hence, after data acquisition, big data goes through preprocessing, or staging. Preprocessing is the process of creating a usable data set that is ready for processing [51]. This data set resulting from preprocessing is considered correct, accurate, and sufficient for processing.
After preprocessing, data analysis is performed over the resulting data set. Data analysis techniques include statistical methods, data mining and parallel algorithms, machine learning, and soft computing.
Application of analytics includes a query interface that is tailored to fit the

specific application requirements. This query interface operates as the control dashboard that dictates the product that the user wants to produce from data. The final step of processing is the data visualization. Visualization is the production of visual summaries of data, processing results, and information. It is considered an important step because the amount of data makes understanding it very difficult [88]. Big data analytics, as identified in [100], is a complex process in which large and varied data sets are examined to uncover information including unknown correlations, hidden patterns, market trends and customer preferences that can help organizations make informed decisions.
Big data analytics often involves predictive models, statistical algorithms, and prolonged what-if analysis that is informed by huge amounts of data. As such, big data analytics requires high storage and processing capabilities to have an effective role in an organization.
Techniques used in big data analytics as listed in [86] are:

- Association rule learning
- Data mining
- Cluster analysis
- Crowdsourcing (although this is more of data collection tool rather analysis)
- Machine learning
- Text analytics

This list is not comprehensive and many other techniques can be added to it, such as bio-inspired techniques, parallel algorithms, and Artificial Intelligence (AI) techniques. According to Ghandomi et al., in [50], most big data analytics research focuses on predictive analysis and structured data. However, there are other applications of big data analytics that focus on unstructured data such as audio, video, images, and unstructured text.
Big data analytics can bring value to an organization by improving customer experience, improving operational efficiency, improving marketing effectiveness, providing new revenue opportunities, and providing competitive advantage over rivals by creating the environment for better informed decision making. Well-informed big data analytics can also improve risk management and analyze failures much quicker than legacy methods.

1.2 Cybersecurity

For many years, security was, and will always be, a major concern for all stakeholders in a computer system. Von Solms et al., in [121] argue that the two terms information security and cybersecurity, although used interchangeably, are not completely identical. In general, information security refers to the protection of information resources. On the other hand, cybersecurity includes protection of information sources in addition to other assets, including the person him/herself. Humans, in information security, take part in the security process. However, in cybersecurity, humans can be targets of cyberattacks, or

even tools in a cyberattack.
The term, network security, can be identified as the controls, policies, and procedures used to protect the network assets from unwanted use (as identified in the policy). These assets can be data in storage and/or in transit, software, or hardware.
Many sources use these three terms interchangeably. However, we believe that they are not identical. In general, the term cybersecurity is considered a larger umbrella with broader focus. As mentioned in [106], the definition of cybersecurity infers a broad spectrum of assets to be secured. These assets can be data, systems, software, hardware, or the humans interacting with, or affected by them. This broad definition, dictates high security requirements because there is a lot to lose. As Google's Eric Shmidt puts it "In our digital age, the issues of cybersecurity are no longer for the technology crowd; they matter to us all. "[106]
Classical cybersecurity systems are mostly focused on the current state of a system and how to maintain it. However, the generation of the huge amounts of data discussed earlier, although challenging, can be considered a big opportunity to detect and deter malicious activity. An intelligent system that is capable of analyzing large amounts of data can be employed to detect anomalies, or protect a systems from unintended usage by a malicious attacker.
As cyber risks grow rapidly, security systems are almost always one step behind. Security risks that were previously considered "low-probability" risks are now growing to be more and more probable. In 1977, when the Data Encryption Standard (DES) was first adopted as a standard, the key length of 56 bits was considered more than adequate [110]. The probability of having it broken by brute-force attack was nearly zero, back then. However, twenty years later, in 1997, Curtin et al. published their paper on the first public DES cracking machine [39]. This, in addition to many research directions that helped in reducing the search space for the key retrieval of DES [29,89,69], pushed for the call of a newer standard for encryption that led to adopting the Advanced Encryption Standard (AES) in 2001[98].
The example of DES and AES mentioned earlier is not a standalone example of a single algorithm considered vital in cybersecurity. In 2017, the number of officially reported vulnerabilities in Microsoft's Windows operating system jumped to 587 vulnerabilities [10]. Although the number seems low, with a user base of around 1 billion users [4], these vulnerabilities can have a massive impact.

2 Related Work

In this section, we will introduce previously published research within the area of our paper. In 2013, Mahmood and Afzal presented a survey in the area of security analytics [84]. The survey focused on introducing the idea of security analytics and suggested an implementation path. The paper also suggests a few future directions such as employing security analytics in the reduction of

false positives in a network intrusion detection system.
Oltsik published in 2013 a white paper arguing that the big data security analytics era is here [92]. The white paper discusses the obstacles in the way of improving organizational security maturity and how the legacy security monitoring tools are holding back the developments in security monitoring. The paper states that new security systems must demonstrate the ability of massive scaling, enhanced intelligence, and tight integration with all information technology assets and leverage automated security intelligence. The paper does not provide a clear pathway or a future direction for proper employment of big data in cybersecurity.
Alguliyev et al., in 2014, presented another survey of the potential applications of big data analytics in information security in [20]. The short survey gave a bird's eye view of the different areas of applications. In addition, it discussed briefly the challenges faced by big data analytics applications in information security.
Talabis et al. published, in 2014, a book on security analytics [113]. The book gave a thorough background information on analytics and its applications in cybersecurity. The book discussed applications in the areas of intrusion detection and incident identification, website security, access and access-misuse, text-mining, and security intelligence. Most chapters came with implementations in R programming language.
Sipola published an article, in 2015, summarizing advances in the area of modeling anomaly detection in knowledge discovery process [107]. Knowledge discovery process is a high-level term for deriving actionable knowledge from databases. The article surveys anomaly detection and fingerprinting techniques based on large networks logs. Due to its limited focus on anomaly detection, the article fell-short in producing insights on future directions in the area.
Abdlhamed et al. published a survey on intrusion prediction systems [14]. The motivation behind the work was that despite the current developments in IDSs, significant high-impact attacks are continuously taking place. Thus, a new way of dealing with these attacks is needed; intrusion prediction systems. The survey examines the concepts of work and methods used in these systems. Prediction methodologies studied include alerts correlation, sequence of actions, statistical methods, probabilistic methods, feature extraction, hidden markov model, bayesian networks, genetic algorithms, artificial neural networks, data mining, along with a few algorithmic methods.
Grahn et al. published in 2017 a survey of the use of analytics in network security [54]. The survey examined certain applications of big data analytics in different directions of network security, with high focus on intrusion detection and prevention. The survey also proposed taxonomy for analytics use in network security.
In 2018, Thirumaran et al. presented, in [117], a short survey on the applications of big data analytics in the area of network security. The survey was short and could not cover all the recent state-of-the-art research in the area of network security.
In 2019, Ullah and Barbar published a thorough review of big data cybersecu-

rity analytic systems [120]. The review adopted a systematic literature review methodology with an architectural perspective. The review presented quality attributes commonly associated with big data cybersecurity analytics. The review also presented common architectural tactics successfully used in such systems. The review was comprehensive and presented important findings in the architectural aspect. However, the focus on architectural aspects has led to skipping important other aspects such as the research impact on cybersecurity goals, novelty of the presented system, and other aspects of these systems.
In 2020, Dias et al. presented an overview of the use of big data analytics in intrusion detection [42]. The chapter provides a thorough summary of research within the area. However, it was solely focused on intrusion detection. The paper summarized the papers reviewed but did not present tangible conclusions. In this paper, we present a thorough review of 56 research papers that tackled the topic. Our review was intended to be a comprehensive review that presents a solid foundation for future research in the area and help researchers easily locate significant research contributions in the field. The paper also aimed at categorizing these papers according to the most current categories of common attacks and threats.

3 Methodology

The focus of this research was to collect and review research in various areas of cybersecurity in which big data and/or big data analytics were employed to improve the outcome. Our research questions were:

1. Which areas of cybersecurity are witnessing developments in employing big data?
2. Which areas of cybersecurity have high potential in employing big data and big data anayltics in the coming few years?

Our search included several recognized databases such as IEEEXplore, Springer, ACM Digital Library, Science Direct, and Wiley. Special care was given to the following venues for their high-impact in research in various areas of cybersecurity:

1. ACM Conference on Computer and Communications Security (CCS)
2. IEEE Symposium on Security and Privacy
3. USENIX Security Symposium
4. Network and Distributed System Security Symposium (NDSS)
5. Computers & Security
6. IEEE Transactions on Information Forensics and Security (TIFS)
7. IEEE Transactions on Dependable and Secure Computing (TDSC)
8. ACM Transactions on Privacy and Security (TOPS)

In addition, we decided to expand our search to include papers from ArXiv Computer Science [5] although not all papers there are peer-reviewed. This

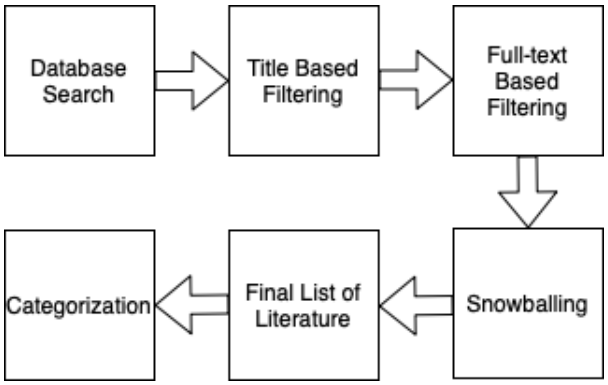

Fig. 2 Literature Selection Process

was motivated by the fact that ArXiv database includes new research that has not been yet published formally in the previously mentioned venues. This can give us a better view to explore future trends in order to respond to our second research question.

The literature selection process and preparation process went through six phases, shown in fig 2.

The first phase was the database search which resulted in 2731 papers found based on our search. The search target was for the past five years, 2015-2019. However, we ended up including papers from 2020. The second phase was title-based filtering which resulted in 1074 papers. The third phase, which was full-text-based filtering, resulted in the selection of 49 paper only. Full-text selection was a very slow and comprehensive process that eliminated papers that did not present noticeable novel solutions. Other papers were excluded because they did not provide proper evaluation of the proposed system. The reduction of the number of papers from 1074 to 49 was expected because the title of the paper does not necessarily imply that it is employing big data techniques to propose a solution to a cybersecurity problem. After this short-listing of papers, we applied a Snowballing technique, explained in [122]. The principle of snowballing is exploring the references of the selected papers for the review and expanding from there to include relevant papers in our review. This resulted in the increase of the selected papers to 56 papers, which was the final number of papers reviewed in our survey as identified by the fifth stage of finalizing the list of papers to be reviewed. Figure 3 shows the distribution of the selected papers over the years of publication. The final stage was the distribution of papers over different categories.

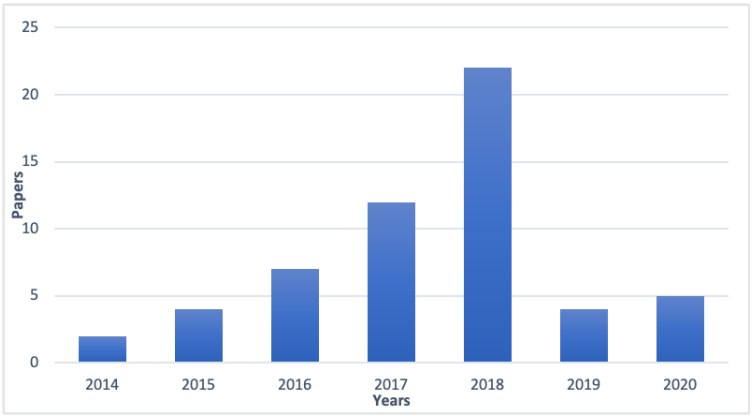

Fig. 3 Selected Papers' Distribution Over Years

4 Applications of Big Data in Cybersecurity

In 2012, a report published by Gartner [82] said "Information security is becoming a big data analytics problem, where massive amounts of data will be correlated, analyzed and mined for meaningful patterns. Investments in additional tools, processes and skills will be required." This shows that the trend of using big data analytics in cybersecurity is not that new. However, the developments taking place in techniques and technologies used have widened the possibilities of implementations. This has turned big data analytics from an "offline" tool to produce reports for security analysts into an integrated "online" component built into security systems to make instant decisions against attacks in real-time.

As cybersecurity has diverse areas, applications of big data analytics in cybersecurity are of diverse nature as well. However, to keep the survey clear and focused, we have subdivided the surveyed papers based on their specific area in cybersecurity, as discussed in section 3. The areas chosen are intrusion and anomaly detection, malware and ransomware, cloud security, code security, spam, and another section listing other directions of research in big data and cybersecurity. Papers in these areas were discussed in a chronological order in each section. The categories were carefully selected based on the most common threats and attacks identified in [3,6,11,1]. In addition, the number of papers falling into each category dictated which categories to choose as the main five categories. An additional sixth category was created to host the review of the papers not particularly fitting in any of the top five ones.

5 Intrusion and Anomaly Detection

5.1 Definitions

An Intrusion Detection System (IDS) is a system designed to detect unauthorized access to networks and resources. IDS can be either host-based, or network-based. A Host-based IDS (HIDS) is resident at the host side and its scope of work is focused on that particular host. This host can be a computer, a server, or any other type of network-connected host. A Network-based IDS (NIDS) resides usually on the edge of the network and has a broader scope of work that covers all traffic incoming our outgoing through the network. IDSs can also be categorized based on their operation into signature-based and anomaly-based. Signature-based IDS has a signature database by which it identifies the scanned item as an intrusion or not. These databases needs to be updated frequently for the IDS to be able to identify the most recent intrusion types. Anomaly-based IDS creates a base-line for what is considered "normal" operation and flags a warning that an anomalous behavior or operation was detected [126].
Although it sounds that "anomaly-detection" is part of IDS, it is not necessarily part of it. Anomaly-detection, in general, is the detection of any anomalous behavior that deviates noticeably from a base-line. This can be in network traffic, a process or software running on a host, machine-learning training data or any other aspect of a system. More information on IDSs and anomaly detection can be found in [126].

5.2 Literature

In 2014, Tan et al. proposed a framework for collaborative intrusion detection [114]. Although the focus was on protecting data stored on the cloud, the proposed system correlates suspicious evidence between different IDSs to improve the efficiency of intrusion detection. The system is based on sharing of traffic information between IDSs. This requires that IDSs are connected either in decentralized or hierarchical arrangements. In decentralized arrangement, each IDS can generate a complete attack diagram of the network by the aggregation of network data received from other IDSs in the collaborative system, and thus, detection happens at each IDS independently. In hierarchical arrangement, one coordinator is responsible for data aggregation, data analysis, and generation of the attack diagram of the network. The proposed system goes a step further and suggests the utilization of network hosts in collecting traffic data to feed the collaborative IDS. The parallel summarization is proposed to be done through MapReduce. The paper does not mention implementation results and hence is considered a conceptual framework rather than a fully tested system with comparable results.
Xu et al. introduced in 2016 a system that supports detection of anomalies

through reduction of the system log data [123]. The proposed system aims to maintain the dependency of events while aggregating the security-related data to maintain high fidelity forensic analysis. The aggregation algorithm exploits the dependency between system events to reduce the number of log entries without impacting the quality of the forensic analysis. After aggregation, an aggressive reduction algorithm is applied, along with the use of exploit domain knowledge to achieve further data reduction. The proposed system is then evaluated on real-world auditing systems using log traces collected over the period of one month. The initial evaluation of the proposed system shows that it can significantly reduce the size of system logs to improve the effeciency of the forensic analysis without compromising accuracy. Another research article, published by Hussain et al. in 2018 [61], also tackled reduction of log data for forensics purposes. The proposed technique is said to reduce the number of records by a factor of 4.6 to 19. The paper also discusses how the proposed method preserves the accuracy of the forensic analysis tasks such as backtracking and impact analysis by preserving the dependence of events. The proposed technique reduced the testing file size by 35 times across all datasets, which makes it on average using 5 bytes per event in memory and capable of analyzing about 1 million events per second. Another article was published in 2018, by He et al. in [59] that characterized the current state of the art log parsers (Zookeeper, Proxifier, BGL, HPC, and HDFS) and evaluated their efficacy on five datasets with over ten million log messages. The study determined that although the accuracy of the parsers is generally high, they are not robust across all datasets. The study proved that these log parsers' efficiency decline when the log files grow to a large scale when run on a single computer. The study proposed a novel Parallel Log Parser (POP) that runs on top of Spark, a large-scale parallel processing platform. The proposed system was evaluated and has demonstrated high accuracy, effectiveness, and efficiency.

Cao et al. introduced, in 2014 as well, a proposed unified end-to-end security testbed and security analytics framework [31]. The two aims of the proposal were to understand the real-world exploitation of known security vulnerabilities, and to detect multi-stage attacks preemptively and stop them. The research employed virtualization techniques to provide the necessary isolation of attacks in linux-based virtual machines. To monitor the system behavior, kernel probes and network packet capturing was used. As this was not a full-paper, the research results were inconclusive on the effectiveness of the proposed model. However, it seemed promising.

Zuech et al. published in 2015 a survey paper discussing state-of-the-art research in the area of intrusion detection and big heterogeneous data[133]. The review discusses particular issues of data fusion, heterogeneous intrusion detection architectures, and Security Information and Event Management (SIEM) systems. The survey also discussed several directions in which the research can go in relation to this area. The review is thorough and detailed, and can be considered a good collection of resources in the area of intrusion detection and big heterogeneous data.

In 2015, Pierazzi et al. proposed a novel framework that is designed to inves-

tigate temporal trends and patterns of security alerts [94]. The purpose of the investigation was to better understand which anomaly detection approach to adopt in identifying relevant security events. The motive behind this research is the fact that the number of security alerts generated by various network defense systems has grown beyond the capability of network administrators to manually inspect the security events. When the proposed framework was tested with several examples, it has shown the ability to extract relevant descriptive statistics that helps in measuring the effectiveness of various popularly used anomaly detection approaches in detecting different types of alerts.

Zhu and Dumitras introduced, in 2016, a malware detection system based on end-to-end approach for automatic feature engineering, named FeatureSmith [131]. The proposed system works by mining documents written in natural language, such as scientific papers, representing and querying the knowledge extracted from these documents in relevance to malware, in a way that is mirroring the human feature engineering process. The research focuses on identifying abstract behaviors related to malware and mapping these behaviors to features that can be tested experimentally. FeatureSmith was built on that concept to detect malware in Andoird-based systems. The system was trained on a large dataset of benign and malicious applications. FeatureSmith acheived 92.5% true positive, and 1% false positive which is comparable to other current malware detection systems. The proposed system was also capable of suggesting informative features that were not available through manually engineered set and to link the features generated to abstract concepts that describe malware behaviors.

Bilge et al. introduced in 2017 a system for prediction of risks of cyber incidents [30]. The proposed system, named RiskTeller, relies on analysis of binary files appearance in logs to predict machines with infection-risk months in advance. The system creates a profile for each machine that captures its usage patterns. Then, the system associates each profile with a risk level using supervised (Random Forest Classifiers) and Semi-Supervised Learning (SSL) methods. The proposed system was evaluated using a year-long data set that includes information about all binaries appearing in the machines of 18 enterprises. The testing phase has shown very good results compared to other techniques.

Also in 2017, Du et al. presented an anomaly detection system that employs deep learning [43]. The proposed system, named DeepLog, is a deep neural network model utilizing Long Short-Term Memory (LSTM) to model a system log as a natural language sequence. The system gets trained to learn log patterns from normal execution and detect anomalies when these patterns deviate from normal execution. The proposed system is updated online to include further normal execution patterns over time. DeepLog also has the capacity of building a workflow from the log data such that when an anomaly is detected, root cause analysis can be implemented by users easily. Testing of the proposed system has shown noticeable improvement in comparison to other anomaly detection methods. DeepLog was also tested on the VAST Challenge 2011 data set and was proven to detect 5 out of the 6 attacks introduced in

the dataset.
Another article published in 2017 by AlEroud and Karabatis introduced a novel contextual framework consisting of several attack prediction models that can be utilized in conjunction with IDSs to detect cyber-attacks [18]. The proposed system employs extractable contextual elements from the network data to create knowledge-based context-aware prediction models that can be applied in combination with other intrusion detection techniques to assist in identifying attack; both known and unknown. The proposed framework focuses on significant dimensions in data. Hence, the expected computational overhead is kept minimum.
Siadati and Memon introduced, in 2017 as well, a system designed to detect anomalies in logins within an enterprise network [105]. In certain types of attacks, the attacker uses credentials stolen from users to login to a network and transfer files between computers. The proposed system addresses this problem by extracting a collection of login patterns using a variation of market-basket analysis algorithm. The resulting login patterns are then used in anomaly detection to detect malicious logins that show in consistency with the login patterns of the organization. The system was tested and has proven operation in real setting. Tests have shown that the system was capable of detecting 82% of malicious logins with 0.3% false positive. The data set used was a collection of millions of logins of a global financial company that was collected over the period of five months. The data set involved a total of 25,450 unique usernames with 33,151 computer names.
In 2018, Cuzzocrea et al. introduced an assessment of various machine learning tools for detection of anomalous behaviors in complex environments [40]. The paper was focused on applying and experimentally assessing machine learning tools to solve security issues in complex environments with a special focus on identifying and analyzing malicious behaviors. The first part of the paper was focused on detection and analysis of Tor traffic. This analysis was done based on a machine learning-based discrimination techniques. The second part was focused on the employment of deep learning in the identification and analysis of Controller Area Network (CAN) bus attacks. The last part of the paper was focused on detection and analysis of mobile malware. This part evaluated the use of structural entropy-based classification in detecting ransomware in Android environments. In general, the paper's results have shown a confirmation of the effectiveness of machine learning in supporting security activities in complex environments.
Shen et al. introduced, in 2018, a system for security events prediction based on machine learning [102]. The proposed system, named Tiresias, is said to not only provide binary results showing whether the attack will happen or not, but to predict the steps that the attacker would undertake. The proposed system employs recurrent neural networks to study previous events and predict the next event that would happen. The proposed system was tested on a large data set of 3.4 billion security events collected through a commercial intrusion prevention system. The system, when tested, demonstrated an accuracy of 93% in predicting the next event that would occur on a machine. The

system's shown stability over time and embeds a self-healing mechanism by which the system detects sudden drops in precision and triggers a retraining episode.

Alsadhan et al. introduced, in 2018, a machine-learning based system for detecting Distributed Denial of Service (DDoS) attacks in IPv6[23]. The study was focused on DDoS attacks performed using Neighbor Discovery Protocol (NDP) that is used in IPv6 as an alternative of Address Resolution Protocol (ARP). NDP protocol was previously identified as a possible tool for DDoS attacks in [124]. The proposed system utilizes machine learning in detecting the use of NDP in DDoS attacks. Several machine learning algorithms were tested in the study out of which decision tree and random forest algorithms proven to give the highest accuracy as compared to other algorithms.

Khan et al. introduced in 2019 a crowd anomaly detection system that focuses on rejecting motion outliers [70]. Crowd anomaly detection is the collaborative work of crowd agents in detecting anomalies. Crowd anomaly detection algorithms usually consume a lot of power and that makes it rarely applicable for battery-powered surveillance cameras or other small-sized mobile devices. The proposed algorithm creates a feature for each superpixel that does not include contribution from its neighbors unless they conform with the dominant direction. The proposed algorithms was implemented on low-power Field-Programmable Gate Array (FPGA). The testing results have shown that the system is capable of processing 126.65 megapixels per second with a maximum of 2.43 nJ of energey required per pixel.

In 2019, Clinque et al. proposed a novel approach to accompany microservices logs with black box tracing to support the decision-making process in troubleshooting [35]. The proposed approach is based on passive tracing of request-response messages of the REpresentational State Transfer (REST) communication model. The paper presents two case studies based on Clearwater IP Multimedia Subsystem (IMS) setup consisting of Docker microservices and a Kubernetes orchestrator deployment hosting tens of microservices. The proposed approach, named MetroFunnel, allows making useful attributions in traversing the logs; more important, it reduces the size of collected monitoring data at negligible performance overhead with respect to traditional logs.

Cinque et al. also presented, in 2020, a log filtering method that is designed to pinpoint interesting events to be followed up by human analysts [36]. The proposed system was implemented on real-life log data collected by a large company operating in the air traffic control domain. The results were compared with a reference filtering technique based on conceptual clustering. This comparison yielded that the proposed method is effective to retain interesting events at very high precision. As the method reduces the need for human experience in reading large amounts of log data, it provides a cheaper solutions to organization who do not have this experience.

In 2020, Kotenko et al. presented an approach to detect network attacks and anomalies based on machine learning and big data technologies [74]. The presented approach is a combination of several layers of data processing. This combination included datasets extraction and decomposition, features vector

compression, training, and classification. Analysis of the input vector is done using various binary classifiers; support vector machine, k-nearest neighbors, Gaussiamn naïve Bayes, in addition to artificial neural networks and decision trees. The proposed approach combined these classifiers into a single weighted ensemble to improve the precision of the attack detection process. This combination is achieved through weighted voting, soft voting, AdaBoost, and majority voting. The proposed approach employs two different architectures of Distributed-IDS based on big data. The first architecture achieves parallel data processing by splitting data into non-intersecting subsets, and assigning a separate parallel thread to each data subset. The second architecture is a client-server one. It is comprised of several client-sensors with a server-collector. Each sensor contains several network analyzers and a balancer. The proposed approach was tested with two different large datasets; the first was IoT traffic including several kinds of attacks (with over 7 million instances), and the second with computer network traffic including host scanning and DDoS attacks (with over 500,000 instances). Results of testing showed good results in terms of accuracy of detection and speed of detection.
In 2020 as well, Yuan et al. presented a log anomaly detector based on an unsupervised online deep neural network, named ADA [127]. ADA employs an adaptive model selection strategy to choose pareto-optimal configurations and thereby utilize resources efficiently. It also utilizes a dynamic threshold algorithm to dictate the optimal threshold based on recently detected events to improve the detection accuracy. Based on testing conducted using the Los Alamos National Laboratory cyber security dataset, ADA accurately detects anomalies with high F1-score of nearly 95% and it is 97 times faster than existing approaches and incurs very low storage cost.
Zuo et al. proposed, in 2020, a learning-based anomaly detection framework for service-provision systems with micro-services architectures [134]. The proposed framework uses service execution logs (temporally) and query traces (spatially). It includes two major parts: logging and tracing representation, and two-stage identification via a sequential model and temporal-spatial analysis. Implementation shows clearly the effectiveness of three essential components in the framework; template extraction, sequential anomaly degree model, and temporal and spatial joint anomaly detection. The proposed framework brings to light a promising branch of research that can be explored further.

5.3 Summary

Anomaly and intrusion detection was one of the most studied areas of big data applications in cybersecurity. Most reviewed papers were focused on implementing machine learning in the detection of anomalies and intrusions. Other studies focused on the reduction of log data to make the anomaly detection faster.

6 Spamming, Spoofing and Phishing Detection

6.1 Definitions

Spamming can be defined as the process of sending unsolicited messages. According to statistics published in [26], spam emails make up 45% of all emails sent around the world every day. This translates to about 14.5 billion emails daily. Spam has been one of the biggest security challenges since the beginning of the Internet.
Spoofing, on the other hand, is a general term for disguising a communication from a malicious source to look like it is coming from a known and trusted source. It has different forms, such:

- IP address spoofing: Creating a packet with a false sender address to hide the actual source or target the spoofed IP address.
- DNS Spoofing: The malicious actor injects false IP addresses in a DNS server to divert traffic intended for the actual server to the spoofed malicious server. This attack is sometimes called DNS poisoning.
- ARP Spoofing: A malicious actor's host pretends to be the owner of the IP address that is rightfully owned by another host on the network.
- MAC Spoofing: A malicious actor's host pretends to be the owner of the MAC address that is rightfully owned by another host on the network.
- Email Spoofing: A malicious actor alters the sender email address in an email to make it look like it is originating from the victim's email.

Other types of spoofing attacks exist as well. Further information on different types of spoofing attacks can be found in [26].
Phishing is a malicious attempt to obtain information or data through disguising as a legitimate trustworthy entity. Phishing can happen through email (usually named phishing), phone (vishing), or messages (smishing, or SMS phishing) among other forms. In addition to the categorization based technology used in phishing, phishing further can be categorized based on its targets, as listed below.

- Phishing: targeting a very wide target base hoping that one or more targets would respond.
- Spear-Phishing: targeting specific individual or a small group with carefully crafted phishing attack.
- Whaling: specific targeting of highly-ranked individual or individuals, such as chief executives, and manager.

Recently, phishing became a powerful tool to achieve malicious actions. An example can be found in the recent hack that targeted the social media service Twitter, in 2020. This attack resulted in compromise of high-profile accounts such as Barack Obama, Joe Biden, and Elon Musk. Twitter announced a few days later that it was a highly orchestrated spear-phishing attack against several employees with high-level access to Twitter internal systems [109]. Further information on phishing, its types, and attack vectors can be found in [34, 19].

6.2 Literature

In 2016, Liao et al. introduced a promotional-infection detection system based on semantic-inconsistency search [80]. Promotional infection can be identified as an attack in which the attacker exploits a vulnerability to inject illicit advertising content. You can be visiting a university's website and an advertisement on adult medication can show up. The proposed system is based on exploiting the semantic gap between the text of the advertisement and the sponsored top-level domains (sTLD). The approach employs natural language processing to identify the bad terms most irrelevant to the sTLDs. Semantic analysis is performed on the search results that are produced by searching the irrelevant bad terms in search engines under the sTLD for suspicious domains. Then, the system will be able to detect truly infected websites. During testing, the system analyzed 100,000 fully qualified domain names (FQDNs) running on 403 sTLDs with initial 30 seed irrelevant bad terms. At the end of the test, the system detected 11,000 infected FQDNs, with a false positive rate of 1.5% and over 90% coverage.
Zhang et al. published, in 2016, a study elaborating the growth and effects of botnets in social media[129]. The study identifies social botnets as a group of social bots, controlled by a botmaster, collaborating to conduct malicious behavior while they mimic normal social media user behavior to reduce the risk of being detected. The study demonstrates the effectiveness of using social botnets on Twitter. The researchers bought 1000 Twitter accounts for 57$ and built a botmaster Java application to control the accounts. The attacks identified were spam distribution, and digital-influence manipulation. The study proposed defense mechanisms against the attacks that can be performed using social botnets. As spam distribution relies on exploiting retweet trees, the study proposes tracking user history of participation in spam distribution and set a specific accumulated suspicious behaviors threshold that when exceed, a user account is labeled as a spammer and suspended. Each user would have a spam score that is updated whenever a user retweets a spam tweet. As for the digital-influence manipulation, the study proposes a method based on [57]. The main idea is to find adequate number of credible users and use their actions alone as a source of digital-influence scores. These actions can be following, mentioning, retweeting, and replying. Thise credible users will be the soul source of calculation of digital-influence score.
Yao et al. introduced, in 2017, a study on automated crowdturfing attacks and defenses in online review systems [125]. As malicious crowdsourcing forums gain popularity in spreading misinformation, ways of automating this malicious task are also gaining traction. The paper identified a new class of attacks using deep learning language models (recurrent neural networks) that is used to generate fake online reviews of various products and services. This class of attacks makes it difficult for detection systems to detect it by controlling the rate of content output to eliminate identification of its signature. These fake reviews, as the study finds, were not only undetected, but also scored well on "usefulness" metrics by users. The study concludes by devel-

oping a novel automated defense system against these attacks by leveraging the lossy transformation introduced by the recurrent neural networks training and generation cycle.

In 2017, Nilizadeh et al. introduced a Twitter analytics system that can identify similar-interest communities and leverage the differences in the propagation between benign and malicious messages on social networks to identify spam and other unwanted content [91]. The presented system, named POISED, was tested on a dataset of 1.3 million tweets collected from 64 thousand users. Testing results have shown malicious message detection with 91% success rate and 93% recall. The proposed system was also compared to three other state-of-the-art spam detection systems and has shown significant improvement over the performance of the other systems. Finally, the paper shows that POISED is resilient to two types of adversarial machine-learning attacks along with its capacity for early detection of spam.

Ikram et al. introduced, in 2017, a system for detecting Facebook like farms [63]. Like farms are businesses that are baed on artificially inflating Facebook's page or post 'likes'. As the number of likes has become a de-facto measure of a page's or business's popularity and success, this kind of shady business grew in popularity. The study aims at filing the gap by a hoenypot-based comparative measurement study of page likes received through advertising and those received through like farms. The analysis is first done based on demographic, temporal, and social characteristics. This has revealed that some like farms are fully operated through bots and does no effort to hide their operations while others try to operate in a stealthier approach by trying to mimic actual users. The study have shown that the fraud-detection algorithms currently employed by Facebook are not actually effective in capturing stealthy like farms that operate by spreading likes over longer time span and like popular pages in a trial to mimic actual users. The proposed system identifies genuine and fake social activity via investigating timeline-based detection of like farms with a special focus on characterizing content generate by Facebook accounts on their timeline as an indicator. The analysis included extraction of features from accounts timelines and categories them into lexical and non-lexical. Like farms accounts have the tendancy of re-sharing content more often and use fewer words with poorer vocabulary with apparent generation of duplicate comments and likes in comparison to normal users. The classifier built using lexical and non-lexical features has shown accuracy of detection of 99% with 93% recall. Comparatively, these results are impressive and development can be done on the proposed system for more efficient deployment and results speed.

In 2018, Jansen et al. introduced a crowdsourcing-based method to detect and localize GPS spoofing attacks [64]. The proposed system, namely Crowd-GPS-Sec, does not require updating the currently available GPS infrastructure, nor the airborne GPS receiver. Instead, the proposed system relies on employing the current GPS infrastructure in the process of crowdsourcing to monitor the air traffic using the GPS location advertisements broadcasted periodically by airplanes. Crowd-GPS-Sec detects and localizes spoofing attacks using an

independent infrastructure on the ground that analyzes continuously the contents and arrival times of these location advertisements. The proposed system was evaluated with real-world data. The test data contained 141,693 unique positions of 142 airplanes. The proposed system achieved attack detection delay of 2 seconds, with an attacker localization accuracy of 150 meters using data from 15 minutes of monitoring time.

Li et al. introduced, in 2018, a machine learning based system to detect malicious calls in a telephone network [78]. The malicious call identification system proposed is based on building a data set by users. The first step was to develop a user interface that enables users to tag malicious calls. This allowed the researchers to collect a data set of 9 billion records over the period of three months. The second step was extracting 29 features from the data that allowed the machine learning based solution to be trained to classify received calls in near-real-time. The researchers did extensive testing with different machine learning approaches. At its best, the proposed system was able to detect 90% of malicious calls and had an accuracy of 99.99% in identifying non-malicious calls. Neural network based implementation of the system cause minimal latency of less than 1ms. Upon further testing, it was found that 10 features out of the 29 were adequate to achieve comparable accuracy.

Gutierrez et al. introduced in 2018 a detection system directed towards new phishing attacks [56]. The proposed system employs a machine learning classifier operating on a large corpus of phishing and legitimate emails dataset. The system, titled Semi-Automated Feature generation for Phish Classification (SAFe-PC), operates by feature extraction, elevating some to higher level features, that are meant to defeat common phishing email detection strategies. The proposed system was trained and tested on two datasets collected from central IT organization of a tier-1 research university and results of phishing identification were compared to those captured by Sophos, a state-of-the-art email filtering tool, and SpamAssassin. The first dataset comprises 37,606 email messages that Sophos has not identified as phishing while the second dataset comprises 388,264 messages that Sophos did identify as phishing. In addition to these phishing datasets, the experiment used legitimate emails from universities, public newsgroups, and publicly available financial emails, thereby keeping the domain of legitimate emails relatively equivalent to the phishing dataset. The experiment has shown that the proposed system caught 70% of the emails that were not caught by Sophos. The researchers have also created an online version of SAFe-PC that can be incrementally retrained with new samples, which improves its detection accuracy.

In 2020, Sun et al. presented an approach to identify hidden security threats using Uniform Resource Locators (URLs)[112]. This research is focused on detecting drive-by-download URLs that many users neglect their security threats. Drive-by-download attacks comprised 93% of the 4.7 million daily web attacks that took place in 2013[48]. The proposed malicious URL identification systems, named AutoBLG, employs machine learning techniques to identify malicious URLs based on a vast number of previously-known malicious URLs databases. AutoBLG is comprised of three phases; URL expansion, filtering,

and verification. In the first phase, the aim is to generate a set of suspecious URLs from already known IP address hosting one malicious URL. Generally, it is likely that the same IP would host more than one malicious URL. HTML content of these malicious URLs is downloaded in this phase as well. In the second phase, the suspicious URLs and HTML contents are fed into the URL filtering module, which significantly reduces the number of suspicious URLs using a bayesian set algorithm, a machine learning algorithm. At the end of the second phase, those URLs that are most similar to the known malicious ones are selected as input for the next phase. At the third phase, the most suspicious URLs obtained through URL filtering are tested to confirm whether they are actually malicious. The confirmation tools employed are a web client honeypot, antivirus software, and online URL reputation checker. The proposed system, AutoBLG, can achieve a high noise filtering of 99% and toxicity range from 1.17 to 16.5%. Compared to crawler-based systems, both the noise filtering and the toxicity of AutoBLG are said to be higher than that of crawler-based systems.

6.3 Summary

As expected, most applications are focused on the detection of spamming, spoofing, and phishing using various big data and machine learning techniques. Special attention in the reviewed studies was given to social media, and social media exploitation detection.

7 Malware and Ransomware Detection

7.1 Definitions

Malware, short for *Malicious Software*, is a collective term for all software and programs written with a malicious intent [118]. An average of 350,000 malware attack detected every day around the world, and about 10 billion attacks in 2018 alone, malware detection has become an important application that requires all possible defense mechanisms [67].
Malware be created in a variety of forms such as backdoors, computer viruses, trojans, worms, hoaxes, logic bombs, etc. The variation in these forms of malware, along with the developed techniques malware creators are using, such as encryption, obfuscation, and dynamic codes, identifying malware has become challenging task. Hence, technologies like big data, machine learning, and cloud computing were called in to the rescue. More information on malware can be found in [97, 126].
Ransomware is a special form of malware in which the malicious actor encrypts all or part of the victim's data and ask for money (i.e. ransom) in return for the decryption key. With a growth in global losses due to ransomware from

350$ million in 2015 to 11.5$ billion in 2019, ransomware became one of the most profitable malware forms for malicious actors [38].
In January 2018, Hancock Health, a hospital in Greenfield, Indiana, woke up to see its information technology infrastructure held hostage by a ransomware attack [13]. A ransomware attack, as the name indicates, is an attack in which all or most of user data is encrypted by a malicious attacker with an encryption key unknown to the victim. The attacker holds the data as a hostage and asks for a ransom in exchange for the decryption key and the decryption tool. Hancock Health hospital was brought to a complete halt due to the attack, as the attacker had encrypted over 1,400 files and renamed them "I am sorry". As any hospital in this day and age, almost everything in the hospital relied on information technology and computerized systems. Although the hospital had regular backups of their data, restoring the backups was not an options, as it would take days or weeks to restore and get systems online again. As the hospital system is considered extremely mission-critical where lives could not be jeopardized. The hospital had to pay $55,000 to get the decryption key and restore its files and its operations to normal as early as possible. This was not the first attack of its kind on a hospital, and will not be the last. In a similar attack earlier in 2016, the attackers received the ransom, but decided not to decrypt all files, and asked for more money to decrypt the rest of the files [108]. Zhao et al. discussed the impact of such an attack on the operations of a hospital in [130].
Another report published in 2017 stated that ransomware attacks brought in around $1 billion in 2016[12]. The reports stated that industries most susceptible to ransomware attacks are education, government, energy, and utilities, and healthcare. Despite that, the highest targets of these attacks were in the industries of finance, retail, healthcare, energy, utilities, government and education, respectively.
Many other examples in almost all industries exist to show us the importance of cybersecurity and why everyone should take it seriously. In some cases, the victim was at no fault, but got the hit anyway. The ransomware attack on Hancock Health was not successful due to an employee opening a malicious email, as in most other ransomware attacks. The attacker used credentials of a partner of the hospital and got access to a portal that is especially built for partner service providers. More information on ransomware attacks can be found in [96].

7.2 Literature

Kwon et al. introduced in 2015 a study that introduces downloader-graph abstraction that captures download activity on end hosts and explores the growth patterns of benign and malicious graphs [76]. Downloader-graphs can support the process of detecting malware download activity that may otherwise remain undetected. The proposed system employed telemetry from anti-viruses and

intrusion-prevention systems to reconstruct and analyze 19 million downloader graphs from 5 million real hosts. The study identified several strong indicators of malicious activity such as growth rate, diameter, and Internet access patterns of downloader graphs. The proposed detection system had a 96% true-positive rate and a 1% false-positive rate. In terms of detection speed, the proposed system proved to be faster by an average of 9.24 days compared to existing commercial anti-virus products.

In 2018, Huang et al. introduced a novel end-to-end tracking of ransomware [62]. The study introduced in the paper was based on data collected for two years. Data about ransomware payments, victims, and operators was collected from various sources including labeled ransomware binaries, victims' ransom payments, victim telemetry, and a large database of Bitcoin addresses annotated with their owners. The study was the first of its kind in terms of studying the ransomware ecosystem, not just small components of it. The study made a correlation between several elements to create a profile of victims and operators. With the wealth of data collected ,the study was able to connect infection timing, Google search trends, payment mechanisms, payment timing, among other elements.

In 2018 as well, Koli introduced a malware detection system named RanDroid [73]. The proposed malware detection system was based on machine learning. The main idea is based on collecting large number of random samples of "goodware" and malware applications to train the machine learning based classifiers. The system was designed to learn feature like permissions, suspicious API calls, dynamic code, reflection code, native code, cryptographic code, and the database. The learning data set in this paper was the top 120 top rated applications that did not include malware, in addition to 175 applications that included malware. Out of the total number of applications, 20 malware-free apps, and 25 malware infecting apps were used for testing. During the testing phase, four different types of classification algorithms were used, out of which Decision-Tree had the highest accuracy in comparison to the other three. Although the paper included a comparison with other Android malware detection systems, the relatively-small data set causes less confidence in the results as a whole.

In 2019, Ugarte-Pedrero et al. published an article discussing how to handle and utilize daily datasets of malware samples [119]. As the number of unique malware samples is rapidly growing, malare detection solutions need to keep up. Security companies collect more than one million unique files per day from its different feeds in order to perform analysis and find new strands of malware. The study guides the reader through a step-by-step analysis of hundreds of thousands of Windows executables collected in one day from the security company feeds. The main aim of the study is to show how a company can employ state-of-the-art techniques for automated processing of samples and perform manual experiments to have a better understanding and documentation of the contents of the dataset. The study concludes with a rough estimate of the human and computer resources needed to make use of the large amount

of data collected.

7.3 Summary

Applications in areas of malware and ransomware were mostly in areas of detection of malware and ransomware. Furthermore, some studies focused on gathering data to further understand malware and ransomware.

8 Code Security

8.1 Definitions

With the global growth in the adoption of open-source software, new security challenges arise. Security challenges in open-source software come from different directions. One direction is malicious actors implant weaknesses in the open-source software and push people to use it. Another direction is that slowness or lack of updates if you're using an open-source software that is no longer maintained by its creator. Another source of threat is having other developers clone parts of the open-source software and use it in another piece of software. If a vulnerability existed in the original code, it will be available in all of the clones.

8.2 Literature

In 2016, Liao et al. introduced an automated Indicators Of Compromise (IOC) detection system, named iACE [79]. The proposed approach relies on the way IOCs are explained in technical article. The approach depends on the predictible way in which the IOCs are described using a set of context terms, such as "download" through suitable grammatical relations. iACE was designed to automatically locate a putative IOC token (such as a compressed file) and its context (such as "malware", or "download") within the text of the technical article. iACE then applies graph mining technique to analyze the the relations of IOCs and the context. The proposed system was run on 71,000 articles published in a period of 13 years from 45 technical blogs and generated 900,000 OpenIOC items with a precision of 95% and coverage over 90%. These results exceed the accuracy of NLP-based techniques. In addition, the proposed system was capable of handling thousands of article per hour.
In 2017, Shu et al. presented a program anomaly detection system that is based on mildly context-sensitive grammar verification [104]. The purposed system's, named Long-span behavior Anomaly Detection (LAD), main feature is its reasoning of correlations among arbitrary events occurring in long

program traces. LAD utilizes a specialized machine learning techniques, constrained agglomerative clustering algorithm purpose-built, to probe normal program behavior boundaries in vast high-dimensional detection space. The prototype was tested and successfully detected all reproduced real-world attacks against sshd, libpcre, and sendmail binary packages. Latency overhead during testing was limited between 0.1 ms to 1.3 ms to profile and analyze a single behavior instance. This single behavior instance consists of tens of thousands of function call or system call events. This small overhead is considered acceptable keeping in mind the size of the search space.

Banescu et al. introduced, in 2017, a system for predicting the resilience of obfuscated code against sybmbolic execution attacks using a machine learning approach [27]. The paper presents a framework for choosing the most relevant features in the software to estimate the effort required by automated attacks to deobfuscate the software. The features are used to build regression models to predict the resilience of different software obfuscation transformations against automated attacks. To train the proposed model, a code generator was implemented to generate a large number of arbitrarily complex random C functions. Open-source software was not used in the training because open-source software, in its majority, does not contain the security checks (such as a license check) that the software that is usually obfuscated contain. Hence, the code generator was used to generate 4608 unobfuscated C programs as the dataset. Testing results have shown that the number of community structures in the graph representation of the symbolic path-constraints have far more impact on the prediction process than other feature, such as cyclomatic complexity. The best model introduced was able to predict the number of seconds of symbolic execution-based deobfuscation attacks with 90% accuracy for 80% of the programs in the dataset. The accuracy of the proposed system relies mostly on the dataset used in training. Hence, future improvements in the accuracy can be achieved when a better real-world dataset is used in training.

Kim et al. presented in 2017 a novel vulnerable code clone discovery system named VUDDY [72]. The proposed system is a largely scalable one that can preprocess one billion lines of code in a little over 14 hours after which it takes only a few seconds to identify code clones. The proposed system includes a security-aware abstraction technique that helps the system to detect common modifications in cloned code. This feature enables VUDDY in detecting variants of known vulnerabilities with relatively high accuracy. During the testing phase, the proposed system is said to outperformed SourcererCC, ReDeBug, DECKARD, and CCFinderX in terms of detection time but with lower number of clones reported.

In 2017, Feng et al. introduced a scalable graph-based bug search for firmware images [47]. With the increase of breaches in IoT devices, vulnerability search in massive IoT ecosystems has become vital. The proposed system addresses the scalability challenges in existing cross-platform bug search techniques and is expected to improve search accuracy. The proposed search engine, named Genius, when tested, has shown significant improvement over previous systems in terms of speed and accuracy. The test was done on a dataset of 33,045

devices which was collected partially from public systems. The average search time was 1 second when performed over 8,126 firmware images of over 420 million functions. The top 50 candidates in the search results contained 38 potentially vulnerable firmware images across five different vendors out of which 23 were confirmed through manual analysis.

Kim et al. introduced, in 2018, a study examining the code-signing Public-Key Infrastructure (PKI) revocation effectiveness [71]. The distributed and closed nature of code-signing PKI makes the process of measuring revocation effectiveness difficult in this ecosystem. Certificate revocation is the tool that is used to eliminate the danger of certificates that have been compromised or issued to malware authors directly. Hence, the effectiveness of the revocation process has to be as high as possible to minimize the threats of abusive certificates. The study collected seven datasets from different sources the totaled to 965,000 certificates. As the revocation process relies on three parts; discovering the abusive certificate, revoking the certificate effectively, and disseminating the revocation information to the clients. The study's main focus was the challenges of discovering and delays in revocation of certificates. The study shows that the erroneous setting of revocation dates can cause the signed malware to remain valid after the revocation of the certificate. The study also demonstrated failures in disseminating the revocation decisions which leaves the users trusting abusive certifications.

Alrabaee et al. introduced in 2018 a system designed to identify free open-source software (FOSS) packages in binaries of which source code is not available [22]. This complicated task is important in malware detection, software infringement detection, digital forensics along with many other security applications. Previous systems relied on practical methods in data mining and database searching. The proposed system, named FOSSIL, incorporates three components; extracting syntactical features of functions by considering opcode frequencies and applying a hidden Markov model statistical test, applying a neighborhood hash graph kernel to random walks derived from control-flow graphs with the goal of extracting the semantics of the functions, and applying z-score to the normalized instructions to extract the behavior of instructions in a function. The components are integrated using a Bayesian network model, which synthesizes the results to determine the FOSS function. This combination gives it high resilience to code obfuscation. The proposed system was tested on three datasets including real-world projects employing FOSS packages, malware binaries utilizing FOSS, and a large repository of malware binaries. The results of testing have shown 0.95 mean precision with 0.85 mean recall. The study has also shown that modern day malware binaries contain 0.10 to 0.45 of FOSS packages.

Alhuzali et al. introduced, in 2018, a system for exploit generation for dynamic web applications [21]. Vulnerability analysis grow more and more difficult with the complexities of multi-tier web applications. The proposed approach is said to overcome the challenges of the dynamic nature of web applications. The proposed system combines dynamic analysis with static analysis to automatically identify vulnerabilities and build working exploits. The implementation

of the proposed approach, named NAVEX, can scale the process of automatic vulnerability analysis and exploits generation to large applications and multiple classes of vulnerabilities. Testing was done on a code base of 3.2 million lines of PHP code and resulted in generation of 204 exploits in the analyzed code. Out of the 204 exploits, 195 were based on injection, and 9 were based on logic vulnerabilities.
In 2019, Salva and Regainia introduced a security software design pattern classification system based on data integration that facilitates security pattern choice [101]. With up to 180 different design patterns available currently, the proper selection of a security design pattern to address a specific design problem can be challenging. The proposed classification exposes relationships among software attacks, security principles, and security patterns. The proposed system provides semi-automatic classification inferred by a data-store that integrates disparate publicly-available security data and generates Attack Defense Trees. These trees illustrates the attack's steps, techniques, sub-attacks, and relevant defense mechanisms. The data-store was created by combining five large databases that include relations among attacks, steps, countermeasures and principles, in addition to security patterns and strong points. Despite the limitation in the proposal's security patterns and classification, the proposed system shows potential of expanding beyond these limitations to become helpful in selecting proper security design patterns.

8.3 Summary

In the area of code security, the applications were in several directions, such as detection of anomalous programs, vulnerability detection in code clones, bug searching, and examination of code signing PKI. In addition, automated exploit generation was an area of interest.

9 Cloud Security

9.1 Definitions

In a simplified terms, cloud computing is the dynamic provision of computing services on-demand [17]. Cloud computing was built on the developments in many technologies such as virtualization, clustering, and grid-computing. It provides the capability to elastically expand (or shrink), vertically or horizontally to accommodate the system processing or storage needs.
With a growth from 30$ billions market in 2013, to over 136$ billion in 2020, cloud computing is becoming the de facto choice for most organizations. This also makes it a big target for malicious actors. The introduction of new systems and new architectures always brings new attack surfaces. Attacks on cloud-based systems doubled in 2019 in comparison to 2018 according to [2].

The same report mentions that the cloud accounted for 20% of investigated security incidents in 2019.
Many security threats in cloud computing are common with non-cloud based technologies such as threats to the network, storage, operating systems, and hardware. However, the unique nature of the cloud brings unique security challenges such as threats to the virtualization layer and the hypervisor, key management, and management interface.
Detailed information on cloud security threats can be found in [75,60].

9.2 Literature

Gai et al. introduced, in 2016, a novel incident analytics framework for cloud-based cybersecurity insurance system [49]. The paper presented a cybersecurity incident analytics framework that employs big data in cloud-based cybersecurity insurance systems. Cybersecurity insurance is gaining popularity as an alternative for financial firms or other high-risk industries to operate in a secure business environment with secured financial transactions. The study proposed a framework named Cost-Aware Hierarchical Cyber Incident Analytics (CA-HCIA) Framework. The suggested framework combines business and technical approaches to support making appropriate decisions related to information technology strategies through the creation of a classified hierarchy that helps in identifying risks. The simulations of the proposed framework has shown reduction in the cost of cybersecurity insurance without compromising the level of security.
In 2018, Nadgowda et al. introduced a different approach to cloud security, named DéjàVu [90]. The proposed framework explores the reuse of existing security solutions as black-box analytics in the cloud. This framework makes data accessible to traditional software by mimicking a system veneer over the data. The proposed system aims to achieve this through re-building a standard native POSIX system interface over data to enable classical (non-cloud) security solutions to run in a black-box fashion without the need for modification. For data collection purposes, the system used an open-sourced agentless-system-crawler to perform platform and system agnostic data collection. The proposed system was tested with state-of-the-art third-party security solutions and has shown reasonable overhead that led to near-real-time execution of security solutions over the DéjàVu platform.
Madi et al. introduced, in 2018 as well, a system for detecting virtual networks isolation breaches [83]. The multi-tenancy configuration of the cloud brings some security challenges in terms of the capability of isolation between the virtual networks that are dedicated for each client. The proposed automated tool, named ISOTOP, was designed to be an offline automated framework for auditing consistent isolation between virtual networks in OpenStack-managed cloud. The proposed tool spans over the infrastructure management, and the implementation layers. The testing results have shown that the proposed tool has achieved its aim successfully by detecting virtual network isolation viola-

tions with the ability to scale to large OpenStack-based data centers.
Majumdar et al. introduced, in 2018, a proactive cloud security auditing system based on learning probabilistic dependencies among events [85]. Security compliance auditing has particular important in the cloud for users to assure that the service provider is taking adequate measures to protect their data. The proposed system provides a proactive approach that prepares for auditing ahead of critical events which can reduce the the response time to a level that is acceptable practically. Such proactive approach can be hindered by manual identification of dependencies among the events. Hence, the proposed system suggests an automated event dependency detection. The system starts first with a log processing technique to prepare raw cloud logs for various analysis purposes, and then designs a learning-based proactive security auditing system named LeaPS+. The proposed system was integrated to OpenStack and tested in an extensive cloud testing environment of about 100,000 virtual machines where it demonstrated practical response time of 6 ms to audit the whole cloud. This shows an improvement of 50% over other existing proactive approaches.

9.3 Summary

Applications explored in this section included the use of big data in building a cloud-based cyebersecurity insurance system, detection of virtual network isolation breaches, cloud security auditing system, and the exploration of the re-use of current non-cloud security analytics systems in the cloud.

10 Other Applications in Cybersecurity

Abraham and Nair proposed in 2015 a predictive cybersecurity analytics framework that is based on non-homogenous markov model [15]. The proposed system was designed to measure the predictive security risk of an enterprise, taking into account the dynamic attributes associated with vulnerabilities that can change over time. The presented system presents a novel attack graph analysis that considers temporal aspects associated with vulnerabilities such as availability of exploits and patches. The proposed system is said to provide a better view of the state of security of an enterprise network through developing a more realistic non-homogenous model that incorporates a time dependent covariate, namely the vulnerability age.
Englehardt and Narayanan published in 2016 a thorough study that analyzed the user tracking habits of 1 million websites on the Internet [44]. The measurement used in the study was done using an open-sourse web privacy measurement tool named OpenWPM. This tool was built on top of Firefox, with automation provided by Selenium [7]. The tool was designed to support par-

allelism, automatic recovery from failure, and comprehensive browser instrumentation. The study collected 15 different types of measurements including cookie-based and fingerprinting-based tracking detection, along with detection of cookie-syncing between different websites. The study have shown that 91.7% of websites used stateless tracking, 9.4% used stateful tracking, while 5% used Ghostery. The tool created by the authors of the study was used to conduct 25 other studies.

Pearce et al. introduced, in 2017, a system to detect Internet connectivity disruptions around the world, named Augur [93]. The proposed system is based on continuously monitoring information about Internet reachability that can show the onset or end of censorship across regions and ISPs. Augur utilizes TCP/IP side channels to measure reachability between two Internet location without directly controlling a measurement vantage point at either location. The dependence of the system on side channels ensures non-implication of the users involved in the communication. The proposed system is said to be scalable and statistically robust method to infer network-layer filtering to perform continuous monitoring of global censorship. Augur was tested by measuring Internet-wide disruptions in 180 countries over the period of 17 days against sites known to be frequently blocked. Testing data have shown that the top country in Internet connectivity disruptions is China with a blocking percentage of 5%.

In 2018, Farris et al. introduced a vulnerability management system, named VULCON [46]. The proposed system is a vulnerability management strategy that is based on two performance metrics; time-to-vulnerability remediation, and total vulnerability exposure. The input to the system consists of vulnerability scan reports, vulnerability metadata, asset criticality, and personnel resources. The vulnerability prioritization relies on the use of mixed-integer multi-objective optimization algorithm. This prioritization is important for patching purposes with proper focus on the performance metrics mentioned earlier. Testing results of the VULCON has shown a 8.97% reduction in the total vulnerability exposure. The proposed system has also shown its capability to determine the monthly resources required to maintain a target total vulnerability exposure score.

Reaves et al. introduced, in 2018, a detailed study characterizing the security of Short Message Service (SMS) ecosystem with public gateways [95]. The study did thorough analysis of about 900,000 text messages sent to public online SMS gateways over the course of 28 months. The study has shown the geographical distribution of spam messages, and the use of SMS to transfer malicious content. The study has also shown that many services sent security-sensitive messages through unencrypted medium, low-entropy solutions were used for One-Time Passwords (OTP)s, and behaviors indicating that public gateways are primarily used for evading account creation policies that require verified phone numbers. The latter finding is considered a serious security threat that can be exploited to combat phone-verification.

Guo et al. proposed, in 2018 as well, a high-fidelity explanation method dedicated for security applications named LEMNA[55]. LEMNA was designed to

generate an interpretable model that explains how an input sample is classified. The proposed system was tested on two popular deep learning applications; malware classifier, and function start detector for binary reverse engineering. Testing results have shown that LEMNA provides a much higher fidelity level in its explanations compared to other existing methods. The study also included practical use cases to help machine learning developers to validate their model's behavior and troubleshoot classification errors.

Gong and Liu introduced in 2018 a novel attack based on large datasets collected from social networks [52]. The attack was designed to leverage innocent user information that is publicly available on social networks to infer missing attributes of the targets. The attack uses publicly available data of social friends and similar user behavioral records such as liked web pages, applications reviewed on Google Play. By combining both sources of data (social friends and behavioral data), inference of private attributes becomes more accurate. The attack was tested on a real world dataset of 1.1 million users. The results shown 57% success rate in inferring the city the user lived in, and the percentage goes up to 90% if the attacker selectively targets users via confidence estimation. Such attack, shows the poor privacy practices of social networks users and calls for increasing levels of awareness.

Barradas et al. introduced, in 2018, a study that analyzes the unobservability properties of three state of the art systems that are used for multimedia protocol tunneling [28]. Multimedia protocol tunneling is the process of creating covert channels by modulating data into the input of widely used multimedia communications applications, such as Skype, to resist censorship. The experimental study conducted on CoverCat, DeltaShaper, and Facet employed machine learning techniques found that employing decision trees can uncover the vast majority of these channels with comparatively lower false positive rates. In the training, a total of 166 features were used in the training process of the classifiers. When building the dataset for Facet, the dataset was built from 1,000 YouTube videos to be used for the covert set, and 1,000 recorded live chat videos on YouTube as the legitimate data set. For CoverCast, the legitimate live-streaming dataset was built from 200 live-streams included in the YouTube-curated list, while the covert set was built from 200 CoverCast live streams from the news websites already available in the CovertCast prototype. The last dataset for DeltaShaper was emulated 300 legitimate bi-directional Skype calls. The study concluded that the existence of manually labeled samples is necessary to detect the covert channels.

Shu et al. introduced, in 2018 also, novel methodology to model threat discovery as a graph computation problem [103]. The proposed methodology enables efficient programming for solving threat discovery problems and equip threat hunters with highly capable new tools for agile codification and threat hypothesis. The proposed system was tested on around a billion records during DARPA's two-week competition. During the testing, the proposed methodology was capable of analyzing and dynamically planning and programming dozens of threat hunting tasks. The proposed methodology, named Threat Intelligence Computing, exhibited strong detection and analytics capabilities

along with high efficiency.

11 Discussions

The rapid growth in the adoption of technology in various aspects of life has significantly intensified the challenges in protecting these systems. With the Internet traffic projected to grow to 161.3 Exabytes per month in 2020 [8], and IoT devices projected growth to 31 billion [81], the attack surface is growing rapidly, and probably out of control. Legacy threat detection technologies are not catching-up with the dynamic nature of threats that are being exploited every day. This is particularly important due to the fact that over half of the 903 million malware instances detected in 2019 were categorized as zero-day threats [9].

Leveraging the benefits of big data in building robust, adaptive, and fast cybersecurity systems is becoming more of a necessity rather than a choice. With the large flows of data, classical detection methods fall behind in terms of accuracy and capacity to detect threats. Most of the current cybersecurity systems rely heavily on logs generated by networking devices, hosts, IDS and IPS (both network- and host-based), etc. These logs can easily be several GBs per day for medium sized networks according to [58]. Without leveraging big data and big data analytics, machine learning, and cloud computing, threat detection systems can easily and quickly fall behind. Hence, the future of cybersecurity is highly connected to big data.

As shown in figure 4, intrusion and anomaly detection takes the lead in terms of number of papers included in the review. This is due to the fact that this has been one of the oldest cybersecurity areas that involve processing of large volumes of data.

The utilization of big data in cybersecurity is, to a great extent, governed by

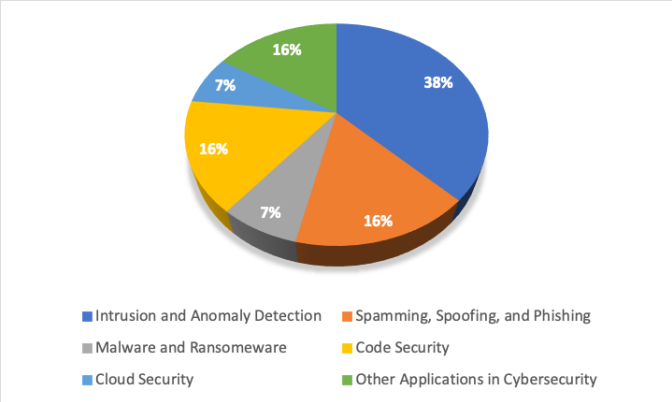

Fig. 4 Percentage of Papers by Category

our capability to maintain the integrity of the big data used in making decisions. One future direction can focus on the protection of machine learning techniques used in cybersecurity applications from being manipulated by malicious actors. Suciu et al. published a paper in 2018 discussing how machine learning techniques could fail [111]. The research presents an adversary model, named FAIL, which focuses on detecting evasions and poisoning attacks.
A technique such as 'Machine Unlearning' introduced by Cao and Young in [32] can be of vital importance for future applications of big data. The main focus of this research direction to make machine-learning based system forget. This can be important in removing unwanted learning data maliciously added to the training dataset.
Several other studies were directed towards protecting big data from manipulation such as [16] which proposed a detection mechanisms for insider attacks on big data systems. Other notable references in big data security can be found in [115,116,128,45].
Recent patents, such as [132,24,25] among others, show a clear interest in cybersecurity services based on big data analytics by large organizations like IBM, Google, and Cisco.
Johnston and Peacock published a paper in 2020 identifying seven pitfalls of using data science in cybersecurity [66]. The seven pitfalls identified in this paper were data source, feature engineering, evaluation metrics, algorithms selection, algorithm convergence, algorithm poisoning, and adversarial machine-learning. This research calls for caution and attention when using machine learning and big data in cybersecurity applications. The paper shows the downfalls of using different type of data sources, along with the impact of algorithm selection on the overall outcome.

12 Conclusions and Future Directions

In this paper we've surveyed a large number of papers suggesting various applications of big data in cybersecurity. These papers tackled different directions in research and were categorized into the following areas:

- Intrusion and Anomaly Detection
- Malware and Ransomware
- Cloud Security
- Code Security
- Spamming, Spoofing, and Phishing
- Other Applications in Cybersecurity

The survey shows clearly that big data analytics represent an integral part of the future of cybersecurity. Future cybersecurity systems will be heavily dependent on processing large amounts of data, whether for detection of malicious activity or intent, or in prevention of malicious activities.
Based on our survey, we identify the following future direction of research leveraging big data in cybersecurity applications:

- Intrusion Detection.
  Although this has been one of the earliest directions explored and mostly published in as seen in this survey, intrusion and anomaly detection remains a very promising area or research. With new attacks being developed in volumes, big data analytics can help in providing the grounds for zero-day attacks detection and deterrence if combined with cloud computing, machine learning, and other inference techniques.
- Code Security The availability of huge amounts of 'good' code in comparison to 'bad' code, enables researchers to train machine-learning based systems to detect and identify vulnerabilities. Comparatively, vulnerability detection in code is easier to achieve than network vulnerabilities' detection. This is due to the fact that features extracted from code are more comprehensive and thorough while feature selection process can highly impact the outcome in IDSs.

- Social Media.
  Many research papers published earlier, such as [53,65,68,52], have shown that social media users tend to 'over-share' data that can be used to violate their privacy. Studies like the ones published in [91,129,63] can be considered a good seed in detecting misuse of social media, or detecting attacks that employ social media as a tool. In addition, studies that give us better understanding of behavior of humans can help in improving security as well, such as [33,99].
- Fraud Detection.
  Although credit card fraud detection is not a new area at all. However, with the collaboration of big data analytics, machine-learning, and other modern techniques can lead to higher accuracy and lower false positives. With the advancements in big data analytics, the features captured from each transaction can be extended to provide higher accuracy.
- Lightweight Cybersecurity.
  In a future where smart cities and IoT are dominant, cybersecurity applications will need to be of lighter weight. Applications that require heavy processing and memory requirements will not be applicable to IoT-based systems. Hence, smart cities should be able to rely on big data analytics, and centralized security controls to manage rapid incident detection and prevention.
- Malware Detection.
  With over 2.3mil new types of malware detected in the first half of 2018 alone [41], the need for malware detection systems are now higher than ever. With such rapid growth in malware, big data analytics can play an important role in detecting and deterring the impact of malware.

Other future directions can include spam email detection, software and data copyright preservation, and privacy preservation.

References

1. 2019 cyber security statistics trends & data: The ultimate list of cyber security stats — purplesec. https://purplesec.us/resources/cyber-security-statistics/. (Accessed on 07/30/2020)
2. 2020 trustwave global security report — trustwave. https://www.trustwave.com/en-us/resources/library/documents/2020-trustwave-global-security-report/. (Accessed on 08/01/2020)
3. 5 cybersecurity threats to be aware of in 2020 — ieee computer society. https://www.computer.org/publications/tech-news/trends/5-cybersecurity-threats-to-be-aware-of-in-2020/. (Accessed on 07/30/2020)
4. Apple reveals windows 10 is four times more popular than the mac. https://www.theverge.com/2017/4/4/15176766/apple-microsoft-windows-10-vs-mac-users-figures-stats. Accessed: 2018-12-03
5. Computer science. https://arxiv.org/archive/cs. (Accessed on 07/30/2020)
6. Cyberthreat trends: 15 cybersecurity threats for 2020 — nortonlifelock. https://us.norton.com/internetsecurity-emerging-threats-cyberthreat-trends-cybersecurity-threat-review.html. (Accessed on 07/30/2020)
7. Github - mozilla/openwpm: A web privacy measurement framework. https://github.com/mozilla/OpenWPM. (Accessed on 03/23/2019)
8. Global_2020_forecast_highlights. https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_2020_Forecast_Highlights.pdf. (Accessed on 07/30/2020)
9. Half of the malware detected in 2019 was classified as zero-day threats, making it the most common malware to date - cynet. https://www.cynet.com/blog/half-of-the-malware-detected-in-2019-was-classified-as-zero-day-threats-making-it-the-most-common-malware-to-date/. (Accessed on 07/30/2020)
10. Microsoft vulnerabilities more than doubled in 2017. https://www.securitynow.com/author.asp?section_id=649&doc_id=740671. Accessed: 2018-12-03
11. Top cybersecurity threats in 2020. https://onlinedegrees.sandiego.edu/top-cyber-security-threats/. (Accessed on 07/30/2020)
12. Ransomware cyber attacks: Which industries are being hit the hardest? https://www.bitsighttech.com/blog/ransomware-cyber-attacks (2017). (Accessed on 12/08/2018)
13. Us hospital pays $55,000 to hackers after ransomware attack — zdnet. https://www.zdnet.com/article/us-hospital-pays-55000-to-ransomware-operators/ (2018). (Accessed on 12/08/2018)
14. Abdlhamed, M., Kifayat, K., Shi, Q., Hurst, W.: Intrusion prediction systems. In: Information Fusion for Cyber-Security Analytics, pp. 155–174. Springer (2017)
15. Abraham, S., Nair, S.: Predictive cyber-security analytics framework: a non-homogenous markov model for security quantification. arXiv preprint arXiv:1501.01901 (2015)
16. Aditham, S., Ranganathan, N.: A system architecture for the detection of insider attacks in big data systems. IEEE Transactions on Dependable and Secure Computing **15**(6), 974–987 (2018)
17. Alani, M.M.: What is the cloud? In: Elements of Cloud Computing Security, pp. 1–14. Springer (2016)
18. AlEroud, A., Karabatis, G.: Using contextual information to identify cyber-attacks. In: Information Fusion for Cyber-Security Analytics, pp. 1–16. Springer (2017)
19. Aleroud, A., Zhou, L.: Phishing environments, techniques, and countermeasures: A survey. Computers & Security **68**, 160–196 (2017)
20. Alguliyev, R., Imamverdiyev, Y.: Big data: big promises for information security. In: Application of Information and Communication Technologies (AICT), 2014 IEEE 8th International Conference on, pp. 1–4. IEEE (2014)
21. Alhuzali, A., Gjomemo, R., Eshete, B., Venkatakrishnan, V.: {NAVEX}: Precise and scalable exploit generation for dynamic web applications. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 377–392 (2018)

22. Alrabaee, S., Shirani, P., Wang, L., Debbabi, M.: Fossil: a resilient and efficient system for identifying foss functions in malware binaries. ACM Transactions on Privacy and Security (TOPS) **21**(2), 8 (2018)
23. Alsadhan, A.A., Hussain, A., Alani, M.M.: Detecting ndp distributed denial of service attacks using machine learning algorithm based on flow-based representation. In: 2018 11th International Conference on Developments in eSystems Engineering (DeSE), pp. 134–140. IEEE (2018)
24. Amini, L., Christodorescu, M., Cohen, M.A., Parthasarathy, S., Rao, J., Sailer, R., Schales, D.L., Venema, W.Z., Verscheure, O.: Adaptive cyber-security analytics (2015). US Patent 9,032,521
25. Baikalov, I.A., Froelich, C., McConnell, T., McGloughlin, J.P., et al.: Cyber security analytics architecture (2016). US Patent 9,516,041
26. Balaban, D.: 11 types of spoofing attacks every security professional should know about — 2020-03-24 — security magazine. `https://www.securitymagazine.com/articles/91980-types-of-spoofing-attacks-every-security-professional-should-know-about` (2020). (Accessed on 08/01/2020)
27. Banescu, S., Collberg, C., Pretschner, A.: Predicting the resilience of obfuscated code against symbolic execution attacks via machine learning. In: 26th {USENIX} Security Symposium ({USENIX} Security 17), pp. 661–678 (2017)
28. Barradas, D., Santos, N., Rodrigues, L.: Effective detection of multimedia protocol tunneling using machine learning. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 169–185 (2018)
29. Biham, E., Shamir, A.: Differential cryptanalysis of des-like cryptosystems. Journal of CRYPTOLOGY **4**(1), 3–72 (1991)
30. Bilge, L., Han, Y., Dell'Amico, M.: Riskteller: Predicting the risk of cyber incidents. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1299–1311. ACM (2017)
31. Cao, P., Badger, E.C., Kalbarczyk, Z.T., Iyer, R.K., Withers, A., Slagell, A.J.: Towards an unified security testbed and security analytics framework. In: Proceedings of the 2015 Symposium and Bootcamp on the Science of Security, p. 24. ACM (2015)
32. Cao, Y., Yang, J.: Towards making systems forget with machine unlearning. In: 2015 IEEE Symposium on Security and Privacy, pp. 463–480. IEEE (2015)
33. Chakraborty, R., Vishik, C., Rao, H.R.: Privacy preserving actions of older adults on social media: Exploring the behavior of opting out of information sharing. Decision Support Systems **55**(4), 948–956 (2013)
34. Chiew, K.L., Yong, K.S.C., Tan, C.L.: A survey of phishing attacks: Their types, vectors and technical approaches. Expert Systems with Applications **106**, 1–20 (2018)
35. Cinque, M., Della Corte, R., Pecchia, A.: Microservices monitoring with event logs and black box execution tracing. IEEE Transactions on Services Computing (2019)
36. Cinque, M., Della Corte, R., Pecchia, A.: Contextual filtering and prioritization of computer application logs for security situational awareness. Future Generation Computer Systems **111**, 668–680 (2020)
37. Clement, J.: Global digital population as of april 2020. `https://www.statista.com/statistics/617136/digital-population-worldwide/`. Accessed: 2020-05-13
38. Crane, C.: 20 ransomware statistics you're powerless to resist reading - hashed out by the ssl store™. `https://www.thesslstore.com/blog/ransomware-statistics/`. (Accessed on 08/01/2020)
39. Curtin, M., Dolske, J.: A brute force search of des keyspace. In: 8th Usenix Symposium, January, pp. 26–29. Citeseer (1998)
40. Cuzzocrea, A., Martinelli, F., Mercaldo, F., Grasso, G.M.: Experimenting and assessing machine learning tools for detecting and analyzing malicious behaviors in complex environments. Journal of Reliable Intelligent Environments **4**(4), 225–245 (2018)
41. DATA, G.: Malware in 2018: The danger is on the web — g data blog. `https://www.gdatasoftware.com/blog/2018/09/31037-malware-figures-first-half-2018-danger-web` (2018). (Accessed on 03/31/2019)
42. Dias, L.F., Correia, M.: Big data analytics for intrusion detection: an overview. In: Handbook of Research on Machine and Deep Learning Applications for Cyber Security, pp. 292–316. IGI Global (2020)

43. Du, M., Li, F., Zheng, G., Srikumar, V.: Deeplog: Anomaly detection and diagnosis from system logs through deep learning. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1285–1298. ACM (2017)
44. Englehardt, S., Narayanan, A.: Online tracking: A 1-million-site measurement and analysis. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 1388–1401. ACM (2016)
45. Fang, W., Wen, X.Z., Zheng, Y., Zhou, M.: A survey of big data security and privacy preserving. IETE Technical Review **34**(5), 544–560 (2017)
46. Farris, K.A., Shah, A., Cybenko, G., Ganesan, R., Jajodia, S.: Vulcon: A system for vulnerability prioritization, mitigation, and management. ACM Trans. Priv. Secur. **21**(4), 16:1–16:28 (2018). DOI 10.1145/3196884
47. Feng, Q., Zhou, R., Xu, C., Cheng, Y., Testa, B., Yin, H.: Scalable graph-based bug search for firmware images. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 480–491. ACM, New York, NY, USA (2016). DOI 10.1145/2976749.2978370
48. Funk, C., Garnaeva, M.: Kaspersky security bulletin 2013. overall statistics for 2013. Kaspersky Lab **10** (2013)
49. Gai, K., Qiu, M., Elnagdy, S.A.: A novel secure big data cyber incident analytics framework for cloud-based cybersecurity insurance. In: 2016 IEEE 2nd International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing (HPSC), and IEEE International Conference on Intelligent Data and Security (IDS), pp. 171–176. IEEE (2016)
50. Gandomi, A., Haider, M.: Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management **35**(2), 137–144 (2015)
51. García, S., Ramírez-Gallego, S., Luengo, J., Benítez, J.M., Herrera, F.: Big data preprocessing: methods and prospects. Big Data Analytics **1**(1), 9 (2016)
52. Gong, N.Z., Liu, B.: Attribute inference attacks in online social networks. ACM Transactions on Privacy and Security (TOPS) **21**(1), 3 (2018)
53. Gou, L., Zhou, M.X., Yang, H.: Knowme and shareme: understanding automatically discovered personality traits from social media and user sharing preferences. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 955–964. ACM (2014)
54. Grahn, K., Westerlund, M., Pulkkis, G.: Analytics for network security: A survey and taxonomy. In: Information fusion for cyber-security analytics, pp. 175–193. Springer (2017)
55. Guo, W., Mu, D., Xu, J., Su, P., Wang, G., Xing, X.: Lemna: Explaining deep learning based security applications. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, CCS '18, pp. 364–379. ACM, New York, NY, USA (2018). DOI 10.1145/3243734.3243792
56. Gutierrez, C.N., Kim, T., Della Corte, R., Avery, J., Goldwasser, D., Cinque, M., Bagchi, S.: Learning from the ones that got away: Detecting new forms of phishing attacks. IEEE Transactions on Dependable and Secure Computing **15**(6), 988–1001 (2018)
57. Gyöngyi, Z., Garcia-Molina, H., Pedersen, J.: Combating web spam with trustrank. In: Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, pp. 576–587. VLDB Endowment (2004)
58. Hale, B.: Estimating log generation for security information event and log management. Retrieved September **15** (2016)
59. He, P., Zhu, J., He, S., Li, J., Lyu, M.R.: Towards automated log parsing for large-scale log data analysis. IEEE Transactions on Dependable and Secure Computing **15**(6), 931–944 (2018)
60. Hong, J.B., Nhlabatsi, A., Kim, D.S., Hussein, A., Fetais, N., Khan, K.M.: Systematic identification of threats in the cloud: A survey. Computer Networks **150**, 46–69 (2019)
61. Hossain, M.N., Wang, J., Weisse, O., Sekar, R., Genkin, D., He, B., Stoller, S.D., Fang, G., Piessens, F., Downing, E., et al.: Dependence-preserving data compaction for scalable forensic analysis. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 1723–1740 (2018)

62. Huang, D.Y., Aliapoulios, M.M., Li, V.G., Invernizzi, L., Bursztein, E., McRoberts, K., Levin, J., Levchenko, K., Snoeren, A.C., McCoy, D.: Tracking ransomware end-to-end. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 618–631. IEEE (2018)
63. Ikram, M., Onwuzurike, L., Farooqi, S., Cristofaro, E.D., Friedman, A., Jourjon, G., Kaafar, M.A., Shafiq, M.Z.: Measuring, characterizing, and detecting facebook like farms. ACM Transactions on Privacy and Security (TOPS) **20**(4), 13 (2017)
64. Jansen, K., Schäfer, M., Moser, D., Lenders, V., Pöpper, C., Schmitt, J.: Crowd-gps-sec: Leveraging crowdsourcing to detect and localize gps spoofing attacks. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 1018–1031. IEEE (2018)
65. John, N.A.: The social logics of sharing. The Communication Review **16**(3), 113–131 (2013)
66. Johnstone, M., Peacock, M.: Seven pitfalls of using data science in cybersecurity. In: Data Science in Cybersecurity and Cyberthreat Intelligence, pp. 115–129. Springer (2020)
67. Jovanovic, B.: Malware statistics – you'd better get your computer vaccinated. `https://dataprot.net/statistics/malware-statistics/`. Accessed: 2020-05-29
68. Jurgens, D.: That's what friends are for: Inferring location in online social media platforms based on social relationships. In: Seventh International AAAI Conference on Weblogs and Social Media (2013)
69. Kelsey, J., Schneier, B., Wagner, D.: Key-schedule cryptanalysis of idea, g-des, gost, safer, and triple-des. In: Annual International Cryptology Conference, pp. 237–251. Springer (1996)
70. Khan, M.U.K., Park, H.S., Kyung, C.M.: Rejecting motion outliers for efficient crowd anomaly detection. IEEE Transactions on Information Forensics and Security **14**(2), 541–556 (2019)
71. Kim, D., Kwon, B.J., Kozák, K., Gates, C., Dumitraș, T.: The broken shield: Measuring revocation effectiveness in the windows code-signing {PKI}. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 851–868 (2018)
72. Kim, S., Woo, S., Lee, H., Oh, H.: Vuddy: A scalable approach for vulnerable code clone discovery. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 595–614. IEEE (2017)
73. Koli, J.: Randroid: Android malware detection using random machine learning classifiers. In: 2018 Technologies for Smart-City Energy Security and Power (ICSESP), pp. 1–6. IEEE (2018)
74. Kotenko, I., Saenko, I., Branitskiy, A.: Machine learning and big data processing for cybersecurity data analysis. In: Data Science in Cybersecurity and Cyberthreat Intelligence, pp. 61–85. Springer (2020)
75. Kumar, R., Goyal, R.: On cloud security requirements, threats, vulnerabilities and countermeasures: A survey. Computer Science Review **33**, 1–48 (2019)
76. Kwon, B.J., Mondal, J., Jang, J., Bilge, L., Dumitraş, T.: The dropper effect: Insights into malware distribution with downloader graph analytics. In: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, pp. 1118–1129. ACM (2015)
77. Laney, D.: 3d data management: Controlling data volume, velocity and variety. META group research note **6**(70), 1 (2001)
78. Li, H., Xu, X., Liu, C., Ren, T., Wu, K., Cao, X., Zhang, W., Yu, Y., Song, D.: A machine learning approach to prevent malicious calls over telephony networks. In: 2018 IEEE Symposium on Security and Privacy (SP), pp. 53–69. IEEE (2018)
79. Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., Beyah, R.: Acing the ioc game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 755–766. ACM (2016)
80. Liao, X., Yuan, K., Wang, X., Pei, Z., Yang, H., Chen, J., Duan, H., Du, K., Alowaisheq, E., Alrwais, S., et al.: Seeking nonsense, looking for trouble: Efficient promotional-infection detection through semantic inconsistency search. In: 2016 IEEE Symposium on Security and Privacy (SP), pp. 707–723. IEEE (2016)
81. Maayan, G.D.: The iot rundown for 2020: Stats, risks, and solutions – security today. `https://securitytoday.com/Articles/2020/01/13/The-IoT-Rundown-for-2020.aspx`. (Accessed on 07/30/2020)

82. MacDonald, N.: Information security is becoming a big data analytics problem. https://www.gartner.com/en/documents/1960615 (2012). (Accessed on 05/13/2020)
83. Madi, T., Jarraya, Y., Alimohammadifar, A., Majumdar, S., Wang, Y., Pourzandi, M., Wang, L., Debbabi, M.: Isotop: Auditing virtual networks isolation across cloud layers in openstack. ACM Transactions on Privacy and Security (TOPS) **22**(1), 1 (2018)
84. Mahmood, T., Afzal, U.: Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools. In: Information assurance (ncia), 2013 2nd national conference on, pp. 129–134. IEEE (2013)
85. Majumdar, S., Tabiban, A., Jarraya, Y., Oqaily, M., Alimohammadifar, A., Pourzandi, M., Wang, L., Debbabi, M.: Learning probabilistic dependencies among events for proactive security auditing in clouds. Journal of Computer Security (Preprint), 1–38 (2018)
86. Maltby, D.: Big data analytics. In: 74th Annual Meeting of the Association for Information Science and Technology (ASIST), pp. 1–6. New Orleans, LA, USA (2011)
87. Marr, B.: How much data do we create every day? https://www.forbes.com/sites/bernardmarr/2018/05/21/how-much-data-do-we-create-every-day-the-mind-blowing-stats-everyone-should-read/. Accessed: 2018-10-22
88. Martha, V.: Big data processing algorithms. In: Big Data, pp. 61–91. Springer, London, UK. (2015)
89. Matsui, M.: Linear cryptanalysis method for des cipher. In: Workshop on the Theory and Application of of Cryptographic Techniques, pp. 386–397. Springer (1993)
90. Nadgowda, S., Isci, C., Bal, M.: Déjàvu: Bringing black-box security analytics to cloud. In: Proceedings of the 19th International Middleware Conference Industry, pp. 17–24. ACM (2018)
91. Nilizadeh, S., Labrèche, F., Sedighian, A., Zand, A., Fernandez, J., Kruegel, C., Stringhini, G., Vigna, G.: Poised: Spotting twitter spam off the beaten paths. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1159–1174. ACM (2017)
92. Oltisk, J.: The big data security analytics era is here. Tech. rep., Enterprise Strategy Group (2013)
93. Pearce, P., Ensafi, R., Li, F., Feamster, N., Paxson, V.: Augur: Internet-wide detection of connectivity disruptions. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 427–443. IEEE (2017)
94. Pierazzi, F., Casolari, S., Colajanni, M., Marchetti, M.: Exploratory security analytics for anomaly detection. computers & security **56**, 28–49 (2016)
95. Reaves, B., Vargas, L., Scaife, N., Tian, D., Blue, L., Traynor, P., Butler, K.R.: Characterizing the security of the sms ecosystem with public gateways. ACM Transactions on Privacy and Security (TOPS) **22**(1), 2 (2018)
96. Richardson, R., North, M.M.: Ransomware: Evolution, mitigation and prevention. International Management Review **13**(1), 10 (2017)
97. Rieck, K., Holz, T., Willems, C., Düssel, P., Laskov, P.: Learning and classification of malware behavior. In: International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 108–125. Springer (2008)
98. Rijmen, V., Daemen, J.: Advanced encryption standard. Proceedings of Federal Information Processing Standards Publications, National Institute of Standards and Technology pp. 19–22 (2001)
99. Rose, C.: The security implications of ubiquitous social media (2011)
100. Rouse, M.: What is big data analytics. https://searchbusinessanalytics.techtarget.com/definition/big-data-analytics. Accessed: 2018-11-24
101. Salva, S., Regainia, L.: A catalogue associating security patterns and attack steps to design secure applications. Journal of Computer Security (Preprint), 1–26 (2019)
102. Shen, Y., Mariconti, E., Vervier, P.A., Stringhini, G.: Tiresias: Predicting security events through deep learning. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 592–605. ACM (2018)
103. Shu, X., Araujo, F., Schales, D.L., Stoecklin, M.P., Jang, J., Huang, H., Rao, J.R.: Threat intelligence computing. In: Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, pp. 1883–1898. ACM (2018)

104. Shu, X., Yao, D.D., Ramakrishnan, N., Jaeger, T.: Long-span program behavior modeling and attack detection. ACM Transactions on Privacy and Security (TOPS) **20**(4), 12 (2017)
105. Siadati, H., Memon, N.: Detecting structurally anomalous logins within enterprise networks. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1273–1284. ACM (2017)
106. Singer, P.W., Friedman, A.: Cybersecurity: What everyone needs to know. Oxford University Press (2014)
107. Sipola, T.: Knowledge discovery from network logs. In: Cyber Security: Analytics, Technology and Automation, pp. 195–203. Springer (2015)
108. Siwicki, B.: Ransomware attackers collect ransom from kansas hospital, don't unlock all the data, then demand more money. Healthcare IT News (2016)
109. staff, G., AP: Twitter hack: Us and uk teens arrested over breach of celebrity accounts — twitter — the guardian. `https://www.theguardian.com/technology/2020/jul/31/twitter-hack-arrests-florida-uk-teenagers`. (Accessed on 08/01/2020)
110. Standard, D.E., et al.: Federal information processing standards publication 46. National Bureau of Standards, US Department of Commerce **23** (1977)
111. Suciu, O., Marginean, R., Kaya, Y., Daume III, H., Dumitras, T.: When does machine learning {FAIL}? generalized transferability for evasion and poisoning attacks. In: 27th {USENIX} Security Symposium ({USENIX} Security 18), pp. 1299–1316 (2018)
112. Sun, B., Takahashi, T., Zhu, L., Mori, T.: Discovering malicious urls using machine learning techniques. In: Data Science in Cybersecurity and Cyberthreat Intelligence, pp. 33–60. Springer (2020)
113. Talabis, M., McPherson, R., Miyamoto, I., Martin, J.: Information Security Analytics: Finding Security Insights, Patterns, and Anomalies in Big Data. Syngress (2014)
114. Tan, Z., Nagar, U.T., He, X., Nanda, P., Liu, R.P., Wang, S., Hu, J.: Enhancing big data security with collaborative intrusion detection. IEEE cloud computing **1**(3), 27–33 (2014)
115. Tankard, C.: Big data security. Network security **2012**(7), 5–8 (2012)
116. Terzi, D.S., Terzi, R., Sagiroglu, S.: A survey on security and privacy issues in big data. In: 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), pp. 202–207. IEEE (2015)
117. Thirumaran, J., et al.: Applications of big data analytics-network security. International Journal for Research in Science Engineering & Technology **5**(1), 55–59 (2018)
118. Tipton, H.: Information Security Management Handbook: Volume IV. CRC Press (2019)
119. Ugarte-Pedrero, X., Graziano, M., Balzarotti, D.: A close look at a daily dataset of malware samples. ACM Transactions on Privacy and Security (TOPS) **22**(1), 6 (2019)
120. Ullah, F., Babar, M.A.: Architectural tactics for big data cybersecurity analytics systems: A review. Journal of Systems and Software **151**, 81–118 (2019)
121. Von Solms, R., Van Niekerk, J.: From information security to cyber security. computers & security **38**, 97–102 (2013)
122. Wohlin, C.: Guidelines for snowballing in systematic literature studies and a replication in software engineering. In: Proceedings of the 18th international conference on evaluation and assessment in software engineering, pp. 1–10 (2014)
123. Xu, Z., Wu, Z., Li, Z., Jee, K., Rhee, J., Xiao, X., Xu, F., Wang, H., Jiang, G.: High fidelity data reduction for big data security dependency analyses. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 504–516. ACM (2016)
124. Yang, X., Ma, T., Shi, Y.: Typical dos/ddos threats under ipv6. In: 2007 International Multi-Conference on Computing in the Global Information Technology (ICCGI'07), pp. 55–55. IEEE (2007)
125. Yao, Y., Viswanath, B., Cryan, J., Zheng, H., Zhao, B.Y.: Automated crowdturfing attacks and defenses in online review systems. In: Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1143–1158. ACM (2017)
126. You, I., Yim, K.: Malware obfuscation techniques: A brief survey. In: 2010 International conference on broadband, wireless computing, communication and applications, pp. 297–300. IEEE (2010)

127. Yuan, Y., Adhatarao, S.S., Lin, M., Yuan, Y., Liu, Z., Fu, X.: Ada: Adaptive deep log anomaly detector. In: IEEE INFOCOM 2020-IEEE Conference on Computer Communications, pp. 2449–2458. IEEE (2020)

128. Zhang, D.: Big data security and privacy protection. In: 8th International Conference on Management and Computer Science (ICMCS 2018). Atlantis Press (2018)

129. Zhang, J., Zhang, R., Zhang, Y., Yan, G.: The rise of social botnets: Attacks and countermeasures. IEEE Transactions on Dependable and Secure Computing (2016)

130. Zhao, J.Y., Kessler, E.G., Yu, J., Jalal, K., Cooper, C.A., Brewer, J.J., Schwaitzberg, S.D., Guo, W.A.: Impact of trauma hospital ransomware attack on surgical residency training. Journal of Surgical Research **232**, 389–397 (2018)

131. Zhu, Z., Dumitraş, T.: Featuresmith: Automatically engineering features for malware detection by mining the security literature. In: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, pp. 767–778. ACM (2016)

132. Zoldi, S., Athwal, J., Li, H., Kennel, M., Xue, X.: Cyber security adaptive analytics threat monitoring system and method (2015). US Patent 9,191,403

133. Zuech, R., Khoshgoftaar, T.M., Wald, R.: Intrusion detection and big heterogeneous data: a survey. Journal of Big Data **2**(1), 3 (2015)

134. Zuo, Y., Wu, Y., Min, G., Huang, C., Pei, K.: An intelligent anomaly detection scheme for micro-services architectures with temporal and spatial data analysis. IEEE Transactions on Cognitive Communications and Networking (2020)

Appendix A Literature Summary Tables

Table 1: Summary of papers in intrusion and anomaly detection section

No	Ref#	Year	Aspect of Big Data	Summary
1	[114]	2014	MapReduce summarization of large volumes of traffic data.	Proposed a collaborative IDS that exchanges large volumes of network traffic information to improve the IDS accuracy. The paper does not show implementation results.
2	[31]	2014	Processing large volumes of network packet captures.	Proposed a unified end-to-end security testbed and security analytics framework built on capturing large volumes of network traffic, and use kernel probes and monitor system behavior. No implementation was presented. Hence, effectiveness not measured.
3	[133]	2015	Big Data processing in SIEM.	A survey paper focused on intrusion detection and big heterogenous data.
4	[94]	2015	Processing large volumes of security alerts.	Proposed a novel framework to investigate temporal trends and patterns of security alerts coming form various hosts and security devices and software. Testing showed that the proposed system's ability to extract relevant descriptive statistics to support effectiveness measurement of anomaly detection approaches.
5	[123]	2016	Reduction of large volumes of system log data.	Proposed a log data reduction algorithm that exploits dependency between system events to reduce the number of log entries without impacting the quality of the forensic analysis. Initial evaluation of the proposed aggregation algorithm showed significant reduction in log file sizes while not compromising quality.
6	[131]	2016	Support the process of mining large volume of documents discussing malware.	Proposed FeatureSmith; an Android-focused malware detection system that feeds its databse from mining of huge volume of documents written in natural language such as scientific papers. The implementation of the proposed system yielded 92.5% true positive and 1% false positive.
7	[30]	2017	Support the processing of very large volume of traffic logs to create machine-specific profiles.	Proposed RiskTeller; a malware infection-risk prediction system based on analysis of appearance of binary files in network traffic. The proposed system was tested in machines in 18 enterprises in a year-long test and showed promising results.

8	[18]	2017	Processing large volumes of network data to build an intrusion prediction model.	Proposed a novel contextual framework that supports the work of IDSs to detect cyber-attacks. The proposed system demonstrated success in detecting both known and unknown attacks.
9	[43]	2017	Support the training of deep learning system using large volumes of log data.	Proposed DeepLog; a deep learning model utilizing LSTM to model a system log as a natural language sequence to detect anomalies. The proposed system was tested on VAST Challenge 2011 data set and sucessfully detected 5 out of 6 attacks.
10	[105]	2017	Support the processing of millions of logins of a global financial company to test the detection algorithm.	Proposed an anomaly detection system design to detect anomalies in logins within an enterprise network using market-bassket analysis algorithm. The implementation of the system detected 82% of malicious logins with 0.3% false-positive.
11	[61]	2018	Support the processing of millions of lines of log data.	Proposed a method for reduction of log datsa for forensic analysis. The proposed system provided a reduction factor of 4.6 to 19 and was capable of analyzing about 1 million events per second.
12	[59]	2018	Support processing a large dataset from log files.	A study to evaluate the efficacy of log parsers (Zookeeper, Proxifier, BGL, HPC, and HDFS) on five datasets. The study proposed a novel Parallel Log Parser (POP) that runs on top of Spark, a large-scale parallel processing platform. The proposed system was evaluated and has demonstrated high accuracy, effectiveness, and efficiency.
13	[40]	2018	Support the training of machine learning model with a large data set.	A study to assess the use of various machine learning tools for detection of anomalous behaviors in complex environments. The findings of the study confirm the effeciency of machine learning systems in detecting security issues.
14	[102]	2018	Support the training of machine learning model with a large data set.	Proposed Tiresias; a security event prediction system based on machine learning. Testing showed a 93% accuracy of the system in predicting the next event that would occur on a machine.
15	[23]	2018	Support the training of machine learning model with a large data set.	Proposed a machine-learning based system for detecting Distributed Denial of Service (DDoS) attacks in IPv6 NDP. Several machine learning algorithms were tested in the study out of which decision tree and random forest algorithms proven to give the highest accuracy as compared to other algorithms.

16	[70]	2019	Support processing a large dataset.	Proposed a a crowd anomaly detection system that focuses on rejecting motion outliers. The proposed algorithms was implemented on low-power Field-Programmable Gate Array (FPGA). The testing results have shown that the system is capable of processing 126.65 megapixels per second with a maximum of 2.43 nJ of energey required per pixel.
17	[35]	2019	Support the processing of millions of lines of log data.	Proposed a novel approach for microservices logs tracing based on passive tracing of request-response messages of the REpresentational State Transfer (REST) communication model.
18	[36]	2020	Support the processing of millions of lines of log data.	Presented a log-filtering method focused at pinpointing interesting events for human analysts to review.
19	[74]	2020	Support the processing and analysis of large data set of IoT and network traffic data.	Proposed an approach to detect network attacks and anomalies based on machine learning and big data technologies. Testing was done on two datasets; IoT (with 7 million instance), and network traffic (500,000 instances) with good accuracy.
20	[127]	2020	Support data processing for a large deep neural network	Presented ADA, a log anomaly detector based on an unsupervised online deep neural network. ADA utilizes a dynamic threshold algorithm to dictate the optimal threshold based on recently detected events to improve the detection accuracy.
21	[134]	2020	Support the processing of millions of lines of log data.	Presented a learning-based anomaly detection framework for service-provision systems with microservices architectures using service execution logs and query traces.

Table 2: Summary of papers in spamming, spoofing, and phishing detection section

No	Ref#	Year	Aspect of Big Data	Summary
1	[80]	2016	Support processing a large dataset of FQDNs and website contents.	Proposed a promotional-infection detection system based on semantic-inconsistency search. The approach employs natural language processing to identify the bad terms most irrelevant to the sTLDs. Testing was done on on 100,000 FQDNs with a false postitive rate of 1.5% and over 90% coverage.
2	[129]	2016	Support processing a large dataset from Twitter.	A study on the creation and defense againt social media botnets. The study proposes a method of calculating influencers score based on credible retweets after detecting bot retweets.

3	[91]	2017	Support processing a large dataset from Twitter(1.3 million tweets).	Proposed POISED; a Twitter analytics system used to detect malicious and spam content propagation. Testing showed a 91% success rate with 93% recall.
4	[125]	2017	Support the training of machine learning model with a very large data set.	Proposed a deep-learning based crowdturfing detection system. The study proposes a recurrent neural network to created an automated defense system against these attacks.
5	[63]	2017	Support the training of machine learning model with a large data set.	Proposed a system to detect Facebook "like farms". The study proved that many like farms are completely operated by bots and does not require human intervention. The machine learning based classifier had an accuracy of 99% in detection with 93% recall.
6	[64]	2018	Support the processing of large volumes of GPS data collected over a specified period of time.	Proposed a crowdsourcing-based method to detect and localize GPS spoofing attacks named Crowd-GPS-Sec. The test data contained 141,693 unique positions of 142 airplanes. The proposed system achieved attack detection delay of 2 seconds, with an attacker localization accuracy of 150 meters using data from 15 minutes of monitoring time.
7	[78]	2018	Support the training of machine learning model with a large data set.	Proposed a machine-learning based malicious call detection system. The system was built on 9 billion call records over the period of 3 months. Testing of the system showed 90% accuracy for malicious calls and 99.99% accuracy to identiy non-malicious calls with an average latency of 1 ms.
8	[56]	2018	Support the processing of large volumes of machine learning data.	Proposed SAFe-PC; a machine learning based phishing detection system. The system was tested on two large data sets provided by a univeristy, and SophoS. The experiment has shown that the proposed system caught 70% of the emails that were not caught by Sophos.
9	[112]	2020	Support the training of machine learning model with a large data set.	Proposed AutoBLG; a malicious URL detection system based on machine learning. When tested, AutoBLG showed 99% noise filtering capacity with a toxicity range from 1.17 to 16.5%.

Table 3: Summary of papers in malware and ransomeware section

No	Ref#	Year	Aspect of Big Data	Summary

1	[76]	2015	Support the processing of 19 million downloader graphs from 5 million hosts.	Proposed a system for detection of malware download activity through downloader-graph abstraction. Testing showed 96% true-positive rate and 1% false positive rate.
2	[73]	2018	Support the training of machine learning model with a large data set.	Prposed RanDroid; asn Android malware detection system based on machine learning. Testing showed that decision-treee classification had the highest accuracy in comparison to other methods.
3	[62]	2018	Support the analysis of large data set of ransomware ecosystem.	A study on ransomware eco system that included data collected over the period of two years from thousands of ransomware attacks. The study ncluded data about ransomeware payments, victims, and payment operators.
4	[119]	2019	Support the analysis of large data set of Windows executabibles.	An article discussing how to handle and utilize daily datasets of malware samples. The study guides the reader through a step-by-step analysis of hundreds of thousands of Windows executables collected in one day from the security company feeds. The main aim of the study is to show how a company can employ state-of-the-art techniques for automated processing of samples and perform manual experiments to have a better understanding and documentation of the contents of the dataset.

Table 4: Summary of papers in code security section

No	Ref#	Year	Aspect of Big Data	Summary
1	[79]	2016	Support the processing of large volumes of data originating from 71,000 articles.	Proposed iACE; automated Indicators Of Compromise (IOC) detection system using graph mining techniques. The IOC generated in testing had 95% accuracy and over 90% coverage.
2	[104]	2017	Support the training of machine learning model with a large data set.	Proposed LAD; a program anomaly detection system that is based on mildly context-sensitive grammar verification. LAD is based on a purpose-built constrained agglomerative clustering machine learning. Latency overhead during testing was limited between 0.1 ms to 1.3 ms to profile and analyze a single behavior instance. This small overhead is considered acceptable keeping in mind the size of the search space.

3	[27]	2017	Support the training of machine learning model with a large data set.	Proposed a system for prediction of reilience of obfuscated code against symbolic execution attacks based on machine learning. Testing showed about 90% accuracy for 80% of the programs tested.
4	[72]	2017	Support the processing of billions of lines of code.	Proposed VUDDY; a vulnerable code clone discovery system with high scalability. During the testing phase, the proposed system is said to outperformed SourcererCC, ReDeBug, DECKARD, and CCFinderX in terms of detection time but with lower number of clones reported.
5	[47]	2017	Support the analysis of large data set of firmware functions.	Proposed a scalable graph-based bug search for firmware images with focus on IoT firmware. The test was done on a dataset of 33,045 devices which was collected partially from public systems. The average search time was 1 second when performed over 8,126 firmware images of over 420 million functions.
6	[71]	2018	Support the analysis of multiple large datasets of certificates.	A study examining the code-signing Public-Key Infrastructure (PKI) revocation effectiveness. The study shows that the erroneous setting of revocation dates can cause the signed malware to remain valid after the revocation of the certificate. The study also demonstrated failures in disseminating the revocation decisions which leaves the users trusting abusive certifications.
7	[22]	2018	Support the processing of billions of lines of code.	Proposed FOSSIL; a system designed to identify free open-source software packages in binaries of which source code is not available. This task is quite important in malware detection. The proposed system was tested on three datasets and has shown 0.95 mean precision with 0.85 mean recall.
8	[21]	2018	Support the processing of millions of lines of code.	Proposed NAVEX; an exploit generation system for multi-tier dynamic web applications using a combination of dynamic and static code analysis. Testing was done on 3.2 million lines of PHP code and sucessfully generated 204 exploits.
9	[101]	2019	Support the generation of Attack Defense Trees from large volumes of publicly-available security data.	Proposed a security software design pattern classification system based on data integration that facilitates security pattern choice. Despite the limitation in the proposal's security patterns and classification, the proposed system shows potential of expanding beyond these limitations to become helpful in selecting proper security design patterns.

Table 5: Summary of papers in cloud security section

No	Ref#	Year	Aspect of Big Data	Summary
1	[49]	2016	Support the processing of large volumes of data.	Proposed CA-HCIA; a novel incident analytics framework for cloud-based cybersecurity insurance system. The suggested framework combines business and technical approaches to support making appropriate decisions related to information technology strategies through the creation of a classified hierarchy that helps in identifying risks.
2	[83]	2018	Support the processing of large dataset of network management audit data.	Proposed a system to detect virtual networks isolation breaches, named ISOTOP. Testing showed the capacity of the proposed system to scale to large OpenStack-based data centers successfully.
3	[85]	2018	Support the processing of large data set of cloud network traffic data.	Proposed a a proactive cloud security auditing system based on learning probabilistic dependencies among events. The proposed system was integrated to OpenStack and tested in an extensive cloud testing environment of about 100,000 virtual machines where it demonstrated practical response time of 6 ms to audit the whole cloud. This shows an improvement of 50% over other existing proactive approaches.
4	[90]	2018	Support the processing of large volumes of data collected for the security solution.	Proposed Dejavu; a framework exploring the reuse of existing security solutions as blackbox analytics in the cloud. The framework creates a layer between the third-party security solutions and the cloud. Testing the system proved its success in tansforming several security solutions into cloud security solutions.

Table 6: Summary of papers in other applications section

No	Ref#	Year	Aspect of Big Data	Summary
1	[15]	2015	Support the processing of large data set of network traffic data.	Proposed a markov-model based cybersecurity analytics framework to measure predictive security risk for an enterprise. The proposed system is said to provide a better view of the state of security of an enterprise network.
2	[44]	2016	Support the processing of tracking data captured from 1 million websites.	Proposed the use of OpenWPM to analyze user tracking habits of 1 million websites. The testing was done to show that 91.7% or websites use stateless tracking among other important results.

3	[93]	2017	Support the processing of large volume data coming from a large number of sources around the world.	Proposed Augur; an Internet-connectivity disruption detector. The system was tested on data from 180 countries over the period of 17 days and was sucessful in showing disruptions to access to most frequently visited sites.
4	[95]	2018	Support the analysis of large data set of SMS.	A study on charicterization of the security of the SMS ecosystem based on analysis of 900k messages. The study shown that many security-sensitive applications sent messages unencrypted, low-entropy solutions used in OTP, among other interesting findings.
5	[55]	2018	Processing large data sets for training on explanation of deep learning results classification.	Proposed LEMNA; a high-fidelity explanation method for security applications. The proposed system was tested on two security functions 1) malware detection, and 2) function start detector for binary reverse engineering. The testing results showed a much higher fidelty level in explanations in comparison to the current methods.
6	[52]	2018	Support processing very large datasets extracted from social media sites.	Introduced a novel attack based on leveraging data freely available on social media to infer missing attributes of targets. The attack was tested on data of 1.1 million users and has shown good success rates.
7	[46]	2018	Support the processing of large volumes of vulnerability scanning reports and data from other sources.	Proposed VULCON; a vulnerability management system based on time-to-vulnerability-remediation and total vulnerability exposure. When tested, VULCON showed 8.97% reduction in total vulnerability exposure.
8	[28]	2018	Support the training of machine learning system from large datasets.	Analyzed the unobservibility properties for three multimedia protocol tunneling technologies. The deep analysis of CoverCat, DeltaShaper, and Facet yielded that the existence of manually labeled samples is necessary to detect the covert channels. The study included large datasets taken from YouTube and live video-chat videos.
9	[103]	2018	Supported the processing of over a billion records of data.	Proposed Threat Intelligence Computing; a novel methodology to model threat discovery as a graph computation problem. Tested with over billion records and proved success in detecting dozens of threat hunting tasks.