Is Our Website Being Attacked? Exercises on Flash Crowds, DDoS, and Fraudulent Resource Consumption

Coby Rem
Johns Hopkins University
crem1@jhu.edu

Clayton Franklin
Johns Hopkins University
cfrank31@jhu.edu

Shashvat Jhaveri
Johns Hopkins University
sjhaver4@jhu.edu

ABSTRACT

With an increasing use of cloud resources, detection of cloud misuse is important and cannot be ignored. With a shortage of cloud security personnel, this paper provides an assignment that familiarizes students with cloud security concepts and how to detect the abuse of cloud services. This goal is expanded by showing that all misuse is not the same, and there are cloud specific attacks that require more research to reliably detect. An outline for a nifty assignment is introduced that will expose students to existing threats such as flash crowd events (FCE) and distributed denial-of-service (DDoS) attacks, as well as emerging non-conventional threats to cloud infrastructure. The assignment offers examples of Fraudulent Resource Consumption (FRC) attacks and activities. The activities expand on why this new cloud-based attack cannot be easily detected and can cause significant harm to a cloud consumer.

KEYWORDS

cloud computing security, flash crowd, DDoS, fraudulent resource consumption

1 INTRODUCTION

As cloud computing becomes commonplace in business related Information Technology (IT) settings, one must be aware of new classes of threats that exist in shared-tenancy "coin-operated" environments. According to a report written by Precedence Research [14], the total cloud computing market in the US has the potential to grow to 450 billion USD by 2032. Unlike traditional IT environments, cloud computing operates on a pay-as-you-go model where cloud consumers only pay for resources and compute used. While in theory this model should pass on the savings of large data centers to the consumer at an affordable price, one must budget and plan correctly as to avoid overspend. This leads into a class of threats called Fraudulent Resource Consumption. Idziorek et al. explains that abuse of utility pricing models is not a new concept and they provide an example of how one could gain unauthorized use of a telephone network using older technology. However, in a cloud environment, the repercussions are different. The cost of this misuse is directly passed to the cloud consumer, or the entity that is responsible for funding the cloud resources being abused. The goal of an FRC attack is to cause excessive spend in a cloud environment, causing an entity to possibly remove their systems from the cloud, or at least reduce their cloud presence. In the worst cases, FRC attacks can cause businesses to fail due to unsustainably high costs [10].

In this nifty assignment, we created a story-driven coding assignment to help students become familiar with cloud computing and understand threats unique to cloud environments. Students will be provided with datasets of network logs and are expected to visualize the traffic, produce algorithms to successfully detect and differentiate FCEs and DDoS attacks, as well as understand why methods to detect these more common threats do not apply to three types of FRC attacks. Students will also model the economic impact FRC attacks may have on a cloud consumer. Students that complete this assignment will have a good understanding of the impact and difficulty in detecting FCEs, DDoS attacks, and FRC attacks, and will hopefully develop a new interest in cloud security.

Contributions that this paper makes are as follows:

- Providing students with hands-on learning experiences with cloud computing security through the lens of threats such as fraudulent resource consumption, an emerging threat in cloud environments.
- Provide a standardized approach to introduce novel concepts in an easy to digest, but challenging way through use of programming, algorithms, and data visualizations.
- Adapting research topics and applying them to real-world situations, further expressing the importance of this topic and the need to explore further underrepresented areas of cloud computing.

2 BACKGROUND

In cloud computing models, there are many facets of security that must be addressed. We will focus on attacks that deem to harm the target by expending resources at high volumes, driving up costs, or affecting service uptime. Perhaps the most well-known network attack is the DDoS attack, which floods the resource with requests until its usage is maxed out, rendering the service unusable. This attack pattern is rather predictable and considerable research has been done to detect and mitigate these, such as detection using Support Vector Machines [16] and mixed machine learning software [19].

Another predictable pattern is an FCE, which consists of a large number of simultaneous client requests to a service. Like DDoS, these events are easier to detect when they occur, as a certain threshold is usually surpassed. However, they can occur due to legitimate circumstances such as sporting events, a viral online post, holiday shopping sales, or major news events [10], unlike DDoS which is usually a deliberate attack. It can be difficult to differentiate between the two patterns. Research has been done on this front as well. Idziorek et al. [10] describes the differences

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
605.731 Class Research Symposium, 2024, Johns Hopkins University
© 2024 Copyright held by the owner/author(s).

in Figure 1a showing what both request patterns look like. Figure 1b further shows that FCEs typically have no change in per-client requests, instead involving a large volume of clients themselves [10]. Entropy detection has been used to differentiate events, as FCEs will have a higher degree of entropy in source IP addresses compared to DDoS [17].

A more problematic attack is Fraudulent Resource Consumption. This type of abuse occurs when cloud resources are expended but not to the point of triggering any alarms or crossing any thresholds. This type of attack is hard to pinpoint as the traffic that is generated appears to be normal, but is in fact artificially driving up cost. The traffic volume may be well below the threshold that would deem it a larger scale attack such as DDoS, but is high enough that the cost increase is "no longer trivial to the target" [4].

Using the paper written by Idziorek et al. as inspiration, we aim to aid students in understanding threats from FRC. Idziorek et al. describe thresholds in which legitimate usage is eclipsed and becomes FRC. Zipf's Law is often used as an anomaly detection metric by comparing traffic to the Zipf's frequency distribution (which is a commonly observed distribution in fields such as natural language [13]), expanding on previous work done by Idziorek et al. [9]. In these cases, traffic patterns in violation of Zipf's Law were a telltale sign of an FRC attack. Spearman's Footrule [5] is also used in both studies as well as by Courtney et al. [4], which is a test to observe changes in web page rankings due to anomalies.

Courtney et al. further expands on these concepts using data science and machine learning to build models that assist in detecting these attacks with greater accuracy that basic algorithms, utilizing Artificial Neural Networks and K-Nearest Neighbor classifiers. However, that study failed to find a "catch-all" method of detecting random, heavy-hitter, and trace-driven FRC attacks, instead finding detection methods that each could detect a specific attack. [4]. Other methods to utilize machine learning to classify such attacks include a study by Rustogi et al., who used an Artificial Neural Network to classify attacks with high degrees of accuracy for attacks that did not follow Zipf's Law [15].

3 RELATED WORK

This assignment address a gap in cloud security education as there are no publicly available LMS-based assignments that allow students to ethically and legally learn to defend against external threats. Although Canvas offers shareable and publicly available learning resources through Canvas Commons, there are only a few assignments related to cloud and no complete assignments related to cloud security [3].

Research regarding network security education has primarily focused on DDoS attacks. One study formed a training sequence involving multiple roles: An attacker to deliver the DDoS attack, a learner to recognize patterns and implement firewall rules, and a victim to recognize the impact these attacks have on organizations [7]. The authors of this study also created their own platform and environment to complete the assignment. Another study outlines the educational aspects of ethical hacking by offering exercises to learn about DDoS and other network attacks, and further emphasizes steps to prevent misuse of such information [18]. Both of these studies utilize Bloom's Taxonomy to evaluate students' learning outcomes based on various categories such as analysis, evaluation, and creation [8]. It should be noted that these studies focused on DDoS attacks and did not address FCEs and FRC attacks. These studies also involved more sophisticated software such as Snort, which is an open-source intrusion detection tool used in the Trabelsi et al. study [18].

This paper's nifty assignment extends to FCEs and FRC attacks and is tool-agnostic. Completing all 4 challenges will give students insight into practical skills required for cloud security, a field that is increasingly important in a digitally-connected world, yet suffers from a shortage of skilled professional workers [2].

4 ASSIGNMENT OVERVIEW

The assignment aims to teach undergraduate students with limited or no cloud security knowledge about legitimate website traffic patterns and malicious attacks that may shut down a website. It will also educate students on 3 approaches for prolonged and non-aggressive attacks that cause undue financial burden for a cloud consumer. Students will participate in hands-on activities to visualize website traffic, programmatically analyze network logs, develop simple algorithms to detect and differentiate between FCEs and DDoS attacks, and report on the monetary strain FRC attacks impose on an organization.

Students will utilize Canvas, a Learning Management System (LMS), to follow along the journey of Alex, a cloud engineer at fictitious e-commerce startup, Swift Mart. As the startup matures and security incidents occur, students will incrementally complete four challenges. Each challenge contains synthesized network logs based on NASA HTTP requests [12]. Each network log will have the same format and be saved in a comma-separated values (CSV) file; this file format allows students to easily parse the files in any software or programming language.

In the first challenge, students will develop software to read a network log that contains baseline network traffic. Next students will visualize the baseline traffic and analyze the data to answer questions about the startup's growth and costs. Students will submit code, visualizations, and answers to the online training platform for evaluation. Subsequent challenges will require students to perform similar data ingestion, visualization, and analysis, but utilizing different network logs that simulate security incidents and may also have to develop additional code to answer more complex questions.

For the next challenge, students will analyze FCE network logs to design a simple algorithm that detects when user request rates pass thresholds where cloud resources become overwhelmed and severe website degradation occurs.

The algorithm from the second challenge will then be applied to the DDoS data set. The same algorithm used to detect FCEs should be able to detect when the DDoS attack crosses into the denial of service region. Students will need to modify the algorithm to differentiate when legitimate, although damaging, FCE requests occur and malicious DDoS attacks are started.

Finally, students will learn about random, heavy-hitter, and trace-driven FRC attacks and how they evade previously created detection algorithms. Students will then model the financial impact of this new cloud-focused attack.

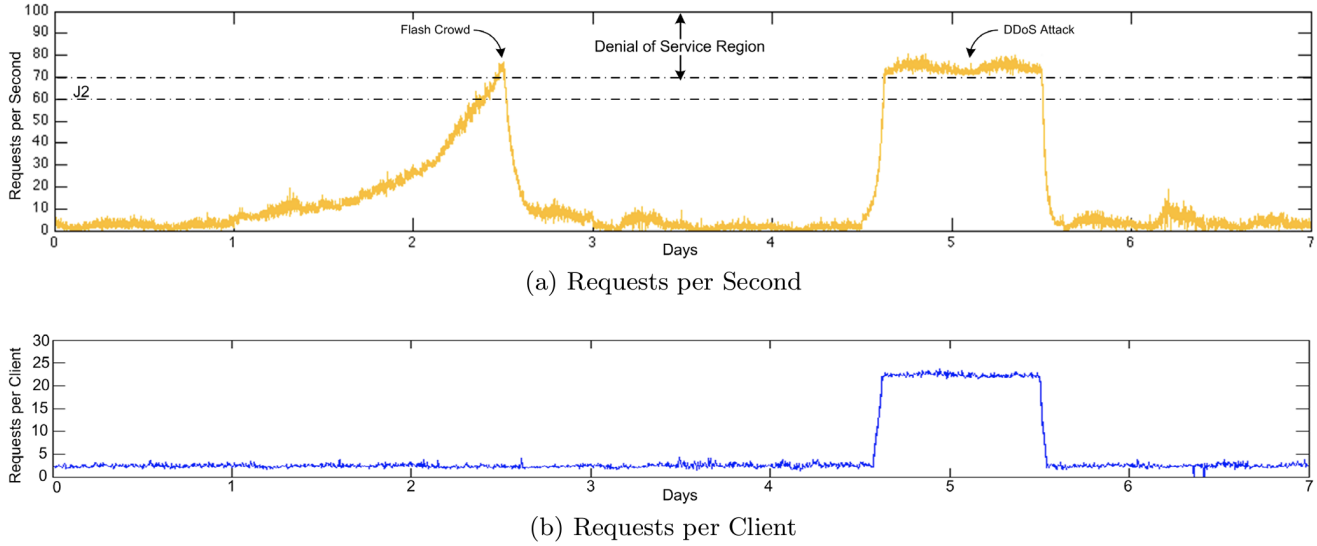

Figure 1: Flash Crowd and Application-layer DDoS Comparison. [10]

Once all challenges are completed, students' code, visualizations, and answers will be evaluated against a predefined rubric to determine if they helped the startup succeed, or if the startup succumbed to external security threats.

5 METHODS

Students' learning is assessed by using the Kirkpatrick Model, a framework for evaluating the effectiveness of training programs, with a focus on levels one and two of the model, Reaction and Learning [11]. Kirkpatrick Model levels three and four, Behavior and Results, are not implemented due to short semester timelines that don't allow for more complex and long term studies.

By hosting the assignment on Canvas, student generated data is used to not only evaluate students' performance, but also to discover if the assignment is effective in its current form or needs modification. The first level of the Kirkpatrick Model, Reaction, is used to evaluate the effectiveness of the training content. A survey is provided at the end of the training to gauge students' immediate reactions to the training. The survey collects quantitative and qualitative data in the form of Likert scale and short-answer questions. Survey elements involve ranking or providing a short response to the following questions:

- Is the student satisfied with the content of the assignment?
- Is the assignment relevant to the student's academic and professional goals?
- Did the student have any issues with the online training platform?
- Were the four challenges engaging?

Feedback is iteratively incorporated into the assignment to improve the training for future students or to gather more applicable feedback.

The second level of the Kirkpatrick Model, Learning, is used to evaluate if students learned from the assignment using two methods. The first method involves manually grading code and visualizations against a predefined rubric. Each challenge will have knowledge checks where students will answer multiple choice, matching, true/false, multiple answers, and fill-in-the-blank questions that are automatically graded. The students' final marks consist of the grades from all four challenges.

The second method used for the Learning level of the Kirkpatrick Model is the use of pre-tests and post-tests to evaluate if students learned new skills after completing the assignment. The results of the pre-tests and post-tests are not used in students' final marks for the course. The tests will be exclusively used as feedback to identify areas of the assignment where knowledge transfer was ineffective and can be improved.

It is expected that all students will complete the assignment with passing grades. Additionally, many students may know or deduce answers to select pre-tests questions, especially questions focused on defining DDoS attacks, and to a lesser extent FCEs. A larger improvement in scores is expected with FRC attack questions, as this attack is newer and not widely known in the cloud industry due to difficulties implementing effective detection methods. If students do not successfully complete the assignment in larger-than-expected rates or there are knowledge gaps in specific topics even after the training, then pre-test, post-test, and survey data can be used to revise the assignment. We hope that students learn the impacts cloud threats have on organization, and consider security best practices to protect critical cloud infrastructure in their careers.

6 RESOURCES NEEDED

To effectively complete this assignment, students will need a computer with Internet access, a Canvas account, and any programming language or software that can perform data manipulation and create

custom visualizations. Educators can assign students to the Canvas course, which includes all necessary assignment material.

7 IMPLEMENTATION AND CHALLENGES

The nifty assignment consists of three sections. First, code was developed to generate the datasets provided to students in each challenge. Second, example code was developed to generate graphs, calculate the maximum requests per second, create detection algorithms for the attack methods, and estimate the costs of FRC attacks. Lastly, a Canvas course was created to deliver the content to students and educators.

7.1 Data Generation

The most challenging part of developing this assignment was generating the network logs required for each cloud threat. The first step was finding data that would be used as a basis for the synthesized datasets. The HTTP logs from NASA Kennedy Space Center that Idziorek et al. utilized was selected to kick-start the data generation instead of generating all of the data from scratch [10, 12]. Once the initial data was downloaded, it quickly became apparent that a way to programmatically manipulate the data was necessary. Python code was developed to unzip, tabulate, remove a few entries that were missing data and the save the data in a CSV format so it would be easier to view and parse. After an informal exploratory data analysis, it was decided to only keep data related to the client's IP address, website visited, and timestamp. The HTTP method, protocol, status code, and number of bytes transferred in the web logs would not be utilized by students to complete the assignment, so the four fields were omitted. Additionally, the timestamps were modified to change the year from 1995 to 2023 to add more realism to the network logs.

After the data was cleaned and converted to a CSV format, the baseline traffic data set was initially generated. Creating this dataset was simple and only required selecting data between two timestamps with no modifications. The data for the FCEs was more complex. The synthesized data had to meet five requirements:

(1) The data needed to have a configurable amount of realistic IP addresses for new users visiting websites.
(2) The requested web pages needed to be randomly sampled from the original network logs.
(3) The attack needed to span between two specified timestamps.
(4) The number of additional web traces per second needed to ramp up to a peak number of requests in a controlled, but also pseudo-randomized fashion.
(5) Similarly, the requests also needed to ramp down back to baseline traffic levels.

Thousands of client IP addresses were generated using the Faker library [6]. A list of unique web pages from the original logs were created and used in each FCE request. To recreate the rise and fall of web requests in an FCE, exponential growth and decay models were implemented to calculate the number of requests needed for each second between two timestamps to reach a peak number of requests. The exponential models allowed for the rate of increase and decrease to be dictated by timestamps and the peak number of requests per second. For example, the increase in requests would be more gradual as an FCE ramps up to 60 requests per second over 5 hours, compared to the sharp decline in requests in the last 30 minutes of the FCE after servers have been overloaded. Once the number of requests at each second of the attack was calculated, a new entry in the web log was appended consisting of a client IP, a randomized web page, and the timestamp. After generating the data, the web logs were sorted in ascending order by timestamp to interweave the fake web logs into the original dataset.

Although it took a substantial amount of time and multiple iterations to correctly implement the exponential models to generate the FCE data, the code was reused to synthesize the DDoS data. The exponential growth and decay was used to start and end the DDoS attack, and a constant elevated number of requests was added for the duration of the DDoS attack. The code to create the requests was also reused for generating the FRC attack data. The cloud threats' intensities and duration can be adjusted by modifying function inputs. In order to verify that the data was generated properly, the network logs needed to be visualized.

7.2 Data Visualization, Detection Algorithms, and Other Calculations

The second component of the assignment involved creating visualizations for all four challenges, writing algorithms to detect and differentiate FCEs and DDoS attacks, calculating the maximum number of requests for each dataset, and calculating the economic impact an FRC attack would have on Swift Mart.

While developing the visualizations, it was difficult to determine where nuisance activity, detection, and denial-of-service thresholds were surpassed. To remediate this issue, horizontal lines were added to the FCE, DDoS, and FRC charts to demarcate critical thresholds. The inclusion of plot thresholds enforces key concepts that distinguish the three cloud threats. For example, an FRC attack must be above the nuisance activity level of 10 requests per second, but under the FCE and DDoS detection threshold of 50 requests per second. If the thresholds were not included in the FRC visualization, students may not understand the dynamic range of an FRC attack. The baseline traffic with no thresholds and the three cloud threats are shown in Figures 2, 3, 4, and 5.

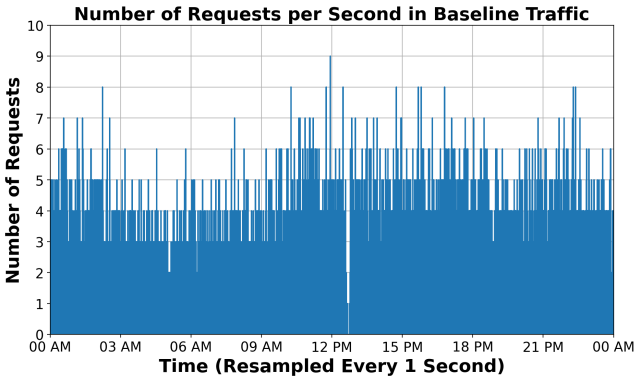

Figure 2: Baseline Traffic: Requests per Second

Another challenge was developing the algorithm to make the distinction between FCEs and DDoS attacks. Idziorek et al. do not

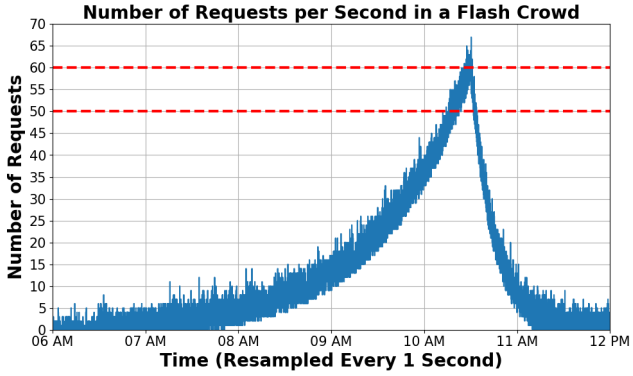

Figure 3: FCE: Requests per Second

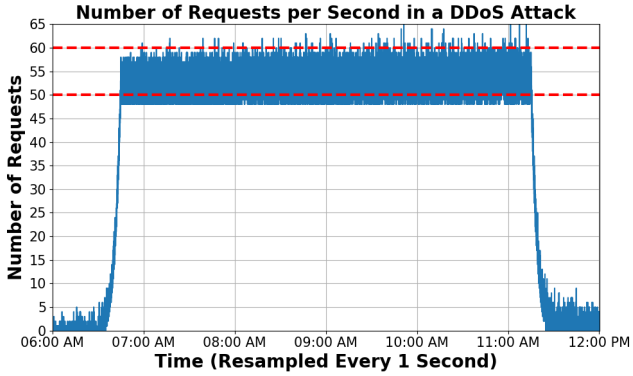

Figure 4: DDoS Attack: Requests per Second

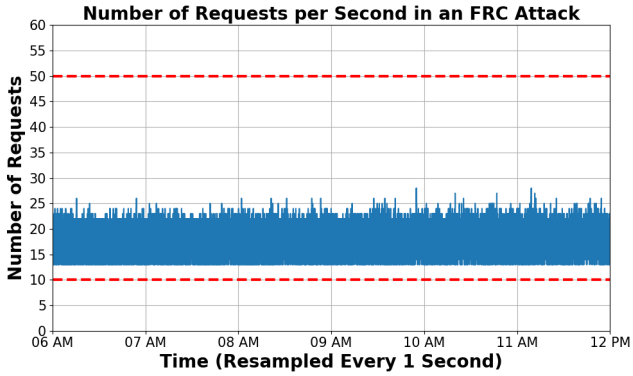

Figure 5: FRC Attack: Requests per Second

specify the resolution required for calculating and plotting the requests per client per time frame. [10] Initially the average requests per client per second was utilized in the detection algorithm and visualizations. With only 50 clients in the DDoS botnet, this resulted in less than 1 request per client per second and poor visualizations as seen in Figure 6. Instead, calculating and plotting the requests per

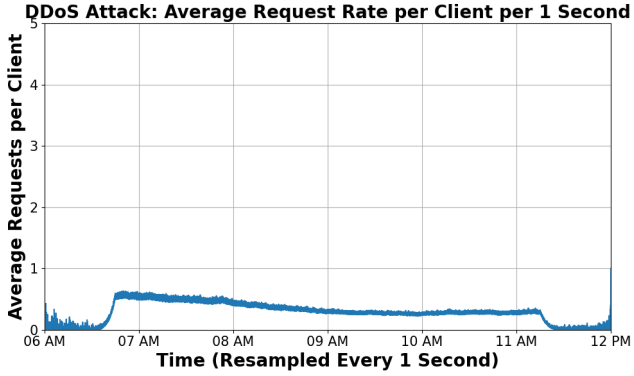

Figure 6: DDoS Attack: Average Requests per Client per Second

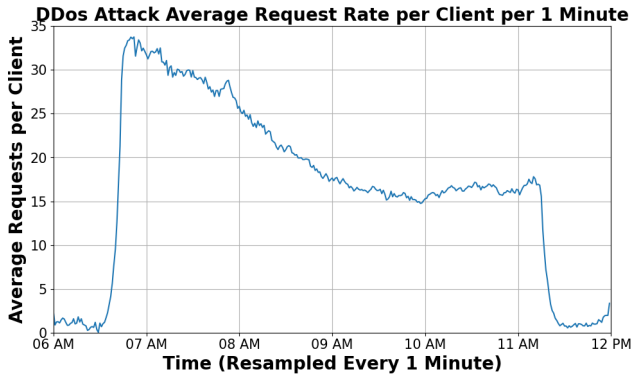

Figure 7: DDoS Attack: Average Requests per Client per Minute

client per minute resulted in more realistic values for the detection algorithm and charts as seen in Figure 7.

To model the economic impact of FRC attacks, a function was created based on Equation 1.

$$T_c = r_{sec} \times r_{size} \times d_{sec} \times t_x \tag{1}$$

Where T_c is the total cost, r_{sec} is the requests per second, r_{size} is the average request size in gigabytes with a default of 0.000015 GB per request, d_{sec} is the duration of the FRC attack in seconds, and t_x is the EC2 egress cost with a default of $0.09 per GB [1].

In addition to calculating the $17.50 cost of the baseline traffic at 5 requests per second over 30 days, the cost of two FRC scenarios was calculated. The first FRC scenario involved a high intensity attack of 45 requests per second over 30 days and the second scenario took place over 90 days at a lower attack intensity of 15 requests per second. Even though the attack intensities and duration differed, the final cost for both scenarios was $157.46. This similarity shows students that a more subtle but longer FRC attack has the same economic impact on a company with a greater ability to evade detection.

7.3 Assignment Creation

The final part of the assignment was to create an assignment in Canvas. Although it was easy to start designing the assignment, many aspects of the assignment creation were challenging and tedious. It was harder than expected to maintain a consistent and engaging story line and questions while balancing the technical aspects of cloud security. Additionally, any changes in the assignment structure, story, or coding resulted in cascading changes through out the course. While the assignment was not tested in a classroom setting, we hope that the assignment's story line and questions are helpful to students. Student data and feedback will also be incorporated into the future versions of the assignment by using the Kirkpatrick Model. The course is accessible for public use through Canvas Commons [3].

8 FUTURE ENHANCEMENTS

Three potential improvements were identified to enhance the students' learning experience.

Educators may adapt the assignment for specific software, programming languages, or tool sets similar to how to Trabelsi et al. utilize Snort [18]. While the data generation and sample code was developed using Python, the assignment makes no reference to required software. Adding instructions tailored to a specific tool may simplify the assignment, especially if software or code templates can be used to teach students about best practices, standardize submissions, and reduce setup time.

Another area of improvement would be to use actual request sizes instead of average request sizes for the economic modelling of FRC attacks. Additional code would be required to map a request size to each synthesized web request. This would provide a more realistic activity compared to using a default 0.000015 GB per request.

The assignment is targeted towards undergraduate students with limited experience in cloud security. Hence, the quiz questions are rudimentary and concise, based on material provided in the LMS and insights from the four challenges. To expand the assignment to graduate students, open-ended questions based on research papers can be implemented in quizzes or Canvas's discussion forum. Graduate students can also research and deploy detection strategies for FRC attacks.

9 CONCLUSION

Network events such as FCEs, DDoS attacks, and FRC attacks continue to threaten the integrity of organizations and websites hosted on cloud platforms. It is important that students interested in the cloud security industry are knowledgeable about such threats to help combat them in any scenario. This assignment aims to provide an introduction to these network events using practical skills. By splitting the assignment into distinct challenges for each event, students have the opportunity to recognize the event patterns, explore impacts on users and organizations, and learn how to detect these threats.

As of this paper's writing, the assignment has not been implemented in a classroom setting. However, in addition to the enhancements mentioned in Section 8, the use of Kirkpatrick Model principles and Canvas to automate the collection of student feedback and grades will allow iterative improvements to the assignment.

REFERENCES

[1] Inc. Amazon Web Services. *Amazon EC2 – Secure and resizable compute capacity – AWS*. en-US. URL: https://aws.amazon.com/ec2/pricing/ (visited on 02/12/2024).

[2] Mike Campfield. "Mind the gap: the cloud security skills shortage". In: *Computer Fraud & Security* 2021.8 (Aug. 2021), pp. 6–10. ISSN: 1361-3723. DOI: 10.1016/S1361-3723(21)00084-1. URL: https://www.sciencedirect.com/science/article/pii/S1361372321000841 (visited on 02/12/2024).

[3] *Canvas Commons*. en. URL: https://community.canvaslms.com/t5/Canvas-Commons/tkb-p/commons (visited on 04/22/2024).

[4] Lauren Courtney et al. "Data Science Techniques to Detect Fraudulent Resource Consumption in the Cloud". In: *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*. Jan. 2021, pp. 0451–0457. DOI: 10.1109/CCWC51732.2021.9375938. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9375938 (visited on 02/12/2024).

[5] Cynthia Dwork et al. "Rank aggregation methods for the Web". en. In: *Proceedings of the 10th international conference on World Wide Web*. Hong Kong Hong Kong: ACM, Apr. 2001, pp. 613–622. ISBN: 9781581133486. DOI: 10.1145/371920.372165. URL: https://dl.acm.org/doi/10.1145/371920.372165 (visited on 02/15/2024).

[6] *faker.providers.internet — Faker 24.11.0 documentation*. URL: https://faker.readthedocs.io/en/master/providers/faker.providers.internet.html (visited on 04/25/2024).

[7] Walter Fuertes et al. "Software-Based Platform for Education and Training of DDoS Attacks Using Virtual Networks". In: *2017 International Conference on Software Security and Assurance (ICSSA)*. July 2017, pp. 94–99. DOI: 10.1109/ICSSA.2017.19. URL: https://ieeexplore.ieee.org/document/8392625/;jsessionid=70FE5A5B0791C7078BF36F29E0444F25 (visited on 04/22/2024).

[8] SHI HUILAN et al. "Educational management in Critical Thinking Training Based on Bloom's Taxonomy and SOLO Taxonomy". In: *2020 International Conference on Information Science and Education (ICISE-IE)*. Dec. 2020, pp. 518–521. DOI: 10.1109/ICISE51755.2020.00116. URL: https://ieeexplore.ieee.org/document/9418900 (visited on 04/22/2024).

[9] Joseph Idziorek and Mark Tannian. "Exploiting Cloud Utility Models for Profit and Ruin". In: *2011 IEEE 4th International Conference on Cloud Computing*. ISSN: 2159-6190. July 2011, pp. 33–40. DOI: 10.1109/CLOUD.2011.45. URL: https://ieeexplore.ieee.org/document/6008690/ (visited on 02/15/2024).

[10] Joseph Idziorek, Mark Tannian, and Doug Jacobson. "Detecting fraudulent use of cloud resources". In: *Proceedings of the 3rd ACM workshop on Cloud computing security workshop*. CCSW '11. New York, NY, USA: Association for Computing Machinery, Oct. 2011, pp. 61–72. ISBN: 978-1-4503-1004-8. DOI: 10.1145/2046660.2046676. URL: https://dl-acm-org.proxy1.library.jhu.edu/doi/pdf/10.1145/2046660.2046676 (visited on 02/12/2024).

[11] Donald Kirkpatrick. *Evaluating Training Programs: The Four Levels*. en-US. 1st. Berrett-Koehler, 1994. ISBN: 978-1-881052-49-4.

[12] NASA. *NASA-HTTP*. URL: https://ita.ee.lbl.gov/html/contrib/NASA-HTTP.html.

[13] Steven T. Piantadosi. "Zipf's word frequency law in natural language: A critical review and future directions". In: *Psychonomic bulletin & review* 21.5 (Oct. 2014), pp. 1112–1130. ISSN: 1069-9384. DOI: 10.3758/s13423-014-0585-6. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4176592/ (visited on 02/15/2024).

[14] Precedence Research. *Cloud Computing Market Analysis*. Englsih. Analysis 1701. Oct. 2023. URL: https://www.precedenceresearch.com/cloud-computing-market.

[15] Rishabh Rustogi et al. "Machine Learning Based Web-Traffic Analysis for Detection of Fraudulent Resource Consumption Attack in Cloud". In: *2019 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*. Oct. 2019, pp. 456–460. URL: https://ieeexplore.ieee.org/document/8909668/ (visited on 02/15/2024).

[16] T. Subbulakshmi et al. "Detection of DDoS attacks using Enhanced Support Vector Machines with real time generated dataset". In: *2011 Third International Conference on Advanced Computing*. ISSN: 2377-6927. Dec. 2011, pp. 17–22. DOI: 10.1109/ICoAC.2011.6165212. URL: https://ieeexplore.ieee.org/document/6165212/ (visited on 02/15/2024).

[17] Srinath Sureshkumar. "Classification of DDoS Attacks and Flash Events using Source IP Entropy and Traffic Cluster Entropy". In: *2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT)*. Sept. 2021, pp. 1–5. DOI: 10.1109/ICECCT52121.2021.9616887. URL: https://ieeexplore.ieee.org/document/9616887 (visited on 02/17/2024).

[18] Zouheir Trabelsi and Latifa Alketbi. "Using network packet generators and snort rules for teaching denial of service attacks". In: *Proceedings of the 18th ACM conference on Innovation and technology in computer science education*. ITiCSE '13. New York, NY, USA: Association for Computing Machinery, July 2013, pp. 285–290. ISBN: 9781450320788. DOI: 10.1145/2462476.2465580. URL: https://doi.org/10.1145/2462476.2465580 (visited on 04/22/2024).

[19] Josy Elsa Varghese and Balachandra Muniyal. "Trend in SDN Architecture for DDoS Detection- A Comparative Study". In: *2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER)*. Nov. 2021, pp. 170–174. DOI: 10.1109/DISCOVER52564.2021.9663589. URL: https://ieeexplore.ieee.org/document/9663589/ (visited on 02/15/2024).