Journal of Internet Services and Applications, 2025, 16:1, doi: 10.5753/jisa.2025.5176

Access Management for Content Delivery Networks: Measurements, Models, and Strategies

Lenise M. V. Rodrigues [**Universidade Federal do Rio de Janeiro e Globo** | ***lenisemvr@ic.ufrj.br***]
Daniel Sadoc Menasché [**Universidade Federal do Rio de Janeiro** | ***sadoc@ic.ufrj.br***]
Arthur C. Serra [**Globo** | ***arthurserra10@gmail.com***]
Antonio A. de Aragão Rocha [**Universidade Federal Fluminense** | ***arocha@ic.uff.br***]

Institute of Computing, Universidade Federal do Rio de Janeiro, Av. Athos da Silveira Ramos, 274 - Cidade Universitária, Rio de Janeiro, RJ, 21941-916, Brazil.

Received: 01 November 2024 • **Accepted:** 14 April 2025 • **Published:** 24 June 2025

Abstract We address the challenge of managing access to Content Delivery Networks (CDNs). In particular, we consider a scenario where users request tokens to access content, and one form of piracy consists in illegally sharing tokens. We focus on mitigating token misuse through performance analysis and statistical access pattern monitoring. Specifically, we examine how illegal token sharing impacts content delivery infrastructure and propose defining acceptable request limits to detect and block suspicious access patterns. Additionally, we introduce countermeasures against piracy, including selective quality degradation for users identified as engaging in illegal sharing, aiming to deter such behavior. Using queuing models, we quantify the impact of piracy on system performance across different scenarios. To validate our model, we perform statistical tests that compare real CDN traffic patterns with the expected request intervals in our proposed framework. These measures—defining access thresholds, quality degradation for unauthorized use, and statistical alignment checks—enhance CDN access management, preserving infrastructure integrity and the legitimate user experience while reducing operational costs.

Keywords: Piracy, Access Management, CDN, Access Patterns

1 Introduction

Over the past decade, there has been a tremendous growth of Internet content streaming platforms. As an example, 43.4% of Brazilian households with TVs now use paid streaming service.[1] Managing access to determine which users have rights to access specific content is a challenge and the central theme of this work. In particular, our focus is on studying the impacts of illegal access token sharing to view live content using the infrastructure of a national Brazilian Content Delivery Network (CDN) that serves millions of users daily.

The CDN aims to estimate the costs associated with nonlegitimate content consumption and to block the corresponding users without impacting legitimate users due to classification errors. In particular, when legitimate users are classified as suspicious, they may face a denial of service due to a false positive. This motivates a thorough analysis before revoking tokens. However, thorough analysis is expensive, consuming resources such as computational power and energy, and causing delays, which can affect legitimate users and can preclude the timely blocking of illegal users [Fett *et al.*, 2023; Patat *et al.*, 2022].

Given the above challenges, we pose the following two key research questions:

- **What is the performance loss of a CDN when serving illegitimate content?** In particular, how does such a loss increase the costs of the CDN and/or affect the quality of service (QoS) for users?
- As the implemented solutions need to be simple, **how do the simplest access management solutions behave in real networks?**

To answer the first question, we turn to queuing models (Section 4). In particular, we consider models like M/M/1, M/M/1 with burst arrivals (to capture malicious users), and M/M/1 with priorities (to capture strategies that favor legitimate users). Through such models, we illustrate some elements involved in the performance of access management systems. To answer the second question, we analyzed data from a national CDN, which serves one of the largest Brazilian content producers, with millions of daily hits (Section 5). We show that access management policies based on thresholds already bring significant results in mitigating the actions of malicious users, with few false positives.

Contributions. In summary, our contributions are threefold.

1. **Queuing models for performance analysis:** We propose queuing models that allow us to understand how system performance varies depending on the level of piracy. Specifically, we analyze both the case in which no countermeasures are in place (Section 4.1) and the case in which such countermeasures are implemented (Section 4.2). For the latter, we introduce a model to compare the cost-benefit of legal versus pirated streaming services. By incorporating service quality degra-

[1] `https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/38306-em-2022-streaming-estava-presente-em-43-4-dos-domicilios-com-tv` (last accessed 29 August, 2024)

dation as a countermeasure, we highlight a viable approach to disincentivizing piracy while maintaining a fair user experience.

2. **Real data analysis:** We assess whether real CDN traffic aligns with the proposed model by conducting statistical tests and applying visual analysis methods to compare observed request intervals with the model's assumptions. We also provide a comprehensive analysis of the time between requests using real data, including an examination of its bursty behavior. Finally, we conduct a longitudinal analysis of recurring suspicious users and find that a majority of flagged tokens correspond to repeat offenders, suggesting that piracy is not an isolated event but rather a structured, ongoing issue.
3. **Threshold-based access management:** We evaluate a simple access management strategy based on defining acceptable request limits per token and per user (Section 5). We demonstrate that this approach effectively detects a significant portion of inappropriate access without affecting legitimate users. We also contrast two different access control mechanisms—one based on IP volumetry per user and another based on request volumetry per user. Our findings indicate that analyzing the number of requests per user provides a more robust way of identifying suspicious behavior.

This work is an extension of our conference paper [Rodrigues *et al.*, 2024]. Among our original contributions, we 1) provide additional discussion on the analytical models (Section 4.1.2), 2) report a more comprehensive analysis of the time between requests using real data, including the analysis of its bursty behavior (Figures 4 and 5) and 3) present additional results about real token reuse in the wild (Section 5).

Outline. The remainder of this paper is organized as follows. Section 2 presents related literature, followed by our methodology in Section 3. Section 4 introduces models and Section 5 measurements. Finally, Section 6 concludes.

2 Related Work

In this section, we present some research lines related to the main themes covered in this paper.

Traffic characterization to CDNs. The security of CDN networks is the subject of numerous works, for example, focusing on statistical aspects related to anomaly detection or practical issues of system implementation, such as Akamai [Gillman *et al.*, 2015; Gonçalves *et al.*, 2020]. Traffic characterization is an important step towards increasing network security. Once "normal" traffic is characterized, one can also attempt to characterize anomalous traffic. While an advantage of characterizing both normal and anomalous traffic is improved accuracy in detecting threats, anomalies with previously unseen signatures may go undetected [Silveira *et al.*, 2010]. For this reason, our approach applies threshold-based methods to classify access attempts, focusing on detecting outliers. This way, we can detect anomalies without having access to ground-truth.

Piracy. The piracy problem in Brazil was studied in [Dent, 2020]. This is a multifaceted problem, involving legal, human, and technological aspects. In this work, we focus on technological aspects, accounting for system performance evaluation and monitoring of real systems. While [Dent, 2020] provides a broad understanding of piracy, including its economic impact, it does not delve deeply into automated detection mechanisms. In this work, we bridge that gap by 1) acknowledging the importance of economic factors (Section 4.2) and 2) proposing automated detection mechanisms (Section 5).

Piracy in streaming. Piracy in streaming has recently gained attention, with content providers prioritizing strategies to mitigate financial losses. In [Doërr, 2024], the discussion focuses on the tactics employed by video pirates to bypass protections, including token sharing. In [Simon and Doërr, 2024], the authors propose token mechanisms to prevent unauthorized access and misuse of CDN resources. These works contribute to the development of mechanisms that prevent unauthorized access. However, some of these solutions are prohibitively expensive for deployment in large-scale systems like the one considered in this study. Our approach builds upon the strategies proposed in [Doërr, 2024; Simon and Doërr, 2024], examining the problem from different perspectives and implementing practical, lightweight mitigation techniques.

Queuing models for detecting misuse of networks. In the context of covert communication, many works are dedicated to understanding the effect of misuse of resources [Jiang *et al.*, 2021]. In these cases, authors focus on the problem of resource theft without owners realizing the problem. In this paper, on the other hand, we primarily consider the effects of resource theft on legitimate users, who have their quality of service affected. The problem of detecting misuse of resources was considered, for example, in [Rufino *et al.*, 2020]. However, while in [Rufino *et al.*, 2020] and [Reza Ramtin *et al.*, 2022] the problem of denial of service was considered, in the present work we consider that pirates will affect the network, but not to the extent of generating denial of service, which would end up being undesirable for pirates as well.

Undue blockings. Among the numerous issues related to the problem of false positives when trying to detect piracy, one of them is distinguishing control traffic from data traffic [Piatek *et al.*, 2008]. In particular, network monitors that transmit information about tokens but do not receive video data need to be differentiated from pirate users. More generally, legitimate users who end up being classified as pirates, for example because they change their IP address several times in a short period, need to be protected. In this dissertation, we have shown with real data that, when filtering out IP changes, the vast majority of users classified as pirates correspond to a significant number of IP address changes in a short period, because the same token (legitimate user) is shared between several IP addresses (pirates). In future work, we intend to relate this data to information from the company's call center, to try to understand in which cases certain users were blocked erroneously.

Table 1 summarizes the main aspects of the related works discussed. While previous studies offer valuable insights into CDN security, piracy detection, and resource misuse, our approach integrates these perspectives, emphasizing real-world data analysis and adaptive access control mechanisms.

Table 1. Comparison of related works

Work	Traffic Analysis	Piracy Analysis	Access Control	Impact on QoS	False Positive Analysis
[Gillman *et al.*, 2015] [Gonçalves *et al.*, 2020]	✓	×	×	×	✓
[Dent, 2020]	×	✓	×	×	×
[Simon and Doërr, 2024] [Doërr, 2024]	×	✓	✓	×	×
[Jiang *et al.*, 2021] [Rufino *et al.*, 2020]	×	×	×	✓	✓
[Piatek *et al.*, 2008]	×	✓	×	×	✓
Proposed solution	✓	✓	✓	✓	✓

3 Methodology

Our methodology is structured into two main parts:

1. **Modeling**: We will first describe the models used to analyze the token usage, request patterns, and detect suspicious behaviors.
2. **Evaluation**: Then, we apply these models to real-world data and evaluate their effectiveness in detecting anomalies and suspicious activity, such as piracy.

3.1 Traffic Modeling

We employ queuing models to analyze the impact of piracy on CDN performance. Our baseline model assumes an M/M/1 queue to represent legitimate user traffic under normal conditions. To incorporate the effects of piracy, we introduce bursty arrival patterns, simulating token-sharing scenarios where multiple users illegitimately access the same content. Additionally, we explore mitigation strategies such as quality degradation for flagged users.

3.2 Evaluation of Threshold-Based Classification

To detect piracy attempts, we establish statistical thresholds based on access patterns observed in CDN logs. We analyze token reuse frequency, request volumes per user, and session duration to identify unusual behaviors indicative of abuse. A key challenge is differentiating between legitimate users with dynamic IP addresses and actual piracy attempts. To address this, we define a threshold-based classification system where users exceeding a predefined number of IP changes or request volumes are flagged as suspicious.

For threshold determination, we use a standard deviation-based approach, ensuring that only extreme deviations from normal access patterns are classified as anomalies. The goal is to minimize false positives while maintaining high detection accuracy.

3.3 Summary

Our methodology integrates analytical modeling, threshold-based classification, and empirical validation to develop an effective framework for detecting piracy in CDNs. The combination of theoretical analysis and empirical evidence aims to provide a comprehensive perspective towards mitigating piracy while preserving the quality of service for legitimate users. The subsequent sections will provide detailed explanations of the steps followed in both the modeling and evaluation stages.

4 Models

In this section, we consider models to capture system performance before and after adopting countermeasures.

4.1 Problem setup and queuing models

Without countermeasures, we face the scenarios illustrated in Figure 1. In the initial scenario (Figure 1(a)) we have legitimate requests being sent to the CDN that responds with content. In the intermediate scenario (Figure 1(b)), we have malicious users via piracy providers generating requests for content that is served by the original content provider's own CDN. Finally, in the last scenario (Figure 1(c)), we have malicious users via piracy providers generating requests that are served by an infrastructure maintained by those piracy providers.

To capture the 3 scenarios, we use queuing models. In this work, we consider the simplest models, aiming to quantify the effects of the three types of pirated content sharing. Therefore, in the original scenario, we considered an M/M/1 queue. In the intermediate scenario, we consider an M/M/1 queue with arrivals in bursts (*batch arrivals*), assuming that each request corresponds to the service of b users (for example, b pirate users). Finally, in the last case, we assume that each request also corresponds to the service of b pirate users. However, if some legitimate users switch to the pirate infrastructure, this may reduce the burden on the legitimate CDN. To capture this phenomenon, we assume that the arrival rate now becomes λ'. Unless otherwise stated, we assume that $\lambda' = \lambda/b$. Therefore, the arrival rate of requests to the CDN in the scenario in Figure 1(c) is equal to λ/b, but the service rate to the users, taking into account the hacker infrastructure, is equal to $\lambda = b \cdot \lambda/b$.

Note that the scenarios in Figures 1(a) and 1(c) are distinguished only by the fact that in the first the arrivals follow a Poisson flow, while in the last they follow a burst flow. Although this model corresponds to a specific instantiation of the problem in question, in which the total rate of users

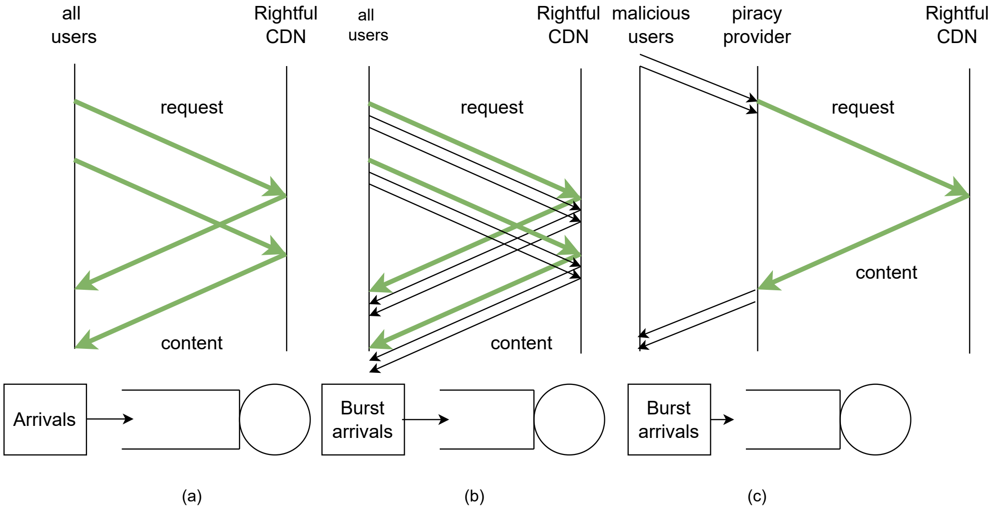


Figure 1. Illustration of the 3 scenarios considered: (a) standard operation; (b) hackers using the CDN for pirated content replication; (c) hackers using their infrastructure to replicate content

served per unit of time remains constant between the scenarios in Figures 1(a) and 1(c), we believe that it already serves as a first step to illustrate the effect of hackers on system performance, taking into account both the legitimate CDN network and the hackers' infrastructure.

Assume that the bursts arrive according to a Poisson flow with rate λ, and each service of each request takes exponentially distributed time with mean $1/\mu$. The average waiting time in an M/M/1 queue with burst arrivals of size b is given by [Ghimire *et al.*, 2014], $T = (1+b)/(2\mu(1-\rho))$ where $\rho = \lambda/\mu$. In the three considered scenarios, the waiting time for customers will be given by the following expressions, respectively,

$$T_{orig} = \frac{1}{\mu(1-\rho)}, \quad (1)$$

$$T_{CDN} = \frac{1+b}{2\mu(1-\rho)}, \quad (2)$$

$$T_{infra} = \frac{1+b}{2\mu(1-\lambda/(b\mu))}. \quad (3)$$

Notation is summarized in Table 2.

4.1.1 Numerical example

Figure 2 illustrates the behavior of the model. In Figure 2 we consider $\lambda = 0.5$, $\mu = 0.6$ and $b = 1$ as our reference scenario. The three systems illustrated in this figure describe the three scenarios listed above, represented by the full (blue), hatched (red), and dotted (black) curves, respectively. As the system's service capacity increases, that is, as μ grows between 0.6 and 4, the difference between the systems' performance decreases. In particular, when $b = 2$ or $b = 4$, the difference between the systems when $\mu = 4$ is practically imperceptible. If the system is over-provisioned, even with the presence of pirates, the system performance remains practically unchanged. Now consider the increase in the number of pirate users per legitimate request. When b increases between 2, 4, and 10, the difference between the red, blue, and black curves increases. This corresponds to the fact that the greater the number of pirate users, the greater the load on the CDN or pirate infrastructure. It is also worth noting that according to the model considered, performance with pirated infrastructure is generally (but not always) between the performance of the original system and the performance of the system that makes exclusive use of the CDN. The model captures the intuition that pirated infrastructure will compensate for overloads on the target CDN.

4.1.2 Comparing scenarios

It follows from Equations (1) to (3) that

$$\max(T_{orig}, T_{infra}) \le T_{CDN}.$$

This means that the worst scenario corresponds to hackers using the infrastructure of the CDN to share illegal copies of the content. It remains to compare T_{orig} and T_{infra}. After algebraic manipulation, it can be verified that $T_{orig} < T_{infra}$ if the following condition is met

$$\frac{1}{\mu-\lambda} < \frac{1+b}{2\mu - 2\lambda/b} \quad (4)$$

which is equivalent to

$$\mu > \lambda \cdot \frac{b+1-2/b}{b-1}. \quad (5)$$

For $b = 2$, the above condition translates to

$$\frac{\lambda}{\mu} < \frac{1}{2}. \quad (6)$$

Indeed, in the example of Figure 2(a) we have $\lambda = 1/2$, hence $T_{orig} < T_{infra}$ for $\mu > 1$, i.e., the curves for T_{orig} and T_{infra} cross at $\mu = 1$. In Figures 2(b) and 2(c), the crossings occur at $\mu = 3/4$ and $\mu = 3/5$, respectively. The derivative of the right-hand side of Eq. (5) with respect to b is always negative, $\frac{d}{db}\frac{b+1-2/b}{b-1} = -2/b^2$, which means that as b grows

Table 2. Notation Table

Symbol	**Description**
	Model
λ	Arrival rate of legitimate users' requests to the CDN.
μ	Service rate of the CDN per request.
ρ	System load, defined as $\rho = \lambda/\mu$.
b	Number of pirate users per legitimate request (token-sharing factor).
T	Average waiting time for a request to be served.
T_{orig}	Average waiting time in the original system (M/M/1 queue without piracy).
T_{CDN}	Average waiting time when pirate users request content directly from the CDN.
T_{infra}	Average waiting time when pirate users access content through an external piracy infrastructure.
p_l	Price of the legal streaming service.
p_i	Price of the illegal streaming service.
β	Fraction of the legal service price charged by the illegal service ($p_i = \beta p_l$).
α	Weight parameter balancing the impact of price and quality on user cost.
C_l	Total cost for a user subscribing to the legal service ($C_l = \alpha T_l + p_l$).
C_i	Total cost for a user subscribing to the illegal service ($C_i = \alpha T_i + p_i$).
P_l	Limiting legal price, below which it is more advantageous for users to choose the legal service.
q	Fraction of legitimate users in the system.
	Evaluation
r	Average number of requests (chunks) per user (token) in a window.
σ	Standard deviation of the number of requests (chunks) per user (token) in a window.
R	Actual number of requests (chunks) per user (token) in a given observation window.

the threshold value for μ above which $T_{orig} < T_{infra}$ decreases, in agreement with Figure 2.

The reasoning behind why $T_{orig} < T_{infra}$ for sufficiently large values of μ, but $T_{orig} \geq T_{infra}$ otherwise, is as follows. When transitioning from the original system to one where hackers leverage their own infrastructure to serve illegal copies of the content, two key factors come into play:

1. the burst arrival rate to the CDN decreases, but, according to our model,
2. the request rate within each burst increases, ensuring that the overall rate of requests per time unit to the CDN remains the same in both the original setup and the hacker-assisted setup.

For lower values of μ, the effect of the reduced burst arrival rate (point 1) is more significant, so the hackers' offloading infrastructure decreases the waiting time for legitimate users. However, as μ increases and the system becomes better provisioned, the second effect, i.e., the increased request rate within each burst (point 2) becomes dominant. In this case, the burst arrivals increase, leading to a degradation in user experience compared to the original system.

4.2 Problem mitigation

One of the simplest countermeasures against piracy is to degrade the quality of service (QoS) of pirate users, to such an extent that it is not worth paying for the pirated service. If it is better to use the original service, rather than the pirated one, there will be no reason to use the pirated service. Degrading QoS is a milder and gentler countermeasure against piracy than simply banning any pirate accounts, especially taking into account false positives.

Let p be the price of the service and T be the average service time for each request. Let C be the user cost, which depends on QoS and price, with the cost associated with QoS being given by the service time,

$$C = \alpha T + p, \tag{7}$$

where α is a constant that balances between the weight of price and QoS. The idea of the model is to capture the fact that the pirate system has lower p but higher T. Let p_i and T_i be the price and service time of the illegal system, and p_l and T_l be the price and service time of the legal system. If the following condition is met, it is worth using the legal system,

$$C_l \leq C_i \Rightarrow \alpha T_l + p_l \leq \alpha T_i + p_i. \tag{8}$$

Assuming that the illegal price is a fraction β of the legal price, $0 \leq \beta \leq 1$, $p_i = \beta \cdot p_l$, the condition is given by

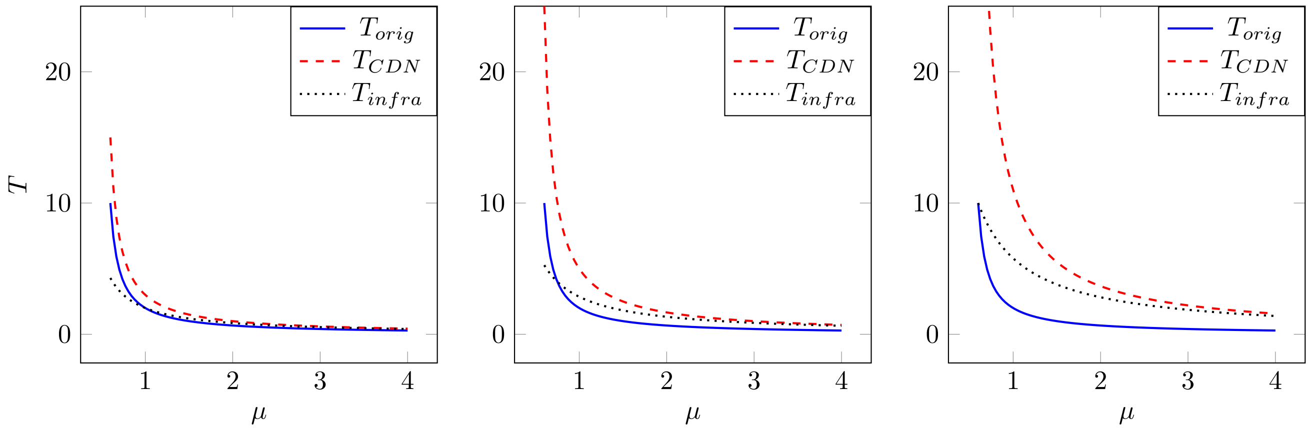
$$p_l \leq \alpha \frac{T_i - T_l}{1 - \beta} = P_l. \tag{9}$$

We refer to the quantity on the right, P_l, as the limiting legal price. If the legal price is below this threshold, it is worth using the legal service.

In the simplest case, we have two queues, with the priority queue serving requests with priority, and without preemption, from customers classified as authentic. In this case, assuming all exponentially distributed services, we have [Harchol-Balter, 2013],

$$T_i = \frac{\rho/\mu}{(1-\rho_l)(1-\rho_l-\rho_i)}, \qquad T_l = \frac{\rho/\mu}{1-\rho_l} \tag{10}$$

where $\rho = \rho_l + \rho_i$, $\rho_l = q\lambda/\mu$, $\rho_i = (1-q)\lambda/\mu$, and q is the fraction of services corresponding to legal customers.



(a) $b = 2$ (b) $b = 4$ (c) $b = 10$

Figure 2. Waiting time experienced by clients in the case of a system with 2, 4, or 10 pirate clients for each legal request.

Figure 3 illustrates the legal price threshold P_l below which it is worth using the legal service (Figure 3(a) varying q and Fig. 3(b) varying μ). To generate the curve, we use $\rho = \lambda = 0.5$, $\mu = 1$ and $\alpha = 2$. We note that as the fraction of legal users, q, increases, the overhead on the illegal users increases and, according to the proposed model, the QoS of pirate users degrades more than the QoS of legal users. Thus, the minimum price below which it is worth paying for the legitimate service increases. Alternatively, as β increases, the price threshold also increases. After all, if the pirated service becomes more expensive, the minimum amount below which it is worth paying for the original service also increases. Finally, as μ increases, Figure 3(b) shows that users have less incentive to use the legal service, as the illegal alternative will already provide similar QoS.

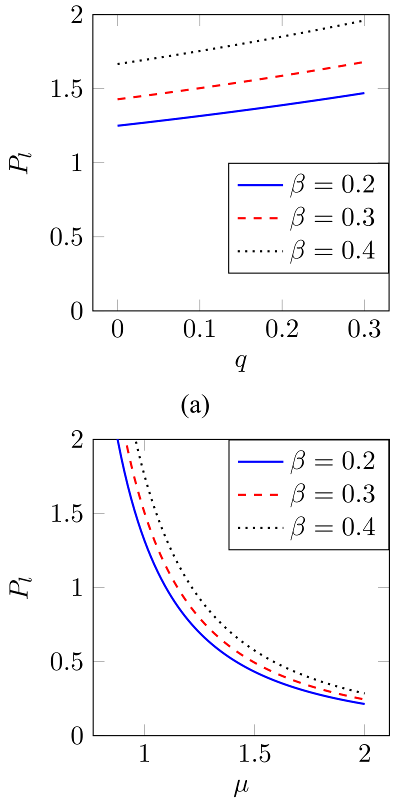
5 Evaluation

In Section 5.1 we discuss practical observations on how to evaluate abusive consumption. Then, we consider two measurement campaigns, in Sections 5.2 and 5.3:

- The first measurement campaign contains data from two servers across one day. It leverages raw data collected from all users accessing the CDN through those two machines. Although the analysis accounts for only two machines, the fact that it uses raw data makes it less scalable than the analysis in our second measurement campaign.
- The second measurement campaign contains data from all CDN, across eight days. It leverages summaries about access patterns per machine. We collect data from each machine and then aggregate, filter, and analyze this aggregated and filtered data to decide, for instance, which users will or will not be blocked.

5.1 How to evaluate abusive consumption in practice?

Next, we present some relevant observations from a practitioner's standpoint on how to evaluate abusive consumption



(a)

(b)

Figure 3. Legal price threshold P_l below which the legal service is worthwhile

in practice:

- Sessions that share tokens are not distinguishable through traditional monitoring tools, i.e., users running different streaming sessions but using the same token may not be distinguishable from the standpoint of certain monitoring tools.
- Sessions do not require additional authentication from a central entity to play videos once they have access to tokens, i.e., once a user has a URL with an access token, the user can play the video from anywhere, as long as this token is valid.
- A video server can be affected by a single, indiscriminately distributed token.

The video delivery process is summarized as follows:

1. A user presses a button in the video player to start watching content.
2. The video player requests, through an API, a URL for that content, containing a token for authentication. If the user has the right to watch that content, the user should be able to receive the URL with the token.
3. In possession of this URL, the video player issues multiple requests for small chunks of video.

In the considered pirate token-sharing scenario of Figure 1(b), for each legitimate user the above steps are repeated b times, one time per pirate user behind each legitimate user. All b requests will leverage the same *access token* but will correspond to different destinations.

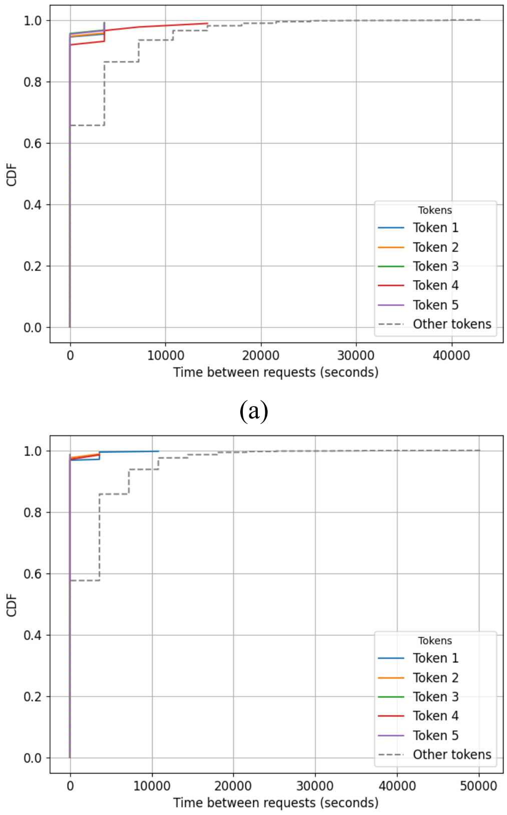
5.2 Micro-benchmarks: real data from two machines across one day

We analyzed the data from the Content Delivery Network (CDN) using a combination of statistical tests and visual analysis methods, relating our results to the considered analytical models introduced in the previous sections. We identified usage patterns by examining the cumulative distribution functions (CDF) of the times between requests and exploring different scenarios. We also applied the Kolmogorov-Smirnov (KS) test, which is commonly used to assess whether data conforms to a theoretical distribution [Berger and Zhou, 2014], and the Burstiness Ratio, often utilized in network traffic studies to identify bursts within the system [Jiang and Dovrolis, 2005].

For this analysis, we focused on the access logs for one day from two CDN machines. The day was selected based on its large audience and because it coincided with an event considered a piracy target by the CDN. The chosen servers are representative of the CDN's overall behavior, demonstrating similar usage patterns and configurations compared to other machines in the network. Furthermore, they serve different types of users, reflecting the variability in the CDN's traffic.

5.2.1 How are tokens reused?

We refer to a request for a token that is already in use as a token reuse. In this section, we focus on requests that correspond to token reuses. If each token were used by a single user, we would expect few reuses, e.g., due to network failures.



(a)

(b)

Figure 4. CDF of time between requests per token on server 1 (a) and server 2 (b).

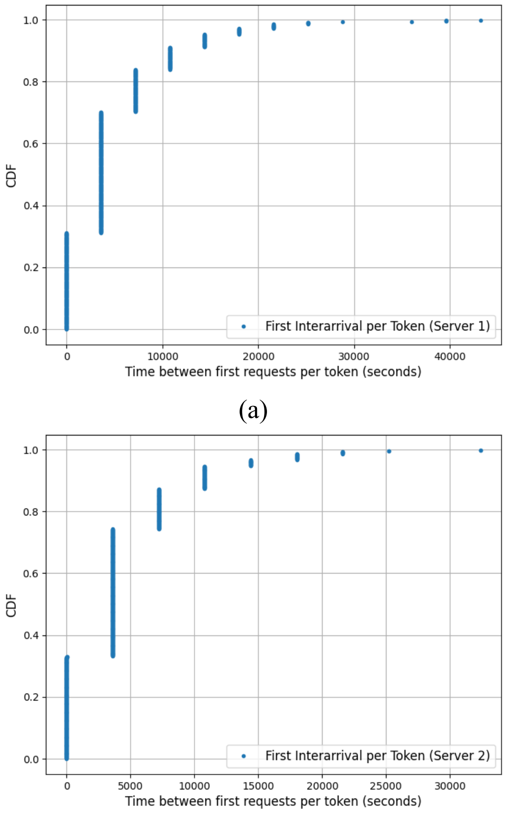
Figure 4 illustrates the cumulative distribution function (CDF) of the time between requests *per token*. It emphasizes the five most accessed tokens, each represented by a distinct line, while the remaining tokens are grouped together and shown as a single dashed line. This arrangement simplifies the comparison of usage patterns between the most popular tokens and the others, enabling us to observe the usage trends of each token over time. In a scenario reflecting legitimate usage, we would expect each token to be used by a single user.

If the use of each token were exclusive to one user, the intervals between requests for the same token should not cluster in short intervals. Even in scenarios involving retries due to failures—typically limited to a maximum of three attempts—these would not significantly impact the curve. Accordingly, we would expect the CDF in this case to not reach high values quickly.

However, when we look at Figure 4, we notice a rapid rise to values close to 1 for very short times between requests, which may indicate bursts, where there are several requests for the same token with very short time intervals between them. In addition, if it were an exponential distribution, what would be expected would be a smooth increase to high values, which is not the case as the CDF goes to high values very quickly. This concentration of short intervals in the distribution of times between requests for the same token is an indication that the same token is accessed repeatedly in extremely close periods. This scenario is not only justified by

Table 3. Approaches for blocking tokens, remembering that each token is associated with a session, and is contained within a URL

	Metric adopted	Blocking	Problem
Approach 1	IPs per user	tokens	there are still (few) false positives left
Approach 2	requests by IP by user	tokens	to be determined



(a)

(b)

Figure 5. CDF of time between first requests for each token on server 1 (a) and server 2 (b).

retries, but is consistent with token sharing and the theoretical model proposed.

In this scenario, the KS test yielded statistical values of 0.6940, for Server 1, and 0.6458, for Server 2. Both servers had a p-value very close to zero, suggesting a rejection of the hypothesis that the data follows an exponential distribution. Additionally, the Burstiness Ratio was above 1, with values of 2.08 for Server 1 and 1.85 for Server 2, indicating a bursty behavior.

5.2.2 How are new tokens issued over time?

Next, we analyze the distribution of token demand over time. We focused on how different tokens are issued, constructing the CDF of the time intervals between the first requests for each token. In Figure 5, we observe a rapid upward trend for both servers, indicating that a significant proportion of the time intervals between requests are short. This suggests a concentrated demand for tokens at specific times.

The results of the KS test showed a statistic of 0.7022 with a p-value of roughly zero for server 1 and a statistic of 0.7437 with a p-value of roughly zero for server 2. These findings reject the hypothesis that the time intervals follow an exponential distribution. Additionally, the Burstiness Ratios were 1.184 for server 1 and 1.112 for server 2, indicating slight variability in the arrival patterns. Together with the KS test results and graphic analysis, this indicates that peaks in demand for new tokens occur with a certain frequency, likely reflecting periods of increased user activity, as we suspected.

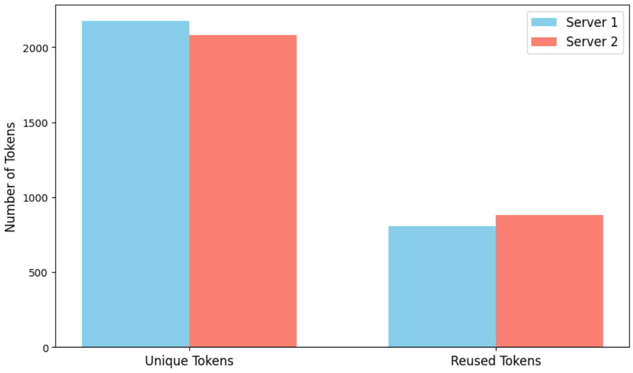


Figure 6. Comparison of unique vs. reused tokens

5.2.3 How the issuing of new tokens compares against token reuse?

Figure 6 shows a comparison of usage between unique tokens and reused tokens. In general, we observe that unique tokens tend to be much more prevalent than reused tokens.

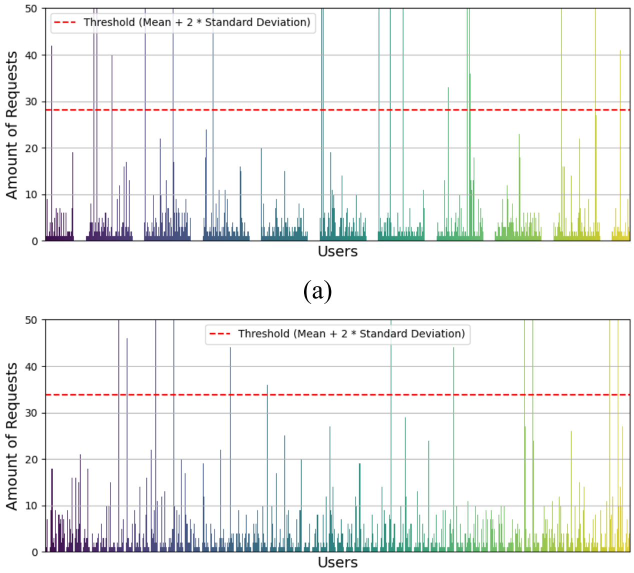
The difference between the number of unique and reused tokens between the two servers influences the CDFs observed above. The fact that server 2 has more reused tokens and fewer unique tokens may explain the difference between the previous CDFs, considering one server and the other. It is also worth noting that server 2 appears to be slightly more susceptible to pirates, as it had more reused tokens and less unique tokens.

5.2.4 Takeaway message

We observed that bursts of token requests, indicating token reuse, are common in real data. The link between request bursts and illegal sharing is a fundamental component of the analytical models discussed in the previous section. While we cannot conclusively prove that all request bursts are due to illegal sharing, their presence suggests that analyzing their impact is crucial for assessing the performance of access control in CDNs. In this section, our focus was on token requests, while in the following sections, we shift our attention to requests for content chunks per token, per user.

5.2.5 Analyzing usage threshold per token and user

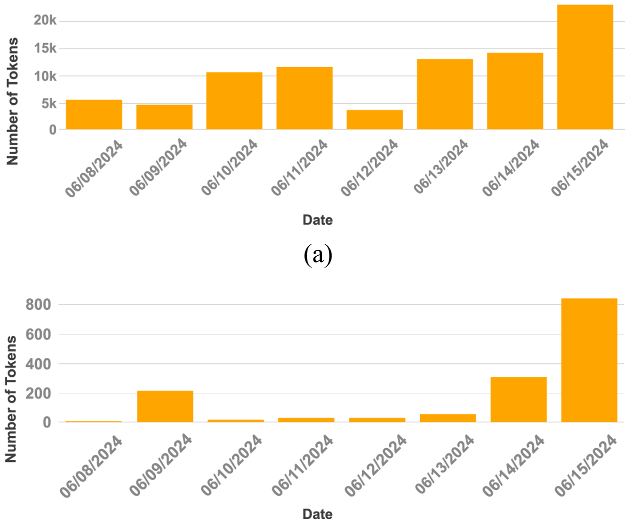
Using data from the same day as in the previous analysis, we analyzed the number of requests for content chunks made by users on two servers. While in the analysis so far we only considered requests for tokens, also referred to as requests



(a)

(b)

Figure 7. Amount of requests per user on (a) server 1 and (b) server 2.



(a)

(b)

Figure 8. Suspicious tokens per day (a) all cases and (b) non-recurring cases.

for master playlists, ignoring the chunks, in this section we consider all requests per token, including those for chunks.

Our goal was to understand if there is a standard threshold for the number of requests for content chunks per user and to identify any users who exceed this threshold, as their behavior may be considered suspicious. To define this threshold, we applied the following empirical rule [Freedman *et al.*, 2007; Triola, 2018; Bluman, 2017]:

$$\begin{cases} R > r + 2 \cdot \sigma, & \text{anomaly detected,} \\ \text{otherwise,} & \text{normal behavior} \end{cases} \qquad (11)$$

where r is the mean and σ is the standard deviation of the number of requests for chunks per token, and R is the actual number of such requests. The thresholds identified were 28.08 for server 1 and 33.88 for server 2. Such similar values observed suggest that there may be a consistent pattern in the normal request behavior for each user. We observed similar behavior for other servers, noting that it would be helpful to analyze this threshold across servers in other CDNs in future studies to gain a better understanding of this pattern under different settings.

In Figure 7 the red dashed line represents the threshold discussed in the previous paragraph. It can be seen that in both images, most users have a number of requests below this threshold, indicating that this calculation seems adequate to capture what would be standard behavior. However, there are some users with values substantially higher than this threshold, which is an indication for a more detailed analysis to understand this behavior, which may indicate content abuse from pirates, and also to understand how simple access control measures can help mitigate these users across the whole CDN, which is the subject of the next section.

5.3 Bird's-eye view: real data from all CDN across eight days

In this section, we illustrate simple access control strategies using data from all servers of a national CDN that serves millions of users. We are focusing on the scenario shown in Figure 1(b), where we aim to block users who are suspected of misusing the legitimate CDN infrastructure. To achieve this, we use two approaches. First, we analyze the number of unique IPs associated with each user (Section 5.3.1). Users linked to many IPs are flagged as suspicious. However, it's important to note that multiple IPs can be legitimate, for example, when a user switches networks. To address this, we also examine the number of requests made by each user (Section 5.3.2). A summary of the considered approaches is presented in Table 3.

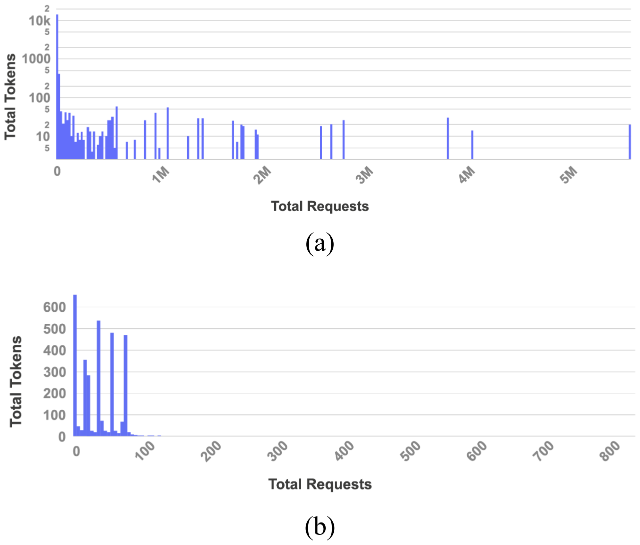
5.3.1 Analysis of suspicious consumption occurrences via volumetry: IPs per user

A token is considered suspicious if it violates certain rules. In this section, we consider rules based on the number of IPs associated with a given token: if a token is associated with many IPs, it is tagged as suspicious.[2] Assuming that the classification has already taken place, the operator can analyze the data related to cases classified as suspicious to understand whether the rules applied are too permissive or too rigid.

As a starting point, we consider statistics related to the detection of suspicious users. Taking as an example the week from 06/08/2024 to 06/15/2024, we characterize the quantiles of the distribution of times that each suspicious user was tagged as such. The 0.64 quantile equals 1, meaning that 64% of users identified as suspects were identified as such only once. In contrast, the 0.66, 0.8, 0.95, 0.99 and 1 quantiles equal 2, 3, 59, 663, and 1397, meaning that some users were identified as suspects hundreds or even more than one thousand times. The above numbers suggest that users tagged as suspicious only once may correspond to false positives, but that others are clearly anomalous.

Next, we consider a longitudinal analysis of the recurrence

[2] The rules will not be exposed in detail in this paper to preserve the confidentiality of the company where the study was carried out.



(a)

(b)

Figure 9. Concentration of tokens by number of requests in (a) suspects and (b) all data.

of suspicious users. Figure 8(a) shows the number of tokens tagged as suspicious, and Figure 8(b) shows the subset of those tokens that are non-recurring. Note that the y axis in Figure 8(a) varies in the range of thousands of tokens, whereas in Figure 8(b), it is in the range of hundreds. So, it is clear that the group of suspects with recurrence is responsible for the largest portion of tokens classified as suspicious.

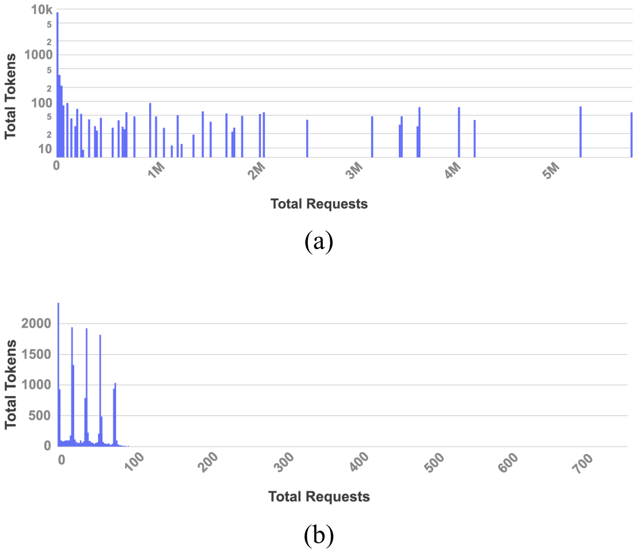
The above results indicate that in the considered week, recurring cases were prevalent. However, analyzing the period from October 2022 to October 2023 we identified that approximately 80% of users that were once classified as suspect were never reclassified. Those users may correspond to false positives, motivating new classification strategies based on the volume of requests per user as opposed to the number of IPs associated with a given token, as further detailed next.

5.3.2 Analysis of occurrences of suspected consumption based on consumption patterns: volumetry of requests per user

Next, we consider the distribution of requests for video chunks. We compare a standard sample distribution with the cases that are flagged as suspects. The data is sampled from 1 out of every 100 users, regardless of whether they have been flagged as suspicious or not, within a specified time window.

Taking the channel with the highest audience as an example, as shown in Figure 9(a), the majority of tokens made below 1M requests, i.e., the mode of the number of requests per token is less than 1M. However, there is a group of tokens that significantly deviates from this threshold. In particular, the tokens with many more requests could be from malicious users or problematic connections, in which multiple attempts exist to access the content. Both cases deserve attention, as they may signal abuse of tokens or a network connection problem. Distinguishing between the two cases is key for determining when and if to block users.

Whereas Figure 9(a) shows all requests marked as suspicious, Figure 9(b) shows a sample from all the data, irrespective of its tag. In this sample, we typically observe a much smaller number of requests per token. This indicates that the number of requests from suspect cases is higher than usual, suggesting that analyzing the request pattern per token can aid in identifying suspicious cases.



(a)

(b)

Figure 10. Concentration of tokens by number of requests for content with a smaller chunk size (compared to Figure 9) in (a) suspicious cases and (b) all cases.

Each content has a different consumption pattern. This is illustrated, for example, through the analysis of an alternative content that, in particular, has a smaller chunk size than the one analyzed in the previous figure. By observing this additional content, we notice that, as in the previous case, Figures 10(a) and Figures 10(b) exhibit very different patterns, allowing us to distinguish between typical and anomalous users. However, given the specificity of this new content, which has a smaller chunk size, it is expected that the typical number of requests will differ from that necessary to consume the previous content. Indeed, comparing Figures 9 and 10 we note that the typical values of total requests per token vary at different ranges. Those results indicate that one needs to analyze the specific consumption patterns per content, e.g., accounting for chunk sizes, to distinguish anomalous traffic.

6 Conclusion

We evaluated minimally intrusive access management to CDNs based on measured access patterns and queuing models. As presented earlier, we had two objectives: to find ways to estimate the costs associated with content consumption and to evaluate measures that could identify suspicious users and mitigate their actions without affecting legitimate users in a real-use scenario. With this in mind, we sought to find measures that could protect the CDN infrastructure without affecting the consumption experience of legitimate users, guaranteeing the efficiency and stability of content delivery from the CDN.

To achieve the first objective, we proposed using the M/M/1, M/M/1 with burst arrivals and M/M/1 with priorities, as a strategy for clarifying aspects present in the performance of systems in different usage scenarios linked to piracy. In addition, we explored the application of these models in a controlled manner, given the large volume of data to be analyzed, so that we could validate the theoretical propositions against their real application scenario.

To achieve the second objective, we analyzed access management strategies in a CDN that serves millions of users. From this analysis, it was possible to discover that simple policies that meet the need not to affect the experience of legitimate users are effective in detecting undue consumption. It is important to note that we identified two groups of interest among the users classified as suspicious by these policies: users classified only once, which could mean a false positive, and recurring users, responsible for the largest share of *tokens* identified in sharing. Although these results were satisfactory, the identification of these groups opens up room for improvement in this classification and this could be a point to explore in future work, e..g, considering the use of machine learning to find suspicious access patterns.

Acknowledgements

This project was sponsored by CAPES, CNPq, and FAPERJ (315110/2020-1, E-26/211.144/2019, and E-26/201.376/2021).

Declarations

Authors' Contributions

LR and DM contributed to the conception of this study. LM performed the experiments and evaluations. LR, DM, AR and AS are this manuscript's main contributors and writers. All authors read and approved the final manuscript.

Competing interests

The authors declare that they do not have competing interests.

Availability of data and materials

Due to the sensitive nature of the information, the specifics of the dataset cannot be shared publicly. This policy helps protect user privacy and ensures the integrity of the CDN's operations, while still allowing for analysis and insights to be derived from the data.

References

Berger, V. W. and Zhou, Y. (2014). Kolmogorov–Smirnov Test: Overview. In *Wiley StatsRef: Statistics Reference Online*. John Wiley & Sons, Ltd. DOI: 10.1002/9781118445112.stat06558.

Bluman, A. G. (2017). *Elementary Statistics: A Step By Step Approach*. McGraw-Hill Education, 10th edition. Book.

Dent, A. S. (2020). *Digital pirates: Policing intellectual property in Brazil*. Stanford University Press. Book.

Doërr, G. (2024). Digital flamenco with video pirates. In *Proceedings of the 2024 ACM Workshop on Information Hiding and Multimedia Security*, MMSec '24, page 1–2, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3658664.3659663.

Fett, D., Campbell, B., Bradley, J., Lodderstedt, T., Jones, M., and Waite, D. (2023). RFC 9449: OAuth 2.0 Demonstrating Proof of Possession (DPoP). DOI: 10.17487/RFC9449.

Freedman, D., Pisani, R., and Purves, R. (2007). *Statistics*. W.W. Norton & Company, 4th edition.

Ghimire, S., Ghimire, R., *et al.* (2014). Mathematical models of Mb/M/1 bulk arrival queueing system. *Journal of the Institute of Engineering*, 10(1):184–191. DOI: 10.3126/jie.v10i1.10899.

Gillman, D., Lin, Y., Maggs, B., and Sitaraman, R. K. (2015). Protecting websites from attack with secure delivery networks. *Computer*, 48(4):26–34. DOI: 10.1109/MC.2015.116.

Gonçalves, C. F., Menasché, D. S., Avritzer, A., Antunes, N., and Vieira, M. (2020). A model-based approach to anomaly detection trading detection time and false alarm rate. In *MedComNet*, pages 1–8. IEEE. DOI: 10.1109/MedComNet49392.2020.9191549.

Harchol-Balter, M. (2013). *Performance modeling and design of computer systems: queueing theory in action*. Cambridge University Press. Book.

Jiang, B., Nain, P., and Towsley, D. (2021). Covert cycle stealing in a single fifo server. *ACM Trans. Modeling and Performance Eval. Computing Systems*, 6(2):1–33. DOI: 10.1145/3462774.

Jiang, H. and Dovrolis, C. (2005). Why is the internet traffic bursty in short time scales? *SIGMETRICS Perform. Eval. Rev.*, 33(1):241–252. DOI: 10.1145/1064212.1064240.

Patat, G., Sabt, M., and Fouque, P.-A. (2022). WideLeak: How Over-the-Top Platforms Fail in Android. In *2022 52nd Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, pages 501–508. DOI: 10.1109/DSN53405.2022.00056.

Piatek, M., Kohno, T., and Krishnamurthy, A. (2008). Challenges and directions for monitoring p2p file sharing networks-or: why my printer received a dmca takedown notice. In *Proceedings of the 3rd conference on Hot topics in security*, pages 1–7. Available at:`https://www.usenix.org/legacy/event/hotsec08/tech/full_papers/piatek/piatek.pdf`.

Reza Ramtin, A., Nain, P., Menasche, D. S., Towsley, D., and de Souza e Silva, E. (2022). Fundamental Scaling Laws of Covert DDoS Attacks. *ACM SIGMETRICS Performance Evaluation Review*, 49(3):20–21. DOI: 10.1145/3529113.3529120.

Rodrigues, L. M. V., Menasché, D. S., Serra, A., and de Aragão Rocha, A. A. (2024). Minimally intrusive access management to content delivery networks based on performance models and access patterns. In *8th International Symposium on Cyber Security, Cryptology and Machine Learning (CSCML 2024)*. DOI: 10.1007/978-3-031-76934-4$_1$2.

Rufino, V. Q., Nogueira, M. S., Avritzer, A., Menasché, D. S., Russo, B., Janes, A., Ferme, V., Van Hoorn, A., Schulz, H., and Lima, C. (2020). Improving predictability of user-affecting metrics to support anomaly detection in cloud services. *IEEE Access*, 8:198152–198167. DOI: 10.1109/ACCESS.2020.3028571.

Silveira, F., Diot, C., Taft, N., and Govindan, R. (2010). Astute: detecting a different class of traffic anomalies. *SIGCOMM Comput. Commun. Rev.*, 40(4):267–278. DOI: 10.1145/1851275.1851215.

Simon, G. and Doërr, G. (2024). Next-generation access tokens to fight CDN leeching. MHV, page 111–112, New York, NY, USA. Association for Computing Machinery. DOI: 10.1145/3638036.3640276.

Triola, M. F. (2018). *Elementary Statistics*. Pearson, 13th edition. Available at:`https://fgsalazar.net/pdf/TRIOLA/TRIOLA-Chap01C.pdf`.