School of Computer Sciences

Universiti Sains Malaysia

Academic Session 2023/2024

Semester 1

CMT321: Management & Engineering of Databases

Assignment 1

Group 7

Name	Matric Number	USM Student Email
Angel Ang	159229	angelang27@student.usm.my
Ching Jia Ying	153463	jiaying01@student.usm.my
Gwee Per Ming	159372	perming@student.usm.my
Lim Chin Feng	157968	limchinfeng@student.usm.my
Yeong Yen Ting	152664	yenting2001@student.usm.my

Lecturer: Ts. Dr. Zainab Ajab Mohideen

Date of Submission: 4 December 2023

Table of Contents

Part 1:

1.1 Introduction

A database system is an integrated collection of related files and information for analysing the data contained within them. The database system is a computer-based documentation system to record and maintain information/data. (Gunjal,2003) In the context of a database, a database management system (DMBS) is aimed to handle the database with an efficient and organised method of describing, keeping, and accessing information in the database. In this case, DBMS is defined as a software application that provides access to data stored in a database. The DBMS interfaces with the application programs so that multiple applications and users can use the data contained in the database. In addition, the DBMS exerts centralised control over the database, prevents fraudulent or unauthorised users from accessing the data, and ensures data privacy.

Social Media platforms such as X (previously known as Twitter) and Facebook need to process a ton of real-time data, which has caused database management issues based on the event detection survey. (Hasan,2017) They have the demand to serve millions of active users, which costs them high usage and complexity of DBMS to handle their database efficiently. (Hellemans,2020) Thus, stream processing technologies are implemented to process and analyse their data rapidly based on user behaviour and trends. It can protect X and facebook from uninterrupted services and server down when providing services to their customers.

In terms of the technical aspect of DBMS on processing technologies, X implemented an event-driven paradigm with Hadoop to achieve the transactional requirements and deal with the highly parallelized system. This is because this approach is well-fit for batch and real-time processing. On the other hand, Facebook acquired a logging-centric strategy with Apache Hadoop that can provide optimised concurrency control and recovery mechanisms. Two different approaches of X and Facebook focused on managing their massive amount of real-time data with the high-end component and most suitable strategies of DBMS for stream processing technology.

1.2 Motivation

X, which is the latest name of Twitter, and Facebook are well-known social media platforms that allow users to share their stories and other content via posts or tweets. For storing real-time data, X and Facebook both use MySQL, a centralised database, and stream processing technology to process the data in real-time. (Pandey, R., Singh, A., Kashyap, A., & Anand, A.,2019) Stream processing enables the real-time processing of data, facilitating the analysis and processing of data within a short timeframe and identifying the trends and patterns in user behaviour by leveraging large volumes of data. (Saurabh, 2022) According to the analysis by team Kepios in October 2023, Facebook has 3 billion monthly active users, almost five times as much as X, which has 6.6 million monthly active users. With such a large amount of monthly active users, both Facebook and X need to have good stream processing technology to prevent the server down.

Hence, we will be analysing the technical aspects of the database management system (DBMS) on stream processing technology especially the transaction management and the architecture as well as the concurrency control of the transaction.

1.3 Related Works of Stream Processing and Real-time Analytics

There are a few researchers that study about the processing technologies in social media. In the research by F Hamami and I A Dahlan, they stated that there are many companies that use data streams to collect information in real time. In fact, there are different kinds of data from various sources such as, social media, Internet of Things and e-commerce. The paper proposes to implement a stream architecture for handling massive incoming data from social media using specific keywords with open source technologies, for example, Apache Kafka combined with Python and MongoDB as NoSQL Database. Data is ingested to NoSQL via the stream architecture. In the system architecture, data is gathered by the producer and published into the Kafka broker with specific topics and the consumer will subscribe to the topic to retrieve the data and store it in the NoSQL Database. (Hamami & Dahlan, 2020)

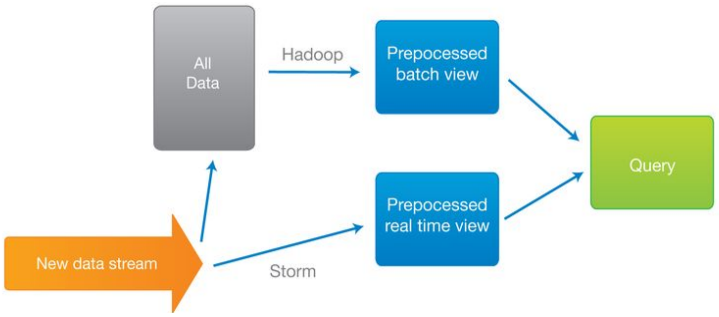
According to the research on Effective and Efficient Event Modelling for Real-Time Online Social Media Networks, they proposed ETree as a solution to real-time data concentrated social media where most of the data is in the form of short and unstructured messages. ETree uses n-gram based content analysis for extracting main information from a stream of short messages. It also uses an incremental and hierarchical modelling technique to create event theme architectures at different granularities. Furthermore, it incorporates an enhanced temporal analysis method to identify underlying relationships among information blocks. (ETree: Effective and Efficient Event Modelling for Real-Time Online Social Media Networks, 2011)

Based on Ilaria Barolini and Marco Patella, they have suggested Real-time Analysis Massive Media Multimedia Streams (RAM3S) framework to aid data analysis on the complicated technicalities of the stream processing of big data platforms like Spark Streaming, Storm, and Flink. It is considered a “middleware” software layer designed to facilitate the integration of multimedia stream analysis techniques with Big Data streaming platforms. It acts as an intermediary between multimedia stream analysis techniques and Big Data streaming platforms, simplifying the deployment of non-parallel techniques. RAM3S is composed of two interfaces: the Analyzer and the Receiver. The Analyzer in the RAM3S framework analyses multimedia objects, while the Receiver decomposes a single input multimedia stream into individual MMObject instances for analysis. (Bartolini & Patella, 2019)

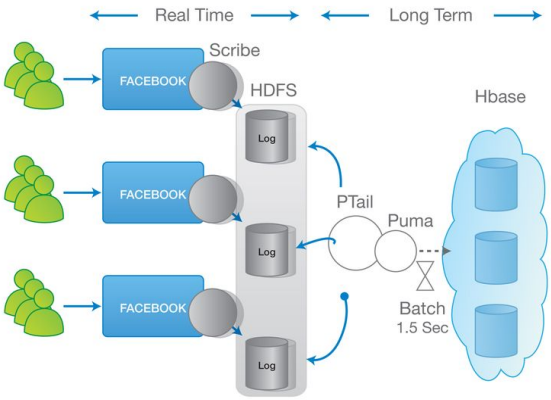
1.4 Comparison Between Facebook and X(Twitter)

	X (Twitter)	Facebook
Transaction and Architecture	Event Driven Paradigm Batch processing using Hadoop, Real-time processing using Storm.	Logging-centric Paradigm Real-time and Batch processing using Apache Hadoop
Concurrency Control	Optimistic concurrency control	Optimistic concurrency control
Database Recovery Method	PowerTrack Recovery streams	Point-in-time recovery (PITR)
Database Security	Security Automation	Defence-in-depth approach

X is using an event-driven approach for the real-time analytics architecture. Hadoop is being used for batch processing while Storm is for real-time processing. Storm is built to be able to parallelize sophisticated analysis for the purpose of transaction parallelization. Since Storm was designed to execute complicated aggregation of data as it is poured into Twitter's server and then sent out to a batch system for a more thorough analysis, it is the perfect fit for Twitter. (Shalom., 2013) Facebook's real-time analytics architecture uses a logging-centric approach, while batch and real-time processing are handled by the Apache Hadoop framework. Logging-centric approaches have less parallelization than event-driven approaches since they provide data-driven ordering and consistency. (Shalom., 2013)



X Real-Time Analytics Architecture



Facebook Real-Time Analytics Architecture

Both X and Facebook implement optimistic concurrency control in the database transaction. ConfigBus, which serves as a front-end for the configuration repository and implements the Git "smart" HTTP protocol, provides programmatic access to the database of X. The system has an inbuilt optimistic concurrency control mechanism that automatically attempts to retry a failed push caused by simultaneous modifications to the repository. (Viswanathan, 2018) Moving on to the optimistic concurrency control in Facebook, it is being used to detect and resolve conflicts. Upon receiving the read-and-write action, the system will assess if the transaction conflicts with another simultaneously processing transaction that has previously been accepted. The transaction will be accepted if all conflicts are resolved to maintain the long-term stability of the database. (Masti, 2021)

Facebook implements a point-in-time recovery which records all the transactions to a previous backup. It is the first line of protection of Facebook's database and it is called "rock" backups (RBUs). It has 2 RBU storage servers and one of the servers will record the binary log of the transaction. The binary log performs point-in-time recovery to the failing server by recovering data changes up to a certain point in time, and it promotes the other server to the main server when one of the servers experiences server failure. (Meta, 2013) For X, there is no official documentation about the recovery method for X's database. However, it offers PowerTrack Recovery streams that enable clients to utilise the real-time PowerTrack for data recovery. (X Developer, n.d.)

It is crucial to protect the database from the threat to prevent data loss and data leakage. X utilises security automation tools to streamline and automate security tasks. It is able to detect suspicious and malicious files as well as vulnerability scanning. It could highly reduce the human effort in protecting the X's database from the threat. (Roth & Harvey, 2018) Facebook implements a defence-in-depth technique as one of its database security methods. It functions as a protective layer that monitors the behaviours of the programme and data in order to identify and flag any irregularities. (Rodriguez, 2021)

1.5 Challenges of Using the Concept on Facebook and X(Twitter)

With the processing technologies implemented by Facebook, the scalability and transactional integrity of the Logging-centric Paradigm for Facebook will face obstacles to managing huge amounts of user data while generating transactional activity. It is difficult to secure faultless transactions in a high-throughput environment that undergoes real-time and batch processing using Hadoop. Furthermore, optimistic concurrency control of transactions has caused conflicts within DBMS. It became harder for Facebook to maintain the consistency of concurrency control when facing a high number of transactions simultaneously. Other than that, the Point-in-time recovery (PITR) method implemented by Facebook raised issues when restoring data to a specific version of DBMS. Thus, inconsistency of data and system crashes happened because of the problems of recovery coordination and the expanding volume of data. (Dinç,2023) Lastly, the high frequency of monitoring programs and data behaviour is a challenge for Facebook as they use a defence-in-depth approach for their database security. This will lead to the risk of a performance downgrade and affect the user experience if no frequent monitoring events happen.

The challenge faced by X with the processing technologies included managing transactions with the rapid growth of the user base. It affected the dependability and consistency of the DBMS for X across concurrent transaction operations. (Molnár,2013) As a result, the maintenance of the DBMS became more complex and difficult. Moreover, X faced the same issue with Facebook, as they both utilised Optimistic Concurrency Control and caused errors. In addition, PowerTrack Recovery streams of X will lead to data loss and system failure in the DBMS. It also caused degradation in the performance of DBMS because of the inability to recover to solve downtime and data retrieval issues. Besides, the Security Automation of X creates difficulties in managing and recognizing suspicious activity. It is because X required continual upgrades and monitoring to ensure protection against new threats.

To sum up, Facebook and X (Twitter) have challenges with their respective technical aspects of DBMS. DBMS's processing technology needs to be improved and enhanced due to the increasing demand for user data to tackle all the challenges and issues that lead to its failure.(Brodie, 1998)

1.6 Conclusion

In conclusion, the analysis of the technical aspects of Database Management Systems (DBMS) and stream processing technologies employed by social media platforms like X and Facebook reveals both innovative strategies and persistent challenges. While X adopts an event-driven paradigm and Facebook utilises a logging-centric approach for real-time analytics, both platforms face the same issue such as optimistic concurrency control, database recovery methods, and security measures. Facebook faces challenges including scalability concerns with the logging-centric paradigm, while X encounters complexities in managing transitions with a growing user processing technologies to ensure data integrity, security, and efficient handling of real-time data, reflecting the evolving landscape of social media platforms and their continuous pursuit of the optimal solutions.

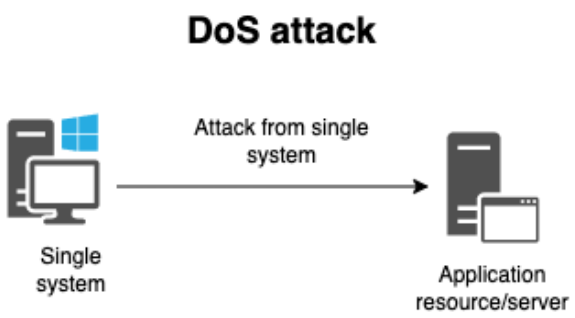
Part 2:

2.1 Introduction

A **Denial-of-Service (DoS) attack** is a cyberattack that attempts to bring down a computer system or network rendering it inaccessible to its intended users (Palo Alto Networks, n.d.). DoS attacks typically overwhelm the target system with an excessive amount of traffic, preventing legitimate users from accessing essential services or resources. For instance, cyber attackers exploit a software vulnerability in the system to consume the RAM or CPU resources of the server that cripples its ability to function (Investopedia, n.d.).

Besides, DoS attacks frequently target the web servers of prominent corporations, encompassing industries from media, financial and commercial fields (Palo Alto Networks, n.d.). Unlike other cyber attacks, it will not lead to data theft or loss but it can inflict significant financial and time-consuming burdens on victims.

Denial-of-Service attacks can have a detrimental impact on organisations. Hence, each organisation should implement effective preventive measures to safeguard against DoS attacks. However, failure to implement preventive measures can lead to the unavailability of resources, damage to reputation and the inability to perform time-critical actions. These effects might cost the organisation the loss of customers or users, failure to satisfy a contract and financial costs to recover the database (BBC Bitesize, n.d.). Therefore, an organisation must proactively adopt preventive approaches to counter DoS attacks to ensure business continuity and protect valuable data.



2.2 Motivation

Denial-of-Service (DoS) attacks on an organisation's digital platform possess a significant threat to the database transaction system of the organisation. Although DoS attacks will not lead to theft and fraud or any loss of information (*What Is a Denial of Service Attack (DoS) ?*, n.d.), it will cause the loss of availability which is also an important database security threat to avoid. DoS attacks overwhelm databases that process real-time data, causing services to crash and users could not use the system effectively. Databases should have continuous operation for authorised users to access the database anytime, however by the threat of DoS attack, the system will be unavailable due to a large amount of requests that flood the target machine to overload the system. As the threat is causing serious negative effects on the database system, it is essential to understand and find solutions to prevent DoS attacks.

As Facebook and X handle a huge number of users, we need to find ways to secure the system from DoS attacks to ensure smooth usage for the users. By studying DoS attacks and ways to overcome the attack, it will be beneficial to both the technology companies and users of the application program to create a safe and reliable environment for using the database of these applications.

2.3 Related Works to Overcome Denial-of-Service (DoS) Attacks

The **Pushback Mechanism** proposed by Ioannidis and Bellovin in 2002 is a mechanism of mitigating attacks in the form of Denial-of-Service, which calls for additional support from the routers in the network to restrict the flow of data to solve the congestion problem brought by DoS attacks. Through a combination of rate-limiting and congestion control strategies, the Pushback Mechanism can distinguish between legitimate traffic and attack packets to ensure that resources are efficiently allocated and crucial services remain accessible. Furthermore, the routers also mitigate DoS attacks by propagating the filters to the upstream routers so that the router resource can be freed to route legitimate traffic (Ioannidis & Bellovin, 2002). The Pushback Mechanism is simple to implement, does not require modifications to the end-user devices and protects large networks.

Network Segmentation is a security strategy that can effectively defend against DoS attacks. Dividing the network into smaller and more manageable segments called a subnet can prevent DoS attacks from overwhelming the entire network (Byos, n.d.). For example, a virtual local area network (VLAN) can be implemented to operate within a single physical network as it can segregate different types of traffic such as user traffic and sever traffic onto different VLANs. This implementation can help prevent DoS attacks from affecting critical resources and services.

Besides that, the **IP Traceback** technique is a reactive mechanism that has been introduced by Savage et al. enabling victims to identify the actual origin of attack packets even if the packets are encrypted, thereby holding attackers accountable for misuse of the internet. By tracing the path characteristics of an IP packet, it reveals the true identity of the attacker. Furthermore, IP Traceback contributes to the mitigation of DoS attacks by isolating the detected sources of attack or by filtering attack packets at a distance from the victim as suggested in the IP traceback-based intelligent packet filtering method (Savage et al., 2000). IP Traceback operates independently of Internet Service Provider (ISP) collaboration for faster identification and mitigation of attack sources without waiting for ISPs to respond.

In 2003, Andersen proposed a new architecture called **Mayday** that integrates overlay networks and lightweight packet filtering to combat DoS attacks. Once client authentication and protocol verification are completed, the overlay nodes will operate to forward the requests to a secure server. This approach protects the server from attacks by implementing simple packet filtering which can be implemented effectively in the backbone routers (Andersen, 2003). Initially, Mayday generalises earlier work on Secure Overlay Services. Mayday has been enhanced by segregating the overlay routing and the filtering to provide a wider range of potent options for each. It offers a variety of schemes that provide different balances between security and performance, along with continuum and support mechanisms that exceed the security or performance of previous systems (Andersen, 2003).

2.4 Challenges of Implementing Related Works to Overcome Denial-of-Service (DoS) Attacks

Based on the related works to overcome Denial-of-Service (DoS) attacks discussed above, here are some challenges that are associated with the related works mentioned.

In a research paper from Pennsylvania State University, the authors mentioned that the challenge of implementing a pushback mechanism is to determine the contributing neighbour and the rate limit of the mechanism should be divided. Legitimate traffic that follows the same route as the attacking traffic is unable to be protected by the pushback mechanism because the active congestion control (ACC) does not differentiate between traffic on the same path.

To implement network segmentation, complexity is one of the challenges to be considered. According to Infrastructure Security from LinkedIn, implementing network segmentation requires an extensive knowledge of the network topology, traffic patterns, and application dependencies on the system (Infrastructure Security, 2023). Moreover, certain devices may not be compatible with micro-segmentation or software tools, necessitating potential upgrades or replacement for legacy systems or components. In 2019, Maxine Holt and Omdia also mentioned that performance bottlenecks might be created due to the increased segregation of IT systems (Holt, 2019).

Apart from the challenges to implementing network segmentation, there are also some remarkable challenges to implementing IP traceback. As mentioned above, IP traceback is a reactive mechanism that has caused a problem where the method must be completed when the attack remains active because once the attack ceases, the method will be ineffective to prevent DoS attacks (Aljifri, 2003). Furthermore, for the traceback to work well, different ISPs need to work together as cooperation is important in the deployment of a traceback scheme. Unfortunately, the current situation shows little evidence of the collaboration among the entities which brought up a major problem. Legal and privacy concerns are making the implementation of IP traceback more challenging (Singh et al., 2016).

Generally, the challenges of implementing the related works to overcome DoS attacks include factors related to the complexity of integration, compatibility of architectures and tools, costs to implement, and the balance between security and performance. To implement methods to overcome DoS attacks, issues such as training to maintain the system and user education should be considered before implementation to ensure the smooth performance of the system and provide a safe and secure environment for the users to utilise transactions of the database system.

2.5 Comparison of Denial-of-Service (DoS) Attack Prevention Techniques

Denial-of-service (DoS) attacks have emerged as major threats to networked computer systems over time. It often exploits vulnerabilities in software protocols to attain significant outcomes with minimal resource input. Also, the internet is particularly prone to flooding DoS attacks, where the attacker makes a little investment that results in a significantly higher resource consumption on the targeted system. However, defending against flooding DoS attacks can be very challenging. The ongoing compromise or trade-off between providing network services to authorised users and preventing malicious users from accessing the system lies at the core of this challenge (Thakur, 2015).

Name of the technique	Malicious behaviour	Good Behaviour Pay-off	Parameter	Protocol	Evaluation Parameter
Game Theoretic Approach	Black holes and falsify route error message	Reputation	Reputation, Cooperation, Utility, Density, Reliability, Distance, Weight Parameters	Utility based Dynamic Source Routing (UDSR)	Mean no of packets dropped
Enforcing Security Using Economical Modeling	Non cooperative nodes.	Reputation.	-	SAR (Secure Auction-based Routing)	Mean no of packets dropped, Reputation,
Repeated Game Theory Approach	Agree to forward packets but fail to do so	Reputation	Cost of forwarding packet, History, Rating, Reputation, Utility, Weight	Repeated Game Theory based on DSR	Number of hops for received packets, Throughput
A Bayesian Game Approach	Non cooperative nodes.	Reputation	-	S-LEACH	Number of packets dropped, Throughput
Strength based Detection and Prevention	Reply of Hello message	Signal Strength	Signal Strength	AODV routing protocol	Total packet received Total packet dropped Packet delivery ratio
An Ant Based Framework	Flooding	Reliability, buffer size	-	Ant-Based Routing Algorithm	-
Protection using KDS	Node replication, Capture nodes.	Mutual Authentication	-	Hybrid Energy-Efficient Distributed clustering (HEED) protocol	Network Lifetime, energy Used
Message Observation Mechanism (MoM)	Content attack, Frequency attack	NML AML	The number of messages and the content of messages.	message observation mechanism (MoM)	Loss Rate of packets, Number of packets
Cooperative game theoretic approach	-	Reputation	cost for attack detection, ratio of correct attack detection to no detection and total detection	Fuzzy Q-learning algorithm	accuracy of defence rate, number of live nodes, energy consumption

Comparison of Techniques in Preventing Denial-of-Service Attacks (Patil & Chaudhari, 2016)

The table above shows all the Denial-of-Service (DoS) Attack prevention techniques. The **Game Theoretic Approach** leverages Utility-based Dynamic Source Routing (UDSR) to identify malicious nodes. This method is indeed a mathematical approach that provides a quantitative framework for modelling DoS attacks and supports decision-making in maintaining the optimal level of network security against the number of resources used (Kumar & Bhuyan, 2019). Using the Game Theoretic Approach can reduce packet losses and unlike Dynamic Source Routing (DSR), which does not respond to nodes that behave poorly. UDSR will assign utility values to nodes based on their performance and network contribution, then prioritise reliable nodes and avoid routing through malicious nodes. While the Watch-list maintains a record of suspected malicious nodes and prevents them from participating in routing decisions. In order to prevent the network from being harmed, the node might be ignored. However, there are disadvantages to this approach, which include the difficulty of identifying false labelling and the challenge of setting the threshold values (Patil & Chaudhari, 2016).

On the other hand, the **Ant-based Framework** is another technique for preventing DoS attacks. It utilises the concept of stateless and stateful signatures to preserve legitimate packets and reject infected ones. The algorithm deployed is called Ant-Based Routing. The DoS Detecting Ants (DDA) detect attacks when reliability varies or when the buffer size exceeds a certain threshold. Meanwhile, the DoS Preventing Ants (DPA) detects discrepancies between the actual and sample packet arrival rates at a node. This approach ensures the attacks on valid packets are unaffected and prevents misleading tagging. The Ant-based Framework reduces resource use when used for DoS prevention and simplifies the process of detecting the source of traffic (Patil & Chaudhari, 2016).

The **Message Observation Mechanism (MOM)** is also a technique that prevents DoS attacks. It is built on top of the spatiotemporal correlation foundation and uses a similarity function to distinguish between the frequency assault and the content attack. The MOM then implements countermeasures such as rekeying and rerouting to isolate any rogue nodes. As the number of rogue nodes rises, this approach reduces packet loss effectively. MOM is ideal for detecting and defending against DoS attacks, while also minimising energy usage since it does not transport packets from malicious nodes further (Patil & Chaudhari, 2016).

2.6 Conclusion

In conclusion, DoS attacks are harmful to database systems and appropriate measures should be taken to prevent DoS attacks. It is a matter of concern that organisations should address the importance of preventing DoS attacks to protect the availability of databases which is the most important property for systems to operate. There are a few methods to overcome DoS attacks, such as Pushback Mechanism, Network Segmentation, IP Traceback and Mayday. However, as each method will benefit the database systems by preventing DoS attacks, they also bring some challenges along in the process of implementing or maintaining. By doing the comparison of the DoS attacks prevention techniques, it is easier to select the most suitable technique that fits the particular system and the specifications of the system. Understanding about DoS attacks will help to increase awareness about the safety of the database system and take steps to protect the database to make sure that the database will be secure and available for authorised users. In the future, preventive measures need to evolve and improve rapidly to match the pace of advancing DoS attacks.

3.0 References

1. Aljifri, H. (2003). IP traceback: a new denial-of-service deterrent? *IEEE Security & Privacy*, *1*(3), 24–31. https://doi.org/10.1109/msecp.2003.1203219
2. Andersen, D. G. (2003). *Mayday: Distributed Filtering for Internet Services.* https://www.cs.cmu.edu/~dga/papers/mayday-usits2003/
3. Bartolini, I., & Patella, M. (2019, November 29). *Real-Time Stream Processing in Social Networks with RAM3S*. Future Internet. https://doi.org/10.3390/fi11120249
4. BBC Bitesize. (n.d.). *Denial of Service (DoS) attacks - Security risks and precautions - Higher Computing Science Revision.* https://www.bbc.co.uk/bitesize/guides/z9fbr82/revision/3
5. Brodie, M. L., Bancilhon, F., Harris, C., Kifer, M., Masunaga, Y., Sacerdoti, E. D., & Tanaka, K. (1990). *Next Generation Database Management Systems Technology. In Deductive and Object-Oriented Databases* (pp. 335–346). Elsevier. https://www.researchgate.net/publication/2709166_Next_Generation_Database_Management_Systems_Technology
6. Byos. (n.d.). *Denial-of-Service (DoS) Attack Prevention: The Definitive Guide.* https://www.byos.io/blog/denial-of-service-attack-prevention
7. Data Reportal. (2023, October). *Global Social Media Stats*. DataReportal – Global Digital Insights; Kepios. https://datareportal.com/social-media-users
8. DİNÇ, S. (2023). *BIG DATA FROM SOCIAL MEDIA PERSPECTIVE: A CASE STUDY WITH FACEBOOK*. Fenerbahçe Üniversitesi Sosyal Bilimler Dergisi, 1, 1–14. https://doi.org/10.58620/fbujoss.1310830
9. *ETree: Effective and Efficient Event Modeling for Real-Time Online Social Media Networks*. (2011, August 1). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/document/6036774
10. Gunjal, B. (2003). *Database System: Concepts and Design.* https://www.researchgate.net/publication/257298522_Database_System_Concepts_and_Design
11. Hamami, F., & Dahlan, I. A. (2020, November 1). *The Implementation of Stream Architecture for Handling Big Data Velocity in Social Media. Journal of Physics: Conference Series*, *1641*(1), 012021. https://doi.org/10.1088/1742-6596/1641/1/012021

12. Hasan, M., Orgun, M. A., & Schwitter, R. (2017). *A survey on real-time event detection from the Twitter data stream. Journal of Information Science,* 4, 443–463. https://www.researchgate.net/publication/315343836_A_survey_on_real-time_event_detection_from_the_Twitter_data_stream
13. Hellemans, J., Willems, K., & Brengman, M. (2020). *Daily Active Users of Social Network Sites: Facebook, Twitter, and Instagram-Use Compared to General Social Network Site Use. In Advances in Digital Marketing and eCommerce* (pp. 194–202). Springer International Publishing. https://www.researchgate.net/publication/341196015_Daily_Active_Users_of_Social_Network_Sites_Facebook_Twitter_and_Instagram-Use_Compared_to_General_Social_Network_Site_Use
14. Holt, M. (2019, May 13). *Security Think Tank: Benefits and challenges of security segmentation*. ComputerWeekly.com. https://www.computerweekly.com/opinion/Security-Think-Tank-Security-segmentation-benefits-and-challenges
15. Infrastructure Security. (2023, March 9). *What are the benefits and challenges of micro-segmentation for network security?* LinkedIn. https://www.linkedin.com/advice/1/what-benefits-challenges-micro-segmentation#:~:text=Complexity%20is%20one%20such%20challenge,lengthy%20and%20error%2Dprone%20process.
16. Investopedia. (n.d.). *Denial-of-Service (DoS) Attack: Examples and Common Targets.* https://www.investopedia.com/terms/d/denial-service-attack-dos.asp
17. Ioannidis, J., & Bellovin, S. M. (2002). *Implementing Pushback : Router-Based Defense Against DDoS Attacks. Network and Distributed System Security Symposium.* https://doi.org/10.7916/d8r78mxv
18. Masti, S. (2021, August 6). *How we built a general purpose key value store for Facebook with ZippyDB*. Engineering at Meta. https://engineering.fb.com/2021/08/06/core-infra/zippydb
19. Matsunobu, Y., Dong, S., & Lee, H. (2020, August 31). *MyRocks: LSM-Tree Database Storage Engine Serving Facebook's Social Graph - Meta Research*. Meta Research.

https://research.facebook.com/publications/myrocks-lsm-tree-database-storage-engine-serving-facebooks-social-graph/

20. Meta. (2013, January 14). *Under the Hood: Automated backups*. Engineering at Meta. https://engineering.fb.com/2013/01/14/web/under-the-hood-automated-backups/
21. Molnár, B., & Vincellér, Z. (2013). *Comparative study of Architecture for Twitter Analysis and a proposal for an improved approach.* https://www.researchgate.net/publication/259469847_Comparative_study_of_Architecture_for_Twitter_Analysis_and_a_proposal_for_an_improved_approach
22. Palo Alto Networks. (n.d.). *What is a Denial of Service Attack (DoS)?* https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos
23. Patil, S., & Chaudhari, S. (2016). *DoS Attack Prevention Technique in Wireless Sensor Networks. Procedia Computer Science, 79, 715–721.* https://doi.org/10.1016/j.procs.2016.03.094
24. R. Pandey, A. Singh, A. Kashyap and A. Anand, *"Comparative Study on Realtime Data Processing System," 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)*, Ghaziabad, India, 2019, pp. 1-7, doi: 10.1109/IoT-SIU.2019.8777499. https://ieeexplore.ieee.org/document/8777499
25. Rodriguez, C. (2021, October 20). *Facebook engineers receive 2021 IEEE Computer Society Cybersecurity Award for static analysis tools*. Engineering at Meta. https://engineering.fb.com/2021/10/20/security/static-analysis-award/
26. Roth, Y., & Harvey, D. (2018, June 26). *How Twitter is fighting spam and malicious automation. Blog.twitter.com.* https://blog.twitter.com/en_us/topics/company/2018/how-twitter-is-fighting-spam-and-malicious-automation
27. Saurabh, S. (2022, November 2). *Stream Processing: Who, How, and Why*. Nexla. https://nexla.com/stream-processing/
28. Savage, S., Wetherall, D., Karlin, A., & Anderson, T. (2000). *Practical network support for IP traceback. ACM SIGCOMM Computer Communication Review, 30(4), 295-306.* https://dl.acm.org/doi/pdf/10.1145/347057.347560

29. Shalom. (2013). *Facebook's vs Twitter's Approach to Real-Time Analytics*. Nati Shalom's Blog. https://natishalom.typepad.com/nati_shaloms_blog/2013/10/facebooks-vs-twitters-approach-to-real-time-analytics.html
30. Singh, K., Singh, P., & Kumar, K. (2016). *A systematic review of IP traceback schemes for denial of service attacks. Computers & Security*, *56*, 111–139. https://doi.org/10.1016/j.cose.2015.06.007
31. Snoeren, A. C., Partridge, C., Sanchez, L. A., Jones, C. E., Tchakountio, F., Kent, S. T., & Strayer, W. T. (2001). *Hash-Based IP Traceback.* https://users.cs.jmu.edu/aboutams/Public/IP%20TraceBack/Hash-Based%20IP%20Traceback
32. Thakur, K. (2015). *Analysis of Denial of Services (DOS) Attacks and Prevention Techniques. International Journal of Engineering Research and Technology, 4(7).*
33. Viswanathan, S. (2018, November 20). *Dynamic configuration at Twitter*. Blog.twitter.com. https://blog.twitter.com/engineering/en_us/topics/infrastructure/2018/dynamic-configuration-at-twitter
34. *What is a denial of service attack (DoS) ?* (n.d.). Palo Alto Networks. https://www.paloaltonetworks.com/cyberpedia/what-is-a-denial-of-service-attack-dos
35. X Developer. (n.d.). *Recovery and redundancy features*. Developer.twitter.com. Retrieved December 1, 2023, from https://developer.twitter.com/en/docs/twitter-api/enterprise/powertrack-api/guides/powertrack_recovery_and_redundancy_features#:~:text=Data%20availability&text=A%20recovery%20stream%20is%20started