Literature Review: Investigate security challenges and solutions in Hadoop-based big data processing.

Abstract:

Big Data encompasses diverse hardware and software technologies within heterogeneous infrastructures. The Hadoop framework is a prominent solution for storing and processing Big Data, delivering rapid and cost-effective capabilities. It plays a pivotal role across various sectors such as healthcare, insurance, and social media. Hadoop's strength lies in its open-source, distributed computing model, allowing data storage and processing across clusters of standard computers. However, the framework's flexibility introduces a number of threats to data security, making it susceptible to attacks. These vulnerabilities need attention to safeguard against potential breaches and ensure the protection of valuable data assets. We have further discussed such security challenges and possible solutions to reduce or eliminate these challenges and vulnerabilities.

Introduction:

Big Data represents an invaluable wealth of historical information crucial for informed decision-making in organizations. Its vastness stems from diverse sources like social media, sensors, GPS, transactions, and online content, offering valuable insights if properly harnessed. The Hadoop framework, synonymous with Big Data, efficiently archives and analyzes this massive data using advanced analytic tools, surpassing traditional methods. However, the colossal growth in data volume, velocity, variety, and veracity poses challenges in managing, securing, and ensuring privacy. Additionally, the emergence of vulnerability as another concern accentuates the importance of addressing data security comprehensively in the realm of Big Data.

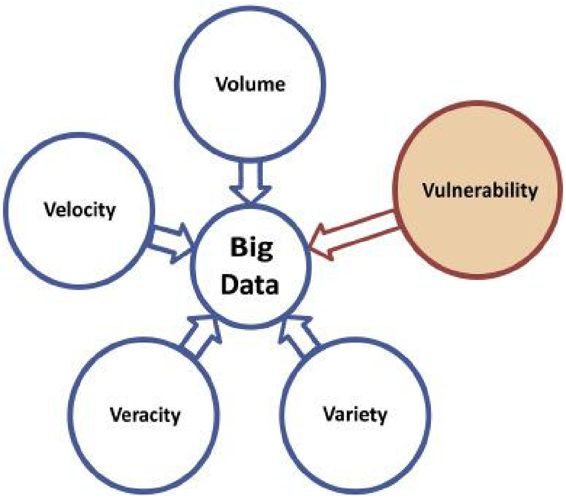

Fig. 1: The 5 Vs of Big Data

Security Challenges:

When we say vulnerability, we essentially mean the security challenges faced in a big data ecosystem. Vulnerabilities in software, online interfaces, or networks can lead to attacks on any component of the Hadoop environment. The Hadoop framework is a sophisticated confluence of hardware, software for distributed computing, application applications, and policies for allocating these resources. The three categories of vulnerabilities are as follows:

1. Technology and Software: For instance, the Hadoop framework is fully written in Java, which is widely used by hackers and connected to a number of security lapses.
2. Configuration/Web Interface: Hadoop is poorly configured due to a number of susceptible default settings, including IP addresses and ports, which are exploited. Hue and other Hadoop web interfaces are prone to XSS scripting attacks.
3. Security/Network Policy: Because the Hadoop architecture is made up of multiple databases, distinct user types in these rules are not configured correctly, necessitating the need for fine-grained policies at both the service and data levels.

Architectural security challenges:

Hadoop was built keeping in mind its on-prem deployment. Hence, sensitive information was stored in isolated clusters where security wasn't an issue and was primarily focused on improving efficiency. However, as Hadoop evolved into Big Data-as-a-Service (BDaaS), took to the cloud, and became surrounded by an ever-growing ecosystem of tools and applications, security became an important topic of discussion.

Security threats and attacks:

The basic principle of security is CIA - Confidentiality, Integrity and Availability.

- Confidentiality: To ensure that only authorized personnel are given access or permission to modify data
- Integrity: To maintain the trustworthiness of data by having it in the correct state and immune to any improper modifications
- Availability: Only authorized users should be able to access data whenever required

Hadoop security progresses through four levels. Initially, at level-1, minimal security operates within trusted environments. Level-2 adds Kerberos authentication(a computer network authentication protocol that allows nodes to prove their identity to each other over a non-secure network) at the perimeter. Moving to level-3, security projects like Apache KNOX, Apache Ranger, and Rhino enhance Hadoop's security framework. Kerberos integrates with AD(Active Directory) and LDAP(Lightweight Directory Access Protocol) for authentication. Each project offers distinct security solutions. However, a transition to level-4 is essential for a unified, centralized security approach within a single project, ensuring comprehensive protection for Hadoop systems.

Some of the most common security challenges include:

1. Attacks:
    a. Impersonation Attacks: These attacks occur when an attacker tries to access resources by impersonating as someone who is authorized to access the

    system. The attacker steals credentials of the authorized user and attacks the Hadoop clusters and other resources. This can cause retrieval of sensitive information for malicious motives.

    b. Denial of Service(DoS) Attacks: DoS attacks disrupt services by flooding targets with excessive traffic or causing system crashes. They deny legitimate users access to expected resources. Two main types are flood attacks, overwhelming servers with traffic, and crash attacks, causing system failure. Examples include DDoS, buffer overflow, SYN flood, and HTTP flood. In Hadoop, the vulnerable Name Node, managing MapReduce tasks and HDFS operations, can be targeted. A DoS attack on the Name Node can halt all MapReduce computations and HDFS read-write operations.
    c. Cross-Site Scripting(XSS): XSS injects malicious code into vulnerable web apps, posing a risk to users rather than the app itself. It comes in two forms: stored (persistent) and reflected. Stored XSS, more severe, injects scripts directly into vulnerable apps, while reflected XSS reflects malicious scripts back to users' browsers via links. Hadoop's web UIs(e.g. Hue) are susceptible to such attacks.
2. Data Security: Hadoop clusters often handle sensitive data, posing risks of unauthorized access, data leaks, or breaches during data transfer(data in transit) or storage(data at rest). Data in transit and at rest might be vulnerable without proper encryption mechanisms, exposing it to interception or unauthorized viewing. Also, ensuring data is compliant to HIPAA, GDPR, CCPA, etc. regulations is challenging given the vast amounts of varied data.
3. Auditing and Monitoring: Establishing comprehensive auditing mechanisms and real-time monitoring to detect suspicious activities, unauthorized access, or potential security breaches across a distributed environment is complex but necessary for security.
4. Internode communication: network security plays an important role in ensuring all nodes such as namenode, datanodes, secondary namenode and client nodes can communicate with each other. Attackers may often hinder the communication between these nodes via node blocks resulting in eavesdropping, DDoS attacks and spoofing.

Solutions:

1. Kerberos mechanism: Kerberos is used to authenticate users while preventing passwords from being sent over the internet. It uses tickets, symmetric key cryptography, and a key distribution center (KDC) to authenticate and verify user identities. Kerberos works by:
    a. Using a ticket-granting server (TGS) to connect the user with the service server (SS)
    b. Storing the password and identification of all verified users in a Kerberos database
    c. Using a shared secret key that is securely kept on both the client and server

   Kerberos has been used since the 1980s and is compliant with modern distributed systems requirements.

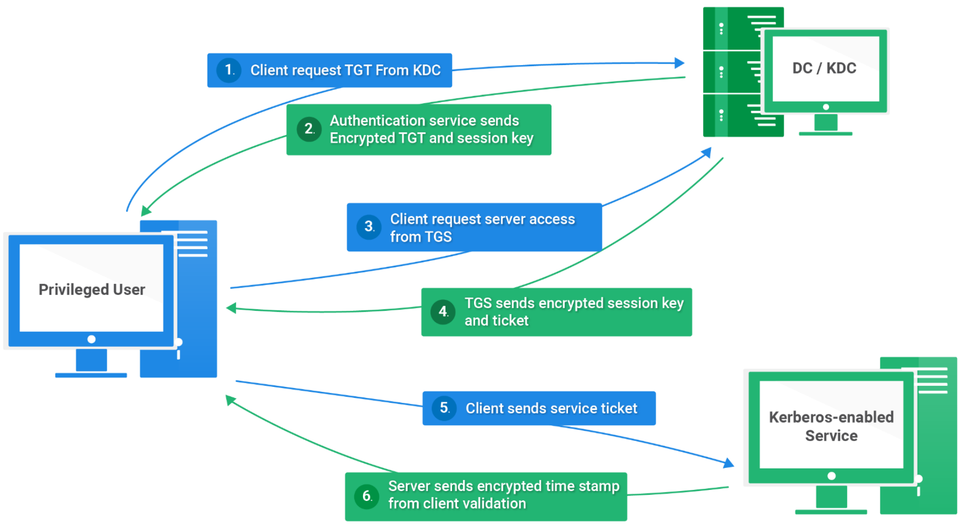
1. Client request TGT From KDC
2. Authentication service sends Encrypted TGT and session key
3. Client request server access from TGS
4. TGS sends encrypted session key and ticket
5. Client sends service ticket
6. Server sends encrypted time stamp from client validation
DC / KDC
Privileged User
Kerberos-enabled Service

Fig.2: Kerberos based authentication flow

2. Apache Knox: It is a security layer that provides a single point of authentication and access to Hadoop data. It acts as a reverse proxy and centralizes authentication and authorization for all endpoints. Knox simplifies Hadoop security for both users who access the cluster data and run jobs, and for operators who access and manage the cluster. Knox creates a perimeter around a Hadoop cluster, allowing the enterprise to extend Hadoop access to new users while maintaining compliance with enterprise security policies. It is a REST API gateway that authenticates users and acts as a single access point for a Hadoop cluster. Knox's native APIs include:
    a. Knox SSO: Provides a normalized SSO token for representing the authenticated user
    b. Knox token service: Takes advantage of the provider pipeline to do token exchange type scenarios

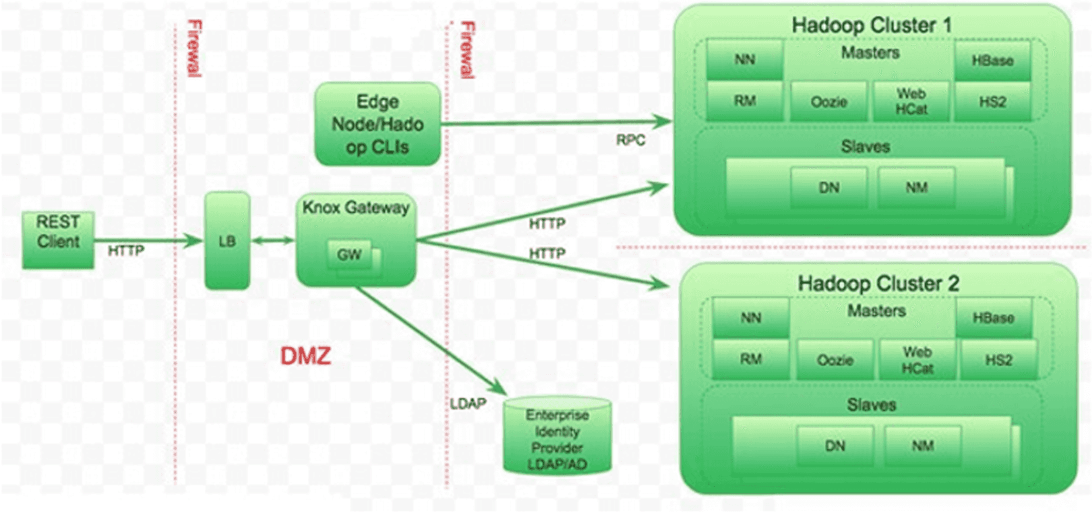

Fig. 3: Apache Knox HTTP request flow

3. Apache Sentry: A role-based, granular Hadoop authorization module is called Apache Sentry. Sentry gives authenticated users and apps on a Hadoop cluster the power to manage and enforce exact degrees of privileges on data. Sentry is now

compatible out of the box with HDFS (limited to Hive table data), Impala, Apache Solr, Hive Metastore/HCatalog, and Apache Hive. Sentry's pluggable authentication engine is intended to be used with Hadoop components. It lets you create authorization rules to verify requests for Hadoop resource access from users or applications. Sentry can facilitate authorization for a large range of Hadoop data formats and is very adaptable.

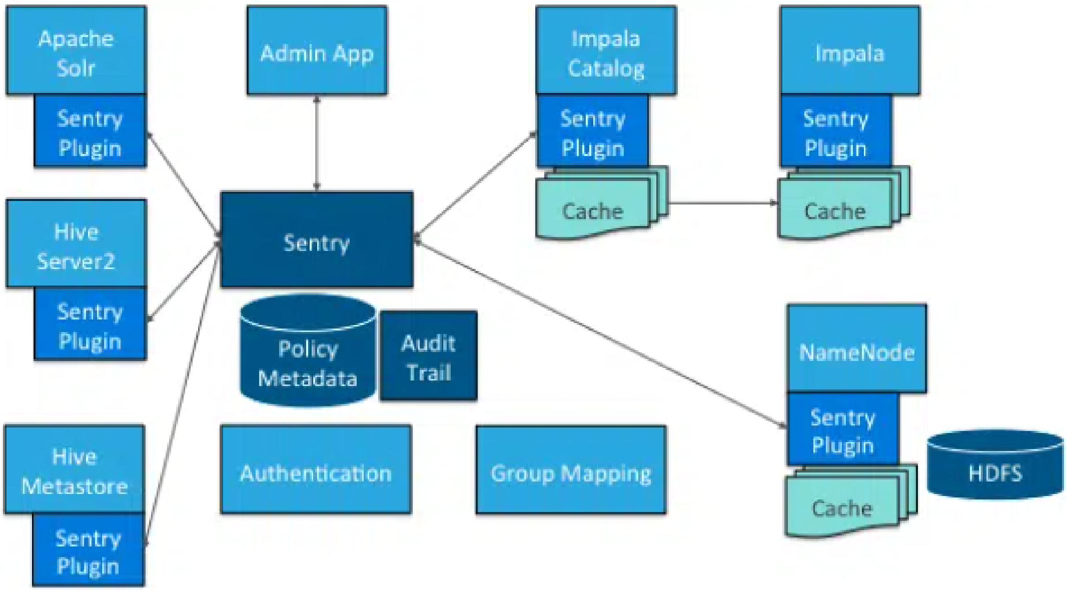

Fig. 4: Sentry Integration with Hadoop Ecosystem

4. Apache Ranger: It is a centralized security framework designed for securing Hadoop clusters. It functions as an authorization system, determining access permissions to various resources within the cluster, like HDFS files or Hive tables, based on predefined policies for authenticated users. Ranger assumes that the user making the request has already been authenticated. It utilizes Kerberos for authentication and Apache Knox to implement role-based access control (RBAC) for authorization purposes. Additionally, Apache Knox supports auditing capabilities for HDFS, Hive, and HBase, while Apache Ranger ensures data protection through wire encryption.

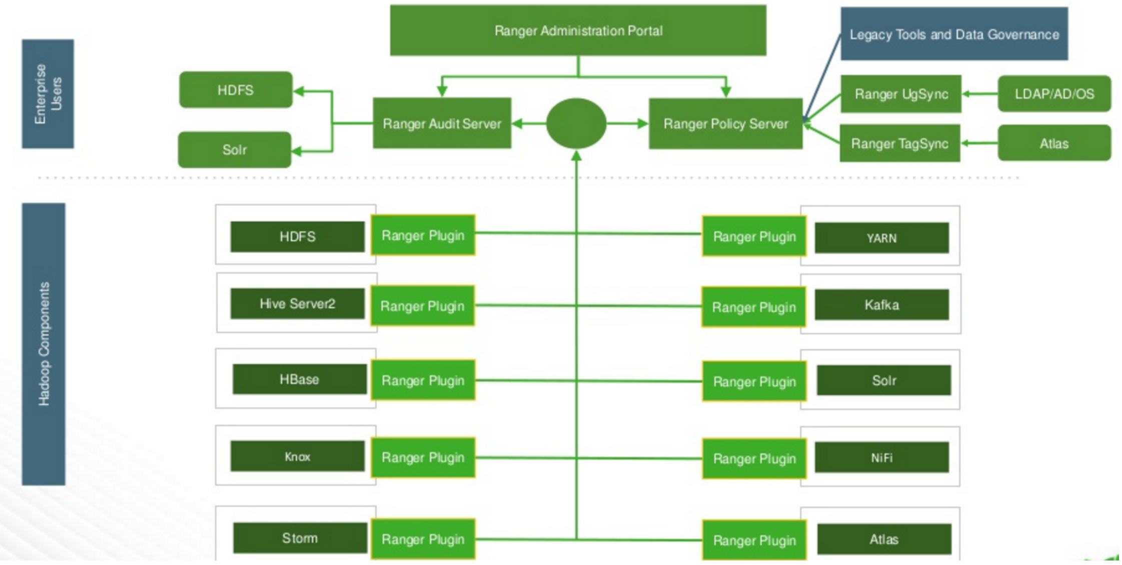

Fig. 5: Apache Ranger Architecture

5. Project Rhino: It is an Intel-developed open-source Hadoop security tool that focuses on bolstering data protection within the Hadoop stack through a single-sign-on (SSO) approach. It builds upon Kerberos and enhances authentication mechanisms. Rhino introduces several key features to fortify Hadoop Apache security:
    a. Enhanced Data Protection: Introduces encryption support across core Hadoop components like HDFS, MapReduce, and Hive for heightened data security.
    b. Token-Based Authentication: Implements a universal token-based authentication framework, separating internal user and service authentication from external mechanisms for improved security.
    c. Standardized Audit Logging: Establishes a standardized and unified log format, facilitating compliance adherence and activity tracking across various aspects of Hadoop Apache.

Conclusion:

Traditional enterprise security tools aren't enough to handle the security challenges posed by Big Data. To address data aggregation risks, organizations must innovate in safeguarding critical data acquisition, maintenance, and analysis methods. Commercial security software often has access to all network data, rendering third-party add-ons ineffective in Big Data environments. Analysts, administrators and engineers working with big data technologies should be provided enhanced security training tailored to specific threats, impact analysis, and mission-critical objectives. Since the large data is in fact a problem, scaling security in a way that matches the scale of data is of extreme significance. Organizations using cloud based environments should efficiently leverage security features provided in cloud services such as AWS or Azure to safeguard their networks and protect their data, both at rest and in transit. Therefore, it is important to build a big data ecosystem that gives high priority to security and privacy while ensuring the performance and reliability of the system is not compromised.

References:

1. Gurjit Singh Bhathal, Amardeep Singh, “Big Data: Hadoop framework vulnerabilities, security issues and attacks”, *Elsevier, Volumes 1–2, January–April 2019, 100002*
2. Hadiqa Amjad, Nimra Jamil, Amna Azeem, and Saba Majeed, “Hadoop Security and Privacy in Big Data”, *Industrial Engineering & Management, Volume 10:9, 2021*
3. Dr. Santanu Koley, “Big data security issues with challenges and solutions”, O*ctober 2019, in book: Big Data Security (pp.95-142), Chapter: 6*
4. Hany Habbak, Khaled Metwally, Ahmed Maher Mattar, “Securing Big Data: A Survey on Security Solutions”, *IEEE, 2022 13th International Conference on Electrical Engineering (ICEENG)*
5. Adluru, Pradeep, Srikari Sindhoori Datla, and Xiaowen Zhang. "Hadoop Eco System for Big Data Security and Privacy." *In 2015 Long Island Systems, Applications and Technology, IEEE, (2015).*

6. What is the CIA Triad? Definition, Importance and Examples
7. Discussion and Comparison of Several Hadoop Security Tools | by Yinyi Soo
8. What Is Kerberos? Kerberos Authentication Explained | Fortinet
9. Apache Ranger
10. Apache Knox