Deloitte.

Analytics & Information Management
March 31st, 2017

Five key principles to secure the enterprise Big Data platform

Hadoop series on best practices for large enterprises – Security

"Security is key for data management especially when working with Big Data "

Sandra Bauer
Director Deloitte

Organizations face the risk of financial, legal and reputational damages if they do not take care of security for their data and IT systems.

Enterprise Data Governance rules, European legal restrictions like the GDPR but also national or industry-specific data privacy laws such as BDSG or TKG in Germany clearly define how data needs to be protected in IT systems in order to avoid data security breaches. Additionally, regulations like the German BSI security law and the international ISO/IEC 27000-series define standards for minimizing threats to IT systems.
It is therefore crucial for enterprises to carefully consider data security especially within their Big Data initiatives. Hadoop in particular poses a data security challenge with its complex ecosystem, ever evolving tool chain and the fact that security was not a top priority for the development of Hadoop in the first place. There is currently no universal security standard in the Hadoop landscape.

Deloitte has worked with numerous clients and Hadoop vendors to resolve these security challenges and offers a wide range of services from security assessments of already existing big data platforms to building a security strategy and implementation support for Big Data platforms.

Five key areas have been identified during projects as crucial for securing Hadoop especially for large enterprises with their strict security, governance and compliance regulations. The five areas are presented in the following to support you in achieving your goal of a secure and well-governed Hadoop environment.

Key Takeaways

Security in the context of Hadoop aims to provide confidentiality, availability and integrity. The following aspects are important cornerstones to any Big Data initiative and are outlined in more detail along with best practices:

- Authentication
- Authorization
- Encryption & Data Masking
- Auditing
- Disaster Recovery & Backup

Authentication: Make sure the user is who he claims to be

Authentication is about verifying the identity of a user or service, so that only legitimate users have access to the data and services of the Hadoop cluster.

Therefore, Hadoop in large organizations needs to be integrated usually with existing authentication systems (e.g. Active Directory). Re-usage of existing services also reduces maintenance efforts and costs. From the user perspective Single Sign-On is important to simplify access and to increase ease of use. In addition, it can also increase security as password hashes do not get repeatedly transmitted over the wire.

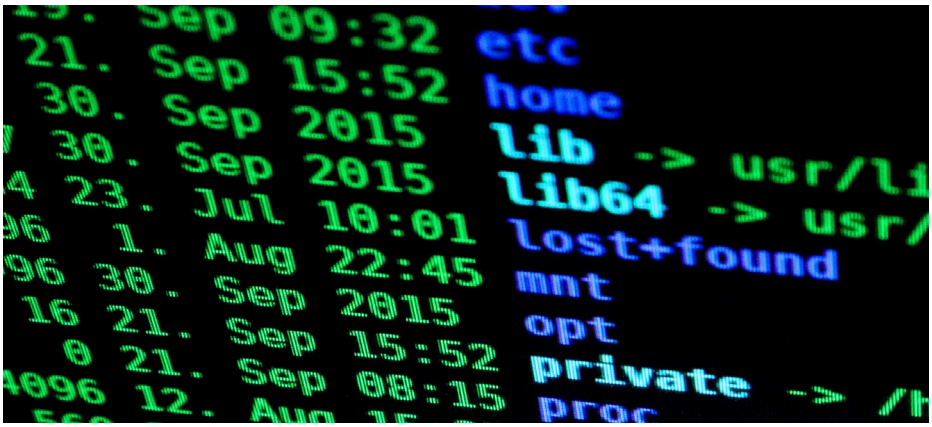
Regarding authentication, there is generally only a minor difference among different Hadoop distributions. Authentication should be reviewed and evaluated carefully end-to-end for each Hadoop component. This review should also include end points such as dashboards and analytical tools.

It is common to authenticate with Kerberos and use LDAP as a backend for Kerberos. Other authentication options exist – depending on the Hadoop component – such as SAML, OAuth and classic HTTP authentication.

Kerberos provides Single Sign-On via a ticket-based authentication mechanism. The SPNEGO protocol, which is supported by all major browsers, extends Kerberos authentication to web applications and portals.
Since Kerberos uses secret-key cryptography, it is crucial to secure intra-cluster communication from the start. It is also important to consider latency and fail-over.
It is recommended to set up a Kerberos instance local to the Hadoop cluster, in order to reduce latency. The local instance should include the technical Hadoop users directly and use synchronization otherwise.

Authentication Key Takeaways

- Review and evaluate carefully end-to-end the available authentication options on the Hadoop component level
- Use Kerberos and SPNEGO whenever possible for high security and Single Sign-On.



Authorization: Manage access to resources and logically separate data for multi tenancy

In almost all areas of business, it is legally required to allow access to information only for key personnel and user groups. Often, even a logical separation by legal entity or corporate structure is required, which can also be accomplished through authorization. Note however, that a required physical separation of data cannot be handled through authorization.

The task of "which users can perform which actions with which resources" is handled through authorization.

In plain Hadoop, access to the Hadoop filesystem HDFS is managed through UNIX access rights where users are assigned to user groups. Obviously, filesystem level access control is not sufficient for the highly complex Big Data environment. Ultimately, we are looking for an integrated authorization solution that spans across Hadoop components and which provides fine-grained access control on a file level, database column or message queue level. Role-based Access Control Lists (ACLs) are a flexible abstraction of the UNIX access rights and include features such as regular expressions for pattern matching.

Common integrated authorization solutions for Hadoop are Apache Ranger/ Knox (Hortonworks) and Apache Sentry (Cloudera, MapR). Whether access control on the column level or message queue level is possible depends on the particular setup. One can expect good integration with commonly used Hadoop components such as HDFS, HBase, Hive, Impala, Solr and Kafka, but might struggle with out-of-the-box access control for more exotic tools such as Storm/Heron. For example, only recently access control for Kafka topics was added to Kafka. Thus, careful planning and extensive knowledge of the Hadoop ecosystem is required.

Authorization Key Takeaways

- It is often legally required to implement fine-grained access control for Hadoop components
- The required breadth and depth of access control must be carefully evaluated on a Hadoop component level
- Access control solutions are available, but don't expect out-of-the-box integration with more exotic Hadoop components. Thus, careful planning and extensive knowledge of the ecosystem is required.

Encryption & data masking: Protect against leakage of data

Internal or external leakage of data is a key business concern. It is challenging to secure business critical data and personally identifiable information in a Big Data environment because data is stored across a long processing chain and in various data formats. Encryption can be mainly distinguished between data in-transit and data at rest encryption.

In-transit and at rest encryption can be challenging at times as more and more information is not file-based in nature, but rather handled through a complex chain of message queues and message brokers. In addition to cluster storage and communication channels, Hadoop components may use machine-local temporary files that consequently also contain sensitive information that must be secured.

Plain Hadoop only provides encryption for HDFS. Some Hadoop components write their temporary files to HDFS, which will automatically be secured, but other Hadoop components might not. However, in plain Hadoop there is no comprehensive cryptographic key management solution let alone Hardware Security Module (HSM) integration.

As with authorization we are looking for a comprehensive, integrated key management solution. Common tools include Cloudera Navigator Encrypt and Key Trustee Server and Ranger Key Management Service (Hortonworks) to set up data at rest encryption.

If data encryption is not required by law or business reasons for the entire dataset, then technically it might be enticing to only encrypt part of data due to performance reasons. Partly encrypted data via data masking techniques can be passed around using format preserving encryption techniques. If the only goal however is to make the data unreadable, but not recoverable, then faster data masking techniques like hash functions can be used. MapR for instance uses format-preserving encryption and masking

techniques maintaining the data format without replacing it with cryptic text supporting faster analytical processing between applications.

Encryption & Data Masking Key Takeaways

- It is often required by regulation or business to store and transmit sensitive data in an encrypted format
- Big Data processing requires information to be passed along a complex process chain, which makes it particularly challenging to apply appropriate encryption techniques in each step
- Comprehensive encryption requires an in-depth understanding of the different Hadoop components and their interplay

Auditing: Ensure compliance through an audit trail

Due to regulation and compliance it is often required to keep an audit trail e.g. to log cluster access and changes in cluster configuration.

Each of the Hadoop distributions offer audit capabilities to ensure that activities of platform users as well as administrator can be logged. Logging should include, but is not limited to, changes of files and folders in the filesystem, modifications of database structures, reconfigurations of the cluster, application exceptions and login attempts to services. A good audit trail allows to identify sources of data and application errors as well as to identify security events.

Every component of a big data platform allows for one or another form of logging either to the local file system or into HDFS. There are two main challenges in the big data world for auditing - on the one side the distributed architecture and on the other side the tight integration of distinct components with each other. Hence, main vendors invested heavily in central management and audit software that collects and combines information from multiple nodes and components.

Administrative changes and service events are usually captured and shown in the Hadoop management UIs. Common management programs are Cloudera Manager, Apache Ambari (Hortonworks) and the MapR Control System. Additionally, they allow to examine log files of different Hadoop components from a central interface without the need to manually collect them from their respective nodes.

Metadata for data lineage, database changes and security events are automatically captured in specialized data governance and metadata frameworks on Hadoop. Two of the most common are the Cloudera Navigator Audit Server and Apache Atlas (Hortonworks). They automatically capture events from the filesystem, database and authorization components of the platform and provide

interfaces to query those events for audit purposes.

It is recommended to write audit logs to both a database for immediate availability via the vendor UI as well as to HDFS for long-term destination.

Additionally, data volume for log files and audit can become huge over time. Therefore, a careful requirement analysis should be done at project start as the cost for storage and analysis of metadata can be significant.

Often organizations have already a metadata or audit framework running for their legacy systems. Some vendors like Informatica, Splunk or IBM support the integration of the mentioned Hadoop metadata tools as source system, which allows to keep the information about data flow in one place.

Auditing Key Takeaways

- Due to regulation and compliance it is often required to keep an audit trail
- Audit logs are the starting point for further analysis
- Audit logs should be stored in a database for immediate availability, and in HDFS for long-term storage

Disaster Recovery and Backup: How to recover from cluster failure

Disaster Recover (DR) enables business continuity for significant data center failures beyond what high availability features of Hadoop components can cover.

Computer systems generally support DR in three ways: backups, replication and mirrors. A backup is usually cold storage which means it is not readily available in due time. Replication aims to provide a close resemblance of the production system by replicating data on a scheduled interval. Replication can also be used within the cluster to increase availability and reduce single points of failure. A mirror is usually an exact copy of the production system with virtually no delay and is setup as a failover instance of the production system.

The Hadoop ecosystem supports different backup mechanisms depending on the component in question.

Distributed filesystems like HDFS and MapR-FS support replication to another remote cluster using a scheduled file copy mechanism. There is as of time of writing no real-time mirroring of file data available for these platforms. On the other side, components like HBase or Kafka support a near real-time mirroring of data. Further, many big data tools need a relational database as backend, which also need to be covered by a backup concept.

The big data vendors provide solutions that make it easier to manage backups of their main systems. Examples for such solutions are the Backup and Disaster Recovery feature in Cloudera Manager, Apache Falcon (Hortonworks) and the MapR Control System.

Disaster Recovery Key Takeaways

- Not all data can be replicated in real-time
- Data should be categorized into critical/non-critical and high/low time to recovery according to business requirements
- A remote cluster offers the shortest time of recovery, but is the most expensive solution

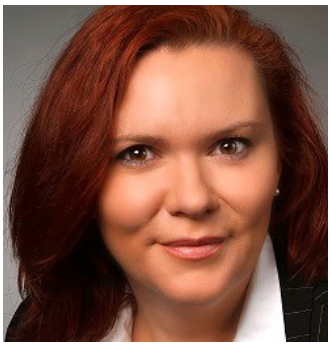
Conclusion

As the Big Data platforms consist of a zoo of ever changing technologies, a good security setup is a major challenge. Experience has shown that it is recommended to consider security aspects of a Hadoop initiative upfront.

For a comprehensive security concept all the topics discussed need to be considered: authentication, authorization, encryption and data masking, auditing, backup and recovery. Different maturity levels can be defined for a security initiative.

Hadoop components should be chosen with regard to security aspects and support by the Hadoop vendor. Some components like Solr, Spark, Kafka or Storm might not be fully supported by a Hadoop vendor out-of-the-box with regard to the specific enterprise security requirements. New security features are constantly being developed and support by the Hadoop vendor underlies change.

There is no one-size-fits-all solution to the specific enterprise security requirements. Deloitte has worked with numerous clients and Hadoop vendors to resolve these security challenges and can help crafting a Hadoop security strategy that is tailored to the specific business and security requirements.



Sandra Bauer
Director
Analytics & Information Management

Mobile: + 49 (0)173 344 3105
Email: sabauer@deloitte.de

Fabian Hefner
Manager
Analytics & Information Management

Mobile: + 49 (0)151 5800 4595
Email: fhefner@deloitte.de

Dr. Tobias Ahnert
Deloitte Analytics Institute

Mobile: + 49 (0)151 5807 0066
Email: tahnert@deloitte.de