WHITE PAPER
JUNE 2016

IT@INTEL

Big Data: Securing Intel IT's Apache Hadoop* Platform

Chandhu Yalla
Big Data Engineering Manager, Intel IT

Angela Gill
Strategic Business Development Manager, Intel Data Center Group

Menka Gupta
Big Data Program Manager, Intel IT

Mohankumar H
Big Data Systems Engineer, Intel IT

Terry McCloskey
Big Data Systems Engineer, Intel IT

Lauri Minas
Contributing Author, Intel IT

Nghia Ngo
Big Data Architect, Intel IT

Sinforoso Tolentino
Big Data Systems Engineer, Intel IT

Darin Watson
Big Data Platform Architect, Intel IT

Using Apache Sentry* and Cloudera Navigator*, we secured our Apache Hadoop* platform at the perimeter, access, visibility, and data levels.

Executive Overview

Intel IT values open-source-based, big data processing using Apache Hadoop* software. Last year we migrated from the Intel® Distribution for Apache Hadoop software to the Cloudera Enterprise* software.

Based on our original experience with Apache Hadoop software, Intel IT identified new opportunities to reduce IT costs and extend our business intelligence capabilities. We use Cloudera enterprise-grade tools to improve the performance, management, and security of our Hadoop platform.

The transition to Cloudera Enterprise especially offered us the ability to boost Hadoop security by replacing perimeter-only security with a far more secure Kerberos*-based platform. Our platform is now protected by centralized and fine-grained authorization and encryption using Apache Sentry* and Cloudera Navigator* Key Trustee capabilities.

Careful planning, thorough testing, proactive communications, and phased implementation helped enable the first big data platform within Intel that is certified to host data that is Intel Top Secret.

To create a more secure Hadoop platform we focused on the following areas:

- **Perimeter.** Guarding access to the cluster itself with authentication and network isolation.
- **Access.** Defining what users and applications can do with data, utilizing permissions and authorization.
- **Visibility.** Reporting about data origin and use, using audits and data lineage management.
- **Data.** Protecting data in the cluster from unauthorized visibility using encryption, tokenization, and data masking.

Our Hadoop security best practices enhance security, privacy, and legal compliance at Intel. They also make it possible for Intel's business groups to broaden their scope for business intelligence analytics with the knowledge that our Hadoop platform has a significantly improved security posture.

Contents

Acronyms

ACL	access control list
DLP	data loss prevention
HDFS	Hadoop Distributed File System
Intel® AES	Intel® Advanced Encryption Standard
Intel® AES-NI	Intel® Advanced Encryption Standard New Instructions
LDAP	Lightweight Directory Access Protocol
POSIX	Portable Operating System Interface
QAS	Quest Authentication Services
SSL	Secure Sockets Layer

Background

Information security is a top priority at Intel. Intel IT is continually looking for ways to increase security while supporting business groups' technology needs. Over the years, big data and advanced analytics have become increasingly important to Intel's business groups as they expand their business intelligence capabilities. As with many emerging technologies, our use of big data—and how we secure it—have evolved as the technology matured.

The Evolution of Big Data at Intel

Intel's approach to security has evolved. Our current security strategy seeks to apply reasonable protections that allow information to flow through the organization. This reduces risk while maintaining a quality user experience. The utilization of business intelligence components such as the Apache Hadoop* big data platform comprises an important component of that strategy.

Our journey with Hadoop has not been a simple one. We conducted many preproduction test cases and migrated only a few projects to production after several months of testing. In compliance with our security implementation strategy, we considered differences between the test environment and the path to production for platforms, networks, and storage.

We formed our big data and analytics strategy in 2011 and evaluated Hadoop in 2012 (see Figure 1). The first Hadoop use case involved working on a big data solution with Telmap in 2013. We invested additional resources to build out the Hadoop preproduction test environment in 2014. With this enlarged testing area, we worked on establishing enterprise standards and guidance for processes on specific platforms. From this point we moved 6 use cases into production and started 12 proof-of-concept use cases. We focused on security business intelligence, attribute reduction systems, Intel's assembly-test-manufacturing ellipses engine, and retail analytics. This journey has yielded 15 active use cases from an investment of USD 500,000 with 12 employees. The results delivered a business value of USD 182 million.[1]

[1] Yalla, Chandhu, The Evolution of Big Data at Intel. 2015 Hadoop Summit. July, 2015.

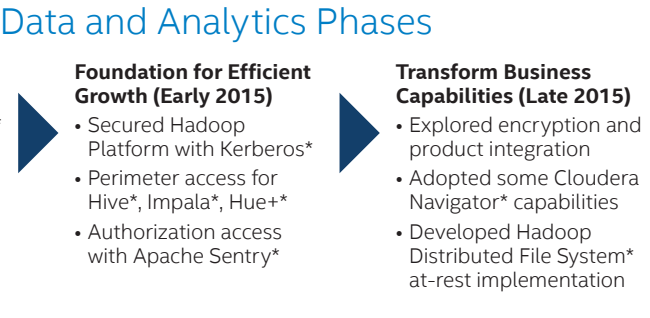

Figure 1. Time line for Intel IT's Hadoop* security implementation.

Our big data platform has evolved quickly to meet the increasing opportunities to deliver business value from big data analytics. Our journey began with deploying a low-cost, on-premises Hadoop platform for business use cases that traditionally landed on expensive massively parallel computing platforms. Migrating to Hadoop 2.0 helped to meet the demand for new high-value business use cases. We are in the process of deploying and growing several use cases that have the potential to deliver a business value of over USD 400 million. One such use case is our SMART WHAT: Marketing Automation, Cloud CRM, and Global Supply Chain Management projects. We continue to work with leading Hadoop suppliers to become familiar with our use cases, which can help move the industry forward and make Hadoop more fully enterprise ready.

Hadoop Security Challenges

Hadoop was developed in 2008. Security was not part of Hadoop's initial development, and it had no authentication of users or services. In 2009, Yahoo focused on adding authentication. But Hadoop still had limited authorization capabilities. In 2013, the Apache Software Foundation[2] launched Project Rhino to add security features to Hadoop.

For many organizations Hadoop has evolved into an enterprise data platform that stores sensitive information. Hadoop has enabled data management that is massively scalable, agile, feature-rich, and cost-effective. But as data that once was in silos is brought together in a vast data lake and made accessible to a variety of users across the organization, new security challenges arise. Addressing these challenges must ensure that following:

- Users who access Hadoop are properly authenticated.
- Authorized Hadoop users can access only the data they are entitled to access.
- Data access histories for all users are recorded in accordance with compliance regulations and for other important purposes.
- Data is protected—both at rest and in transit—through enterprise-grade encryption.

Big data that resides within a Hadoop environment may contain sensitive financial data, proprietary corporate information, and personally identifiable information such as the names, addresses, and Social Security numbers of clients, customers, and employees.

A Closer Look at Apache Hadoop*

Hadoop is an open source software framework for distributed storage and distributed processing of very large data sets on computer clusters built from Intel® Xeon® processor-based servers. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework.

Conventional supercomputer architecture relies on a parallel file system where computation and data are distributed using high-speed networking. In contrast, the core of Apache Hadoop consists of a storage part, known as the Hadoop Distributed File System*, and a processing part called MapReduce*. Hadoop splits files into large blocks and distributes them across nodes in a cluster. To process data, Hadoop transfers packaged code for nodes to process in parallel based on the data that needs to be processed. This approach allows data locality nodes to manipulate the data they have access to—resulting in faster and more efficient data processing.

[2] For information about Apache Software Foundation, visit www.apache.org

Due to the sensitive nature of this data and the potential damage that can occur if it falls into the wrong hands, it must be adequately protected. At Intel we have specific security requirements for storing Intel Restricted and Intel Top Secret information. To that end, we developed a Hadoop security strategy along with best practices to help keep sensitive data secure.

Intel IT is building the technology foundation for analytics across our entire data center infrastructure. Intel® architecture-based hardware and software security tools help block known threats, identify compromises, and expedite remediation to significantly enhance information security. Intel® Xeon® processors include security features like Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI) to boost cryptography performance, thereby allowing more data to be securely stored, transmitted, and analyzed.

The integration of Hadoop security features has posed some substantial challenges. Intel's Enterprise Security guidelines are among the most stringent in the industry in terms of authorization, authentication, encryption, management, and governance. In our initial implementations of Hadoop and Hive* (Hadoop's data warehouse), we did not include security controls—users could access data objects without authentication or authorization. We created a temporary solution where we explicitly isolated each application to store its business data in separate folders on the file system. Then we applied storage-level Portable Operating System Interface (POSIX) access control list (ACL) controls to the folders. This temporary solution was challenging to administer from the beginning; as business groups began to request access control based on support roles, the administration challenges multiplied. The POSIX permissions gave us an all-or-nothing control over users and groups. If a user belonged to multiple groups and assumed multiple "personas" (such as data stewards, data owners, data engineers, business analysts, and so on), the all-or-nothing model simply was not sufficient. The challenges were compounded when the authentication was handled by the Lightweight Directory Access Protocol (LDAP) protocols while authorization was managed at the storage layer. With that approach, we could not meet business requirements because we granted access to either everyone or no one.

Big Data, Big Opportunities for Business Value

Industry leadership is built by predicting emerging needs, delivering new experiences to meet those needs, and executing better than the competition. Part of Intel's path to achieving industry leadership has been through the analytics made possible by big data. Transforming Intel's processes with big data has helped identify new growth opportunities and enhance operational performance. This allows Intel to move faster and more efficiently than the competition.

The IT industry is being reimagined with mobile devices, the Internet of Things, and wearable technology in mind. At Intel, big data helps identify opportunities to reduce time to market and improve delivery of products that match customer needs. Big data helped deliver over USD 351 million in incremental revenue and reduce time to market for new products by up to 21 weeks.[i]

Intel invested in Cloudera as part of our belief that big data analytics will continue shifting the business paradigms in every industry. A year later, we are more certain than ever that our roadmap will bring significantly improved capabilities to enterprise analysis workflows and add new context to every business decision. Together, Intel's products and the Cloudera Enterprise* software are enabling businesses from every industry to solve untapped market challenges and deliver new industry leading profitability.

[i] How Intel's CIO Helped the Company Make $351 Million blog

Hadoop Security Solution

Part of the challenges in implementing Hadoop security features stem from its openness and flexibility. Hadoop can ingest data of any type. This can lead to sensitive data being added to a data set without triggering the appropriate security governance. Also, the Hadoop Distributed File System* (HDFS*) can use lower-cost storage on clustered servers instead of using legacy storage systems. The downside is that the built-in compliance controls that legacy storage systems provided are no longer available.

As shown in Figure 2, our Hadoop platform security strategy focuses on tightening the security of four areas: perimeter, access, visibility, and data. We use an out-of-the-box solution to minimize our team's support burden, instead of building a custom solution. We are using Cloudera Enterprise* software 5.4 Apache Sentry* and Cloudera Navigator* to provide the main security controls.

We conducted a technical evaluation of the Apache Sentry product and conducted impact analysis to determine refactoring costs of existing projects. We decided to break this effort into a phased approach. As illustrated in Figure 1, we divided the perimeter and access controls to be addressed in the first half of 2015 and the visibility and data controls in the second half of 2015. We also clearly defined a scope for each control layer and itemized a quarterly deliverable against each layer.

In the first half of 2015, we launched a project to enable LDAP for authentication and fine-grained controls for authorization using Apache Sentry. During the project, we faced several critical challenges that threatened a potential reset and an impact on our deployment time line. For example, to support Apache Sentry's features we had to incorporate multi-domain authentication with Kerberos*[3] and Apache Sentry, establish Secure Sockets Layer (SSL) certificate trusts between the front-end system and the back-end cluster, and replace Beeline* with the Hive command-line interface. Each of these tasks involved significant rework from developers. Our team worked closely with Cloudera to identify the root cause of several issues and develop patches. We also faced challenges with testing new Apache Sentry features in a resource-constrained environment.

[3] For information about Kerberos, visit www.kerberos.org

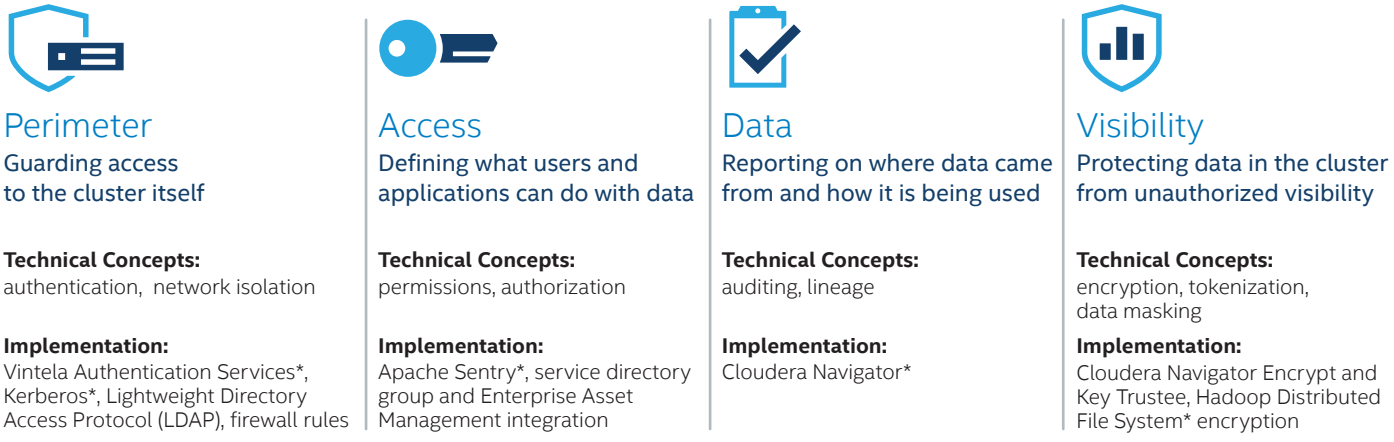

Figure 2. Intel IT's strategy for enterprise-ready Hadoop* security focuses on securing the perimeter, access, visibility, and data.

The team set up a scaled-down parallel Hadoop platform for the testing. In order to test, developers needed to make code changes to adopt the new design and syntax. In the interest of solving the logistics and time challenges, we came up with an innovative idea to minimize manual testing procedures. We developed automation scripts that refactored the old syntax with new syntax. Additionally, the scripts verified data and code integrity with the preproduction environment. This approach helped us move faster and deliver business-critical projects successfully with minimal disruption.

With a strong momentum in first half of 2015, we entered the second half of the year with a sharp focus on enabling visibility and data controls. The team began exploring Cloudera Navigator's features such as auditing and lineage. We enabled the collection service for collecting metadata from the platform periodically and stored it in a PostgreSQL database. We later switched to MySQL* because that was Cloudera's recommended database. We used the audit and lineage capabilities to help ensure appropriate access was granted and to identify unauthorized individuals who attempted to access data.

"With AES-NI, our preliminary performance evaluation with TestDFSIO shows negligible overhead for writes and only a minor impact on reads (~7.5 percent) with data sets larger than memory."

—Alex Gonzalez
Cloudera

We also enabled encryption zones for data at rest. Encryption zones are directories in HDFS whose contents are automatically encrypted on write and decrypted on read. All subsequent directory contents are encrypted. HDFS is integrated into Cloudera Navigator Key Trustee. This enables Cloudera Navigator to support the rapid read/write rates that our Hadoop workloads require. Fast read/write access is important for us to support our performance service-level agreements with Intel business groups. In addition to the Cloudera Navigator integration, HDFS uses hardware acceleration enabled by the Intel AES-NI instruction set. This provides our users with a query response that is an order-of-magnitude faster than with software implementations of Intel® Advanced Encryption Standard (Intel® AES).

Cloudera tested Navigator Key Trustee with Intel AES and Intel AES-NI. Cloudera's Alex Gonzalez commented, "With AES-NI, our preliminary performance evaluation with TestDFSIO shows negligible overhead for writes and only a minor impact on reads (~7.5%) with data sets larger than memory."[4] With this information, we defined the guidance to set up our Hadoop project with encryption zones enabled only if the project data was sensitive. This approach will avoid minimal overhead of the encryption/decryption process during data read.

At the end of 2015, we successfully rolled out encryption capability to production with one project in which data was classified as Intel Restricted.

[4] Cloudera Engineering blog. blog.cloudera.com/blog/2015/01/new-in-cdh-5-3-transparent-encryption-in-hdfs

Perimeter Security Requirements

Before implementing Apache Sentry, we had implemented basic security measures. We had mapped users to groups, assigned and locked down permissions by group, and enforced strong passwords. Hadoop runs on an OS and most of the software runs in a Java* virtual machine. We locked down the OS and Java virtual machine according to security best practices. For example, we enabled network layer firewalls, disabled root remote access, forced secure shell key pair login, used limited Sudo*, restricted root access, and removed non-required services.

The goal of perimeter security is to guard access to the cluster itself. We needed to preserve user choice of the Hadoop service, such as Impala* or Spark*. To secure the perimeter, we set a rule that all services must conform to centrally managed authentication policies.

In building our security perimeter, we started with some standards. Hadoop now supports industry-standard Kerberos to block access to non-authenticated users. With LDAP integration and a directory service, Hadoop can integrate with centralized user and identity management systems. We built our perimeter access with LDAP for Hive, Impala, and Hue*, and implemented authorization with Apache Sentry.

For the infrastructure, we installed a Quest Authentication Services (QAS) on all Hadoop nodes. User login access is controlled by adding a directory service group in the QAS user's file. QAS will return the user's information and directory service group memberships when queried by Hadoop or by the HDFS. Under Impala with Apache Sentry and LDAP implementation, the slave nodes (also known as data nodes) also verify a user's authorization directly to the directory service.

As illustrated in Figure 3, when a user logs in, he or she authenticates with the directory service, and Kerberos acts as a trusted third party and provides the user with a Kerberos ticket. The Kerberos ticket validates the user's identity. The user then accesses the needed services using the Kerberos ticket.

Finally, we use ACLs to grant access to directory service groups for both files and directories. The ACL entry types are the POSIX ACLs representations of owner, group, and others. POSIX ACLs are enabled on both HDFS and OS disks to control access to each project directory. Directory service group entitlements are also used on Apache Sentry, Hue, Cloudera Manager, Hive, and Impala. Directory service, Kerberos, LDAP, and firewall rules provide user authentication and network isolation regardless of the services used.

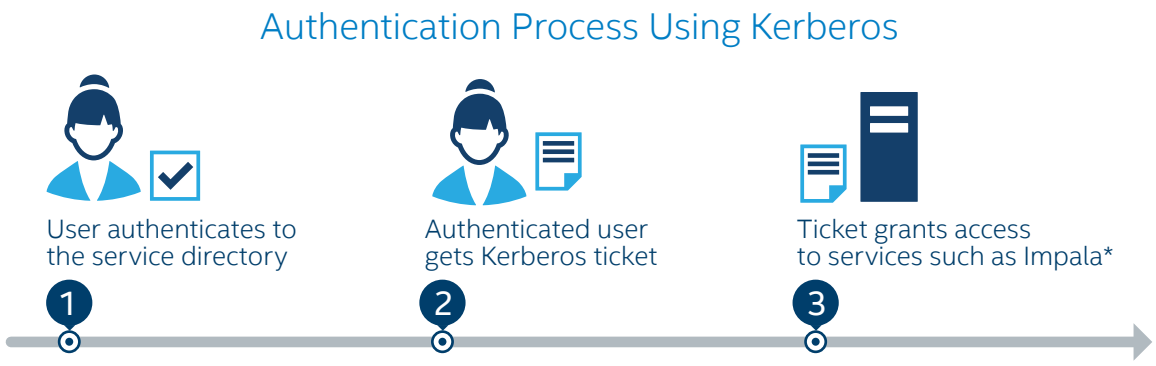

Figure 3. The authentication process with Kerberos* tickets helps secure the Hadoop* perimeter.

Access Security Requirements

The goal of access security is to help ensure that users who access Hadoop can access only the data they are entitled to access, with the same policies applied consistently regardless of how they access the Hadoop platform.

Identity Management

The first level of control is to manage identity. We use single sign-on technology to improve the user experience. After the user signs on, we use other tools to enforce access controls and use multifactor authentication controls to manage access. These controls help maintain a trustworthy computing environment by complying with Intel's Enterprise Security guidelines.

Permissions are set for users, groups, or roles by defining ACLs. We developed a role-based access control code and submitted it to the open source Apache Rhino project. The code was then merged into Apache Sentry for authentication and Hadoop security.

With Apache Sentry, when Intel employees want to access the Hadoop platform, they open a browser window on their device and then sign in using their corporate identity. Intel employees have a single identity (username/password combination) that they can use to access multiple applications. Single identity improves security and the user experience, because employees do not need to keep track of multiple passwords and thus are not tempted to write them down to remember them.

Access Controls

While managing identities is important, access controls are also necessary. We manage access controls through enrollment and entitlement workflows. We use a request/approval, role-based, and attribute-based process to manage access to Hadoop data. A context-aware access control framework grants access to the data, taking into consideration factors beyond user group membership.

For example, we can use network location to inform an access decision. In some cases it may be important that access is granted only if the user is connected to the corporate network. If we detect that the user is off-network, we can ask for a multifactor authentication to provide better assurance of identity.

After we authenticate users against services, we then need to validate the user's authorization and control data access. As illustrated in Figure 4, user access is managed and controlled using Intel's internally developed Enterprise Access Management application. Enterprise Access Management entitlements are mapped to directory service groups, which are then mapped to Apache Sentry roles.

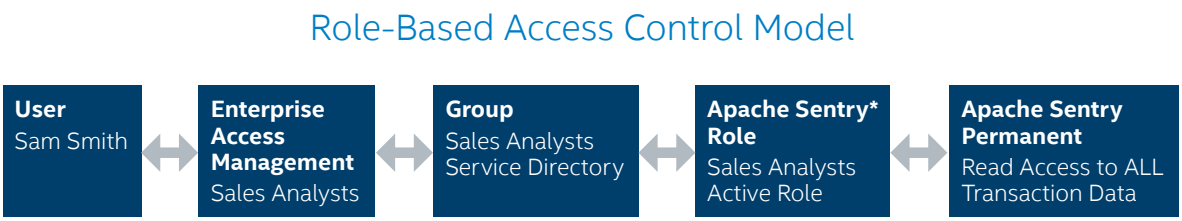

Figure 4. We use a role-based access control model that combines Apache Sentry* roles and directory service groups to grant permissions for Hive* and Impala* objects.

Apache Sentry enforces roles across Hive and Impala objects. Permissions are specified by Administrators, both top-level and delegated. An example of an Apache Sentry role is "allow sales analysts read access to the transaction table." A common enforcement code is used for consistency with access to the Hadoop components such as Hive, Impala, MapReduce*, Pig*, or HDFS.

Apache Sentry is configured to use the directory service to determine a user's group assignments. Group assignment changes in the directory service are automatically detected, resulting in updated Apache Sentry role assignments. Apache Sentry provides unified authorization through the following:

- Fine-grained, role-based access control for Impala, Hive, and Search
- Impala and Hive permissions synchronized in the HDFS for all other components such as Spark and MapReduce

Visibility Security Requirements

A crucial component of any security model is the capability to monitor, measure, and audit the security process. This is done to verify that the enterprise's security model is working as expected and that any suspect or actual security breach or non-compliance is quickly detected and resolved.

Visibility security includes understanding where a report's data came from and discovering more data like it. It also includes verifying that data and access comply with policies for audit, data classification, and lineage. We deployed Cloudera Navigator for visibility security and metadata management. In addition to providing data visibility, Cloudera Navigator also provides a centralized audit repository, performs discovery, and automates lineage. We use the Linux* Logwatcher utility to support additional Linux OS-level security monitoring.

Cloudera Navigator provides us with the following capabilities:

- Review and trigger alerts based on who has been accessing what data
- Playback access using access control permissions in Apache Sentry
- Life-cycle management of policy-based data purges

Data Security Requirements

In addition to securing the perimeter and providing access and visibility security controls, we also protect data in the cluster from unauthorized visibility. Data security essentially is enterprise-grade encryption for data at rest and in motion.

Data encryption and tokenization help us protect the data, while data loss prevention (DLP) techniques help keep sensitive information from unauthorized visibility. For some use cases, we combine encryption and DLP to provide a higher level of protection.

Due to compliance regulations—including the Health Insurance Portability and Accountability Act, Payment Card Industry Data Security Standard, and internal policies—we need to protect data from more than just unauthorized users. We have extended the protection to clear-text access over the wire using SSL, and at-rest data using Linux encryption or HDFS encryption.

Encryption and Tokenization

We encrypt both structured data and unstructured data (such as entire files). Our encryption model is a hybrid in that some encryption is done on-premises and some off-premises.

HDFS implements transparent, end-to-end encryption of data that is read from and written to HDFS without requiring changes to application code. Because the encryption is end-to-end, data can be encrypted and decrypted only by the client (end user). We also use HDFS to encrypt database columns and indexes and to implement application-level policies on file systems.

As discussed earlier, Apache Sentry supports the specification of HDFS directories as encryption zones. Encryption zones are directories in HDFS whose contents are automatically encrypted on write and decrypted on read. All subsequent directory contents are encrypted.

Strong encryption capabilities require secure key management. Our data encryption keys are stored on-premises. For encryption key management, we use Cloudera Navigator Key Trustee, a "virtual safe-deposit box" for managing encryption keys, certificates, and passwords. HDFS never directly handles or stores sensitive key material—thus it never compromises the HDFS daemons themselves (which could potentially cause leaks of sensitive Intel material). Multitenant encryption is facilitated with tenant-specific keys. Separation of tenant data is handled using key access restrictions.

When tokenization is required for data residency for some fields or data elements, they are sent to the Hadoop platform. The use of tokenization requires an on-premises database, because an unencrypted index is created.

Data Loss Prevention

DLP helps us prevent the transfer of data containing certain types of information. It has two aspects: detection and action. Detection means that we look for certain keywords or phrases (depending on policy), such as "top secret." If we detect a prohibited word or phrase, we instigate appropriate actions, such as informing the user of security policies and encrypting, quarantining, deleting, or unsharing a document.

Logging and Monitoring

Our third area of data security controls is logging and monitoring. We use Apache Sentry as the focal point for logging, monitoring, alerting, responding to information security violations, and using advanced analytics to improve our information security environment. We have established controls that detect when information security violations, anomalies, or events occur. When any of these occur, alerts are sent to the appropriate IT staff, initiating appropriate responses to correct the situation. Actionable alerts are immediately sent to our IT security information and event management system for incident response and remediation purposes. Security data is correlated and monitored for anomaly detection. All logging and monitoring also must comply with our corporate Privacy Principles and policy as this activity may directly impact employees.

When it comes to auditing, Cloudera's solution is Cloudera Navigator. Cloudera Navigator records Hadoop activity details that include the following:

- A timestamp
- The object that was accessed
- Details of the operation performed on an object
- The user's identity and IP address
- The service instance through which the data was accessed

6 Key Planning Areas for Securing Apache Hadoop*

- Find the correct skills and give the team time to ramp up.
- Build use cases to drive the project.
- Implement with a phased approach.
- Work with a supplier.
- Evaluate and test the ecosystem.
- Use automation scripts where possible.

Best Practices for Securing Hadoop

To minimize risk with Hadoop data, we established the following best practices:

- Develop a security strategy and build a Hadoop security reference architecture that reflects that strategy.
- Balance risk and productivity.
- Implement Hadoop distribution security controls.
- Keep up with Hadoop and security technology development.

Our successful implementation to secure Hadoop required careful planning in the following key areas:

- **Find the correct skills and give the team time to ramp up.** Technology is important but software engineers with the right skills are essential. We formed a small team that could implement the total solution: the platforms, the code, and network security tools. When picking the team, we made certain the team could communicate, execute, and evaluate itemized changes in the phases necessary for our Hadoop security implementation.
- **Build use cases to drive the project.** We prioritized integration of components and capabilities according to use cases, rather than trying to enable all the security functions at once. We maintained a balance of scope, timelines, testing, and resources. If it was not clear what functions to enable, we prioritized only those specified by the current set of use cases. From there the team built and expanded feature sets as needed. For preproduction implementation we used component and capability integration use cases. This helped the team measure and check the capabilities in the phased implementations.
- **Implement with a phased approach.** Progressing from our strategy to production-ready Hadoop security took 12 months. We divided the security capability implementation into phases. This helped us verify that each stage had successfully implemented software and rules before we started layering on other software.
- **Work with a supplier.** To determine the best methods for implementing data security, we consulted with Cloudera on a number of topics for securing Hadoop, such as data compression, data security, and data transfer standards, the metadata backup and restore process, user accessibility guidelines (using Hue), Apache Sentry integration requirements, and resource management guidelines. This saved us significant time in the project.
- **Evaluate and test the ecosystem.** In each phase, we ran tests to check support for the capability integration, impact analysis, and customer testing. We verified that the identity, authentication, tracking, and network tools were configured to support the Hadoop distribution.
- **Use automation scripts where possible.** In the third and fourth phases, we developed custom automation scripts to support the code and platform changes. We used custom scripts for the Hive metastore service and ACLs. We captured the user management, permissions, and partitioning schemas. This reduced the manual configuration and overhead of management of the security code and features with each new big data project.

Results

Intel IT is committed to protecting Intel's intellectual property and the personally identifiable information of customers and employees. We have established a method for balancing risk and productivity to achieve the appropriate level of risk tolerance. While risk may seem difficult to quantify, Intel has developed methods based on industry standards to calculate information security risk as a function of threat, vulnerability, and consequence in the context of risk scenarios. Risk scenarios are developed in conjunction with subject matter experts and information security specialists. These risk scenarios allow us to measure the probability that a specific architecture will open up a vulnerability or produce a negative business impact.

Intel IT has benefited from a Hadoop distribution that has open source as its core, not proprietary software. Using open source software components helps minimize total cost of ownership. Starting with a small test project helped keep the focus on design and scalability of the security solution.

We completed the first-quarter 2015 deliverable by implementing a secure platform with LDAP and SSL for authentication, a fine-grained authorization and role-based access controls for authorization. In the fourth quarter we completed extending security features to all big data ecosystem capabilities with Kerberos. All connections now require users to provide user ID, password, and SSL certificate in order to connect to our platform. In addition, users can access Hive or Impala objects (like a database or tables) only where permission is granted. We addressed the immediate security needs and closed existing Hadoop security gaps. These actions reduced the overhead of managing the ACLs for Hive and Impala databases and tables as well as for HDFS. Finally, data-at-rest encryption allows compliant storage of financial and Intel Top Secret information.

We can now easily manage complex authorization hierarchies that stem from multiple geographic regions, business segments, and personas. Our sales and marketing business units that requested the fine-grained access and authorization control are satisfied with our Hadoop security implementation.

After successful implementation in 2015, we plan to continue our security journey by enabling the following:

- Full life-cycle certificate management integration
- Column- and row-level data security
- Compliance reports for specific sets of events, such as repeated access attempts or other abnormal activity
- Data redaction for sensitive data (masking)

The approach we used to secure our Hadoop platform proved successful. We applied possibility thinking and assumed a measured risk to implement Apache Sentry in our platform to comply with Intel's Enterprise Security guidelines. This sets us up for continued success and brings us one step closer to meeting our Enterprise Security guidelines while still delivering to the business groups all the features expected of a tier-1 enterprise system with the agility and cost advantages of an open source Hadoop. Fully meeting Intel's Enterprise Security guidelines will become reality in the near future for our Hadoop platform.

Conclusion

Intel IT gained valuable experience from securing our Hadoop distribution. Although securing Hadoop was complex and involved multiple layers of change, the implementation was successful. Careful planning, thorough testing, proactive communication, and efficient management of scope, schedules, and skills helped us successfully implement increased security.

This is the first project that uses Apache Sentry to increase the security of the big data Hadoop environment within Intel. Now that we have established the best practices, guidance, and process for strengthening the security of our Hadoop platform, we will share our knowledge with other Intel organizations and the industry as a whole.

We performed a thorough impact analysis and understood the scope of the project. Communication and a clear articulation of the reward for the business against the risk were necessary; in fact, that was the only way we could gain Intel management support. We sized the projects, clearly documented the milestones, worked in close collaboration with Cloudera, and remained agile to quickly change course when necessary. We also developed reusable automation, migration, and test scripts that can be used for future projects. At Intel, Hadoop security is becoming a reality.

For more information on Intel IT best practices, visit **intel.com/IT**.

Receive objective and personalized advice from unbiased professionals at **advisors.intel.com**. Fill out a simple form and one of our experienced experts will contact you within 5 business days.

IT@Intel

We connect IT professionals with their IT peers inside Intel. Our IT department solves some of today's most demanding and complex technology issues, and we want to share these lessons directly with our fellow IT professionals in an open peer-to-peer forum.

Our goal is simple: improve efficiency throughout the organization and enhance the business value of IT investments.

Follow us and join the conversation:

- Twitter
- #IntelIT
- LinkedIn
- IT Center Community

Visit us today at **intel.com/IT** or contact your local Intel representative if you would like to learn more.

Related Content

Visit **intel.com/IT** to find content on related topics:

- How Intel's CIO Helped the Company Make USD 351 Million blog
- How Intel Implemented a Low-Cost Big Data Solution in Five Weeks paper
- How Intel IT Successfully Migrated to Cloudera Apache Hadoop* paper
- Intel IT Best Practices for Implementing Apache Hadoop* Software paper
- Integrating Apache Hadoop* into Intel's Big Data Environment paper

Product Information

cloudera® Cloudera Enterprise: cloudera.com

Printed in USA ♻ Please Recycle 0616/LMIN/KC/PDF