BI + Security + Hadoop

Delivering Secure BI in the Age of Hadoop

Josh Klahr
VP of Product
josh@atscale.com
@jklahr
HELLO!

My name is

Josh

Product Manager
Aka NOT a Security Expert
A Typical BI-on-Hadoop RFP

- Support for [Tableau] [Excel] [Qlik] [Etc]
- Interactive query performance
- Security
- Security
- Security
These aren’t the kind of headlines anyone wants
A good foundation is important
A foundation for data platform security

➔ **Authentication** - is this user who they say they are?
➔ **Authorization** - what is this user able to do/access?
➔ **Data Protection** - how is data kept hidden/obfuscated?
➔ **Auditing** - who accessed what data and when?
You’re in good hands.
A typical Hadoop security layout...

Authentication: Active Directory, LDAP

Application-level authentication

Service Principal
- Data Application
- BI Tool
- Command Line Tool

Kerberos, *almost always*

Hadoop Security: Authorization Auditing
- Hive Metastore
- SQL Engines (SparkSQL/Hive/Impa/)
- Encrypted Data
- Data
- HDFS

User A did this
User B did this
Congrats on locking down your data, but...

Year over year, **self-service access on Big Data grew by 15%**

However, **most companies (53%) still suffer from a lack of self-service access to Big Data**

Self-service access is one of the best ways to drive value on Big Data. **Companies who have self-service are 32% more likely to succeed**

But...

As soon as you move data off of your secure, governed Hadoop cluster, all bets are off.
Hadoop security when you start moving data...
The goal...

… deliver a BI-on-Hadoop experience that takes advantage of the Hadoop security foundation (instead of undermining it).
AtScale Raw Data Query

1. Client submits cube-scoped query to AtScale engine
2. AtScale authenticates the user using LDAP/AD
3. Query Service submits query to SQL Engine as end-user
4. SQL Engine applies data-level ACLs
5. Query is distributed to query workers
The challenge... 

… how to create a multi-user data “cache” that takes advantage of summarized data while still respecting the underlying Hadoop cluster security policies.
AtScale Adaptive Cache Query

1. Client submits cube-scoped query to AtScale engine
2. AtScale authenticates the user using LDAP/AD
3. Query Service submits canary query to SQL Engine as end user
4. SQL Engine applies data-level ACLs
5. Canary query returns failure or success with 0 rows
6. If Canary Query passes, cache query submitted as system
7. SQL Engine applies data-level ACLs
8. SQL Engine Query executed on AtScale Workspace
Checklist for success

- Support for Kerberos-secured cluster connectivity
- Integration with LDAP and Active Directory for user management
- Delegated authorization/impersonation for all end-user queries
- TLS (Transport Layer Security) support for all data connections
- Ability to work with encrypted data sets
- Any and all optimizations (cache/aggregates) respect data level ACLs
Thank You!

Questions? Comments? Discussion?

Josh Klahr
VP of Product
@jklahr