Introduce Apache Kerby to Apache Hadoop

Kai Zheng

Apache Directory PMC, Apache Hadoop committer

Intel senior software engineer

Agenda

- Introduce Apache Kerby
- Kerberos Integration Challenges in Hadoop
- How Apache Kerby Can Help Hadoop
- Q/A

An Overview: Apache Kerby

- Apache Kerby, as an Apache Directory sub project, is a Java Kerberos binding. It provides a rich, intuitive and interoperable implementation, library, KDC and various facilities that integrates PKI, and token (OAuth2) as desired in modern environments such as cloud, Hadoop and mobile.

- The Apache site: http://directory.apache.org/kerby/
- Github project site: https://github.com/apache/directory-kerby
- Apache Directory: http://directory.apache.org/
- Kerby Developers List: kerby@directory.apache.org(subscribe@directory.apache.org)

A Kerberos Client Library

- A Java Kerberos binding (besides c, python!)
- A strong Kerberos client library, full functional, compatible and flexible to talk with any KDC
- KrbClient API: request Ticket Granting Ticket(TGT), Service Granting Ticket(SGT) via all means (password, credential cache, keytab, JWT token, X509 certificate)
- Kadmin API: administrate KDC backend locally and remotely
- Keytab and credential cache utilities
- JAAS and GSSAPI support based on Kerby library

Also a KDC Server Implementation

- Offers a Simple KDC server, embeddable, and lightweight with memory or json file based backend
- Offers a standalone KDC, powered by Netty network support and equipped with LDAP and Zookeeper based back ends
- Offers a KDC server abstract, customizable, easy to develop your own KDC and plugin your own KDC backend

Nice and Strong ASN-1 Backed

- Model driven framework, quite straightforward to implement your own TYPEs given you know the ASN-1 definition
- All these types are written up upon it (*even not too much knowledge*):
  - core Kerberos codec(130+), CMS(50+), X509(70+)
- Both BER and DER are supported
- Extensively tested, good performance

More Means: Not Just Password

- It plays well with classical Kerberos protocol using password
- Also aims to support mechanisms: PKI, OTP and token (OAuth 2)
- JWT token is already supported and available to use
- Anonymous PKINIT is out, PKINIT with X509 is on going (the 1st Java library that supports PKINIT!)

Target Environments

- Look forward, no legacy, so we can move!
- Targets for modern environments:
  - Hadoop: more authentication means, easier to access
  - Cloud: all kinds of API, token and PKI support, easy to integrate
  - Mobile: core library self-contained, easy to port and migrate

Overall Status

- Highly involved and backed by Apache Directory community
- More than 1 years, 2 candidate releases, approaching 1.0.0
- 8+ PMCs/committers
- Diverse contributors contributing codes
- 8+ known users or projects powered

Agenda

- Introduce Apache Kerby
- Kerberos Integration Challenges in Hadoop
- How Apache Kerby Can Help Hadoop
- Q/A

In Hadoop, why Kerberos

- Kerberos is the right approach adopted for Hadoop security
  - ➢ Symmetric encryption, mutual authentication
  - ➢ Flexible SASL QoP, authentication (privacy) by default
  - ➢ Command line (kinit, SSO) + Browser (SPNEGO)
  - ➢ Mature, available in Linux/Windows + J2SE
- With Kerberos, never beg for a secured connection (SSL) !
- Kerberos incurs deployment overhead, let's make it easy
  - ➢ Apache Kerby is just the first step, a bootstrap !
- Want to support more mechanisms other than Kerberos ?
  - ➢ It's possible, leveraging Apache Kerby
  - ➢ It's doable, involving limited change

Lacking a Java Kerberos Library

- Java lacks a comprehensive Kerberos library. The Kerberos support in Java/JRE is
  - Limited, lacking full encryption and checksum types
  - Hidden from GSSAPI/SASL layers

Dynamic Application/Container Provisioning

- How to provisioning dynamic applications or containers with security enabled securely?
  - How to prepare the principals
  - How to configure the runtime environment (MIT Kerberos client package?)
  - How to distribute the credentials
- Typically seen: YARN, Slider, Streaming frameworks

Integration, Management and HA

- Not able to plugin customized KDC back end
- Hard to integrate Kerberos accounts into existing management system, better to provide kadmin side library in Java
- Easier readable logs for big data talents
- Familiar and reusable way to support high availability and failover?

Agenda

- Introduce Apache Kerby
- Kerberos Integration Challenges in Hadoop
- How Apache Kerby Can Help Hadoop
- Q/A

How Apache Kerby Can Help Hadoop

- Kerby is on the way to solve these challenges, though the progress is fairly good, still far from ideal
- Integrate with dominant authentication methods in enterprise, cloud and internet companies
- TokenPreauth: A brand new token authentication mechanism
- Kerby KDC: High efficient, high availability, auto-failover

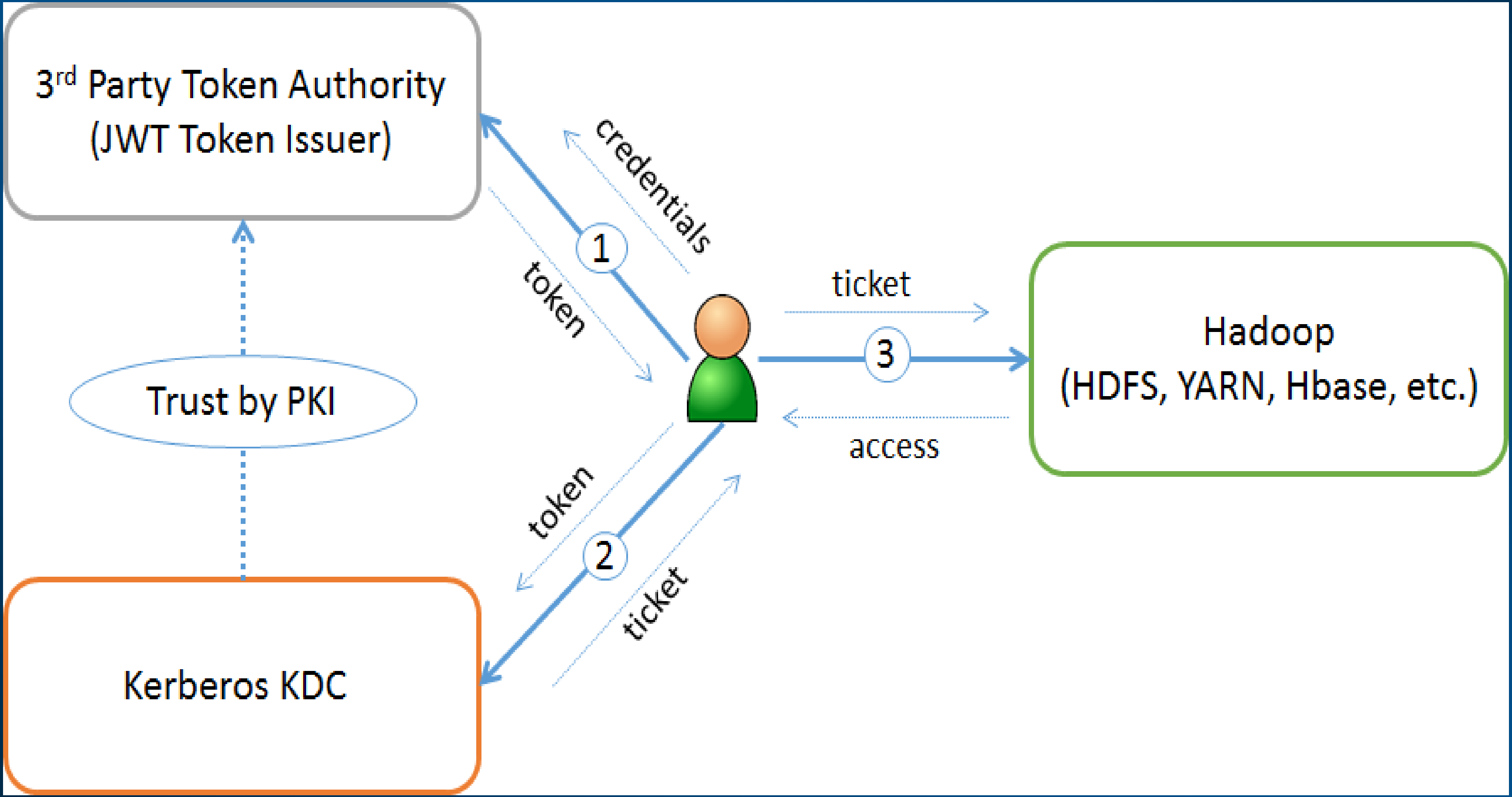
Kerberos Pre-Authentication mechanisms

- Kerby’s important targets: preauth mechanisms
- FAST: the preauth framework, providing secure channel and facilities for new mechanisms
- PKINIT: authentication using x509 certificate
- OTP: using One Time Password
- Token: using a JWT token, but more than authentication, also carrying identities and authorization attributes

Good: JDK-8044085, our extension proposal accepted and committed: allowing querying authorization data field of service ticket.

TokenPreauth mechanism

Allows user to authenticate to KDC using 3rd party tokens instead of password



TokenPreauth mechanism (cont’d)

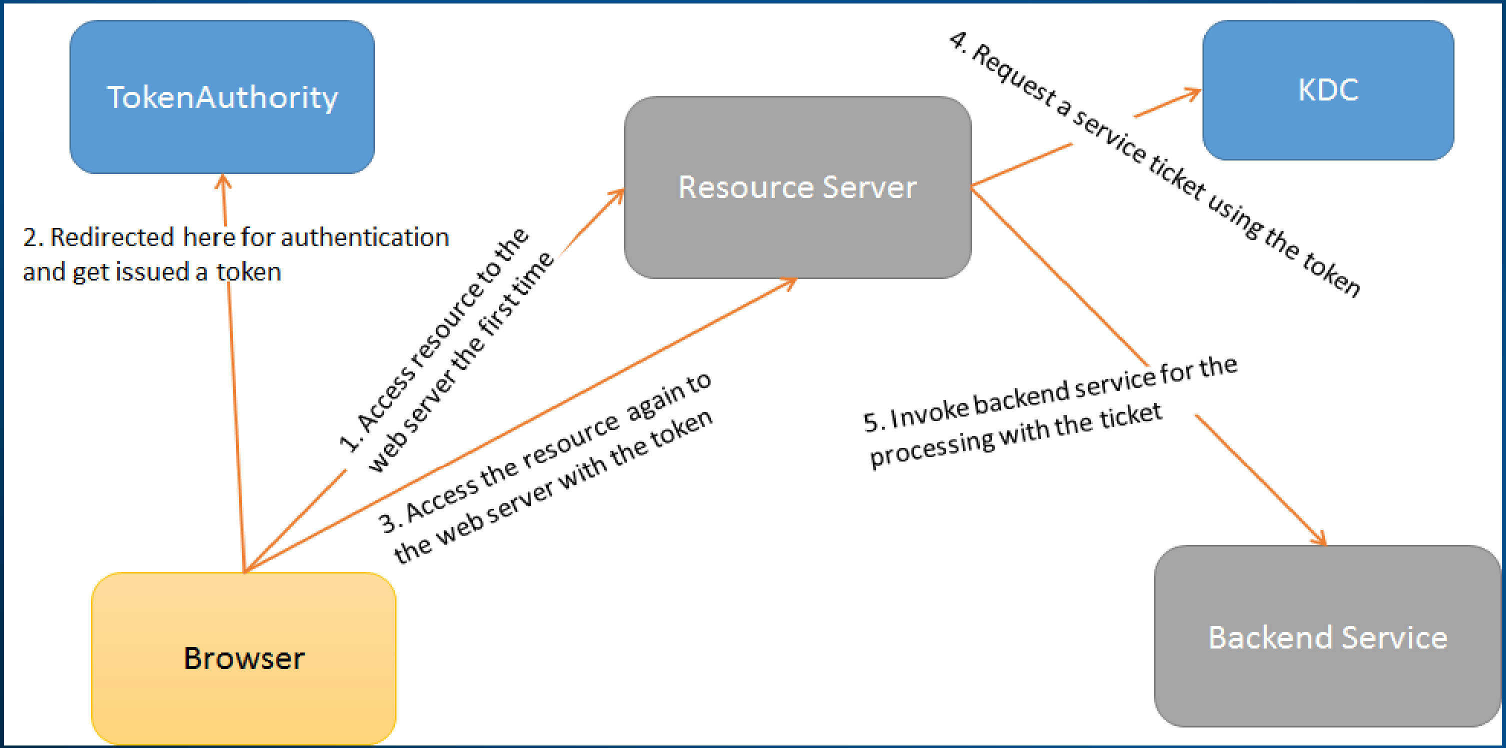
- Defines required token attribute values based on JWT token, reusing existing attributes
- Support Identity Token and Access Token

TokenPreauth mechanism (cont'd)

- Client principal may exist or not during token validating and ticket issuing
- kinit –X token=[Your-Token], by default ref. ~/.kerbtoken
- How token being generated may be out of scope, left for token authority
- Identity Token -> Ticket Granting Ticket, Access Token -> Service Ticket
- Ticket lifetime derived from token SHOULD be in the time frame of the token
- Ticket derived from token may be not renewable

Access Token profile for Kerberos

- Based on TokenPreauth, allow Access Token to be used to request Service Ticket directly in AS exchange
- Should be useful to support OAuth 2.0 Web flow to favor Resource Server accessing Kerberized backend service


TokenAuthority
Resource Server
KDC
Browser
Backend Service
1. Access resource to the web server the first time
2. Redirected here for authentication and get issued a token
3. Access the resource again to the web server with the token
4. Request a service ticket using the token
5. Invoke backend service for the processing with the ticket

TokenPreauth, why it matters

- Token and OAuth are widely used in Internet, cloud and mobile, more and more popular
- It allows Kerberized systems to be supported in token's world
- Also allows Kerberized systems to integrate other authentication solutions thru token and Token Authority, without modification of existing codes.
- May help Kerberos evolve in both cloud and big data platform
- Make extra sense for Hadoop, supporting token across the ecosystem without performance impact

Kerby KDC, Hadoop oriented

- A standalone KDC solution, with BigData in mind:
  - Netty powered
  - Zookeeper cluster for the back end
  - Multiple KDCs active to active for clients
  - Auto failover
  - Admin server with remote Kadmin API and facility

TokenPreauth, how it is going

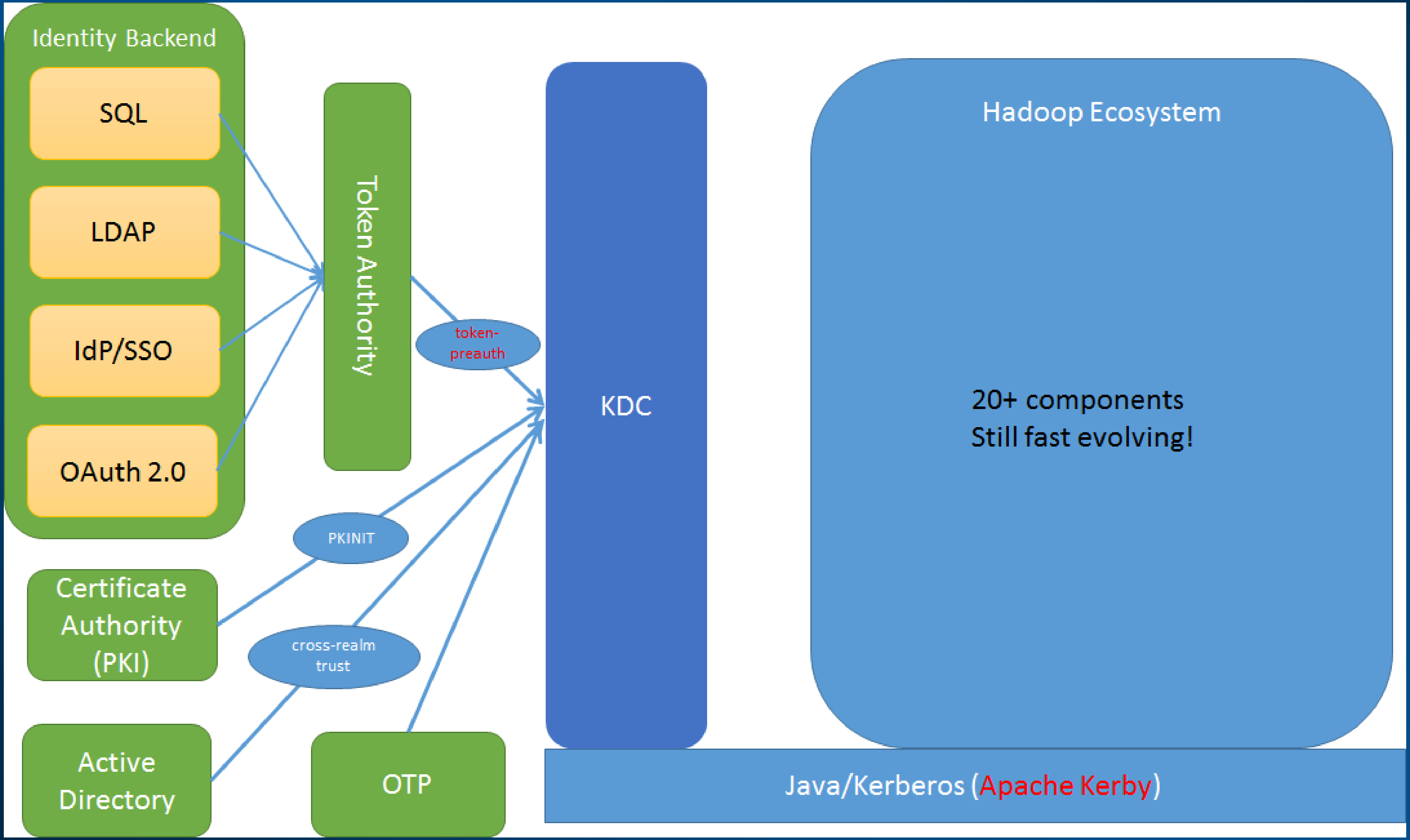
- MIT likes this, collaborating with MIT, standardize the mechanism, having two initial drafts under MIT team's review. Original plan:
  - Submit to KITTEN before IETF 92
  - Implement in next major MIT Kerberos release
- It's pending for sponsoring. We're evaluating and desiring more input and feedback
- PoC done targeting for Hadoop, thru MIT Kerberos, Java, Kerby, and Hadoop (HADOOP-10959)
- Implemented and available in Apache Kerby now!

Near Term

- Introduce Apache Kerby into Hadoop (3.0)
- Refresh and complete security tests
  - HADOOP-12911 Upgrade Hadoop-miniKDC, no dependency and low overhead
  - Complete security tests, secured mini clusters
  - Easy to support, doable in all components, including HBase, Hive, …
  - Facility tools and helpers

Long Term, Kerby-rized Hadoop

- Let’s combine all of these together



Hadoop Authentication Server (HAS)

- Even longer, think about

Hadoop Authentication Server (HAS)

?

- Kerberos is essentially a protocol, or secure channel, doesn't have to be that complex to most or normal users, hiding the details
- How about leveraging Kerberos as a secure channel, Apache Kerby as an internal authentication hub, think about HAS as the system level authentication service ?

Agenda

- Introduce Apache Kerby
- Kerberos Integration Challenges in Hadoop
- How Apache Kerby Can Help Hadoop
- Q/A