Cloudera Professional Services

Security Engagement Prerequisites

**Prepared by: CenturyLink**

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision History** | | | |
| **Version** | **Author** | **Description** | **Date** |
| 0.1 | Mubashir Kazia | Initial version | 2014-10-01 |
| 0.2 | Eddie Garcia | Updates to encryption-at-rest | 2014-10-02 |
| 0.3 | Ben Spivey | Moved to new template | 2014-10-04 |
| 0.4 | Ben Spivey | Minor grammar changes | 2014-10-07 |
| 0.5 | James Kinley | Updated to new template and merged with authentication prerequisites | 2015-03-30 |
| 0.6 | Mubashir Kazia | Replaced scary language around using AD with sensible language, removed the incorrect bash principal/keytab creation script, fixed links, added section for HDFS encryption and clearly explained we need 2 servers for KTS and two for KMS. | 2015-05-22 |
| 1.0 | James Kinley | Replaced architectural considerations with security reference architecture, updates from field feedback, added checklist | 2015-10-21 |
| 1.1 | James Kinley | Added note about Kerberos supported encryption types with older Windows server versions | 2016-01-06 |
| 1.2 | Mubashir Kazia |  | 2016-01-29 |
| 1.3 | Dave Beech | Reworded sentence concerning requirement for local Unix accounts | 2016-04-25 |

Contents

1 Introduction 5

2 Professional Services Prerequisites 5

3 Introduction to Hadoop Security 6

3.1 Background 6

3.2 Kerberos Principals and Keytabs 6

3.3 Identity Management 7

3.4 Group Membership 8

4 Security Reference Architecture 9

4.1 Logical Architecture 9

4.2 Architecture Summary 9

5 Authentication and Authorization Prerequisites 13

5.1 License Keys 13

5.2 Root/sudo access 13

5.3 Authentication Packages 13

5.4 Identity Management Integration 13

5.5 Active Directory OU and OU user 14

5.6 Enable SSL/TLS on Active Directory 14

5.7 Principals and Keytabs 14

5.8 Active Directory Bind Account 15

5.9 Active Directory Groups for Privileged Users 15

5.10 Active Directory Groups for Cloudera Manager and Navigator Role 16

5.11 Sentry and Navigator Databases 16

5.12 Active Directory Test Users and Groups 16

5.13 Multiple Realms/Active Directory Domains 16

5.14 SAML configuration 16

5.15 TLS/SSL Certificates 17

5.16 Firewall 18

5.17 Windows Registry Key Setup 19

5.18 Admin/Root Access for Installing ODBC Drivers 19

6 Encryption Prerequisites 20

6.1 Navigator Key Trustee 20

6.2 HDFS Encryption 21

6.3 Navigator Encrypt 22

7 Security Prerequisites Checklist 23

Appendix A: Example MIT KDC Configuration 25

Appendix B: Example Kerberos Client Configuration 26

Appendix C: Example Custom Kerberos Keytab Retrieval Script 27

Appendix D: Example Scripts for Creating Principals and Keytab Files in Active Directory 28

# Introduction

This document describes Cloudera's Security Reference Architecture for configuring authentication, authorization, and encryption in an Enterprise Data Hub (EDH). Its aim is to explain to the reader Cloudera's recommended approach for setting up Hadoop security in enterprise environments and the associated prerequisite steps that need to be completed prior to a Cloudera Professional Services engagement.

* **Note**: Should the reader identify any potential exceptions to following the Security Reference Architecture (for example, having to use a local MIT KDC instead of integrating directly with Active Directory) then these should be communicated to Cloudera prior to any Professional Services engagement.

# Professional Services Prerequisites

Prior to Cloudera Professional Services arriving onsite the customer must complete the Cloudera Professional Services Prerequisites[[1]](#footnote-1) and the prerequisites from this document.

# Introduction to Hadoop Security

## Background

Hadoop security leverages Kerberos[[2]](#footnote-2) to perform user authentication and prevent malicious user impersonation. Kerberos is a network authentication protocol designed by Massachusetts Institute of Technology (MIT) to provide strong authentication for client/server applications. Kerberos forms the basis for many authentication systems such as MIT Kerberos on Linux and Microsoft Active Directory on Windows.

Without Kerberos enabled, Hadoop makes no effort to verify that a user is who they say they are. Users can easily impersonate other users to circumvent any authorisation controls that have been put in place (e.g. HDFS file permissions).

With Kerberos enabled, users must first authenticate with a Kerberos Key Distribution Centre (KDC) to obtain a valid Ticket-Granting-Ticket (TGT). This ticket is used by Hadoop to ensure that a user is authenticated. Microsoft's Active Directory provides a Kerberos KDC as well as LDAP directory services.

Kerberos authentication coupled with Hadoop's authorisation mechanisms, such as HDFS posix permissions and Sentry, gives a robust security model to protect access to data in a cluster.

## Kerberos Principals and Keytabs

A user in Kerberos is called a **principal**, and a principal is made up of three components: primary, instance, and realm:

* The **primary** component is an arbitrary string that is typically a user's operating system username or the name of a service (e.g. hdfs);
* The **instance** component is an optional string that qualifies the primary. It is separated from the primary by a slash (/).
  + For example, Cloudera Manager's principal qualifies its primary as an admin instance (e.g. cloudera-scm/admin@REALM.COM) and the Hadoop service principals use the instance to define the host on which a service runs (e.g. hdfs/fully.qualified.domain.name@REALM.COM);
* The **realm** is used to logically define a related group of principals. Large organizations commonly create distinct realms to delegate administration of groups within the enterprise. A Hadoop cluster deployment will typically have its own realm. Realms, by convention, are written in uppercase.

The Hadoop service principals have the form:

service\_name/fully.qualified.domain.name@REALM.COM, where service\_name is the username of an existing Unix account that is used to run a Hadoop process, such as hdfs or yarn. These are commonly referred to as Service Principal Names (SPN).

Users who want to access the Hadoop cluster will also need to have Kerberos principals. User principals typically have the form: username@REALM.COM, where username refers to the username of the user's Unix account. User principals typically do not include the instance component.

There are two ways to authenticate a principal in Kerberos: password login or keytab. Users will typically authenticate via a password login and Hadoop processes will authenticate via keytab files.

A **keytab** is a file containing one or more Kerberos principals and their encrypted keys. A keytab file is used to authenticate a principal to Kerberos without human interaction. Because having access to a keytab file allows one to act as the principal(s) it contains, the keytab files should be tightly secured (i.e. readable by a minimal set of users, stored on the local disk, and not included in the host backup, unless access to those backups is as secure). Every Hadoop process running in the cluster has it own keytab file since the principals are unique to each service and host.

## Identity Management

A secure Hadoop cluster requires that all users (including service users) have Unix accounts available to every cluster host. These accounts are used to own the executing Hadoop processes and job tasks. For example, the hdfs service account will own the NameNode and DataNode processes, and a given user account will own the executing tasks for a submitted job.

Cloudera recommends that all the service accounts be created locally on the hosts in the cluster. There is no additional step to create these accounts, as installation of Cloudera software will automatically create them.

Cloudera recommends that all user accounts and groups be managed in Active Directory (or other identity management system) and the cluster's Linux hosts be configured to cache these accounts using integration software such as SSSD[[3]](#footnote-3), Centrify, or QAS/VAS. This provides fast lookup for user and group information that is critical for proper functioning of Hadoop Authorization.

* **Note**: Prior to a Cloudera Professional Services engagement the customer should have their desired identity management solution and Linux identity integration in place.

## Group Membership

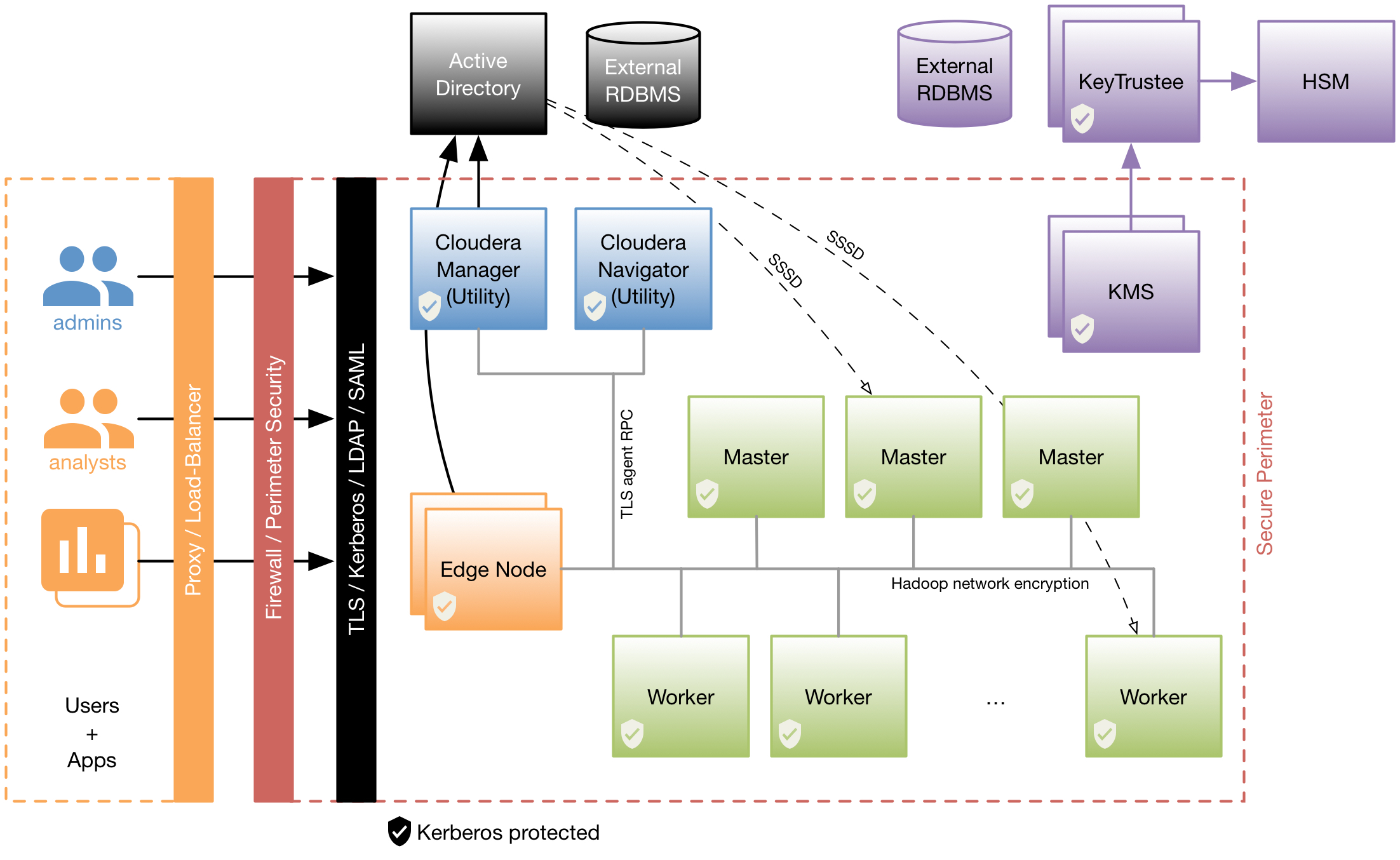
By default Hadoop retrieves user's group membership information from Linux, which in turn can use local groups or groups defined in the central identity management system, as discussed above.

Hadoop can also be configured to retrieve group membership information directly from LDAP. Although, care must be taken when choosing this option because it will incur additional network traffic and may cause degraded cluster performance. This can also cause significant additional load in some cases on the backend LDAP servers and is not recommended.

# Security Reference Architecture

This section describes Cloudera's Security Reference Architecture for configuring authentication, authorization, and encryption in an Enterprise Data Hub (EDH). Not every customer will be able to setup Hadoop security inline with these recommendations (e.g. due to individual security policies) so areas where alternate approaches can be taken are highlighted.

## Logical Architecture



## Architecture Summary

The Security Reference Architecture focuses on four key elements:

* **Perimeter security**: guarding access to cluster services and only exposing specific interfaces to external business users, applications, and networks (firewalls and authentication)
* **Access control**: controlling what data users and applications can access, and what compute resources they can utilise (authorisation and multitenancy)
* **Data visibility**: reporting on where data is and how it is being used (auditing and lineage)
* **Data protection**: protecting the visibility of sensitive data (encryption)

**Firewall**

The first line of defence for managing access to cluster services is a firewall. A correctly configured firewall will secure the cluster perimeter and only allow access to specific services, such as Cloudera Manager, Navigator, and the edge node interfaces. A full list of services and ports can be found in Cloudera's online documentation: <http://www.cloudera.com/content/www/en-us/documentation/enterprise/latest/topics/cm_ig_ports.html>.

**Authentication**

Cloudera recommends integrating directly with Active Directory for cluster authentication. In this architecture:

* All service and user principals are created in Active Directory
* All cluster hosts are configured to authenticate with Active Directory using krb5.conf
* Cloudera Manager will connect directly to Active Directory to create and manage the service principals for the CDH services. To do this Cloudera Manager uses its own Active Directory account that has the privileges to create other accounts within a given Organisational Unit (OU)
* All service and user principals are authenticated by Active Directory
* All cluster hosts cache user accounts and groups from Active Directory using SSSD (or other Linux/AD integration tool)
* Active Directory is used to authenticate users with Cloudera Manager, Cloudera Navigator, Hue, and Impala. Cloudera Manager, Navigator, and Hue also support SAML authentication for Single Sign-On (SSO)
* **Note**: If it is not possible to create Cloudera Manager its own Active Directory account with the required privileges, then the CDH service principals will need to be created manually. The corresponding keytab files can then be securely stored on the Cloudera Manager host before being imported into Cloudera Manager using a Custom Kerberos Keytab Retrieval Script.  
    
  Please refer to the appendix for example scripts for manually generating the CDH service principals in Active Directory, and an example Custom Kerberos Keytab Retrieval Script.
* **Note**: If it is not possible to integrate directly with an Active Directory KDC for Kerberos authentication then the alternative is to setup a local MIT KDC. Cloudera Manager will connect directly to the local MIT KDC to create and manage the CDH service principals.   
    
  A cross-realm trust can then be established between the local MIT KDC and a central Active Directory KDC for user authentication. If it is not possible to setup a cross-realm trust then a Hadoop administrator must create and manage all users accounts in the local MIT KDC.  
    
  <http://www.cloudera.com/content/www/en-us/documentation/enterprise/latest/topics/cm_sg_kdc_def_domain_s2.html>  
    
  <https://www.centos.org/docs/5/html/5.2/Deployment_Guide/s1-kerberos-server.html>

**Authorisation and multitenancy**

* HDFS Posix permissions and extended ACLs are used to secure directories and files
* Sentry is used to provide fine-grained access control to Hive/Impala/Solr objects
* Static service pools (cgroups) are used to divide up compute resource at the CDH service level
* Yarn/Impala dynamic resource pools are used with ACLs to control user and group access to compute resource
* Active Directory user accounts and groups being cached in Linux are used for authorisation in HDFS, Sentry, and Yarn

**Auditing and lineage**

* Cloudera Navigator is used for auditing, lineage, and business metadata capabilities

**Encryption**

* If encrypting data at rest is a requirement, Cloudera recommends the use of HDFS-level encryption, which can be applied only to HDFS directories where it is needed
* Cloudera Navigator Key Trustee is used for enterprise-grade reliable key storage. The default Java keystore should be used for test purposes only
* If network-level encryption is a requirement then this can be enabled for data sent from client user interfaces, service-to-service communication (e.g. RPC), and Cloudera Manager Agent communication. This protection uses industry-standard protocols such as SSL/TLS. Even when network-level encryption is not a requirement for data itself, Cloudera recommends that certain interfaces are encrypted so that the end-user or the service account credentials are not transmitted in clear text.

# Authentication and Authorization Prerequisites

## License Keys

License keys to any new components have to be acquired by the customer from Cloudera. For Encryption at rest with Navigator KTS and/or Cloudera Navigator an EDH license is required.

## Root/sudo access

Additional packages will be installed from the OS repository. Root access or sudo access to root will be needed to install additional packages.

## Authentication Packages

The following Kerberos package and its dependencies will be installed on all hosts and the OpenLDAP package and its dependencies will be installed on the Cloudera Manager Server host.

| **OS** | **Packages to be Installed** |
| --- | --- |
| RHEL/CentOS 5 & 6 | openldap-clients on the Cloudera Manager Server host  krb5-workstation on ALL hosts |
| SLES | openldap2-client on the Cloudera Manager Server host  krb5-client on ALL hosts |
| Ubuntu or Debian | ldap-utils on the Cloudera Manager Server host  krb5-user on ALL hosts |

## Identity Management Integration

All cluster hosts should be integrated with the customer's internal Active Directory infrastructure using SSSD, Centrify, Quest Authentication Services (QAS), or similar tool. It is also recommended that a mechanism be put in place to allow users to login to the edge nodes only. Only admin users should be able to login to the cluster hosts.

* **Note**: If Centrify, QAS, or similar tool is chosen then the customer may have to purchase licenses from the respective vendor. This should be planned in advance as it may take some time to procure license and configure the tool.
* **Note**: Care should be taken to ensure that the identity management tool does not associate the service principal names (SPNs) with the host principals when the hosts are joined to the Active Directory domain. For example, Centrify by default associates the HTTP SPN with the host principal, so the HTTP SPN should be excluded when the hosts are joined to the domain. More information can be found here <https://docs.centrify.com/en/pdf/css/suite2015/centrify-Cloudera-guide.pdf>

* **Note**: When using SSSD to integrate with AD, AD should support POSIX schema. This is available in AD version 2008 onwards. For other versions of AD additional AD components to support POSIX schema may need to be installed. Unique IDs should be assigned to LDAP attributes for uidNumber and gidNumber

## Active Directory OU and OU user

The customer should create a separate, dedicated OU in Active Directory for the cluster, and an account provided that has privileges to create additional accounts in that OU. More information on how to setup the account for delegation can be found here <http://blog.cloudera.com/blog/2014/07/new-in-cloudera-manager-5-1-direct-active-directory-integration-for-kerberos-authentication>

## Enable SSL/TLS on Active Directory

The customer should enable SSL/TLS in Active Directory and Cloudera Manager should be able to connect to Active Directory on the LDAPS port (TCP 636).

## Principals and Keytabs

When implementing the recommended direct-to-AD approach all required service principals and keytabs are created and managed by Cloudera Manager. If the direct-to-AD approach cannot be used then the customer should manually create the service principals in Active Directory and provide associated keytab files.

The accounts will have the User Principal Name (UPN) set to service/fqdn@REALM. The account will also have the Service Principal Name (SPN) set to service/fqdn. The principal name in the keytab file will be the UPN of the account. The keytab files will have the naming convention servicename\_fqdn.keytab.

The following principals need to be created for each host they run on:

accumulo/fqdn@REALM

flume/fqdn@REALM

hbase/fqdn@REALM

hdfs/fqdn@REALM

hive/fqdn@REALM

HTTP/fqdn@REALM

httpfs/fqdn@REALM

hue/fqdn@REALM

impala/fqdn@REALM

kms/fqdn@REALM

mapred/fqdn@REALM

oozie/fqdn@REALM

sentry/fqdn@REALM

solr/fqdn@REALM

spark/fqdn@REALM

webhcat/fqdn@REALM

yarn/fqdn@REALM

zookeeper/fqdn@REALM

## Active Directory Bind Account

The customer should create an Active Directory account that can be used for LDAP binding in Cloudera Manager, Navigator, Hue and Impala for LDAP(s) authentication.

## Active Directory Groups for Privileged Users

Groups should be created in Active Directory and corresponding members should be added for authorized users, HDFS admins, and HDFS superuser groups.

The purpose of these groups:

* Authorized users: a group consisting of all users that need access to the cluster
* HDFS Admins: group of users that need to run HDFS administrative commands
* HDFS superusers: group of users that have superuser privilege (r/w access to all data and directories in HDFS)

It is not recommended to put regular users into the HDFS superuser group. Instead, an account that administrators escalate to should be part of the HDFS superuser group.

## Active Directory Groups for Cloudera Manager and Navigator Role

Groups should be created in Active Directory and corresponding members should be added for the roles that will be used for role-based access in Cloudera Manager and Navigator.

The Cloudera Manager roles and their definitions are available here:

* <http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cm_sg_user_roles.html>

Cloudera Navigator roles and their definitions available here:

* <http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cn_user_roles.html>

## Sentry and Navigator Databases

Create a separate database for storing Sentry and/or Navigator metadata. Admin credentials for these databases will be requested during Sentry and/or Navigator installation.

For an up-to-date list of supported databases visit: <http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cdh_ig_req_supported_versions.html?scroll=topic_2_unique_5>

For an up-to-date list of supported databases for Navigator visit:

<http://www.cloudera.com/content/cloudera/en/documentation/core/latest/topics/cn_iu_reqts.html#concept_oqr_j4k_np_unique_1>

## Active Directory Test Users and Groups

At least one Active Directory user and group should be provided to test HDFS permissions, Sentry, and Yarn resource pools.

## Multiple Realms/Active Directory Domains

If users from multiple Kerberos realms will be using the cluster, then at least a one-way trust is needed so that users from external domains are trusted in the realm the cluster is secured.

## SAML configuration

If SAML authentication is required then the following steps must be completed:

1. Generate a private key and certificate pair for signing/encrypting SAML data
2. Retrieve the SAML metadata XML file from your IDP. This file must contain the public certificates needed to verify the sign/encrypt key used by the IDP per the SAML Metadata Interoperability Profile
3. Get the entity ID that should be used to identify the Cloudera Manager/Navigator/Hue instance
4. Know how the user ID is passed in the SAML authentication response:
   1. As an attribute. If so, what identifier is used?
   2. As the NameID
5. Know the method by which the Cloudera Manager/Navigator/Hue role will be established:
   1. From an attribute in the authentication response?
      1. What identifier will be used for the attribute?
      2. What values will be passed to indicate each role?
   2. From an external script that will be called for each request
      1. The script takes user ID and $1
      2. The script sets an exit code to reflect the assigned role (see below)
      3. A negative value is returned for a failure to authenticate

### Script exit codes:

* 0 - Administrator
* 1 - Read-only
* 2 - Limited operator
* 3 - Operator
* 4 - Configurator

### Additional SAML packages for Hue

For SAML authentication with Hue these additional packages will need to be installed on the Hue host:

* openssl and swig from the package repository
* xmlsec1 package from <http://www.aleksey.com/xmlsec/>
* Python packages for pysaml2 and djangosaml2 from Github

## TLS/SSL Certificates

Communication between Cloudera Manager and Cloudera Agents can be encrypted so that sensitive information such as Kerberos keytab files is not communicated from Cloudera Manager to the Agents in clear text.

There are different levels of protection available (levels 1 to 3). Level 3 (the highest level) requires each Cloudera Agent to have its own certificate. If this level of protection is required then the customer should provide certificates for each node in the cluster.

As a minimum, certificates are needed for Cloudera Manager and Edge Nodes to secure the web interfaces. Either the CN or the subjectAltName of the certificate should match the fully qualified DNS hostname of the host the certificate belongs to. If one or more subjectAltName is present in the certificate then at least one subjectAltName should be same as the CN in Subject of the certificate. If haproxy or Load Balancer will be used for accessing cluster services such as Impala, Solr etc., then the DNS name of the hosts where haproxy is running or the Load Balancer DNS name (VIP) should also be specified in the subjectAltName of all the certs that’ll be part of load balanced services. As a best practice Cloudera recommends that the certs used has a corresponding RSA keys of size 2048 bits and signed with SHA-256 hashing algorithm. The certs should either have no restrictions or have these properties set

keyUsage = digitalSignature, keyEncipherment

extendedKeyUsage = serverAuth, clientAuth

If the CA that signs these certificates is internal, then the customer will also need to provide the certificate chain of the CA that signs these certificates.

The same certificates can also be used to encrypt Cloudera Manager, Hue, Hadoop web interfaces, HiveServer2, Impala JDBC/ODBC, and Yarn encrypted shuffle. A node does not need more than one cert.

* **Note**: If any external services such as LDAPS or SAML use certificates from an internal CA, then the public certificate of the Root CA and any intermediate CA in the chain should be provided. Intermediate CAs will only be trusted if the corresponding server does not provide full certificate chain in the TLS/SSL connection.

## Firewall

If perimeter security is in place and there is a firewall in between the cluster and Active Directory, then the following ports should be opened to all cluster hosts.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Component** | **Service** | **Qualifier** | **Port** | **Protocol** | **Access Requirement** |
| Kerberos | KRB5 KDC Server | Secure | 88 | UDP/TCP | External |
|  | KRB5 Admin Server | Secure | 749 | TCP | Internal |
| LDAP | LDAP non-ssl |  | 389 | TCP | External |
|  | LDAP SSL | Secure | 636 | TCP | External |
|  | AD Global Catalog |  | 3268 | TCP | External |
|  | AD Global Catalog | Secure | 3269 | TCP | External |

## Windows Registry Key Setup

If Windows clients require access to HiveServer2 and Impala via JDBC then the following registry key should be added to the Windows clients:

**Windows XP:**

Add a new DWORD\_VALUE called “allowtgtsessionkey”, and assign it a value of 1.

HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos\   
REG\_DWORD name: allowtgtsessionkey   
Value: 1

**Windows 2000, Windows Vista and above:**

Add a new DWORD\_VALUE called “allowtgtsessionkey, and assign it a value of 1.

HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Lsa\Kerberos\Parameters REG\_DWORD name: allowtgtsessionkey  
Value: 1

## Admin/Root Access for Installing ODBC Drivers

Clients that need access to HiveServer2 or Impala via ODBC will need admin/root privileges on the client machine to install the ODBC drivers and to configure DSNs.

# Encryption Prerequisites

## Navigator Key Trustee

### Key Trustee Server minimum requirements

The customer should allocate two servers to host the Key Trustee Server that meets the minimum system requirements. Latest minimum requirements for KeyTrustee Server can be found at <http://www.cloudera.com/documentation/enterprise/latest/topics/key_trustee_install.html>

* **Note**: The minimum storage requirements cannot be allocated to one partition/volume. KTS installation needs space in /usr, /opt and /var/lib, /var/log and /var/run

### Navigator Key Trustee parcel download

The customer should acquire Navigator Trustee Repository parcels by downloading from Cloudera website’s downloads section at <http://www.cloudera.com/downloads.html> .

### Package dependencies

The customer should be aware of the required package dependencies (covered in the Navigator Key Trustee Installation and Maintenance Guide).

### Backup and Failure Options

The customer should be aware of the available backup and fail-over options for Key Trustee Server.

### Firewall

Network access through firewall can be found here <http://www.cloudera.com/documentation/enterprise/latest/topics/key_trustee_install.html#concept_jhm_kmm_3q_unique_1__section_qjc_tgb_kq_unique_1>

The software is downloaded from Cloudera’s repository (https://archive.cloudera.com) so the customer must be able to access this URL. Optionally, if email is configured, outbound TCP traffic over port 25 (SMTP) will be required.

### TLS

If the customer would like to protect Navigator Key Trustee traffic with TLS, the following criteria should be met:

* Customer should provide a certificate, which has been properly signed by a trusted CA
* Customer should provide the private key used to generate Certificate Signing Request (CSR) for the above certificate
* Customer should provide the chain file for the CA that signed the above certificate
* The DNS hostname of the Key Trustee Server should either match the Common Name (CN) of the above certificate or match one of the hostnames in subjectAltName. If one or more subjectAltName is present in the certificate then at least one subjectAltName should be same as the CN in Subject of the certificate.
* The 'SSL Verification Commands' below have been run and returned successfully

### TLS verification

If the customer would like to encrypt Key Trustee Server traffic with TLS, they should provide a CA-signed PEM certificate. This certificate should match the DNS hostname of the Key Trustee machine and must be signed by a trusted authority (or CA).

The following command should complete successfully without error:

openssl verify -CAfile chain-file.pem certificate.pem

## HDFS Encryption

### Key Management Server (KMS) minimum requirements

The customer should allocate a pair (a primary and a failover) of servers (VM preferred) to host KMS

Servers that meets the minimum system requirements. The hardware requirements for KMS Server are same as KTS servers except the KMS servers need an additional 10 GB space in /opt volume as CDH packages are also installed on KMS Servers.

* **Note**: The minimum storage requirements cannot be allocated to one partition/volume. KTS installation needs space in /usr, /opt and /var/lib, /var/log and /var/run

## Navigator Encrypt

### Navigator encrypt minimum requirements

The minimum requirements for Navigator Encrypt can be found at <http://www.cloudera.com/documentation/enterprise/latest/topics/navigator_encrypt_install.html>

### Downtime Required

HDFS data will not be accessible during the encryption process, please submit any necessary downtime requests.

### Navigator Encrypt binaries download

The customer should acquire Navigator Encrypt Repository packages by downloading from Cloudera website’s downloads section at <http://www.cloudera.com/downloads.html>.

### Package Dependencies

The customer should be aware of the required package dependencies explained at <http://www.cloudera.com/documentation/enterprise/latest/topics/navigator_encrypt_install.html>

### Firewall

Customer will need to allow TCP traffic to the Navigator Key Trustee Server. It is explained in the network requirements section at <http://www.cloudera.com/documentation/enterprise/latest/topics/navigator_encrypt_install.html>

### TLS

If a self-signed (or non-CA signed certificate) is being used on the Key Trustee Server, the Encrypt servers should be updated to trust the new certificate.

# Security Prerequisites Checklist

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Task** | **Completed Date** | **Completed By** |
| **Authentication and Authorization** | | | |
| ☐ | Opened required firewall ports |  |  |
| ☐ | Acquired license keys from Cloudera |  |  |
| ☐ | Setup root/sudo access on the cluster hosts |  |  |
| ☐ | Installed required authentication packages on all cluster hosts |  |  |
| ☐ | Installed and configured chosen Linux/AD integration tool (e.g. SSSD) on all cluster hosts |  |  |
| ☐ | Created Active Directory OU and OU user account for Cloudera Manager |  |  |
| ☐ | Enabled SSL/TLS on Active Directory |  |  |
| ☐ | [Exception] Created CDH service principals and keytab files (only required if not using the recommend direct-to-AD architecture) |  |  |
| ☐ | Created Active Directory Bind Account |  |  |
| ☐ | Created Active Directory groups for privileged users |  |  |
| ☐ | Created Active Directory groups for Cloudera Manager and Navigator roles |  |  |
| ☐ | Created databases for Cloudera Navigator and Sentry |  |  |
| ☐ | Created Active Directory test users and groups |  |  |
| ☐ | Configured cross-realm trusts for multi-realm authentication |  |  |
| ☐ | [SAML only] Created certificates and gathered required information from IDP for SAML authentication |  |  |
| ☐ | Generated SSL/TLS certificates |  |  |
| ☐ | [Windows clients only] Created Windows registry keys |  |  |
| ☐ | Installed Hive and/or Impala ODBC drivers on client machines |  |  |
| **Encryption** | | | |
| ☐ | Completed Cloudera Navigator Key Trustee Server prerequisites (covered in the Installation and Maintenance Guide) |  |  |
| ☐ | Provisioned Key Management Server (KMS) hosts for HDFS-level encryption |  |  |
| ☐ | [Navigator Encrypt only] Installed required Encrypt packages |  |  |
| ☐ | Opened firewall ports to Navigator Key Trustee Server |  |  |

# Appendix A: Example MIT KDC Configuration

# /var/kerberos/krb5kdc/kdc.conf

[kdcdefaults]

 kdc\_ports = 88

 kdc\_tcp\_ports = 88

[realms]

 HADOOP.COMPANY.COM = {

  master\_key\_type = aes256-cts

  acl\_file = /var/kerberos/krb5kdc/kadm5.acl

  dict\_file = /usr/share/dict/words

  admin\_keytab = /var/kerberos/krb5kdc/kadm5.keytab

  supported\_enctypes = aes256-cts:normal aes128-cts:normal

  max\_life = 1d 0h 0m 0s

  max\_renewable\_life = 7d 0h 0m 0s

 }

* **Note**: When integrating with Microsoft Active Directory (AD) the AES encryption types are not supported until Windows Server 2008. For older versions include arcfour-hmac in supported\_enctypes.   
    
  Weak encryption types should not be supported, e.g. des\* and arcfour-hmac-exp

# Appendix B: Example Kerberos Client Configuration

# /etc/krb5.conf

[libdefaults]

 default\_realm = HADOOP.COMPANY.COM

 ticket\_lifetime = 1d 0h 0m 0s

 renew\_lifetime = 7d 0h 0m 0s

 forwardable = true

[realms]

 HADOOP.COMPANY.COM = {

  kdc = utilityhost.company.com

  admin\_server = utilityhost.company.com

  default\_domain = company.com

 }

[domain\_realm]

 .company.com = HADOOP.COMPANY.COM

 company.com = HADOOP.COMPANY.COM

[logging]

 default = FILE:/var/log/krb5libs.log

 kdc = FILE:/var/log/krb5kdc.log

 admin\_server = FILE:/var/log/kadmind.log

# Appendix C: Example Custom Kerberos Keytab Retrieval Script

The script should take two arguments: a destination file to write the keytab to, and the full principal name to retrieve the key for.

#~/bin/bash

# Custom Kerberos Keytab Retrieval Script

# This script is called by Cloudera Manager Server. It takes two arguments:

#   A destination file to write the key to, and

#   The full principal name to retrieve the keytab for

# CM will input /tmp for DEST

DEST=$1

# CM will input principal name in format <service>/<fqdn>@REALM

PRINC=$2

# BASEDIR is the location on Cloudera Manager Server host where the keytabs are stored

# BASEDIR=/etc/cloudera-scm-server/service\_keytabs

BASEDIR=/root/cdh\_keytabs

# Parse PRINC to determine keytab filename

SERV=${PRINC%%/\*}

NORM=${PRINC%%@\*}

FQDN=${NORM#\*/}

# Keytab filenames should be in format <service>\_<fqdn>.keytab

FILE=${BASEDIR}/${SERV}\_${FQDN}.keytab

echo ${FILE}

if [ ! -e ${FILE} ] ; then

  # Keytab not found

  echo "Keytab not found: ${FILE}"

  echo "Keytab not found: ${FILE}" > ${BASEDIR}/error.log

  exit 1

fi

cp ${FILE} ${DEST}

exit 0

# Appendix D: Example Scripts for Creating Principals and Keytab Files in Active Directory

**Windows**

The following bash script generates the appropriate DOS commands for creating principals and extracting keytab files from Active Directory.

#!/bin/bash

set -x

#hbase/gs-hue-1.ent.cloudera.com@QA.CLOUDERA.COM

#hdfs/gs-hue-1.ent.cloudera.com@QA.CLOUDERA.COM

while read a; do

user=${a%%/\*}

host=${a%@\*}

host=${host#\*/}

shost=${host%%.\*}

uid="${user}\_$shost"

realm=${a##\*@}

if (( ${#uid} > 19 )); then

new\_shost=`echo $shost | sed 's|-||g' | sed 's|centos|cent|g' | sed 's|cento|cent|g' `

if [ "$user" = "zookeeper" ]; then

new\_uid="zk\_$new\_shost"

elif [ "$user" = "impala" ]; then

new\_uid="imp\_$new\_shost"

elif [ "$user" = "mapred" ]; then

new\_uid="MR\_$new\_shost"

elif [ "$user" = "hbase" ]; then

new\_uid="hb\_$new\_shost"

else

new\_uid="${user}\_$new\_shost"

fi

if (( ${#new\_uid} > 19 )); then

echo name too long \"$new\_uid\"

exit 1

else

uid="$new\_uid"

fi

fi

echo "dsadd user \"CN=$uid,CN=Users,dc=qa,dc=cloudera,dc=com\" -upn \"$user/$host\" -fn \"$user\" -ln \"$shost\" -display \"${uid}\" -pwd Cloud3r@457"

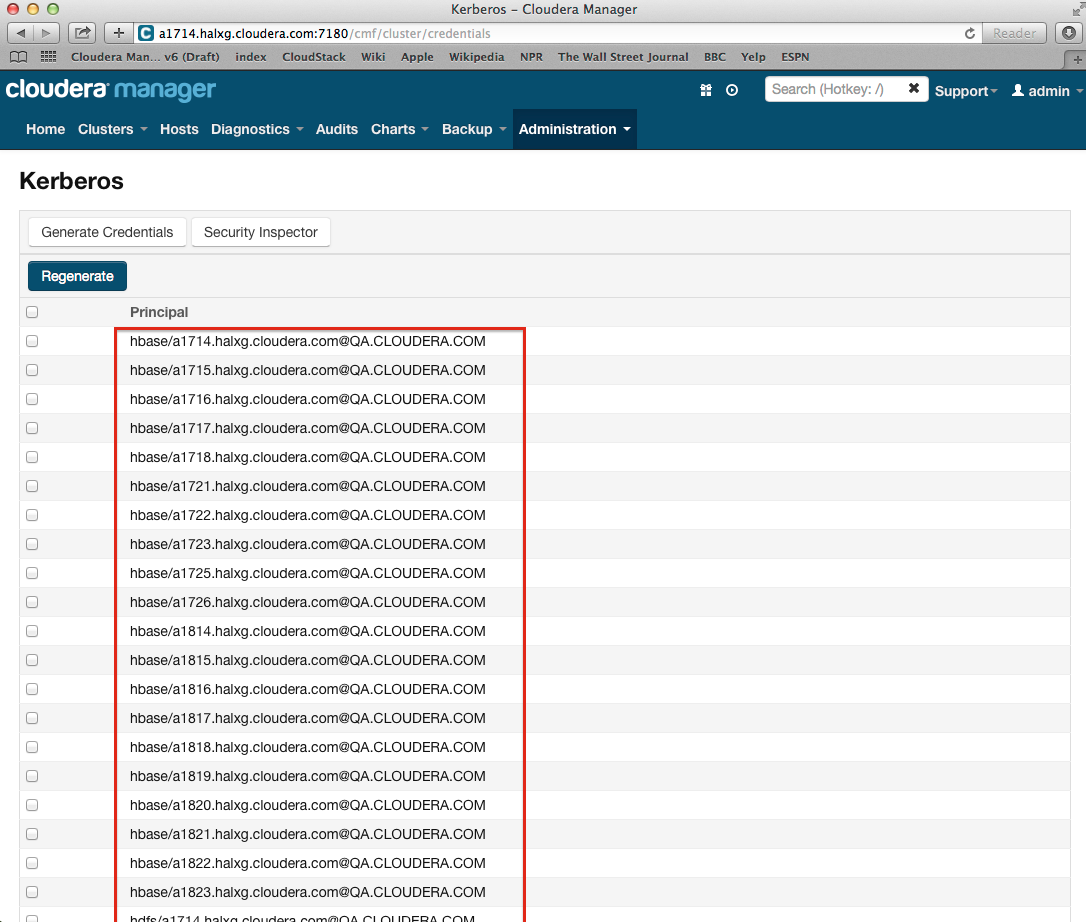
echo "ktpass /out ${user}\_$shost.keytab /crypto All /princ $a /mapuser ${uid} /ptype KRB5\_NT\_PRINCIPAL /mapop add /pass Cloud3r@457"

done

#dsadd user "CN=hbase\_gs-hue-50-2,CN=Users,dc=qa,dc=cloudera,dc=com" -upn "hbase/gs-hue-50-2.ent.cloudera.com" -fn "hbase" -ln "gs-hue-50-2" -display "hbase\_gs-hue-50-2" -pwd Cloud3r@457

#ktpass /out hbase\_gs-hue-50-2.keytab /crypto All /princ hbase/gs-hue-50-2.ent.cloudera.com@QA.CLOUDERA.COM /mapuser hbase\_gs-hue-50-2 /ptype KRB5\_NT\_PRINCIPAL /mapop add /pass Cloud3r@457

This script is run against a list of principals stored in a file. If you have previously setup Hadoop security with a local MIT KDC, then the list of principals can be obtained from Cloudera Manager: Administration > Kerberos:



The resulting DOS commands should be saved to a file and copied to the Active Directory host where they can be executed.

After the DOS commands have been run, the extracted keytab files can be copied to the Cloudera Manager host and the Custom Kerberos Keytab Retrieval Script configured in Cloudera Manager. See Appendix C: Example Custom Kerberos Keytab Retrieval Script.

1. Cloudera Professional Services Prerequisites v1.8 [↑](#footnote-ref-1)
2. <http://web.mit.edu/Kerberos/#what_is> [↑](#footnote-ref-2)
3. <https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/SSSD-Introduction.html> [↑](#footnote-ref-3)