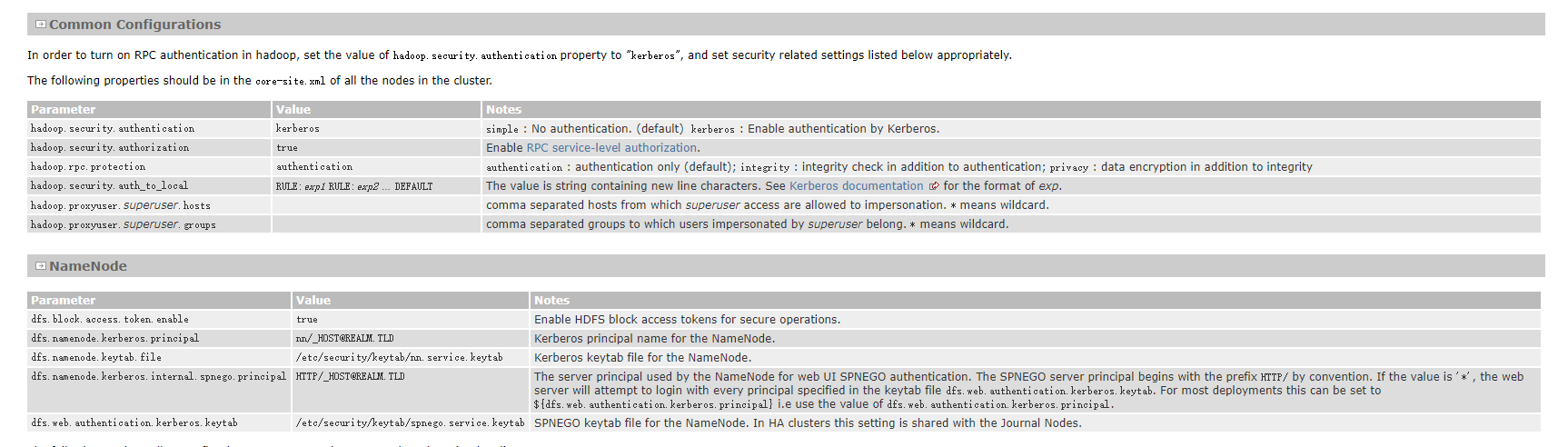
<https://blog.csdn.net/scgaliguodong123_/article/details/46523569>

官方文档：

<http://hadoop.apache.org/docs/r2.7.5/hadoop-project-dist/hadoop-common/SecureMode.html>



# 大数据平台搭建利器 Ambari 之 Kerberos 集成之路

源:https://www.ibm.com/developerworks/cn/opensource/os-cn-ambrri-kerberos/index.html

表 1.Kerberos 相关的术语

| **术语** | **简述** |
| --- | --- |
| KDC（Key Distribution Center） | 在启用 Kerberos 的环境中，KDC 用于验证各个模块 |
| Kerberos KDC Server | KDC 所在的机器 |
| Kerberos Client | 任何一个需要通过 KDC 认证的机器（或模块） |
| Principal | 用于验证一个用户或者一个 Service 的唯一的标识，相当于一个账号，需要为其设置密码（这个密码也被称之为 Key） |
| Keytab | 包含有一个或多个 Principal 以及其密码的文件 |
| Relam | 由 KDC 以及多个 Kerberos Client 组成的网络 |
| KDC Admin Account | KDC 中拥有管理权限的账户（例如添加、修改、删除 Principal） |
| Authentication Server (AS) | 用于初始化认证，并生成 Ticket Granting Ticket (TGT) |
| Ticket Granting Server (TGS) | 在 TGT 的基础上生成 Service Ticket。一般情况下 AS 和 TGS 都在 KDC 的 Server 上 |

Kerberos 的认证过程如下图所示。我们可以简单的理解为，一个 User 或者一个 Service 会用 Principal 到 AS 去认证，AS 会返回一个用 Principal Key 加密的 TGT，这时候只有 AS 和这个 Principal 的使用者可以识别该 TGT。在拿到加密的 TGT 之后，User 或者 Service 会使用 Principal 的 Key 来解密 TGT，并使用解密后的 TGT 去 TGS 获取 Service Ticket。在 Kerberos 认证的集群中，只有拿着这个 Service Ticket 才可以访问真正的 Server 从而实现自己的业务逻辑。一般我们将 TGT 的存放文件，称为 Kerberos Confidential 文件，默认的存放目录为/tmp，文件名则由 krb5cc 和用户的 id 组成，例如“/tmp/krb5cc\_0”为 root 的 confidential 文件。

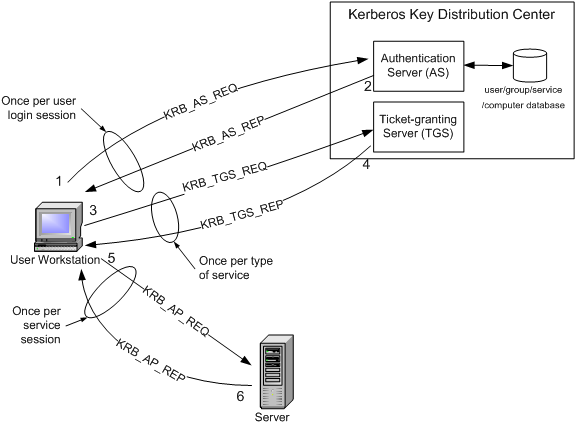


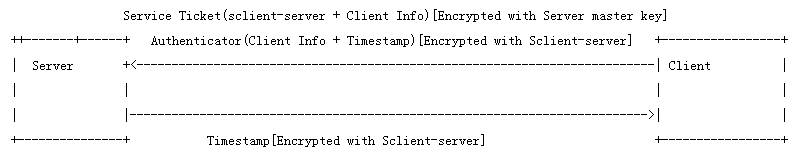
表 2. 常用的 Kerberos 命令

| **命令名** | **命令简介** |
| --- | --- |
| Kadmin | 用于管理 Kerberos 相关配置的，需要用 Principal 登录（需要 Principal 的 Key），可以在任一个 Kerberos Client 机器上使用 |
| Kadmin.local | 同 Kadmin，不过这个命令供 root 使用（只能在 KDC Server 上用） |
| Kinit | 生成初始化 ticket，也就是 TGT，并保存到一个 cache 文件，一般存放在/tmp 目录 |
| Klist | 用于列出 Kerberos 的 Confidential 信息，默认会列出/tmp 目录下的 cache 文件中的 Confidential 信息，也可以列出 keytab 文件的 ticket 信息 |
| Ktutil | 用于管理 keytab 文件 |

## 认证过程

首先 Client询问KDC，我想访问某个Server，然后KDC会将会话密钥Sclient-Server用Client master key加密后传送给Client；与此同时，KDC也会将会话密钥Sclient-Server连同Client的基本信息打包用Server master key加密也发给Client，并经Client转发给Server，至此Client与KDC的交互完成。

然后，Client用自己的master key解密KDC传过来的第一个包，解密后获得会话密钥Sclient-Server，并用这个密钥加密自己的的信息和时间戳打包后传送给Server，此时Client开始和Server交互，如下图：



Server会收到两个数据包，一个用会话密钥加密，一个用自己的master key加密，Server先用自己的master key解密获取会话密钥和一份关于Client的信息，然后Server拿到解密后获取到的会话密钥再解开另外一个数据包，获得另一份关于Client的信息和时间戳，至此Client和Server的交互完成。

https://blog.csdn.net/kkdelta/article/details/46633557

## 编译安装kerberos

<http://orchome.com/322>

## kerberos使用

kadmin.local -q “addprinc root/admin” 会提示输入密码创建管理员账户

使用：

Kadmin -p root/admin 提示输入密码登录管理员账户在kadmin:下输入各种命令，如下：

$kadmin:list\_principals 查看认证用户

$kadmin:addprinc user1 新增普通用户

$kadmin:delprinc user1 删除用户

$kadmin:exit 退出kerberos环境

抽取秘钥并将器存储在本地keytab文件/etc/krb5.keytab中。这个文件由超级用户拥有，所以必须是root用户才能在kadmin shell中执行以下命令::

$kadmin.local -q “ktadd kadmin/admin”

$klist -k /etc/krb5.keytab

## 安装Ambari

### 在线安装

参考文档（含配置详解）

<https://www.cnblogs.com/boyzgw/p/6525313.html>

实验过程中由于没有配置如下步骤造成安装集群不成功

**2.3 开启NTP服务(所有集群上节点都需要操作)**

1 yum install ntp

2 systemctl is-enabled ntpd

3 systemctl enable ntpd

4 systemctl start ntpd

**2.4 检查DNS和NSCD(所有节点都要设置)**  
　　ambari在安装时需要配置全域名，所以需要检查DNS。在节点里用 Name Service Caching Daemon (NSCD)

1 vi /etc/hosts

1 192.168.190.21 namenode www.namenode.zgw

2 192.168.190.22 datanode1 www.datanode1.zgw

3 192.168.190.23 datanode2 www.datanode2.zgw

4 192.168.190.24 SecondNamenode www.SecondNamenode.zgw

　　每台节点里配置FQDN，如下以主节点为例

1 vi /etc/sysconfig/network

1 NETWORKING=yes

2 HOSTNAME=www.namenode.zgw

### 离线安装

配置

[JAAS/GSS-API/SASL/Kerberos简介](http://www.nosqlnotes.com/technotes/distributed-system-with-kerberos/)

<http://www.nosqlnotes.com/technotes/distributed-system-with-kerberos/>

## hadoop用户权限管理

<https://my.oschina.net/hetiangui/blog/143364>

## [Hadoop服务层授权控制](https://www.iteblog.com/archives/983.html)

https://www.iteblog.com/archives/983.html

windows下和linux下编译

<https://imaidata.github.io/blog/kerberos_client/>

<https://blog.csdn.net/dxl342/article/details/55510659>

LDAP授权管理

https://blog.csdn.net/cheng\_fangang/article/details/40143261