## BCID：一种新的HADOOP安全认证方法

### BCID背景

如今hadoop已经在业界非常广泛的使用，关于Hadoop的安全也日益受到人们的关注。其中用户认证是Hadoop安全中非常重要的一个环节。Hadoop原生的用户认证体系安全性是比较弱的，因为Hadoop作为开源软件，更多专注于系统的通用性和性能。但如果要作为成熟的商业软件，提供服务，健壮用户安全认证体系显然是非常重要的。

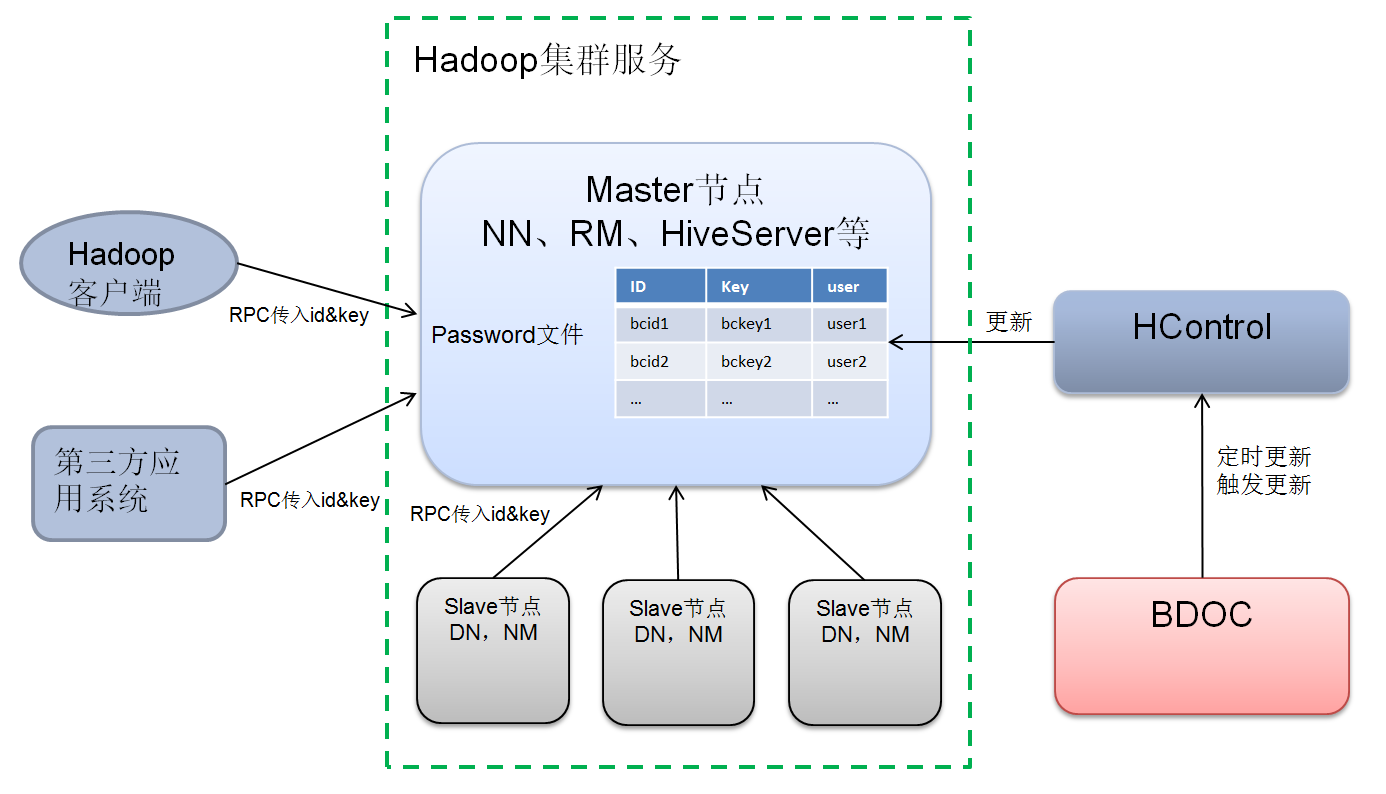
目前主流的大数据产品用户安全认证的方案大概有两类：

1. Hadoop原生支持的Kerberos认证：Hadoop原生自带支持Kerberos认证。基于Hadoop的一些大数据产品（包括CDH，华为等）也使用Kerberos做安全认证。主要优点是代码开源，但是部署使用比较繁琐。
2. 基于AccessId和SecurityKey的认证：为每个用户分配一组或多组AccessId和SecurityKey，用户使用AccessId和SecurityKey对系统进行访问。目前主流的大数据服务AWS和ODPS都采用这种用户认证方案。这种方案的优点在于使用非常方便，但是AWS和ODPS都是自成体系的系统，而不是基于hadoop，并不开源。

BCID是我们苏研自研的一种Hadoop安全认证方法，使用AccessId和SecurityKey认证，类似于AWS和ODPS的用户认证方案。BCID在安全性上并不逊于Kerberos，但是在易用性和性能上能够达到更优。

### BCID认证原理

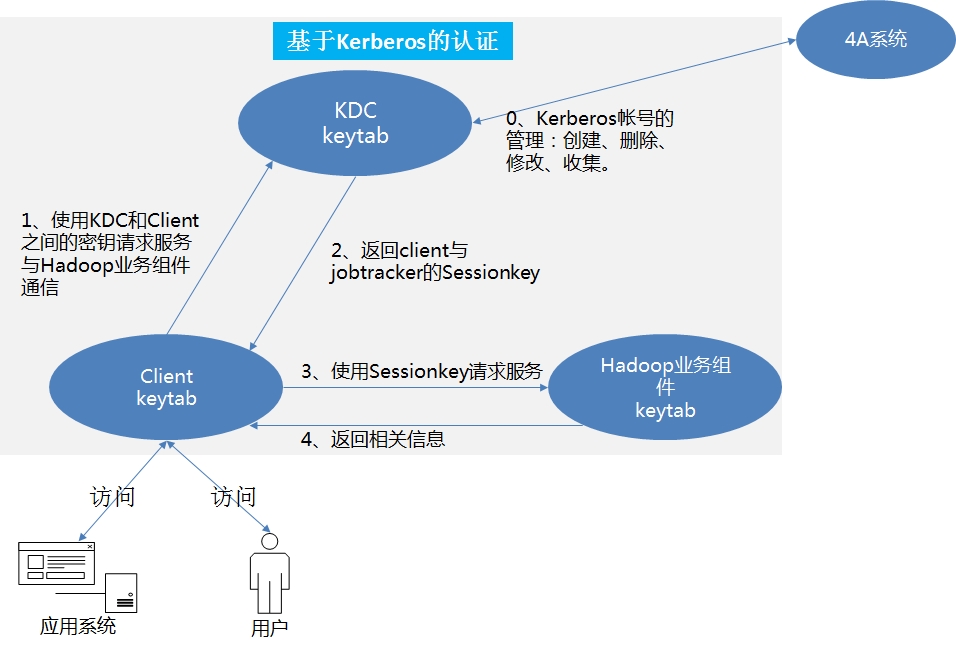
BCID是基于AccessId和SecurityKey的认证方法，其原理如下图所示：



1. BDOC（大数据运营管理平台）中会维护一套用户体系，包括每个用户的bcid和key信息。用户及管理员可以通过前台web页面对用户认证信息进行增删改查的操作。
2. BDOC会通过HControl对Hadoop集群内的用户认证信息进行同步，同步的机制包括定时更新和触发更新两种。在Hadoop的Master节点上（包括Namenode、ResourceManager、HiveServer等）会保存一个Password文件（bdoc-password.txt）。每次HControl同步BDOC的用户信息到这些节点的Password文件中。
3. Password文件中每行记录<id, key, user>的三元组，其中id是唯一值，每个user可以对应多组id&key，key的值会进行加密。
4. Master节点会定时读取Password文件，将用户认证信息更新加载到内存中。每当处理rpc请求时，会解析出请求中传入的id&key信息，并进行用户认证。如果id&key信息错误或者未传入id&key，则拒绝服务。
5. Hadoop客户端及第三方应用需要访问Hadoop集群时，需要将id&key信息加到配置中，通常可以通过修改配置文件（core-site.xml）或者添加命令行参数。Hadoop客户端向服务端发起请求时，会将配置中的id&key序列化到请求的HEADER中。用户所使用的id&key可以在BDOC的web页面中获取到。
6. 集群内的节点（Datanode和NodeManager）与Master节点通信时也需要传入id&key信息。在Password文件中会有一组特殊的admin用户信息。集群内各节点需要在配置文件中配置admin用户对应的id&key。这样既保证了集群内的安全认证，又方便集群统一配置管理。

### BCID认证对比Kerberos认证

Kerberos是开源hadoop原生自带的用户认证体系。基于开源的Hadoop产品（CDH、华为）通常采用Kerberos来做自身的用户安全认证。下图是Kerberos认证的原理展示。



将从以下几个方面对Kerberos与BCID认证机制进行对比：

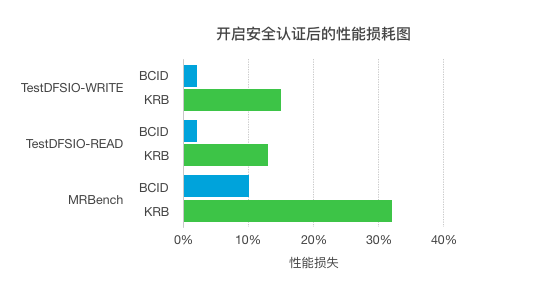
1. 安全性

|  |  |  |
| --- | --- | --- |
|  | Kerberos | BCID |
| 客户端请求认证 | TGT（ticket-granting ticket） | AccessId，SecurityKey |
| 集群内节点是否认证 | 是 | 是 |
| 集群内请求是否认证 | 是（通过Token） | 是（通过AccessId，SecurityKey） |
| 加密算法 | CRC32、DES | HMAC-SHA |
| 对请求客户端的限制 | 必须是Kerberos认证的节点 | 任何能够访问到集群的节点 |

通过以上对比看出，Kerberos认证与BCID认证安全性的强度是接近的。BCID对访问集群的客户端并没有限制，允许任何能够访问到集群的节点进行访问，目的是为了更方便用户的使用以及第三方应用的接入，AWS和ODPS的方案也是基于同样的原因。

1. 对hadoop性能的影响

测试Kerberos与BCID认证对hadoop性能的影响，测试case包括hadoop标准benchmark中的DFSIO-Write、DFSIO-Read以及MRBench，结果如下：



从结果上看到，开启BCID认证后对hadoop性能的影响远小于开启Kerberos认证。

BCID认证性能好于Kerberos认证的原因在于：

1. BCID认证中，用户认证信息实际上是保存在了服务节点的内存中，认证的行为实际是在内存中进行的；而Kerberos认证需要依赖于第三方节点KDC，这样产生了更多的网络通信开销。
2. BCID认证中，客户端访问服务端前只需要加载本地配置中的id&key信息；kerberos认证中，客户端发起请求之前，需要先与KDC通信获取有效的Ticket，从而产生因网络通信产生的延迟
3. 集群部署运维易用性及可用性

Hadoop集群使用kerberos认证的运维难度是很大的。部署并使用kerberos需要做的运维工作包括：

1. 需要部署KDC服务
2. 集群内所有节点，以及需要访问集群的客户端节点都需要部署kerberos客户端并进行相应配置。
3. 集群内每个用户对应每个节点需要在kdc中生成认证信息（keytab文件），并分发到每个节点下。
4. 集群每新增节点，新增用户，都需要重新生成keytab，并进行分发
5. 集群各节点启动前需要执行命令获取ticket
6. 集群各节点重启前需要执行命令重新获取ticket。
7. 客户端每次访问集群前，需要执行命令重新获取ticket
8. 集群每个节点，以及访问集群的客户端之间需要时间同步
9. 集群各个节点上需要额外增加大量配置

需要注意的是，以上工作需要手动执行，并且目前除了9以外，无法通过运维管理工具HControl完成。

相比之下，hadoop集群使用BCID认证的运维难度小很多。使用BCID认证只需要通过HControl增加若干个配置项即可。

关于可用性，使用kerberos认证是存在单点问题的，即一旦KDC发生故障，对集群的任何访问就无法正常进行认证，从而导致集群无法服务。此时即使集群本身开启了HA也无效。

使用BCID认证，由于认证是内嵌到Master节点进行的，只要集群是开启HA的，就不会存在单点问题。一旦一台Master节点失败，服务会自动切换到StandBy Master，用户认证服务也随之切换，从而不受影响。

1. 集成BDOC以及第三方应用难度

BDOC作为基于Hadoop的大数据运营管理平台，本身提供了统一的基于BCID的用户认证管理服务，可以无缝支持BCID认证。但是对于Kerberos认证机制，目前暂时无法支持。

对于第三方应用，接入Kerberos认证或者BCID认证，都需要做相应适配工作：

第三方应用使用Kerberos认证：

1. 第三方应用通常部署在集群外的节点，这样需要对部署节点环境有一定依赖，并且需要为应用创建相应kerberos用户认证信息
2. 应用本身可能需要考虑到ticket过期问题，即需要额外实现逻辑去更新ticket。

第三方应用使用BCID认证，第三方应用只需要在访问hadoop集群的配置中增加用户id&key配置参数即可。目前已有第三方应用成功接入BCID认证的案例。

1. 总结

总结Kerberos认证与BCID认证各方面的对比：

|  |  |  |
| --- | --- | --- |
|  | Kerberos认证 | BCID认证 |
| 安全性 | 高 | 高 |
| 性能影响 | 非常高 | 低 |
| 集群运维难度 | 高 | 低 |
| 高可用 | 无 | 集群自身高可用 |
| 应用集成难度 | 中 | 低 |

### BCID认证相关配置

1. Server节点（NN，RM）上core-site.xml 中配置（可通过HControl进行配置）：

|  |  |  |
| --- | --- | --- |
| 参数 | value | 默认值 |
| hadoop.security.enable.bdoc.authentication | 是否开启bdoc认证 | False |
| hadoop.security.user.mapping.class | 用户认证实现类（通常默认） | org.apache.hadoop.security.PasswordFileBasedUserMapping |
| hadoop.security.bdoc.password.file | bdoc密码文件名（通常默认） | bdoc-password.txt |
| hadoop.security.bdoc.password.interval | bdoc密码文件自动刷新时间间隔（通常默认） | 30000 |
| hadoop.security.system.user | 认证系统用户名（通常默认） | bdi |

1. Server节点上bdoc-password.txt文件示例（使用BDOC可自动配置）：

|  |
| --- |
| accessid1,429cbdf9a80f9c1ebe85fe1eac61c0da,user1  accessid2,8675cf213a438b99ae2dbb8c39b4142f,user2  accessid3,bc704e7363933c47ba2fa460f9840e86,user3  admin,21232f297a57a5a743894a0e4a801fc3,bdi |

此处系统用户对应的id和key都是admin

1. 集群内所有节点core-site.xml中需要增加配置：

|  |  |
| --- | --- |
| 参数 | Value |
| hadoop.security.bdoc.access.id | admin （password文件中系统用户对应的id值） |
| hadoop.security.bdoc.access.key | admin （password文件中系统用户对应的key值） |

1. 客户端访问集群需要在其core-site.xml中添加配置或者通过命令行传入参数：

|  |  |
| --- | --- |
| 参数 | Value |
| hadoop.security.bdoc.access.id | 相应用户id值 |
| hadoop.security.bdoc.access.key | 相应用户key值 |