Hadoop DelegationToken

Hadoop安全机制

1. Kerberors认证
2. Token机制
3. Delegation Token

HDFS NameNode delegation tokens用于允许MapReduce Job访问HDFS资源，

hdfs fetchdt

ERROR: Must specify exactly one token file

fetchdt retrieves delegation tokens from the NameNode

fetchdt <opts> <token file>





DFSClient的使用如下：

proxyInfo = NameNodeProxiesClient.createProxyWithClientProtocol(conf,

nameNodeUri, nnFallbackToSimpleAuth);

this.dtService = proxyInfo.getDelegationTokenService();

this.namenode = proxyInfo.getProxy();

ClientProtocol proxy = createNonHAProxyWithClientProtocol(nnAddr, conf,  
 UserGroupInformation.getCurrentUser(), true, fallbackToSimpleAuth);

getCurrentUser，从

String fileLocation = System.getenv(HADOOP\_TOKEN\_FILE\_LOCATION);  
if (fileLocation != null) {  
 File source = new File(fileLocation);  
 Credentials cred = Credentials.readTokenStorageFile(  
 source, conf);  
 loginUser.addCredentials(cred);  
}

FileSystem =>

DFSClient <=

this.ugi <= UserGroupInformation.getCurrentUser

Job#getInstance

#JobContext#init

this.ugi <= UserGroupInformation.getCurrentUser

this.credentials.mergeAll(ugi.getCredentials)

对Token的验证：

Text dtService

this.dtService = ProxyInfo.getDelegationTokenService

this.namenode:ClientProtocol = proxyInfo.getProxy();

客户端与NameNode通信进行的验证

SaslRpcServer.SaslDisgestCallbackHandler

#getPassword

AbstractDelegationTokenSecretManager

#retriableRetrievePassword

#retrievePassword

#checkToken

SaslDataTransferServer

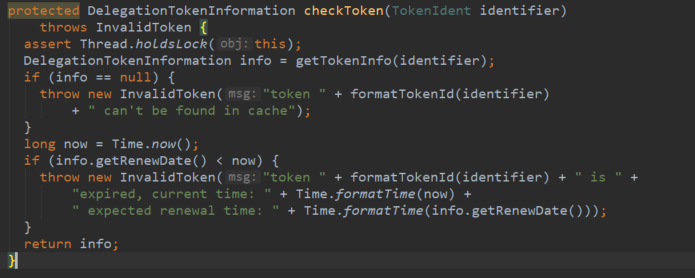
#buildServerPassworld





SaslRpcClient#getServerToken

return tokenSelector.selectToken(  
 SecurityUtil.*buildTokenService*(serverAddr), ugi.getTokens());



export HADOOP\_TOKEN\_FILE\_LOCATION="/hadoop/yarn/local/usercache/hdfs/appcache/application\_1536503067245\_0003/container\_e49\_1536503067245\_0003\_01\_000002/container\_tokens"

FINAL\_CONTAINER\_TOKEN\_FILES

https://blog.cloudera.com/blog/2017/12/hadoop-delegation-tokens-explained/