Before..!
in /etc/hosts

- Add following lines to /etc/hosts file
  - 127.0.1.2 server.myhostname.com # - kerberos server
  - 127.0.1.3 kafka1.myhostname.com # - kafka broker 1
  - 127.0.1.4 kafka2.myhostname.com # - kafka broker 2
  - 127.0.1.5 client1.myhostname.com # - kafka producer

Kerberos Installation on Centos 6.X I
Kerberos Server Side

- Install Kerberos system
  - Download the following Cento 6.X rpm packages relevant to kerberos system from http://mirror.centos.org/ and make sure that you pick the rpms compatible with your system(version and arch.)
    - krb5-libs
    - krb5-server
    - krb5-workstation
- Replace the krb.config file sitting in /etc with this krb5.conf and kdc.conf in /var/kerberos/krb5kdc/ with this kdc.conf . These configs are tested ok with Centos 6.8.
  - Realm given in the files above “MYREALM.COM”
- Create a new realm
  - “sudo /usr/sbin/kdb5_util create -s”
  - Provide password when prompted
  - Thus, Key Distribution Center(KDC) created

Kerberos Installation on Centos 6.X II

Kerberos Server Side

- Need to write the admin keytab based on the path specified in the kdc.conf file for the kerberos services to start, type the following
  - “kadmin.local -q “ktadd -k /var/kerberos/krb5kdc/kadm5.keytab kadmin/admin kadmin/changepw“ “
- Start kadmind and kdc services with following commands
  - ”/sbin/service krb5kdc start”
  - “/sbin/service kadmin start”
- Add principal to KDC
  - Use the “kadmin.local” tool to add/delete/modify users (principles)
  - To assign an admin role to a user named “user1”, use the following command “addprinc user1/admin”
  - Create /var/kerberos/krb5kdc/kadm5.acl if not available and put */admin * line in it
- Type the following command “kinit user1/admin” to get a ticket for your client machine
- Life and renewable time can be set in the kdc.conf with max_life, max_renewable_life variables

Kerberos Installation on Centos 6.X III

Zookeeper, Kafka Broker-Producer Side

- Need to generate keytabs for each users not to enter passwords every time service is requested
- Type the following commands on the kerberos server

Kafka Broker - Zookeeper Side

kafka1

- kadmin.local -q “addprinc -randkey kafka/kafka1.myhostname.com@MYREALM.COM”
- kadmin.local -q “ktadd -k /home/kafka/kafka1.keytab kafka/kafka1.myhostname.com@MYREALM.COM”

kafka2

- kadmin.local -q “addprinc -randkey kafka/kafka2.myhostname.com@MYREALM.COM”
- kadmin.local -q “ktadd -k /home/kafka/kafka2.keytab kafka/kafka2.myhostname.com@MYREALM.COM”

zookeeper

- kadmin.local -q 'addprinc -randkey zookeeper/kafka1.myhostname.com@MYREALM.COM’
- kadmin.local -q ‘ktadd -k /etc/security/keytabs/zk.keytab zookeeper/kafka1.myhostname.com@MYREALM.COM’

Kafka Producer Side

kafka-client

- kadmin.local -q 'addprinc -randkey kafka-client/client1.myhostname.com@MYREALM.COM’
- kadmin.local -q ‘ktadd -k /home/kafka-client/kafkaclient.keytab kafka-client/client1.myhostname.com@MYREALM.COM’

Start up - Zookeeper and Kafka Broker with SASL/Kerberos

- Use 0.9.0.1 version of Kafka since authentication and authorization with SASL/Kerberos is implemented later than version 0.9.0.0
- Authentication using SASL/PLAIN which is a simple username/password authentication mechanism used with TLS is implemented in version 0.10.0
- This security implementation mentioned below is based on SASL/Kerberos with no TLS . If desired that could be done.
  - Pass the name of the JAAS file as a JVM parameter to zookeeper
    - export KAFKA_OPTS= "Djava.security.auth.login.config=/etc/kafka/zookeeper_kerberos_jaas.conf "
  - Use modified zookeeper properties file based on SASL
  - Enter the following
    - “bin/zookeeper-server-start.sh config/zookeeper_new.properties
  - Type the following to pass jass file for kafka as jvm parameter
    - export KAFKA_OPTS="-Djava.security.auth.login.config= /etc/kafka/kafka_server_kerberos_jaas.conf "
  - Use modified kafka broker properties file
  - Start kafka-broker “bin/kafka-server-start.sh config/server_sasl_kerberos.properties”
  - To Start 2. kafka-broker type as follows
    - export KAFKA_OPTS="-Djava.security.auth.login.config= /etc/kafka/kafka_server2_kerberos_jass.conf
    - “bin/kafka-server-start.sh config/server_sasl_kerberos.properties”

Kafka Producer with Pentaho

- Need to use Pentaho version 5.4 no later than this version
- There seems to be a issue regarding JAAS on Pentaho version of which is greater than 5.4. please see the link.
- Issue solved in this post. The solution can be implemented
- Open the ktr file provided with pentaho.
- Set the necessary properties in “Props” step for kafka procuder api
- Assign the parameters to the values desired in params tab in kafka producer step

Autorization and ACLs

- bin/kafka-acls.sh --authorizer kafka.security.auth.SimpleAclAuthorizer --authorizer-properties zookeeper.connect=kafka1.myhostname.com:2181 --add --allow-principal User:kafka-client --operation Write --topic test
- bin/kafka-acls.sh --authorizer kafka.security.auth.SimpleAclAuthorizer --authorizer-properties zookeeper.connect=kafka1.myhostname.com:2181 --add --allow-principal User:kafka-client --operation Read --topic test
- bin/kafka-acls.sh --authorizer kafka.security.auth.SimpleAclAuthorizer --authorizer-properties zookeeper.connect=kafka1.myhostname.com:2181 --add --allow-principal User:kafka-client --producer --topic test
- bin/kafka-acls.sh --authorizer kafka.security.auth.SimpleAclAuthorizer --authorizer-properties zookeeper.connect=kafka1.myhostname.com:2181 --add --allow-principal User:kafka-client --consumer --topic test --group test-consumer-group