

```
#####  
# How to setup cross realm trust between two MIT KDC  
#####
```

setup cross realm trust between two MIT KDC to access and copy data of one cluster from another if the cross realm trust is setup correctly.

2 clusters with 2 HDP versions(2.4 and 2.5) and Ambari version(2.2.2.0)  
Cluster 1:

```
192.168.0.47 texas-1.texas.com  texas-1  
192.168.0.46 texas-2.texas.com  texas-2  
192.168.0.45 texas-3.texas.com  texas-3  
Cluster 2:
```

```
192.168.0.48 utah-1.utah.com  utah-1  
192.168.0.49 utah-2.utah.com  utah-2  
192.168.0.50 utah-3.utah.com  utah-3
```

Both the clusters are kerberized with MIT KDC.

```
#####  
# Please configure /etc/hosts file on both the clusters to have FQDN  
#####
```

On both clusters /etc/hosts file should look like below:

```
192.168.0.47 texas-1.texas.com  texas-1  
192.168.0.46 texas-2.texas.com  texas-2  
192.168.0.45 texas-3.texas.com  texas-3  
192.168.0.48 utah-1.utah.com  utah-1  
192.168.0.49 utah-2.utah.com  utah-2  
192.168.0.50 utah-3.utah.com  utah-3
```

```
#####  
# Configure krb5.conf:  
#####
```

Configure the [realm] section to add another cluster's KDC server details this is required to find KDC to authenticate user which belongs to another cluster.

Example on Cluster1:

```
[realms]  
  TEXAS.COM = {  
    admin_server = texas-1.texas.com  
    kdc = texas-1.texas.com  
  }  
  UTAH.COM = {  
    admin_server = utah-1.utah.com  
    kdc = utah-1.utah.com  
  }
```

3.2 Configure [domain\_realm] section to add another cluster's domain <-> realm mapping.

```

[domain_realm]
.texas.com = TEXAS.COM
texas.com = TEXAS.COM
.utah.com = UTAH.COM
utah.com = UTAH.COM
3.3 Configure [capaths] to add another cluster's realm
[capaths]
    TEXAS.COM = {
        UTAH.COM = .
    }

```

On Cluster 1, the krb5.conf should look like below:

```

[libdefaults]
renew_lifetime = 7d
forwardable = true
default_realm = TEXAS.COM
ticket_lifetime = 24h
dns_lookup_realm = false
dns_lookup_kdc = false
#default_tgs_enctypes = aes des3-cbc-sha1 rc4 des-cbc-md5
#default_tkt_enctypes = aes des3-cbc-sha1 rc4 des-cbc-md5
[logging]
default = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log
kdc = FILE:/var/log/krb5kdc.log
[realms]
TEXAS.COM = {
    admin_server = texas-1.texas.com
    kdc = texas-1.texas.com
}
UTAH.COM = {
    admin_server = utah-1.utah.com
    kdc = utah-1.utah.com
}
[domain_realm]
.texas.com = TEXAS.COM
texas.com = TEXAS.COM
.utah.com = UTAH.COM
utah.com = UTAH.COM
[capaths]
    TEXAS.COM = {
        UTAH.COM = .
    }

```

Note – Please copy modified /etc/krb5.conf to all the nodes in Cluster 1

Similarly on Cluster2, the krb5.conf should look like below:

```

[libdefaults]
renew_lifetime = 7d
forwardable = true
default_realm = UTAH.COM
ticket_lifetime = 24h
dns_lookup_realm = false
dns_lookup_kdc = false

```

```

#default_tgs_etypes = aes des3-cbc-sha1 rc4 des-cbc-md5
#default_tkt_etypes = aes des3-cbc-sha1 rc4 des-cbc-md5
[logging]
default = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log
kdc = FILE:/var/log/krb5kdc.log
[realms]
UTAH.COM = {
    admin_server = utah-1.utah.com
    kdc = utah-1.utah.com
}
TEXAS.COM = {
    admin_server = texas-1.texas.com
    kdc = texas-1.texas.com
}
[domain_realm]
.texas.com = TEXAS.COM
texas.com = TEXAS.COM
.utah.com = UTAH.COM
utah.com = UTAH.COM
[capaths]
    UTAH.COM = {
        TEXAS.COM = .
    }

```

Note – Please copy modified /etc/krb5.conf to all the nodes in Cluster 2

```

#####
# Modify below property in hdfs-site.xml on a cluster from
# where you want to execute distcp command # (specifically speaking – client side )
#####
dfs.namenode.kerberos.principal.pattern=*

```

```

#####
# Add a common trust principal in both the KDCs.
# Please keep same password for both the principals
#####
On Cluster 1 and 2, execute below commands in kadmin utility:
addprinc krbtgt/TEXAS.COM@UTAH.COM
addprinc krbtgt/UTAH.COM@TEXAS.COM

```

```

#####
# Configure auth_to_local rules on both the clusters:
#####
On Cluster1, append auth_to_local rules from Cluster2
Example on Cluster 1:

```

```

RULE:[1:$1@$0](ambari-qa-hadoop@TEXAS.COM)s/.*/ambari-qa/
RULE:[1:$1@$0](hdfs-hadoop@TEXAS.COM)s/.*/hdfs/
RULE:[1:$1@$0](spark-hadoop@TEXAS.COM)s/.*/spark/
RULE:[1:$1@$0](.*@TEXAS.COM)s/@.*//
RULE:[2:$1@$0](dn@TEXAS.COM)s/.*/hdfs/
RULE:[2:$1@$0](hive@TEXAS.COM)s/.*/hive/
RULE:[2:$1@$0](jhs@TEXAS.COM)s/.*/mapred/
RULE:[2:$1@$0](jn@TEXAS.COM)s/.*/hdfs/

```

```
RULE:[2:$1@$0](nm@TEXAS.COM)s./*/yarn/
RULE:[2:$1@$0](nn@TEXAS.COM)s./*/hdfs/
RULE:[2:$1@$0](rm@TEXAS.COM)s./*/yarn/
RULE:[2:$1@$0](yarn@TEXAS.COM)s./*/yarn/
DEFAULT
RULE:[1:$1@$0](ambari-qa-utah@UTAH.COM)s./*/ambari-qa/
RULE:[1:$1@$0](hdfs-utah@UTAH.COM)s./*/hdfs/
RULE:[1:$1@$0](spark-utah@UTAH.COM)s./*/spark/
RULE:[1:$1@$0](.*@UTAH.COM)s/@.*//
RULE:[2:$1@$0](dn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](hive@UTAH.COM)s./*/hive/
RULE:[2:$1@$0](jhs@UTAH.COM)s./*/mapred/
RULE:[2:$1@$0](jn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](nm@UTAH.COM)s./*/yarn/
RULE:[2:$1@$0](nn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](rm@UTAH.COM)s./*/yarn/
RULE:[2:$1@$0](yarn@UTAH.COM)s./*/yarn/
```

On Cluster2, append auth\_to\_local rules from Cluster1

Example on Cluster 2:

```
RULE:[1:$1@$0](ambari-qa-utah@UTAH.COM)s./*/ambari-qa/
RULE:[1:$1@$0](hdfs-utah@UTAH.COM)s./*/hdfs/
RULE:[1:$1@$0](spark-utah@UTAH.COM)s./*/spark/
RULE:[1:$1@$0](.*@UTAH.COM)s/@.*//
RULE:[2:$1@$0](dn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](hive@UTAH.COM)s./*/hive/
RULE:[2:$1@$0](jhs@UTAH.COM)s./*/mapred/
RULE:[2:$1@$0](jn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](nm@UTAH.COM)s./*/yarn/
RULE:[2:$1@$0](nn@UTAH.COM)s./*/hdfs/
RULE:[2:$1@$0](rm@UTAH.COM)s./*/yarn/
RULE:[2:$1@$0](yarn@UTAH.COM)s./*/yarn/
DEFAULT
RULE:[1:$1@$0](ambari-qa-hadoop@TEXAS.COM)s./*/ambari-qa/
RULE:[1:$1@$0](hdfs-hadoop@TEXAS.COM)s./*/hdfs/
RULE:[1:$1@$0](spark-hadoop@TEXAS.COM)s./*/spark/
RULE:[1:$1@$0](.*@TEXAS.COM)s/@.*//
RULE:[2:$1@$0](dn@TEXAS.COM)s./*/hdfs/
RULE:[2:$1@$0](hive@TEXAS.COM)s./*/hive/
RULE:[2:$1@$0](jhs@TEXAS.COM)s./*/mapred/
RULE:[2:$1@$0](jn@TEXAS.COM)s./*/hdfs/
RULE:[2:$1@$0](nm@TEXAS.COM)s./*/yarn/
RULE:[2:$1@$0](nn@TEXAS.COM)s./*/hdfs/
RULE:[2:$1@$0](rm@TEXAS.COM)s./*/yarn/
RULE:[2:$1@$0](yarn@TEXAS.COM)s./*/yarn/
```

```
#####
# Add common user principal to both the KDCs. Execute below commands
# on both the KDC, please keep same password for both the principals.
#####
```

For Cluster 1:

Login to kadmin

Execute below command to add a user principal

```
addprinc hadoop@TEXAS.COM
```

For Cluster 2:

Login to kadmin

Execute below command to add a user principal

```
addprinc hadoop@UTAH.COM
```

```
#####
```

```
#Login to Cluster 2, do a kinit and try to access hdfs files of Cluster 1
```

```
#####
```

Example:

```
hdfs dfs -ls hdfs://texas-2.texas.com:8020/tmp
```

```
Found 8 items
```

```
drwx----- - ambari-qa hdfs          0 2016-07-29 23:24 hdfs://texas-2.texas.com:8020/tmp/ambari-qa
drwxr-xr-x   - hdfs          hdfs          0 2016-07-29 22:02 hdfs://texas-2.texas.com:8020/tmp/entity-
file-history
drwx-wx-wx   - ambari-qa hdfs          0 2016-07-29 23:25 hdfs://texas-2.texas.com:8020/tmp/hive
-rwxr-xr-x   3 hdfs          hdfs          1414 2016-07-29 23:50 hdfs://texas-
2.texas.com:8020/tmp/id1aac2d44_date502916
-rwxr-xr-x   3 ambari-qa hdfs          1414 2016-07-29 23:26 hdfs://texas-
2.texas.com:8020/tmp/idtest.ambari-qa.1469834803.19.in
-rwxr-xr-x   3 ambari-qa hdfs          957 2016-07-29 23:26 hdfs://texas-
2.texas.com:8020/tmp/idtest.ambari-qa.1469834803.19.pig
drwxr-xr-x   - ambari-qa hdfs          0 2016-07-29 23:53 hdfs://texas-
2.texas.com:8020/tmp/tezsmokeinput
```

Note – texas-2.texas.com is the Active Namenode of Cluster 1.

You can try copying files from Cluster 2 to Cluster 1 using distcp

Example:

```
[hadoop@utah-1 root]$ hadoop distcp hdfs://texas-1.texas.com:8020/tmp/test.txt /tmp/
02/02/16 16:20:27 INFO tools.DistCp: Input Options: DistCpOptions{atomicCommit=false,
syncFolder=false, deleteMissing=false, ignoreFailures=false, maxMaps=20, sslConfigurationFile='null',
copyStrategy='uniformsize', sourceFileListing=null, sourcePaths=[hdfs://texas-
1.texas.com:8020/tmp/test.txt], targetPath=/tmp, targetPathExists=true, preserveRawXattrs=false}
02/02/16 16:20:27 INFO impl.TimelineClientImpl: Timeline service address: http://utah-
3.utah.com:8188/ws/v1/timeline/
02/02/16 16:20:27 INFO client.RMProxy: Connecting to ResourceManager at utah-
3.utah.com/192.168.0.50:8050
```

02/02/16 16:20:28 INFO hdfs.DFSClient: Created HDFS\_DELEGATION\_TOKEN token 20 for hadoop on 192.168.0.47:8020  
02/02/16 16:20:28 INFO security.TokenCache: Got dt for hdfs://texas-1.texas.com:8020; Kind: HDFS\_DELEGATION\_TOKEN, Service: 192.168.0.47:8020, Ident: (HDFS\_DELEGATION\_TOKEN token 20 for hadoop)  
02/02/16 16:20:29 INFO impl.TimelineClientImpl: Timeline service address: http://utah-3.utah.com:8188/ws/v1/timeline/  
02/02/16 16:20:29 INFO client.RMProxy: Connecting to ResourceManager at utah-3.utah.com/192.168.0.50:8050  
02/02/16 16:20:29 INFO hdfs.DFSClient: Created HDFS\_DELEGATION\_TOKEN token 24 for hadoop on ha-hdfs:utah  
02/02/16 16:20:29 INFO security.TokenCache: Got dt for hdfs://utah; Kind: HDFS\_DELEGATION\_TOKEN, Service: ha-hdfs:utah, Ident: (HDFS\_DELEGATION\_TOKEN token 24 for hadoop)  
02/02/16 16:20:29 INFO mapreduce.JobSubmitter: number of splits:1  
02/02/16 16:20:29 INFO mapreduce.JobSubmitter: Submitting tokens for job: job\_1469916118318\_0003  
02/02/16 16:20:29 INFO mapreduce.JobSubmitter: Kind: HDFS\_DELEGATION\_TOKEN, Service: 192.168.0.47:8020, Ident: (HDFS\_DELEGATION\_TOKEN token 20 for hadoop)  
02/02/16 16:20:29 INFO mapreduce.JobSubmitter: Kind: HDFS\_DELEGATION\_TOKEN, Service: ha-hdfs:utah, Ident: (HDFS\_DELEGATION\_TOKEN token 24 for hadoop)  
02/02/16 16:20:30 INFO impl.YarnClientImpl: Submitted application application\_1469916118318\_0003  
02/02/16 16:20:31 INFO mapreduce.Job: The url to track the job: http://utah-3.utah.com:8088/proxy/application\_1469916118318\_0003/  
02/02/16 16:20:31 INFO tools.DistCp: DistCp job-id: job\_1469916118318\_0003  
02/02/16 16:20:31 INFO mapreduce.Job: Running job: job\_1469916118318\_0003  
02/02/16 16:20:43 INFO mapreduce.Job: Job job\_1469916118318\_0003 running in uber mode : false  
02/02/16 16:20:43 INFO mapreduce.Job: map 0% reduce 0%  
02/02/16 16:20:52 INFO mapreduce.Job: map 100% reduce 0%  
02/02/16 16:20:53 INFO mapreduce.Job: Job job\_1469916118318\_0003 completed successfully  
02/02/16 16:20:53 INFO mapreduce.Job: Counters: 32

File System Counters

- FILE: Number of bytes read=0
- FILE: Number of bytes written=142927
- FILE: Number of read operations=0
- FILE: Number of large read operations=0
- FILE: Number of write operations=0
- HDFS: Number of bytes read=346
- HDFS: Number of bytes written=45
- HDFS: Number of read operations=12
- HDFS: Number of large read operations=0
- HDFS: Number of write operations=2

Job Counters

- Launched map tasks=1
- Other local map tasks=1
- Total time spent by all maps in occupied slots (ms)=14324
- Total time spent by all reduces in occupied slots (ms)=0
- Total time spent by all map tasks (ms)=7162
- Total vcore-seconds taken by all map tasks=7162
- Total megabyte-seconds taken by all map tasks=7333888

Map-Reduce Framework

- Map input records=1
- Map output records=1
- Input split bytes=118
- Spilled Records=0
- Failed Shuffles=0

Merged Map outputs=0  
GC time elapsed (ms)=77  
CPU time spent (ms)=1210  
Physical memory (bytes) snapshot=169885696  
Virtual memory (bytes) snapshot=2337554432  
Total committed heap usage (bytes)=66584576  
File Input Format Counters  
Bytes Read=228  
File Output Format Counters  
Bytes Written=45  
org.apache.hadoop.tools.mapred.CopyMapper\$Counter  
BYTESSKIPPED=0  
SKIP=1  
Note – texas-1.texas.com is the Active Namenode of Cluster 1.