**Hadoop Zookeeper Tunnel**

Zookeeper is the interface which eCFD DA uses to control Hadoop analysis. There are three Zookeeper servers "(called an ensemble) and the servers are aware of each other. As long as a critical mass of servers is available, the ZooKeeper service will also be available. There is no single point of failure." (Reference: <http://hortonworks.com/hadoop/zookeeper/>). For our situation all of the servers are not directly accessible from the BEN. Access is through hdedge.ca.boeing.com for all Hadoop services. We set up a tunnel using ssh port forwarding which sends all Zookeeper communication back and forth between ecfd-dev and the Zookeeper servers using hdedge as the pass-through.

Here is the portion of the .ssh/config for the tunnel for user ecfd on ecfd-dev-01 server:

Host zookeeper\_tunnel

User svcecfd

Hostname hdedge.ca.boeing.com

IdentityFile /boeing/sw/ecfd/.ssh/zookeeper\_tunnel

HostKeyAlias zookeeper\_tunnel

ServerAliveInterval 15

ServerAliveCountMax 3

ExitOnForwardFailure yes

LogLevel ERROR

ServerAliveInterval pings the ssh server if there has been no activity for the specified period. ServerAliveCountMax tries that many times and shuts down the connection if there is no response. ExitOnForwardFailure shuts the process down if the tunnel is not established.

The ssh tunnel is initiated from ecfd-dev using the following command:

sudo -u ecfd ssh -f -N \

-L mho101-8:2181:mho101-8:2181 \

-L mho101-9:2181:mho101-9:2181 \

-L hdo101-13:2181:hdo101-13:2181 \

-L mho101-8:60000:mho101-8:60000 \

-L mho101-9:60000:mho101-9:60000 \

-L hdo101-13:60020:hdo101-13:60020 \

-L hdo101-15:60020:hdo101-15:60020 \

-L hdo101-16:60020:hdo101-16:60020 \

-L mho101-9:60030:mho101-9:88 \

zookeeper\_tunnel

More ports may be necessary for master and region failover servers.

Port 88 is used for Kerberos activities. Notice that the port on the client side (ecfd-dev) is 60030, not 88. That is because port 88 is a privileged on the client side and ssh refused to create the tunnel with it unless it is created by root. This command will be run by ecfd. The solution is to use port 60030 on the client side. A modification in /etc/krb5.conf is required. Add ':60030" to the computers in the definition of admin\_server and kdc.

It is possible to set up the tunnel using names that are local to ecfd-dev (such as localhost1, localhost2, ...) But the log files indicate that traffic comes back from Zookeeper which contains fully qualified names of Zookeeper servers which the application running on ecfd-dev uses to communicate back to Zookeeper. So we set up the tunnel using the names. In addition, the Zookeeper master and port must be specified (server mho101-9.ca.boeing.com and port 60000).

The names of the Zookeeper servers were translated to localhost IP addresses in cfd-dev, /etc/hosts, using the loopback address:

127.0.0.8 mho101-8.ca.boeing.com mho101-8  
127.0.0.9 mho101-9.ca.boeing.com mho101-9  
127.0.0.13 hdo101-13.ca.boeing.com hdo101-13  
127.0.0.15 hdo101-15.ca.boeing.com hdo101-15  
127.0.0.16 hdo101-16.ca.boeing.com hdo101-16

A SSH key was set up on hdedge to permit the tunnel to be established. It is a no password key. The authorized\_keys file on hdedge contains the following parameters on the key to restrict access:

command="/data/home/svcecfd/bin/zookeeper\_tunnel.sh",permitopen="mho101-8:2181",permitopen="mho101-9:2181",permitopen="hdo101-13:2181",permitopen="mho101-8:60000",permitopen="mho101-9:60000",permitopen="hdo101-13:60020,permitopen="hdo101-15:60020,permitopen="hdo101-16:60020",no-agent-forwarding,no-X11-forwarding,no-pty

Setting up a SSH tunnel does not run any commands on the target system. That is the reason for the -N flag in the SSH command above. /data/home/svcecfd/bin/zookeeper\_tunnel.sh is only run if a user tries to open a tunnel and run a command. The script just prints an error message and exits. The tunnel is not established.

"permitopen" restricts the host:port combinations which can opened using the key. It appears the parameter restricts the key to be used to only forward ports that are specified. All the host:port combinations that are to be forwarded must be defined using separate permitopen parameters. A subset of them can be used when creating a tunnel.

The websites below contain information on SSH tunneling:  
<https://chamibuddhika.wordpress.com/2012/03/21/ssh-tunnelling-explained/>  
<http://unix.stackexchange.com/questions/109749/tunnels-to-the-same-port-number>

Information on client to Hadoop/Zookeeper communication including port numbers:  
<http://blog.cloudera.com/blog/2013/07/guide-to-using-apache-hbase-ports/>

Information on how to keep the tunnel open:  
<http://superuser.com/questions/37738/how-to-reliably-keep-an-ssh-tunnel-open> <https://drupal.star.bnl.gov/STAR/comp/sofi/facility-access/ssh-stable-con>