APPENDIX E

Incast Utility Scripts, IXIA Config

This appendix includes scripts used for setting up and creating Incast events in the lab. “fail-mapper.sh” fails relevant Hadoop mappers (VMs), thus causing a shift in traffic. “find-reducer.sh” determines the location of relevant reducers. “tcp-tune.sh” and “irqassign.pl” help prepare the servers for the lab environment.

fail-mapper.sh

```
#!/bin/bash

URL=http://jobtracker.jt.voyager.cisco.com:50030/jobdetails.jsp?jobid=job_201211051628
_
RURL=http://jobtracker.jt.voyager.cisco.com:50030/taskdetails.jsp?tipid=task_201211051
628_
JOB_ID=$1
JOB_ID2=$2
RLOG=_r_000000
MLOG=_m_0000
W=0
counter=0

for i in {0..77};
do
   printf -v MID "%03d" $i
   wget $RURL$JOB_ID$MLOG$MID
   wget $RURL$JOB_ID2$MLOG$MID
done

grep attempt taskdetails.jsp?tipid=task_201211051628_$JOB_ID* | awk -F '<' '{print
$6}' | cut -c 67-78 |
sort -u | sed 's/vm*/vm-/'> vmnames1

grep attempt taskdetails.jsp?tipid=task_201211051628_$JOB_ID2* | awk -F '<' '{print
$6}' | cut -c 67-78 |
 sort -u | sed 's/vm*/vm-/'> vmnames2

'rm' task*

diff --suppress-common-lines vmnames1 vmnames2 | grep ">" | sed 's/> //' > vmnames
while [ $W -lt 96 ]
do
   wget $URL$JOB_ID
   W1=`grep "jobtasks.jsp?jobid=job_201211051628_$JOB_ID&type=map&pagenum=1"
jobdetails.jsp?jobid=job_201
211051628_$JOB_ID | awk -F 'align="right">' '{print $2}' | cut -c 1-5`
```

```
   rm jobdetails.jsp?jobid=job_201211051628_$JOB_ID
   if [ -z $W1 ]; then
      W=0
   else
      W=${W1/\.*}
   fi
   echo -e "\n currently at ******** $W \n"
done

echo "physical breakdown"
for i in {1..15..2}
do
  let j=$i+1
  for z in {1..5}
    do
      unset HOSTS
      unset HOSTS2
      if (( $i < 10 ))
        then
          HOSTS=($(cat vmnames | grep r0$i-p0$z | cut -c 9-13))
          HOSTS2=($(cat vmnames | grep r0$j-p0$z | cut -c 12-13 | awk '{printf
"vm-%02d\n", $1+7}'))
      if (( $j == 10 ))
        then
          unset HOSTS2
          HOSTS2=($(cat vmnames | grep r$j-p0$z | cut -c 12-13 | awk '{printf
"vm-%02d\n", $1+7}'))

      fi
      fi

      if (( $i > 10 ))
        then
          HOSTS=($(cat vmnames | grep r$i-p0$z | cut -c 9-13))
          HOSTS2=($(cat vmnames | grep r$j-p0$z | cut -c 12-13 | awk '{printf
"vm-%02d\n", $1+7}'))
      fi

     if [[ ! -z $HOSTS ]]; then
        for h in ${HOSTS[@]}
          do
            if [ -z "$WW" ]; then
              printf -v WW  "virsh destroy $h"
              printf -v WW1  "virsh start $h"
            else
              printf -v WW "virsh destroy $h ; $WW"
              printf -v WW1 "virsh start $h ; $WW1"
            fi
          done
     fi

     for h2 in ${HOSTS2[@]}
      do
        if [ -z "$WW" ]; then
          printf -v WW "virsh destroy $h2"
          printf -v WW1 "virsh start $h2"
        else
          printf -v WW "virsh destroy $h2 ; $WW"
          printf -v WW1 "virsh start $h2 ; $WW1"
        fi
      done

     if [ ! -z "$WW" ] && (( $counter < 10 )); then
      counter=$counter+1
```

```
      printf -v HOSTN "%02d" $i
      printf "ssh  -o StrictHostkeyChecking=no r$HOSTN-p0$z.hosts.voyager.cisco.com \"
$WW1 \""
      ssh  -o StrictHostkeyChecking=no r$HOSTN-p0$z.hosts.voyager.cisco.com " $WW "
     fi

  unset WW
  unset WW2
  done
done

'rm' task*
```

find-reducer.sh

```
#!/bin/bash

URL=http://jobtracker.jt.voyager.cisco.com:50030/jobdetails.jsp?jobid=job_201211051628
_
RURL=http://jobtracker.jt.voyager.cisco.com:50030/taskdetails.jsp?tipid=task_201211051
628_
JOB_ID=$1
RLOG=_r_000000
MLOG=_m_0000
W=0
counter=0
while [ $W -lt 100 ]
do
   wget $URL$JOB_ID
   W1=`grep "jobtasks.jsp?jobid=job_201211051628_$JOB_ID&type=map&pagenum=1"
jobdetails.jsp?jobid=job_201211051628_$JOB_ID | awk -F 'align="right">' '{print $2}' |
cut -c 1-5`
   rm jobdetails.jsp?jobid=job_201211051628_$JOB_ID
   if [ -z $W1 ]; then
      W=0
   else
      W=${W1/\.*}
   fi
   date;
   echo -e "\n currently at ******** $W \n"
done

while [ "$Reducer1" == "" ]
do
   wget $RURL$JOB_ID$RLOG
   Reducer1=`grep "<td>attempt_201211051628_$JOB_ID$RLOG"
taskdetails.jsp?tipid=task_201211051628_$JOB_ID$RLOG | awk -F "</*td>" '{print $4}' |
cut -c 17-28`
   Reducer=`echo $Reducer1 | cut -c 1-7`
   RACK=`echo $Reducer1 | cut -c 2-3`
   POD=`echo $Reducer1 | cut -c 6-7`
   'rm' taskdetails.jsp?tipid=task_201211051628_$JOB_ID$RLOG
done

EVENODD=`expr $RACK % 2`
echo -e "\n reducer is at Physical host ***** $Reducer $Reducer1 \n"
if [ $RACK -eq 13 -o $RACK -eq 14 ]; then
   if [ $EVENODD -eq 1 ]; then
     ssh -o StrictHostkeyChecking=no $Reducer.hosts.voyager.cisco.com "date ; tcpdump
-s128 -i eth2 -n -w bla3"
   else
```

```
      RACK=`expr $RACK - 1`
      printf -v RACKID "%02d" $RACK
      ssh -o StrictHostkeyChecking=no r$RACKID-p$POD.hosts.voyager.cisco.com "date ;
tcpdump -s128 -i eth3 -n -w bla3-eth1"
   fi
else
   if [ $EVENODD -eq 1 ]; then
     ssh -o StrictHostkeyChecking=no $Reducer.hosts.voyager.cisco.com "date ; tcpdump
-s128 -i eth0 -n -w bla3"
   else
      RACK=`expr $RACK - 1`
      printf -v RACKID "%02d" $RACK
      ssh -o StrictHostkeyChecking=no r$RACKID-p$POD.hosts.voyager.cisco.com "date ;
tcpdump -s128 -i eth1 -n -w bla3-eth1"
   fi
fi
```

tcp-tune.sh

```
#!/bin/bash

date >> /tmp/setup_tcp.log
# setting tcp send and receive rules.
echo "setting rmem wmem default and max..." >> /tmp/setup_tcp.log
echo 524287 > /proc/sys/net/core/rmem_default
echo 524287 > /proc/sys/net/core/wmem_default
echo 33554432 > /proc/sys/net/core/rmem_max
echo 33554432 > /proc/sys/net/core/wmem_max
echo 33554432 > /proc/sys/net/core/optmem_max
echo 3000000 > /proc/sys/net/core/netdev_max_backlog
echo "setting tcp rmem and tcp_wmem..." >> /tmp/setup_tcp.log
echo "33554432  33554432  33554432" > /proc/sys/net/ipv4/tcp_rmem
echo "33554432  33554432  33554432" > /proc/sys/net/ipv4/tcp_wmem
echo "33554432  33554432  33554432" > /proc/sys/net/ipv4/tcp_me
```

irqassign.pl

```
#!/usr/bin/perl
use strict;
use POSIX;

# Open a logfile.
my $log;
open($log, '>>/tmp/irq_assign.log') or die "Can't open logfile: $!";
print $log strftime('%m/%d/%Y %H:%M:%S', localtime), ": Starting run.\n";

my %irqmap = (
   79 => 2, # Start of eth1
   80 => 200,
   81 => 8,
   82 => 800,
   83 => 20,
   84 => 2000,
   85 => 80,
   86 => 8000,
   87 => 2,
   88 => 200,
   89 => 8,
```

```
   90 => 800,
   91 => 20,
   92 => 2000,
   93 => 80,
   94 => 8000,
   95 => 2, # End of eth1
   62 => 1,    # Start of eth0
   63 => 100,
   64 => 4,
   65 => 400,
   66 => 10,
   67 => 1000,
   68 => 40,
   69 => 4000,
   70 => 1,
   71 => 100,
   72 => 4,
   73 => 400,
   74 => 10,
   75 => 1000,
   76 => 40,
   77 => 4000,
   78 => 40,   # End of eth0
);

foreach my $irq (sort(keys(%irqmap))){
   my $fh;
   open($fh, "+>/proc/irq/$irq/smp_affinity") or die "Can't read $irq: $!";
   my $value = <$fh>;
   chomp($value);
   print $log "Current value of IRQ $irq = $value, setting to $irqmap{$irq}.\n";
   truncate($fh, 0);
   seek($fh, 0, 0);
   print $fh $irqmap{$irq};
   close($fh);
}

# And for good measure, enable forwarding.
my $fh;
open($fh, "+>/proc/sys/net/ipv4/ip_forward") or die "Can't read ip_forward: $!";
my $value = <$fh>;
chomp($value);
print $log "Current value of ip_forward = $value, setting to 1.\n";
truncate($fh, 0);
seek($fh, 0, 0);
print $fh '1';
close($fh);
```

VM configuration

```
   for z in {1..16..2};
   do
      for i in {1..5};
      do
         printf -v RACK "%02d" $z;
         ssh r$RACK-p0$i.hosts.voyager.cisco.com " virsh setvcpus vm-01 4 --maximum
--config;
         virsh setvcpus vm-08 4 --maximum --config";
      done;
   done
```

```
for z in {1..16..2};
do
   for i in {1..5};
   do
      printf -v RACK "%02d" $z;
      ssh r$RACK-p0$i.hosts.voyager.cisco.com " virsh setmaxmem vm-01 24576000;
      virsh setmaxmem vm-08 24576000";
   done;
done

for z in {1..16..2};
do
   for i in {1..5};
   do
      printf -v RACK "%02d" $z;
      ssh r$RACK-p0$i.hosts.voyager.cisco.com " virsh setvcpus vm-01 4 --config;
      virshsetvcpus vm-08 4 --config";
   done;
done

for z in {1..16..2};
do
   for i in {1..5};
   do
      printf -v RACK "%02d" $z;
      ssh r$RACK-p0$i.hosts.voyager.cisco.com " virsh setmem vm-01 20480000
--config;
      virsh setmem vm-08 20480000 --config";
   done;
done
```