Network Analysis of ESX to Virtual Center Communications

Well Well Well !!!!!

Guys am back from a long vacation (well to me the vacation is the packaging a new thing). Well we can talk about my next venture definitely some time.

Now people were asking me what are you doing these days and I said folks gimme some time so that I can come up with the best shot as usual.

Now I thought letz post this twist. I was bogged down with this since a long time and now got all the big shots in my hand.

Well sorry about a late post. I was completely busy with my new packaging and product and also with the vCenter Orchestrator training.

Now let me start on this. This is pretty big one but a cool stuff.

Issue is as below

Intermittent ESX host disconnect alarms from the Virtual Center. Typical disconnect/reconnect times are less than 5 minutes.

Action Plan Taken as below

Verified/Validate all physical network connections have specific speed/duplex settings. Verified physical switch ports are not showing signs of errors. Verified utilization of switch ports are below maximums. Virtual Center management configuration is per specification. ESX Host servers are configured to matching specifications. Network trace captures have been collected and analyzed.

Action needed finally

Need to answer the questions below surrounding proper application responses, network transmission patterns, and general what the network patterns should be, and how the disconnect alarms are triggered.

Open Questions

1. How exactly does the ESX Host determine that it is no longer communicating with the VC server?
2. What is the exact process that restarts ‘hostd’ or other services to re-establish ESX to VC communications?
3. What is the timeline of ESX Host determining there is an issue and the restart of the ‘hostd’ process to re-establish VC communications?
4. What is the normal timeline for VC to ESX Host heartbeats?
5. What is the normal timeline for ESX to VC Heartbeats?
6. Why is there a small and large UDP 902 packet from each ESX host?
7. Why is the small UDP 902 packet always 71 bytes?

8. Why is there variation in the order of the large/small pattern?

9. Why do different ESX hosts use a different large UDP 902 size?

10. Why does the size of the large UDP 902 packet for a given ESX host change?

11. Does the TCP 902 traffic from the VC server play any role in the hostd restart?

12. Why do we see gaps as long as 3,384 seconds with no VC to ESX TCP 902 traffic?

Backup Information provided on the next few pages showing the analysis of the network captures.

The two baseline traces of "normal ESX traffic" provided some interesting data for consideration.

Trace 4506 A:

• 2-hour trace taken on 4/24 from 8:50 AM to 10:50 AM

• Contains UDP/TCP 902 traffic for 11 ESX hosts

Trace 4506 B:

• 2-hour trace taken on 4/24 from 8:50 AM to 10:50 AM

• Contains UDP/TCP 902 traffic for 8 ESX hosts

Here is a summary of the UDP/TCP 902 characteristics seen in these baseline traces, sorted by ESX host IP address:

19 ESX hosts were seen in the two trace files as expected:

Here are some observations for the UDP 902 traffic:

• The UDP 902 traffic is one-way from the ESX host to the VC server as expected

• The delta time between UDP 902 heartbeats is very consistent with no gaps found in either trace file

• The minimum delta is consistently 10.0 seconds and the maximum observed delta was 10.9 seconds

• Each ESX host sends UDP 902 packets with the same two sizes, a large packet and a small packet

• The small UDP 902 packet was always 71 bytes for all 19 ESX hosts

• The large UDP 902 packet varies from 320 bytes minimum to 1,490 bytes maximum, but each host only uses one large size in these traces

• The typical pattern is to alternate large and small UDP 902 packets, but we do see some exceptions as shown below:

Note the double small and double large UDP 902 packets highlighted below:

Here are some follow-up questions for the UDP 902 traffic:

- Why is there a small and large packet from each ESX host?
- Why is the small packet always 71 bytes?
- Why is there variation in the order of the large/small pattern?
- Why do different ESX hosts use a different large size?

For the earlier trace with the ESX host disconnect, we see an interesting pattern of UDP 902 packet sizes:

- For 1,910 seconds from the start of the trace to the gap in UDP 902 traffic, we see a pattern of 71 bytes / 75 bytes
- For 411 seconds during the gap we see no UDP 902 traffic
- For 265 seconds just after the gap, we see a pattern of 71 bytes / 125 bytes
- For 1,015 seconds to the end of the trace, we see a pattern of 71 bytes / 190 bytes

The baseline trace A shows this ESX host sending a pattern of 71 bytes / 710 bytes. We should try to understand the reason for the different sizes.

Note the changes in UDP 902 packet sizes around the gap in traffic:

Here are some observations for the TCP 902 traffic:

- The TCP 902 traffic is initiated by the VC server and is bi-directional as expected
- The number of TCP 902 connections per ESX host varies from 2 to 8 during the 2-hour trace
- There is a wide variation of packets and bytes per ESX host
- There are large time gaps ("max delta" on chart) with no TCP 902 traffic from the VC server

Here are some follow-up questions for the TCP 902 traffic:

- In the earlier "ESX host disconnect" trace, there was a ~310 second gap from the last VC server TCP 902 packet and the syslog message showing a hostd restart
- If the absence of TCP 902 traffic is triggering the hostd restart, the timeout value should be less than 310 seconds
- Note that during the baseline traces, every ESX host had gaps larger than 310 seconds with no triggered hostd restart ß Max gap was 3,384 seconds

We should try to understand how hostd decides to trigger a service restart. The baseline traces make it hard to see how the TCP 902 traffic could be the trigger.

Now this is the time to answer these questions. Look below for the answers.

1. How exactly does the ESX Host determine that it is no longer communicating with the VC server?

Ans. When the ESX Host does not respond back within ~20 seconds from the VC initiated Hostsync process.

2. What is the exact process that restarts 'hostd' or other services to re-establish ESX to VC communications?

Ans. The vmware-watchdog process.

3. What is the timeline of ESX Host determining there is an issue and the restart of the 'hostd' process to re-establish VC communications?

Ans. It checks every minute for hostd to be running, -u 60, and if it finds that it's not running after 5 failed checks, -q 5, it will restart hostd.

4. What is the normal timeline for VC to ESX Host heartbeats?

Ans. Heartbeat is initiated from the ESX Host.

5. What is the normal timeline for ESX to VC Heartbeats?

Ans. Every 10 seconds.

6. Why is there a small and large UDP 902 packet from each ESX host?

Ans. The message sent to VC from vpxa is a serialization of VpxHeartbeatMsg object. However, it’s size depends on the quick status size. In the smallest situation, no quick status needs to be sent. The quick status depends on the machine status such as HA.

7. Why is the small UDP 902 packet always 71 bytes?

Ans. See answer to 6 above.

8. Why is there variation in the order of the large/small pattern?

Ans. Depends on running time status of ESX. In case a host keeps adding new VMs or removing new VMs, it will have more status data.

9. Why do different ESX hosts use a different large UDP 902 size?

Ans. It depends on running state of the ESX host.

10. Why does the size of the large UDP 902 packet for a given ESX host change?

Ans. It depends on running state of the ESX host.

11. Does the TCP 902 traffic from the VC server play any role in the hostd restart?

Ans. TCP traffic from VC is used to transport remote RDP call, (in our term, VMOMI call). It does not play a role.

12. Why do we see gaps as long as 3,384 seconds with no VC to ESX TCP 902 traffic?

Ans. When ESX has no status change for example, all VMs are powered off, and no status change on ESX, (both Host or VMs), there is no sync needed between VC and ESX. The only traffic will be udp traffic to VC server to indicate that the Host is alive.