Network Analysis and Troubleshooting

Basics of network troubleshooting

- When you suspect that the failure you're experiencing may be related to the network, the following are considered basic troubleshooting procedures:
  - Make sure that the correct IP address and default gateway has been assigned to the machine
  - Test basic network connectivity
  - Ensure that name resolution is correctly configured (DNS or /etc/hosts)
  - Revising the firewall rules on the local machine as well as any other firewalls in the packets path
- Each of those steps has its own command(s), which we are going to examine in this section

Packet INternet Grouper (PING)

- The most basic yet important network troubleshooting command
- It is used to check the status of hosts or complete network segments
- Some challenges involve having some hosts and gateways intentionally ignore PING requests (like using a firewall)
- It can be used also to test name resolution by providing the hostname instead of the IP address.
- Command options:
  - -c *number* send the specified number of requests instead of indefinitely
  - -s *number* specifies the MTU size to use (by default 1500 on standard Ethernet) specifying a larger MTU size forces fragmentation, which may aid you in troubleshooting congested network problems for example
- Command result is either a timeout message if the remote host did not (or would not) reply, or a sequence of replies.
- The time value expressed in milliseconds can be a basic measure of network speed

Trace your packets (traceroute)

- It outputs the number of gateways that packets travel through until they reach their final destination
- The syntax is `traceroute` *`hostname/IP address`*
- The mechanism by which traceroute works is simple: it just sends a packet (3 packets actually) to the hostname with a low TTL (Time to Live). When it receives the packet, each gateway decreases the TTL until it reaches zero. When that happens, the gateway sends an ICMP "time exceeded" message back to the originating host. The originating host determines the source of the ICMP message and resolves it using DNS to a hostname.
- The process starts by sending a packet with a TTL of 1, the first gateway (hop) decreases the TTL by 1 to make it 0. The second packet has its TTL set to 2, the first gateway decreases it to 1 while the second gateway decreases it to 0 and sends the ICMP message.
- The process goes on until the number of hops is equal to the number of TTL
- When a star (*) is returned in the time field of the response, this means that the router did not supply a response. This may be due to either a network congestion, or because the host is actually down. Additionally, some firewalls are configured to drop expired-TTL packets. It could also be that the gateway did not return back the error packet until traceroute stopped waiting.

Analyze your host networking (netstat)

- The netstat command includes a full set of tools to analyze and display rich information about networking on the server. In the following we examine the most important usages:
  - netstat –i: it displays information about the network interfaces on the system. The output is similar to that of ifconfig –a. The RX and TX counters show the total number of received and sent packets since machine boot. The errors counter is also important. They can be safely ignored if they are less than 1% of the total packets. A high error rate on a single machine may indicate a problem with the network card, while a high error rate on a number of machines indicates a failure in the network or media.
  - netstat: with no arguments, it displays the status of TCP and UDP ports on the system. Only ports with active connections are displayed. If –a was added, the *listening* ports (no active connections yet) are displayed. Ports are displayed by name as per the /etc/services mapping, and hostnames are displayed by their DNS name. You can override this behavior by adding –n to the command arguments; which displays hostnames and ports numerically. If you want to list only the listening ports, use –l argument. Use –p to display the PID of the associated processes.

Connection states

- The Send-q and Receive-q represent the queue of packets waiting to be sent or received. Those numbers should be close to zero on a normal host.
- The TCP connection state may be ESTABLISHED (active connections), LISTEN (waiting for a connection), or TIME_WAIT (in the process of closing a connection).
- If the TCP state is SYN_SENT, this means that the process is trying to contact a non-accessible host on the network
- If you find a lot of SYN_WAIT connection states, this means that the machine is unable to handle the number of connections requested. It may need some kernel tuning or it may be under some sort of an attack.

The routing table

- To display the routing table, use netstat –r. You can add –n to display the output in numerical format.
- The flags indicate the type of host:
  - U: up, active
  - G: gateway
  - H: host
- There should always be a default gateway for any machine connected to the network (with some exceptions like backbone routers). The default route is displayed either by the word *default* or by the IP address 0.0.0.0

Monitoring *live* network activity

- On Linux, netstat cannot be used for this. You will have to use `sar`, which is part of the sysstat package.
- It may be used to capture network activity for a specific period of time. The syntax is as follows:
  `sar –n DEV` *`time count`*
- The DEV here is written literally, it is not a reference to a device. The time is the period – in seconds – through which the command will gather the statistics (for example 5 seconds). The count is the number of times the command will run. Accordingly sar –n DEV 5 12 will gather statistics every five seconds, twelve times (1 minute). The noticeable fields of the output are rxbyte/s and txbyte/s for the number of received and send bytes per second respectively, which indicates the current bandwidth used by the machine.
- You can use ping with a large MTU size and observe the output of sar –n DEV to have a good overview of your network performance

Sniffing on the network

- Packet sniffers are tools that let you inspect the packets travelling from and to (or just pass by) your network card. They usually work according to a filter that specifies which packets to capture.
- Even on a switched network, a lot of information can be gathered using network sniffers.
- Normally, a network interface will only allow broadcast/multicast packets to the software layer (the kernel). But if it is working in promiscuous mode, it will relay each and every packet arriving even if it is not intended for the current host.
- For this reason, all services/applications that sends credentials across the network should use secure protocols (like SSL) in order not to be captured by a network sniffer.
- Packet sniffers have the capability of reading streams of packets and displaying them in human-readable format.

`tcpdump`, the default network sniffer

- It can be installed using package manager (yum or apt-get)
- Can display packet information on the screen or to a file in libpcap format
- The -i option to specify the interface on which it will work. Otherwise it will work on the first available interface
- The –A to display information in ASCII human-readable format
- The –n to switch on numerical display (not use DNS)
- The –v or –vv to display more verbose output
- The –s (with –w) to capture full packet details instead of the default header information
- The –r switch to read the information written in the file