Volume

1

INFOSEC4FUN/UNIX4FUN DOCS

IT Troubleshooting Documentation

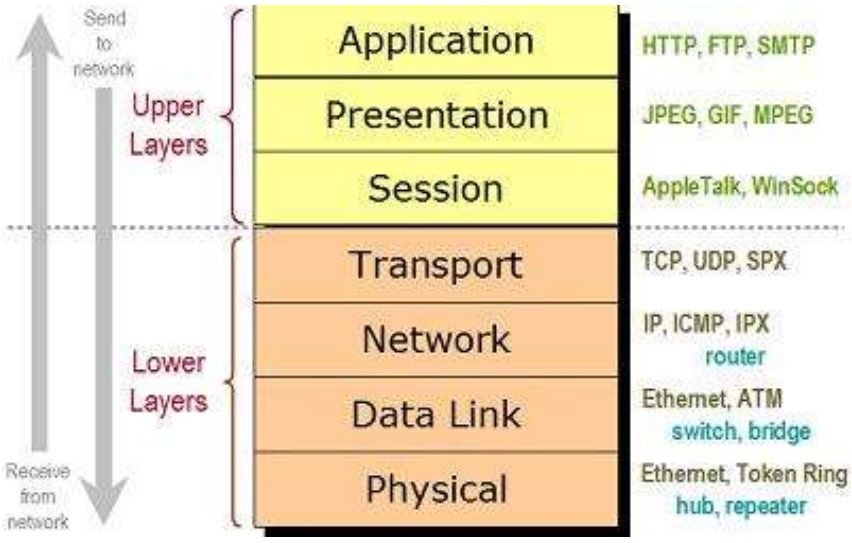
Troubleshooting the Stack

Chapter 1

OSI Model

Troubleshooting should always start with the basics.

To provide effective methodologies for troubleshooting network and ITrelated issues it's important to understand the communications of the OSI model. From the initial source where an electrical signal is produced to the point where the application is presented to the user, understanding these steps is an integral part of determining a solution to an issue.



Physical Layer

The physical layer of the OSI model is where electrical and physical specifications are defined. There is a direct relationship defined between the device and the transmission medium. This layer includes copper or fiber cable, power, voltage, hubs, repeaters network adapters and more.

This layer determines the protocol to establish and terminate the connection between two connected nodes.

Troubleshooting the physical layer - This should be the first step for any network related issues. Many times overlooked, an issue at this layer can be disguised by an application not working properly. Example, a UNIX server has two network interfaces, one in a DMZ, the other connection to the internal network. You notice you can no longer connect to a server in the DMZ from the internal network. While there may be many scenarios as to why this no longer works, it could just be a bad cable or interface card on the server or switch. Initially starting at Layer one and looking for errors can save hours of troubleshooting headaches.

On the server look at whether the interface is up and data is flowing using, tcpdump, or ifconfig or ethtool from the CLI. Other examples, bad switch port, cable not seated properly, etc. If you see data flowing, you can then move on to the next layer.

Ethtool Example:

```
Unix-Firewall# ethtool eth0
Settings for eth0:
        Supported ports: [ TP ]
        Supported link modes:   10baseT/Half 10baseT/Full
                                100baseT/Half
100baseT/Full
                                1000baseT/Full
        Supports auto-negotiation: Yes
        Advertised link modes:  10baseT/Half 10baseT/Full
                                100baseT/Half
100baseT/Full
                                1000baseT/Full
        Advertised pause frame use: No
        Advertised auto-negotiation: Yes
        Speed: 1000Mb/s
        Duplex: Full
        Port: Twisted Pair
        PHYAD: 1
        Transceiver: internal
        Auto-negotiation: on
        MDI-X: Unknown
        Supports Wake-on: g
        Wake-on: g
        Link detected: yes
```

Data Link Layer

The data link layer provides reliable link connectivity between two directly connected nodes by detecting and correcting errors that may occur on the physical layer.

This layer is divided into two sub layers:

Media Access Control (MAC), is responsible for controlling how computers on an network gain access to the data to transmit it

Logical Link Control (LLC) provides control error checking and packet synchronization.

Troubleshooting the Data Link Layer - You can see if the device is directly connected by looking for the MAC address and doing an ARP command (arp –a).

You may have to ping the device first from the CLI to create an ARP entry. Again this works if the device is directly connected and there are no additional hops to the device from where you are troubleshooting. Example below is for a device directly behind a firewall off of Eth3 interface.

If you see the MAC address for the device you are troubleshooting you can move on to the next layer..

```
Unix-Firewall# ping 192.168.1.1
PING 192.168.112.4 (192.168.112.4) 56(84) bytes of data.
64 bytes from 192.168.1.1: icmp_seq=1 ttl=64 time=0.278 ms
64 bytes from 192.168.1.1: icmp_seq=2 ttl=64 time=3.68 ms
64 bytes from 192.168.1.1: icmp_seq=3 ttl=64 time=0.182 ms
--- 192.168.1.1 ping statistics ---
11 packets transmitted, 11 received, 0% packet loss, time
10004ms
rtt min/avg/max/mdev = 0.132/0.844/3.857/1.381 ms

Unix-Firewall# arp -a 192.168.1.1
? (192.168.1.1) at 00:10:7G:28:1D:3A [ether] on eth3
```

Network Layer

The network layer provides the functional means of transferring variable length data sequences (called datagrams) from one node to another connected to the same network. This layer determines the method of routing the data.

Routing protocols such as multicast group management, network layer information and error detection, and network-layer assignment all function at this layer.

Troubleshooting the Network layer - You can search for the device on the network by using various commands at the CLI. Traceroute, Nslookup, DIG to name a few. Traceroute should show the connection route hop the data travels to reach the destination. If you are able to successfully resolve the device and show sufficient routing information to it you can move on to the next layer.

- Can you ping your default gateway
- Are you IP settings correct
- Can you tracert a well know IP/DNS address / URL
- Has your network adaptor been assigned an APIPA address
- Are your DNS / DHCP settings correct
- What out do you get from ipconfig, ping, tracert

```
C:\>tracert -d 192.168.1.1
Tracing route to 192.168.112.4 over a maximum of 30 hops
  1    <1 ms    <1 ms    <1 ms  10.100.2.253
  2     1 ms    <1 ms    <1 ms  10.10.1.254
  3     1 ms    <1 ms    <1 ms  192.168.1.1


Trace complete.

C:\>nslookup
Default Server:  dns.company.com
Address:  10.10.2.1

> webserver
Server:  dns.company.com
Address:  10.10.2.1

Name:    webserver.company.com
Address:  192.168.1.1
```

Transport Layer

The transport layer controls the reliability of a given link through flow control, segmentation/de-segmentation, and error control. This layer covers the transmission of data streams using connection and connectionless oriented protocols like TCP/UDP, IPX, and end to end encryption with IPsec.

Troubleshooting the Transport layer - You can verify the data is flowing between two nodes using a specific protocol. If it is a specific TCP port you can try to telnet to the device on its port number.

Example, to verify the http port is listening properly on a web server you can telnet to its IP address on port 80.

```
Telnet 192.168.1.1:80
Trying 192.168.1.1...
Connected to 192.168.1.1.
Escape character is '^]'.

GET /index.htm HTTP/1.1
host: www.esqsoft.globalservers.com
```

Your output should look similar to this if the port is listening properly and the HTTP server is functioning as expected.

```
 HTTP/1.1 200 OK
Date: Mon, 18 Apr 2005 16:38:00 GMT
Server: Apache/1.3.27 (Unix)  (Red-Hat/Linux)
mod_ssl/2.8.12 OpenSSL/0.9.6 PHP/4.1.2 mod_perl/1.27
FrontPage/5.0.2.2623
Last-Modified: Thu, 01 Jul 2004 01:16:05 GMT
ETag: "158e008c-182c-40e365d5"
Accept-Ranges: bytes
Content-Length: 6188
Connection: close
Content-Type: text/html

<html>
<head>
<title>...

...lots of HTML code here...

</body></html>
Connection closed by foreign host.
```

Session Layer

The session layer deals with the creation and control for connections between computers to higher layers (FTP, Telnet, CIFS) It establishes, manages and terminates the connections between the local and remote application.

Authentication Netbios, and NFS function at this layer.

Troubleshooting the Session layer - You can verify connectivity at this layer by using some of the windows nbtsat commands to verify session connections of a device.

```
C:\>nbtstat -a localcomputer

Local Area Connection:
Node IpAddress: [10.10.1.32] Scope Id: []

           NetBIOS Remote Machine Name Table

       Name               Type         Status
    ---------------------------------------------
    localcomputer      <20>  UNIQUE      Registered
    localcomputer      <00>  UNIQUE      Registered
    company            <00>  GROUP       Registered
    company            <1E>  GROUP       Registered

    MAC Address = 00-1C-C4-8F-A8-08
```

Presentation Layer

This layer provides independence from differences in data representation (e.g., encryption) by translating from application to network format, and vice versa. The presentation layer works to transform data into the form that the application layer can accept. This layer formats and encrypts data to be sent across a network, providing freedom from compatibility problems. It is sometimes called the syntax layer.

This layer looks at things like JPEG, MPEG, MIDI, QUICKTIME and other files of the same nature. Most of your troubleshooting

Troubleshooting the Session layer - will be in conjunction with the applications that create them (at the application layer).

Application Layer

This layer deals with network services that interact with the user such as http, ftp, email, DNS etc. Problems related to browsers, ftp programs, email and network or internet programs can start here.

Troubleshooting the Application Layer – this layer can be the first layer to check with general computer type troubleshooting. An example would be a program that doesn't function properly or you get an error message when the program runs.

If you start at this layer for computer related troubleshooting, then you would work your way down the stack.
Some examples:

Unable to access a web page

- Can you view a web page in your browser?

Email issues

- Can you send or receive email

Unable to authenticate to an application

- Are the username and or password correct?

Website not showing up correctly

- Are your Internet Explorer connection settings correct?

Other Troubleshooting methods:

Get as much information as possible regarding the issue.

Source IP address of the device having issues
Destination IP address of the device in question
Type of application having issues?
Can you recreate the issue?
Is there a network diagram showing the data flows?
Is there a firewall installed on the source device?
Is there a firewall that the data flow has to traverse?
What has recently changed?