Troubleshooting

1.Find out what the problem is, which should have EAR(expected, actual, reproduction)

2.Triage the Situation, make sure the system is still in as good of a condition under the circumstances.

3.Examine as best you can inc. Logs.

4.Diagnose problem using elimination, successive refinement, bisection. You can use elimination, bisection, or successive refinement.

5/6. Test the changes and "Cure"(Deploy)

TCP/IP

4 Layer System:

Application Layer(Like Youtube)

TCP(Transport Layer): Wants to know how it's going to get to it's location(also makes it a packet at that point)

Internet (IP) Layer: Uses IP addresses to send the packets through a network interface to an IP(Which works like a house address)

Network Interface: This is how a packet get's transferred(Ie. Wifi, Ethernet, Etc.)

TCP Handshake: (A > B) SYN, (B > A) ACK-SYN, (A > B) SYN.

UDP is stateless and has minimal overhead, packet has addressing and data with dupe protection.

Vlans are their own networks, to traverse VLANs

Launching a Service

1. Define the ready list by Must have features, would be nice features, bugs and regressions, assertions and approvals.

2. Work on the ready list

3. Launch the beta with staging environments(dev-QA-UAT-beta-prod)

4.Launch production

5.Capture Lessons by making sure mistakes won't be repeated again.

Launching a Service (cont)

6.Repeat

Linux Commands

```
cat - concatenate files and
print the standard output
id - prints real and effective
user and group IDs
sudo- executes commands as a
root user
adduser - add a user or group to
the system
ctlr+l - clear terminal
ls - list the directory
touch - changes the file
timestamps
chown - change the file owner
and group
chgrp - change the group
ownership
chmod - change file access
permissions
apt-get/apt- get packages/-
install applications
tree- like ls but listing the
content of directories
ln- make a symbolic link between
files
dig- performs dns lookups
host - resolve a hostname into a
IP address
whois - tells you who owns the
ip/hostname
```

Containers and VM's

Networking

RING Topology

It is called ring topology because it forms a ring as each computer is connected to another computer, with the last one connected to the first. Exactly two neighbours for each device.

STAR Topology

In this type of topology all the computers are connected to a single hub through a cable. This hub is the central node and all others nodes are connected to the central node.

TREE Topology

It has a root node and all other nodes are connected to it forming a hierarchy. For example if in an office in one department ring topology is used and in another star topology is used, connecting these topologies will result in Hybrid Topology (ring topology and star topology).

Types:

IAS(Infrastructure as a Service), Vendor gives Infras, you do everything else on your own

PAS(Platform as a Service), run an application using framework or environment specific to a vendor

SAS(Software as a Service) basically Youtube/Github.

VM's:

HVM(Needs to pause VM to use hardware such as a sata controller)

PV(Paravirtualization) just sends the I/O calls directly to the VMM.

Containers:

Basically VM's but with no set resources.