AMAZON INTERVIEW ROUNDS

1) **Applied through** - Referral/Amazon Job portal/LinkedIn
2) **Position selected** - Cloud support engineer 1 (Deployment)
3) **Screen In exam** - Yes (Mostly on Linux , networking and behaviour) which will take max of 3 hours to complete
4) **Number of rounds** - 4 (2 technical, 2 managerial/technical based on amazon principles)
5) **Amazon portal for job apply** - https://account.amazon.jobs/en-US/login?relay=%2Fen-US
6) **Each Round Duration in average** - 90 to 120 minutes

Round 1 As you are a devops engineer they will test on the following
- CICD basics to advanced (CICD Flow, types of pipelines, master slave config)
- GIT integration with jenkins and service account (Webhook, Service account for each tool)
- GIT commands, and few technical questions on git (What is git push, pull, what will happen if MR fails)
- Kubernetes questions like architechture in your organization, cluster management etc.
- Ansible playbook script like httpd start stop
- Linux commands

Round 2 Purely on Networking
- What is DNS , HTTP and HTTPS protocol (53, 80, 443). SSH, NFS, SFTP , FTP (22, 2049, 22,21)
- How the packets are transferred from one point to other(Internet and intranet)
- WIFI enablement and device connectivity
- The ports for some requests
- Router and DHCP working
- Linux commands and networking commands related to linux. (firewall cmd, nmcli connection modify)
- Commands to open and block the ips in linux. (iptables -I INPUT -s xxx.xxx.xxx.xxx -j DROP)

Round 3 Managerial(Amazon principles) STAR [HERO]

- - Ownership (Saw your friend and colleague in problem and helped)
  - Deliver results (When you have not only met the goal but exceeded expectation)
  - Earn trust (Critical feedback from colleague and how you worked)

  Round 4 Managerial (Amazon principles)
  - Customer obsession (Difficult situation with customer and how you faced)
  - Bias for action (You worked on deadline and did not had option to take decison)
  - Learn to curious (Your team challenged you to think different and how you have tackled)

Important links to Prepare:

- **Networking :**
  - Make sure you have a big picture understanding of how networks work - https://www.youtube.com/watch?v=9BGWrLiT9qs
  - TCP/IP - 3-way handshake - https://support.microsoft.com/en-in/help/172983/explanation-of-the-three-way-handshake-via-tcpT-ip
  - HTTP - https://hpbn.co/
  - SSL/TLS - https://blog.cloudflare.com/keyless-ssl-the-nitty-gritty-technical-details/ ; https://www.ssl.com/article/ssl-tls-handshake-overview/
  - IP addressing: https://www.cisco.com/c/en/us/support/docs/ip/routing-information-protocol-rip/13788-3.html ; https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/dd379547(v=ws.10)
  - NAT: https://www.slashroot.in/linux-nat-network-address-translation-router-explained ; https://netfilter.org/documentation/HOWTO/NAT-HOWTO.html#toc6.1
  - SNAT: https://netfilter.org/documentation/HOWTO/NAT-HOWTO-6.html#ss6.2
  - DNS: https://www.verisign.com/en_US/website-presence/online/how-dns-works/index.xhtml ; https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/dd197427(v=ws.10)
  - DHCP: https://www.youtube.com/watch?v=0Dp7YoR0SLE ; https://vinitpandey.com/2015/06/17/dora-process-of-dhcp/
  - ARP: https://networklessons.com/cisco/ccnp-route/arp-address-resolution-protocol-explained
  - MSS/MTU: https://www.imperva.com/blog/mtu-mss-explained/
- **Devops:**
- **Docker:**
  - Containerizing applications: https://docs.docker.com/engine/examples/
  - Customizing docker daemon: https://docs.docker.com/config/daemon/
  - Logging drivers: https://docs.docker.com/config/containers/logging/configure/
  - Storage drivers: https://docs.docker.com/storage/storagedriver/select-storage-driver/
  - Networking modes: https://docs.docker.com/network/
  - Troubleshooting: https://docs.docker.com/toolbox/faqs/troubleshoot/
- **Container References:**
  - https://training.play-with-docker.com/
  - https://www.katacoda.com/
  - https://docker-curriculum.com/
- **Kubernetes:**
  - K8's architecture and components of master and worker nodes: https://www.tutorialspoint.com/kubernetes/kubernetes_architecture.htm ; https://x-team.com/blog/introduction-kubernetes-architecture/

    - Troubleshooting: https://kubernetes.io/docs/tasks/debug-application-cluster/troubleshooting/
- **CI/CD:**
    - Concept: https://medium.com/@nirespire/what-is-cicd-concepts-in-continuous-integration-and-deployment-4fe3f6625007
    - CICD Tools overview: https://www.digitalocean.com/community/tutorials/ci-cd-tools-comparison-jenkins-gitlab-ci-buildbot-drone-and-concourse
    - Troubeshooting: https://openedx.atlassian.net/wiki/spaces/TE/pages/102367626/Troubleshooting+Jenkins+Builds
    - Overview of Build and release: https://wiki.cdot.senecacollege.ca/wiki/Overview_of_the_Build_and_Release_Process
- **Configuration Management:**
    - Introduction to Ansible: https://www.tutorialspoint.com/ansible/ansible_introduction
    - Chef overview: https://docs.chef.io/chef_overview.html
    - Troubleshooting Cookbooks: https://docs.chef.io/errors.html
- **Infra as code:**
    - Terraform introduction: https://wiki.cdot.senecacollege.ca/wiki/Overview_of_the_Build_and_Release_Process