# Upskilling Plan Schedule

After the proposal is submitted on Friday 4th April 2025, and assuming it is accepted at the proposal presentation during the following week, the team will have approximately two weeks of uninterrupted time for upskilling. During that time, it is expected that team members self-study using the resources provided below and help each other learn where someone has more knowledge on a required subject than others.

General Resources:

Linux From Scratch: <https://www.linuxfromscratch.org/lfs/>

Build your own Linux distro to learn more about how Linux works.   
Tangentially related to the project, time-consuming, but might be worth your time.

Linux Journey: <https://linuxjourney.com/>

Beginner guides and tutorials for Linux. A good free resource with which to start.

Arch Wiki: <https://wiki.archlinux.org/>

The Arch Linux wiki houses a ton of useful Linux information.

A lot of the information isn’t Arch-specific and can easily be applied to other distros.

FreeCodeCamp: <https://www.freecodecamp.org/>

Contains tons of free tutorials from industry professionals. Videos and written guides on many subjects, including Linux and networking.

LinuxCommand.org: <https://linuxcommand.org/>

A guide to the Linux Command Line and shell scripts.

GNU/Linux Desktop Survival Guide: <https://www.togaware.com/linux/survivor/>

Tangentially related to the project but contains some good information if you want to use Linux as a regular operating system on your machine.

Linux as a router (Fedora): <https://fedoramagazine.org/use-fedora-server-create-router-gateway/>

A guide on configuring Fedora Server as a router gateway.

Windows Users:

WSL: <https://learn.microsoft.com/en-us/windows/wsl/install>

Virtual machine package that runs Linux on and integrates into Windows.

WSL terminals can then be run to use Linux packages like a native OS.

Can be installed from the Windows Store for ease, and multiple distros can be installed at once.

Mac Users:

Homebrew: <https://brew.sh/>

Linux-like package manager that runs in Terminal and allows CLI (“Command LIne”) packages such as iPerf or nmap to be installed easily.

Asahi Linux: <https://asahilinux.org/>

Dual-boot Linux support for Apple Silicon Macs. There are multiple distros to choose from, with the flagship being Fedora Asahi Remix, based on Fedora Linux.

Linux Distros:

Distros in bold are the focus operating systems for this project. However, it is worth researching their derivates and upstream versions where applicable, as this may give you a better understanding of these Linux distros.

**Fedora Linux**: <https://fedoraproject.org/>

Fedora is effectively the unstable version of Red Hat Enterprise Linux (which is owned by IBM). It has become Red Hat’s testbed and upstream distro in recent years. Works well for general users.

Uses the DNF package manager, which is considered slow but powerful.

Debian: <https://www.debian.org/>

A stable, point-release Linux distro often used for servers. Binaries and packages on Debian aren’t usually the latest versions as a trade-off for stability.   
Uses the APT (“Advanced Package Tool”) package manager, which is used by many other distros.  
Its versions are named after Toy Story characters.

Rocky Linux: <https://rockylinux.org/>

Based on Fedora and Red Hat Enterprise Linux; the spiritual successor to CentOS. Often used for servers and has long-term support.

**Ubuntu**: <https://ubuntu.com/>

Based on Debian Unstable (aka Debian Sid), Ubuntu is a popular distro for general use and great for beginners. It is also a commercialised distro – while it is free to use, Ubuntu’s parent company Canonical offers various upsells, especially for enterprise.

Red Hat Enterprise Linux: <https://www.redhat.com/en>

Also known as RHEL (pronounced “rall”, as in “rally”), Red Hat Enterprise Linux has been the de-facto enterprise server OS for many years.   
Red Hat was purchased by IBM, with both companies earning a controversial reputation over the years.   
Red Hat Enterprise Linux is typically a paid-only Linux distro.  
Despite all of this, you can make a free Red Hat account, earn industry-recognised Red Hat certifications, and even get a free individual subscription (i.e. not for commercial use) to use RHEL.

**Kali Linux**: <https://www.kali.org/>

Designed for penetration testing. Usually comes pre-installed with a lot of hacking tools, making it quick and easy to start penetration testing.

Based on Debian Testing, which is the stage between Debian Stable and Debian Unstable.

Tools:

Tools in bold are the focus of this project. However, D-ITG may cause problems (particularly on Fedora) because it is outdated by 8 years. Other tools are listed to give you points of comparison in terms of how networking tools work and when to use which tool.

**iPerf**: <https://iperf.fr/>

“iPerf3 is a tool for active measurements of the maximum achievable bandwidth on IP networks.”  
Could be a good choice for the primary tool.

**D-ITG**: <https://github.com/jbucar/ditg>

“D-ITG is a platform capable to produce traffic at packet level accurately replicating appropriate stochastic processes for both IDT (Inter Departure Time) and PS (Packet Size) random variables.”

Nmap: <https://nmap.org/>

“Nmap ("Network Mapper") is a [free and open source](https://nmap.org/npsl/) utility for network discovery and security auditing.”

qperf: <https://github.com/rbruenig/qperf>

“A performance measurement tool for [QUIC](https://quicwg.org/) similar to iPerf.”

Netperf: <https://hewlettpackard.github.io/netperf/>

“Netperf is a benchmark that can be used to measure the performance of many different types of networking. It provides tests for both unidirectional throughput, and end-to-end latency.”  
Could be a good choice for the primary tool.

sockperf: <https://github.com/Mellanox/sockperf>

“sockperf is a network benchmarking utility over socket API that was designed for testing performance (latency and throughput) of high-performance systems.”  
Could be a good choice for detailed latency and jitter analysis.

My Traceroute: <https://www.cloudflare.com/learning/network-layer/what-is-mtr/>

“My Traceroute, or MTR, combines traceroute and ping to measure a network path's health.”

Wireshark: <https://www.wireshark.org/>

“The world's most popular network protocol analyser.”