

**Project Proposal**

Network Performance Evaluation of Linux Based Operating Systems in a Physical Environment

Date: 02/04/2025

Version: 1.6

Client: Dr. Raymond Lutui

Prepared By:

Supervisor: Daniel Vaipulu

Project Manager: Thomas Robinson

System Architect: Win Phyo

System Architect: Nathan Quai Hoi

Network Engineer: Zafar Azad

Network Engineer: Larissa Goh

Network Engineer: Charmi Patel

Network Engineer: Kylie Afable

# Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Note |
| 13/03/2025 | 0.1 | Nathan | Initial document creation. |
| 15/03/2025 | 1.0 | Nathan | Initial versioning and header. |
| 17/03/2025 | 1.01 | Nathan, Zafar | Formatting, basic information for early sections, and some placeholders. |
| 18/03/2025 | 1.02 | Thomas | Update date and name. |
| 20/03/2025 | 1.03 | Zafar, Win | Formatting, start of methodology comparison, new team member, disclaimer, and start of references. |
| 24/03/2025 | 1.04 | Zafar | Addition of methodology information and references. |
| 25/03/2025 | 1.05 | Nathan | Formatting and addition of cost information. |
| 26/03/2025 | 1.1 | Thomas, Nathan, Win | Reformatting of document sections, terms of reference, and upskilling information. |
| 27/03/2025 | 1.2 | Nathan, Win, Zafar | Reformatting sections into paragraphs, documents added to appendix, additions and changes to methodology, and summarising some sections with reference to appendices. |
| 28/03/2025 | 1.3 | Nathan, Thomas | Formatting and placeholders for missing information. Version bumped to 1.2 with properly written changelog (should be 1.3 – fixed next day). |
| 30/03/2025 | 1.4 | All Team Members | Completion of draft proposal with revisions of all major sections. |
| 31/03/2025 | 1.5 | All Team Members | Improvement based off feedback, reformatting document and layout. |
| 02/04/2025 | 1.6 | All Team Members | Continued improvement based off feedback, and further reformatting. |

Table of Contents

[Changelog 2](#_Toc194547093)

[Executive Summary 4](#_Toc194547094)

[Terms of Reference 4](#_Toc194547095)

[Rationale 4](#_Toc194547096)

[Project Objectives 4](#_Toc194547097)

[Project Scope 5](#_Toc194547098)

[Out of Scope 5](#_Toc194547099)

[Key Stakeholders 5](#_Toc194547100)

[Technical Infrastructure 5](#_Toc194547101)

[Skills Analysis 5](#_Toc194547102)

[Upskilling Plan Schedule 6](#_Toc194547103)

[Deliverables 6](#_Toc194547104)

[Success Criteria 6](#_Toc194547105)

[Project Management Methodology 6](#_Toc194547106)

[Project Phases 7](#_Toc194547107)

[Deliverables 7](#_Toc194547108)

[Team Contract 7](#_Toc194547109)

[Team Schedule 8](#_Toc194547110)

[Risk Register 8](#_Toc194547111)

[Issue Log 9](#_Toc194547112)

[Project Plan 9](#_Toc194547113)

[Estimated Costs 9](#_Toc194547114)

[Labour Breakdown 10](#_Toc194547115)

[Project Feasibility 10](#_Toc194547116)

[Appendices 11](#_Toc194547117)

[Appendix A – Disclaimer 11](#_Toc194547118)

[Appendix B – Scope Statement 12](#_Toc194547119)

[Appendix C – Stakeholder Register 15](#_Toc194547120)

[Appendix D – Stakeholder Management Strategy 16](#_Toc194547121)

[Appendix E – Network Diagram 17](#_Toc194547122)

[Appendix F – Skills Analysis Matrix 18](#_Toc194547123)

[Appendix G – Upskilling Plan Schedule 19](#_Toc194547124)

[Appendix H – Team Contract 23](#_Toc194547125)

[Appendix I – Agenda and Meeting Minutes 25](#_Toc194547126)

[Appendix J – Risk Register 40](#_Toc194547127)

[Appendix K – Risk Management Plan 43](#_Toc194547128)

[Appendix L – Issue Log 44](#_Toc194547129)

[Appendix M – Milestone Report 45](#_Toc194547130)

[Appendix N – Work Breakdown Structure 47](#_Toc194547131)

[Appendix O – Gantt Chart 50](#_Toc194547132)

[Critical Path Analysis 53](#_Toc194547133)

[Appendix P – Estimated Cost Breakdown 54](#_Toc194547134)

[Appendix Q – Labour Breakdown Table 54](#_Toc194547135)

[References 55](#_Toc194547136)

# Executive Summary

This project evaluates the IPv4 and IPv6 network performance of three Linux-based software routers using iPerf/D-ITG. Performance evaluation, encompassing throughput, delay, jitter, and packet loss, will be conducted across 12 packet sizes on a network comprised of four computers.

The project, estimated at 300-360 hours, acknowledges potential risks such as Linux networking experience and hardware limitations, and includes a cost analysis for mentor support and necessary equipment.   
Some of the risks and constraints to consider include lack of team experience with Linux networking, possible hardware failure, and potential budget restraints.   
The total project cost is budgeted at NZD$125,721.31, which covers the hardware, labour, and tools needed for the project.

# Terms of Reference

Our client, Dr. Raymond Lutui, has requested an evaluation of the network performance of three Linux-based operating systems (Fedora, Ubuntu, and Kali Linux) configured as software routers, to determine which has the best performance and to compare the data with his previous evaluations. Despite IPv4's acknowledged scalability constraints and IPv6's intended role as a replacement (Suryaningrat et al., 2016), a lack of real-world performance data hinders network professionals from confidently strategising and executing the migration to IPv6 in their software-defined networks (Narayan et al., 2016).   
This project addresses the lack of real-world data with thorough network evaluation: measuring throughput, delay, jitter, and packet loss across 12 different packet sizes using iPerf and D-ITG, over both IPv4 and IPv6, in a four-PC network topology. The project is estimated to take 300-360 hours, commencing on April 4, 2025, with final deliverables due on October 31, 2025.

# Rationale

The transition to IPv6, while crucial for internet expansion and advanced features, introduces increased data overhead. This can create performance bottlenecks, particularly in resource-constrained software routers. To address this, we will conduct a controlled experiment measuring the performance differences between IPv4 and IPv6. Using TCP and UDP protocols across 12 packet sizes on four Linux systems configured as software routers. We will evaluate on 3 different Linux operating systems which include, Ubuntu, Fedora, and Kali. We will analyse throughput, delay, jitter, and packet loss. The resulting empirical data will provide valuable insights for network performance comparing IPv4 to IPv6.

# Project Objectives

The goal of this project is to evaluate the performance of 3 latest Linux based Operating Systems distributions – Fedora, Ubuntu, Kali Linux. These operating systems need to be configured as software routers before running the evaluations. Tools such as iPerf or D-ITG are to be employed to generate the evaluation traffics.

Objectives to be achieved include:

* Setup a network with 4 computers, 2 clients and 2 servers (servers configured as routers)
* Use TCP and UDP as transmission protocols.
  + Evaluate on IPv4
  + Evaluate on IPv6
* Performance evaluation:
  + Each evaluation should run a minimum of ten times.
    - Evaluate for throughputs.
    - Evaluate for any delays.
    - Evaluate for jitter.
    - Evaluate for any packet loss.
  + Any evaluation that falls outside the 95% confidence interval needs to be re-run
* Each evaluation will range from a minimum of 128 through to 1536 Bytes to assess performance under different conditions.
* Set up physical machines with Fedora, Ubuntu, and Kali Linux distributions configured as software routers.
* Conduct performance evaluations to evaluate network throughput, delay, jitters, and packet loss for TCP and UDP transmissions on IPv4 and IPv6 protocols.
* Follow the hybrid waterfall-scrum methodology for systematically evaluating phases and activities including requirement analysis, evaluation planning, evaluation execution and evaluation closer.

# Project Scope

This project encompasses the comprehensive performance evaluation of three Linux-based operating systems (Fedora, Ubuntu, and Kali Linux) configured as software routers. The scope includes configuring and evaluating all three operating systems across a four-PC network topology (one sender, two routers, one receiver), with evaluation of both TCP and UDP protocols over IPv4 and IPv6. Performance assessments will measure throughput, delays, jitter, and packet loss across 12 different packet sizes ranging from 128 to 1536 bytes. Each configuration will undergo 10 evaluation runs to ensure statistical validity, with additional runs conducted for any results falling outside the 95% confidence interval. The project includes detailed analysis and documentation of all findings, complete data logging, and delivery of comparative performance statistics across all evaluated operating systems. For more complete details on the scope statement, please refer to Appendix B.

## Out of Scope

This project focuses specifically on performance evaluation and does not include hardware procurement, initial operating system installation, nor physical environment setup.

The team will not be responsible for implementing performance improvements based on findings, providing ongoing maintenance, nor supporting the evaluation environment beyond the project duration.

Evaluation is limited to the specified operating systems, network topology, and protocols as outlined in the scope statement.

## Key Stakeholders

The project client is Raymond Lutui, with Daniel Vaipulu acting as supervisor.   
The project team consists of seven members: Kylie Afable, Zafar Azad, Larissa Goh, Nathan Quai Hoi, Charmi Patel, Win Phyo, and team lead Thomas Robinson.   
For a complete and detailed list of stakeholders, please refer to Appendix C, and for the Stakeholder Management Plan, please refer to Appendix D.

## Technical Infrastructure

A diagram of a router

AI-generated content may be incorrect.

The project would involve 4 computer hardware provided by the client, in which two will be configured as routers, while the other two will be used as sender and receiver. For a detailed representation of the technical infrastructure layout of the evaluation environment, please refer to Appendix E.

# Estimated Costs

The table below shows the estimated cost of the project.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cost Category | Description | Quantity | Unit Cost (NZD) | Total Cost (NZD) |
| Hardware | Supermicro 521R-T Mini Server SVRSPM34170 | 4 | $2,686.65 | $10,746.60 |
| Hardware | Philips 243V7QJAB/79 Monitor | 4 | $161 | $644.00 |
| Hardware | ASUS Vivobook Go 15 L510KA-EJ599W Laptop | 7 | $667 | $4,669.00 |
| Hardware | Rapoo X130PRO wired keyboard and mouse combo | 4 | $23.14 | $92.56 |
| Network Equipment | Cruxtec 0.5m Cat7 Ethernet Cable RS7-005-BK | 3 | $5.45 | $16.35 |
| Network Equipment | TP-Link TG-3468 32-bit Gigabit PCIe Network Adapter | 6 | $22 | $132.00 |
| Labour | Refer to labour breakdown table | - | - | $109,420.80 |
| Software Tools | iPerf | 4 | $0 | $0.00 |
| Software Tools | D-ITG | 4 | $0 | $0.00 |
| **Total Estimated Cost** |  |  |  | **$125,721.31** |

(PB Technologies, 2025).

# Labour Breakdown

We have included a full breakdown of labour costs for this project in Appendix F. It covers each role needed in the project, including a mentor, project manager, network engineers, and system architects, along with how many hours they will contribute and the associated costs. Based on current industry rates (sourced from PayScale, 2025), the total cost for labour sums up to **NZD$ 109,420.80.**

## Skills Analysis

All team members have existing Linux and networking experience but will need to upskill in specific networking tools such as iPerf and D-ITG.   
Some team members will need to upskill in certain Linux skill areas such as BASH scripting. Therefore, an upskilling plan schedule has been created.  
For a complete skills analysis matrix, please refer to Appendix F.

## Upskilling Plan Schedule

**Need just table for the Schedule – TODO: Thomas**

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. For the complete upskilling plan schedule, please see Appendix G.

## Deliverables

The project will deliver a comprehensive performance evaluation package including complete evaluation logs, statistical analysis of performance metrics, comparative analysis across all evaluated operating systems, and a final report detailing findings and methodology.   
All raw data and configuration documentation will be provided to enable verification and potential future analysis.

## Success Criteria

The project will be considered successful upon delivery of valid performance data for all three operating systems, using both TCP and UDP protocols over IPv4 and IPv6, with all evaluation results falling within the 95% confidence interval.   
Success includes comprehensive documentation of methodology, complete raw data logs, and a final analysis comparing performance across all evaluated configurations.

# Project Management Methodology

|  |  |  |  |
| --- | --- | --- | --- |
| **Methodology** | **Core Approach** | **Pros** | **Cons** |
| Waterfall | A fixed set of phases, where each phase must be completed before moving to the next phase. (Atlassian, n.d.) | Waterfall outlines a clear project structure; this demonstrates clear cost and goals. Due to the fixed nature of the methodology, tracking is linear and easier. The upfront planning approach also minimises risk factors as most of them are accounted for in the initial planning phase. Waterfall approach also expects fewer delays that can occur from additional requirements. (Atlassian, n.d.) | Due to the fixed nature of the methodology, its benefit can also be a drawback depending on the nature of the project. Projects that require continual interactions with end users and the team to review current direction and course correction do not fit in the waterfall approach. Its fixed approach ultimately limits flexibility. (Atlassian, n.d.) |
| Scrum | Working quickly and collaboratively while promoting an environment that allows changes during the development cycle. (Asana, 2025) | The scrum framework is the suggested approach for software development projects that prioritises customer needs constantly changing throughout the development lifecycle.  The phases of this approach are broken down into sprints. Once a sprint is completed, review and feedback commence to improve the efficiency of the next sprint. (Asana, 2025) | Due to the agile approach being constantly changed and reviewed after each sprint, limitation on resource planning can occur due to the unclear end objective.  In most cases, the dev team can only have sight on a few sprints ahead.  Agile is also difficult to measure due to its nature of change which can also result in scope creep. (Asana, 2025) |
| Software testing lifecycle | The core approach to (stlc) showcases the sequences of phases that occur during the testing process of software. |  |  |

The chosen methodology uses a hybrid Waterfall-Scrum approach because our project requires both structured documentation and iterative development. Waterfall is well-suited for the planning and execution phases, where we need clear documentation, risk management, and defined deliverables. This ensures that our project remains well-organised and aligned with stakeholder expectations. However, since our implementation and evaluation phases involve setting up and optimising multiple operating systems, Scrum allows us to work in sprints: configuring, evaluating, and refining each OS before moving to the next. Our phases are based on the Software Testing Life Cycle (STLC). These include the 6 key phases which our management methodology is applied to, the phase break down are as below. (Katalon, 2025).

## Project Phases

1. **Requirement Analysis Phase**

* Project initiation, project scope, objectives, and deliverables
* Team contract, project charter
* Work Breakdown Structure (WBS)

1. **Evaluation Planning**

* Define evaluation data, scenarios and environment
* Resource planning
* Training requirements.

1. **Evaluation Case Development**

* Approved evaluation plan.
* Defined evaluation case scenarios.
* Validate with client.

1. **Evaluation Environment Setup**

* Establish Hardware and Software
* Prepare Evaluation Data
* Configure Evaluation Environment

1. **Evaluation Execution phase**

* All evaluations are performed, and results are documented
* Implement evaluation case failure protocol
* Analyse Performance Metrics to determine Operating System Performance

1. **Evaluation Close**

* Prepare an excel spreadsheet of all the data for each operating system.
* Document Closure Report
* Client Feedback

## Deliverables

For project-related documents, please refer to the scope statement in Appendix B.

The product-related deliverables include:

* Results of the evaluation for each Linux OS in a spreadsheet.
* Logs for the evaluation
* Results comparison document

# Team Contract

This section covers all the team members and the assigned roles. For complete details on team contract, please refer to Appendix H.

|  |  |  |
| --- | --- | --- |
| Member Name | Team Role | Assigned IP Version |
| Thomas Robinson | Project Manager | IPv4 |
| Win Phyo | System Architect | IPv4 |
| Zafar Afrad | Network Engineer | IPv4 |
| Kylie Afable | Network Engineer | IPv6 |
| Larissa Goh | Network Engineer | IPv6 |
| Nathan Quai Hoi | System Architect | IPv6 |
| Charmi Patel | Network Engineer | IPv6 |

Each Team members are responsible for allocating 12 – 15 hours per week as weekly commitment for the project. All team members are to complete the assigned tasks on timely manner, while constantly communicating with the team while adhering to the code of conduct and ethic.

# Team Schedule

The table below shows the frequency along with the time, date, location for our meetings.

A screenshot of a computer screen

AI-generated content may be incorrect.

All the meeting minutes and agendas are attached in Appendix I.

# Risk Register

Risks are identified during planning and reviewed weekly during team meetings. Team members raise or are assigned risks, assess their causes and potential impact of each, and define suitable mitigation strategies. If a risk occurs, it’s escalated to the Issue Log where it’s tracked separately and monitored until it’s closed. Refer to Appendix J & K for the complete Risk Register and Risk Management Plan.

Below are two scope-related risks that have the potential to impact the project if not managed.

|  |  |  |
| --- | --- | --- |
| Risk | Description | Mitigation |
| **Tool replacement due to compatibility issues** | One of the client’s chosen tools D-ITG, has shown compatibility issues as it is not available in the Fedora package repositories and cannot be compiled from source. This creates a risk of inconsistent data as Fedora will lack D-ITG metrics that will be captured on Ubuntu and Kali. | The team will conduct additional evaluations with D-ITG on Fedora to determine feasibility. If it remains non-functional, iPerf will be used to maintain consistency across all operating systems. This will be documented and communicated to stakeholders to stay within the scope. |
| **Incorrect router configuration affecting evaluation results** | The project requires the configuration of Ubuntu, Fedora, and Kali as routers. Since routing directly affects packet flow, any error from misconfiguring routing tables or IP forwarding can compromise result accuracy. | The team will follow best practices for Linux router setups and validate configurations during the analysis phase. A checklist will be used for consistency, and a peer review process will ensure each router is configured correctly. |

# Issue Log

Issues are logged when risks manifest or when unexpected problems arise during the project. Each issue is assigned to a team member for resolution and tracked until resolved. The issue log is also reviewed during weekly meetings, and priority is escalated where needed to prevent impact on project delivery. Refer to Appendix L For the complete Issue Log.

Project Plan

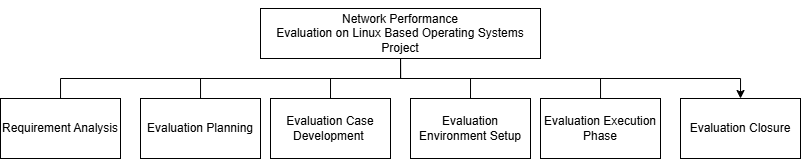
**Milestones**

This section includes the main milestones of the project, as well as internal milestones to help track the progress of the project, and meeting deadlines. The following table includes the important external milestones. For the complete Milestones Report, please refer to Appendix M.

|  |  |  |
| --- | --- | --- |
| **Date** | **Milestone Type** | **Milestone** |
| 04/04/2025 | External | Project Proposal Submission |
| 06/06/2025 | External | Mid-Term Review |
| 17/08/2025 | Internal | Completion of Ubuntu Evaluation |
| 14/09/2025 | Internal | Completion of Fedora Evaluation |
| 12/10/2025 | Internal | Completion of Kali Evaluation |
| 31/10/2025 | External | Final Poster |
| 31/10/2025 | Internal | Portfolio |

**WBS, Tasking, Scheduling**

This section displays a high-level view of the Work Breakdown Structure.



For the complete and detailed version of the WBS and Gantt Chart, please refer to the Appendix N and O, respectively.

# Project Feasibility

**Required Infrastructure:**

Equipment:

We will need 4 computers where 2 will act as routers while the other 2 will be the sender and receiver. Each router should have network cards to configure IPv4 and IPv6.

**Tools and Technology:**

Software:

Three Linux-based operating systems will be used, configured as routers. These operating systems are commonly used for networking tasks. We will be using Ubuntu, Fedora, and Kali Linux.

Network Performance Evaluation tools:

The tool ‘iPerf’ will be used to measure network performance. IPv4 and IPv6 will be evaluated using TCP and UDP transmission protocols. 12 variations of packet sizes will be used: 128, 256, 384, 512, 640, 768, 896, 1024, 1152, 1280, 1408, and 1536 Bytes.

**Skills That Are Required:**

* Familiarise themselves with Linux, software routing, and performance evaluation.
* Understanding IPv4 and IPv6 network principles.
* Adequate knowledge in programming languages.
* Problem solving with debugging and troubleshooting.
* Understanding how to configure routers and is comprehends the results from the network performance tools.

**Roles:**

*Project Manager:*

Looks after the project and ensures everyone and the project are all running smoothly.

*System Architect:*

Creates the general framework and ensures that it aligns with the projects main goal.

*Development Team Member:*

Builds, configures and maintains the network.

**Rationale/Justification:**

Our project aims to evaluate the performance of IPv4 and IPv6 protocols in different Linux based operating systems. This is done by using the routers for network configuration. With these tools, technologies, and required infrastructure, it helps us understand how both these protocols perform in different networking environments. This is important as IPv6 is on the rise.

Please refer to Project Purpose and Rationale.

# Appendices

## Appendix A – Disclaimer

Auckland University of Technology

Bachelor of Computer & Information Sciences

Research & Development Project

Disclaimer: Network Performance Evaluation on Linux Based Operating Systems

Clients should note the general basis upon which the Auckland University of Technology undertakes its student projects on behalf of external sponsors:

While all due care and diligence will be expected to be taken by the students, (acting in software development, research or other IT professional capacities), and the Auckland University of Technology, and student efforts will be supervised by experienced AUT lecturers, it must be recognised that these projects are undertaken in the course of student instruction. There is therefore no guarantee that students will succeed in their efforts.

This inherently means that the client assumes a degree of risk. This is part of an arrangement, which is intended to be of mutual benefit. On completion of the project, it is hoped that the client will receive a professionally documented and soundly constructed working software application, some part thereof, or other appropriate set of IT artefacts, while the students are exposed to live external environments and problems, in a realistic project and customer context.

In consequence of the above, the students, acting in their assigned professional capacities and the Auckland University of Technology, disclaim responsibility and offer no warranty in respect of the “technology solution” or services delivered, (e.g. a “software application” and its associated documentation), both in relation to their use and results from their use.

## Appendix B – Scope Statement

|  |
| --- |
| **Project Title: Network Performance Evaluation on Linux Based Operating Systems**  **Date: 30/03/2025** **Prepared by:** Win Phyo & Thomas Robinson |
| **Project Justification**  This project is designed to evaluate the network performance of IPv4 and IPv6 on the latest versions of three popular Linux-based operating systems, so that the client may compare the data with historical data to observe changes in performance over time and between operating systems.  Both IPv4 and IPv6 have their own benefits and drawbacks, of which this project intends to evaluate.  With this data, the client can have a better understanding and comparison of how each of the chosen operating systems handles network traffic, which ultimately helps to make adequate decisions on optimising network infrastructure. |
| **Product Characteristics and Requirements**  **Functional Requirements:**   1. Configure Fedora, Ubuntu, and Kali Linux as software routers 2. Implement both TCP and UDP transmission protocols 3. Support both IPv4 and IPv6 addressing 4. Set up a four-PC network topology (1 sender, 2 routers, 1 receiver) 5. Configure dual NICs on router PCs and single NICs on sender/receiver PCs 6. Perform three complete evaluation rounds, one for each operating system 7. Generate network traffic using iPerf or D-ITG 8. Evaluate with 12 specified packet sizes (128 to 1536 Bytes) 9. Run each evaluation configuration at least 10 times 10. Identify and re-run evaluations falling outside 95% confidence interval 11. Record throughput performance across all evaluation scenarios 12. All evaluations must be run 10 times each with all evaluations at 95% confidence 13. Record throughput performance across all evaluation scenarios 14. Measure network delays in all evaluation configurations 15. Calculate jitter values throughout evaluations 16. Track packet loss rates for all evaluation cases 17. Log all evaluations 18. Generate statistical averages using the evaluation tool 19. Compile final statistical data for comparison and analysis   **Non-Functional Requirements:**   1. Results must fall within a 95% confidence interval 2. Statistical validation of evaluation results 3. Consistent evaluation methodology across all platforms and evaluation rounds 4. Evaluation environment must support the full range of packet sizes 5. Network infrastructure must handle required throughput levels 6. Measurement tools must be precise enough to capture microsecond-level jitter 7. Evaluation environment must maintain consistent conditions across evaluation rounds 8. Hardware configurations must remain stable throughout all evaluations 9. Evaluation tools must produce repeatable results 10. Complete logs of all evaluation runs must be preserved 11. Detailed recording of evaluation configurations and parameters 12. Final deliverable must include both raw logs and statistical summaries 13. Evaluation must be organised into three distinct rounds/sprints (one per OS) 14. Consistent evaluation procedures must be followed across all sprints 15. Comparison data must allow for direct analysis between operating systems   **Out of Scope:**   1. Acquisition of the four PCs required for evaluation 2. Physical installation and configuration of hardware 3. Initial operating system installations 4. Network facility and power supply 5. Physical security of the evaluation environment 6. Environmental controls of the evaluation facility 7. Long-term maintenance of the evaluation environment 8. Ongoing support for the configured systems 9. Application of findings to production networks 10. Performance optimisation recommendations beyond evaluation results 11. Evaluation of operating systems beyond the specified three distributions 12. Network configurations other than the specified topology 13. Evaluation of protocols beyond TCP and UDP 14. Performance evaluation at speeds exceeding gigabit Ethernet 15. Addressing performance issues identified during evaluation 16. Implementing changes to improve network performance 17. Troubleshooting underlying hardware issues |

|  |
| --- |
| **Summary of Project Deliverables**  **Project management-related deliverables:**   * Meeting Agendas * Meeting Minutes * Project Charter * Team Contract * Stakeholder Register * Stakeholder Management Strategy * Scope Statement * Project Proposal * Risk Register * Issue Log * Communication Management Plan * Work Breakdown Structure (WBS) * Gantt Chart * Critical Path Analysis * Skills Analysis * Upskilling Plan Schedule * Estimated Cost Breakdown * Milestone Report * Lesson-Learned Report * Mid-Term Review * Final Product Presentation   **Product-related deliverables:**   * Complete evaluation plans detailing procedures for all configurations * Configuration documentation for router setup on each operating system * Scripts or configuration files used for evaluation * Spreadsheet of performance evaluation results * Raw evaluation logs from all execution runs (10 runs per configuration) * Data analysis documentation with statistical validation |
| **Project Success Criteria:**   * Delivery of all specified project deliverables * Statistical validity of all evaluation results * Clear presentation of comparative performance across operating systems * Adherence to project timeline and budget constraints * Comprehensive documentation allowing for evaluation reproducibility |

## Appendix C – Stakeholder Register

**Stakeholder Register for** **Network Performance Evaluation on Linux Based Operating Systems - Physical Environment**

**Version: 1.0**

**Prepared by:** Nathan Quai Hoi **Date: 11/03/2025**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Name** | **Position** | **Internal /External** | **Project Role** | **Contact Information** |
| **Nathan Quai Hoi** | Team Member | Internal | System Architect | wgk6332@autuni.ac.nz |
| **Win Phyo** | Team Member | Internal | System Architect | ddk8093@autuni.ac.nz |
| **Zafar Azad** | Team Member | Internal | Network Engineer | ftk8708@autuni.ac.nz |
| **Thomas Robinson** | Team Leader | Internal | Project Manager | cgr2690@autuni.ac.nz |
| **Larissa Goh** | Team Member | Internal | Network Engineer | xhm5236@autuni.ac.nz |
| **Charmi Patel** | Team Member | Internal | Network Engineer | fhv9735@autuni.ac.nz |
| **Kylie Afable** | Team Member | Internal | Network Engineer | cjq7738@autuni.ac.nz |
| **Daniel Vaipulu** | Project Mentor | Internal | Project Mentor | daniel.vaipulu@aut.ac.nz |
| **Raymond Lutui** | Client | Internal | Client | raymond.lutui@aut.ac.nz |
| **Ramon Lewis** | IT Support | Internal | IT Support | ramon.lewis@aut.ac.nz |
| **Dr. Ramesh Lal** | Lecturer | Internal | Lecturer | ramesh.lal@aut.ac.nz |
| **Prof. Catherine Shi** | Lecturer | Internal | Lecturer | catherine.shi@aut.ac.nz |
| **Olivia Tang** | Lecturer | Internal | Lecturer | olivia.tang@aut.ac.nz |

## Appendix D – Stakeholder Management Strategy

**Stakeholder Management Strategy for Network Performance Evaluation on Linux Based Operating Systems**

**Prepared by: Nathan Quai Hoi** **Date: 19/03/2025**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Level of Interest** | **Level of Influence** | **Potential Management Strategies** |
| **Daniel Vaipulu** | High | High | Daniel is very resourceful and relaxed. He makes sure to engage the project at a reasonable pace. We will need to keep him updated with what is happening regularly. |
| **Raymond Lutui** | High | High | Raymond is very relaxed, while also having high expectations. We will need to keep him updated with what is happening. |
| **Thomas Robinson** | High | Medium | Thomas would like to ensure all team members are on the same page and keep track of the project tasks. He develops plans and allocates tasks to achieve goals. |
| **Ramon Lewis** | Low | High | Ramon would like to know details of our setup for evaluation and would like to communicate via Teams chat. |

## Appendix E – Network Diagram

A diagram of a router

AI-generated content may be incorrect.

The figure above shows the network diagram representing the technical infrastructure of the evaluation environment. Two computers will have the same Linux OS installed and configured as routers. A third computer will be the sender, and the fourth will be the receiver. There will be three different networks in total, and each computer will have its own static IP address.

**Note:** Router 1 and Router 2 computers will each have two Network Interface Cards.

## Appendix F – Skills Analysis Matrix

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Skills Matrix** | | | | | | | | | | |
| **Team Members** | **Linux Installation and Configuration** | **Linux as Router Configuration** | **BASH Scripting** | **Subnetting IPv4** | **Subnetting IPv6** | **Understanding of Networking Protocols (TCP/UDP)** | **iPerf Usage** | **D-ITG Usage** | **Basic Network Performance Metrics (Throughput, Delay, Jitter, Packet Loss)** | **Basic Data Analysis Skills** |
| **Kylie Afable** | **2** | **0** | **2** | **1** | **1** | **1** | **0** | **0** | **0** | **0** |
| **Zafar Azad** | **2** | **0** | **0** | **3** | **1** | **3** | **0** | **0** | **2** | **1** |
| **Larissa Goh** | **2** | **0** | **0** | **1** | **0** | **1** | **0** | **0** | **1** | **1** |
| **Nathan Quai Hoi** | **2** | **0** | **1** | **2** | **1** | **1** | **0** | **0** | **0** | **0** |
| **Charmi Patel** | **2** | **0** | **0** | **1** | **0** | **1** | **0** | **0** | **0** | **1** |
| **Win Phyo** | **2** | **0** | **0** | **2** | **0** | **1** | **0** | **0** | **1** | **1** |
| **Thomas Robinson** | **3** | **0** | **2** | **2** | **1** | **2** | **0** | **0** | **1** | **2** |
| **Total** | **15** | **0** | **5** | **12** | **4** | **10** | **0** | **0** | **5** | **6** |
| **Average** | **2.14** | **0.00** | **0.71** | **1.71** | **0.57** | **1.43** | **0.00** | **0.00** | **0.71** | **0.86** |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  | **Key** | | | |  |  |  |  |
|  |  |  | **0** | **1** | **2** | **3** |  |  |  |  |
|  |  |  | Unskilled | Low-Skilled | Competent | Expert |  |  |  |  |

## Appendix G – 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.”

## Appendix H – Team Contract

**Project Name: Network Performance Evaluation on Linux Based Operating Systems**

**Project Team Member Names and Sign-off:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Date** | **Sign-off on Team Contract** |
| **Nathan Quai Hoi** | **18/03/2025** |  |
| **Win Phyo** | **18/03/2025** |  |
| **Zafar Azad** | **18/03/2025** | **ZA** |
| **Thomas Robinson** | **18/03/2025** |  |
| **Larissa Goh** | **18/03/2025** |  |
| **Charmi Patel** | **18/03/2025** |  |
| **Kylie Afable** | **20/03/2025** | **KA** |

**Code of Conduct:** As a project team, we will:

* Work proactively, anticipating potential problems and working to prevent them
* Keep other team members informed of information related to the project
* Focus on what is best for the whole project team
* See the team project through to completion

**Code of Ethics:**

* Maintain integrity, transparency, and accountability in all actions.
* Respect and value diversity, fostering an inclusive team environment.
* Commit to excellence and collaborative efforts for project success.

**Participation:**

* Be honest and open during all project activities
* Encourage diversity in teamwork
* Provide the opportunity for equal participation
* Be open to new approaches and consider new ideas
* Have one discussion at a time
* Let the team know well in advance if a team member must miss a meeting or may have trouble meeting a deadline for a given task

**Communication:**

* Decide as a team on the best way to communicate various information
* Focus on solving problems, not blaming people
* Present ideas clearly and concisely
* Meet and communicate frequently to discuss project progress
* Arrange additional meetings as needed
* Keep discussions on track
* Honor meeting timeframes
* Read communications (emails, meeting minutes, action items, etc.) from each other
* Respond to each other in a timely manner (i.e., within 24 hours)

**Problem Solving:**

* Encourage everyone to participate in solving problems
* Only use constructive criticism
* Hold each other accountable for meeting the standards
* Consequences for behaviours impacting Team performance

**Meeting Guidelines:**

* It is mandatory to attend meetings every Thursday with the mentor/ client.
* Any additional meeting date and time must be scheduled in advance.
* Create and distribute an agenda before each meeting.
* Set expectations for meeting schedules and respect allocated time for each agenda item.
* Assign responsibilities for documenting meeting minutes, tracking action items, and following up on tasks between meetings.
* Encourage feedback on meeting processes and suggest periodic reviews to assess the effectiveness of meetings and adjust guidelines as needed.

**Consequences for behaviour impacting Team performance:**

* Tasks will not be accomplished on time due to constant absence.
* Poor communication among team members can impact work efficiency.
* Project will not meet the deadline if individual tasks are not completed on time.
* Teams cannot work effectively if team members do not respect other team members or constantly have conflicts.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Appendix J – Risk Register | | | | | | | | | | | | | | | | | | | | |
| **Prepared by: Larissa Goh** | | | | |  |  | | **Date: 19/03** | |  | | |  |  | |  | | |  |
| **No.** | **Rank** | | **Risk** | | **Description** | **Category** | | **Root Cause** | | **Triggers** | | | **Mitigating Action** | **Risk Owner** | | **Probability** | | | **Impact** |
| 1 | Medium | | Long distance between members | | The huge distances between majority of the team may create challenges when organising meetups. | Logistical | | Members located in different regions | | Meetup planning, travel expenses, travel time, unreliable transport | | | Advanced weekly schedules in place, online meetings scheduled where needed | Project Manager | | Medium | | | High |
| 2 | Medium | | Varying levels of technical knowledge among members | | All members have varying levels of technical skill required for this project. This disparity may create workload imbalances and affect team cohesion. | Training & Knowledge | | Different backgrounds and technical experience | | Imbalanced task completion. Reliance on members | | | Ongoing commitment to upskilling. Creating a skill matrix and detailed upskilling plan. | Team | | High | | | High |
| 3 | High | | Unsuitable operating system | | Selecting an unsuitable operating system for our project. This may lead to major issues during our evaluation phase, cause compatibility issues, or have performance limitations. | Technical | | Lack of research | | software and hardware incompatibility | | | In depth research and assessing project requirements. Comparing OS options | System Architect | | Medium | | | High |
| 4 | High | Group member quitting | | Group member potentially quitting during any stage of the project. | | | Team Members | | Personal reasons | | Personal reasons | Urgent meeting to discuss changed task delegations and effect on project timeline. | | | Project Manager | | Medium | High | | |
| 5 | Low | Poor communication | | Miscommunication or lack of clarity among group members | | | Team Members | | Missed updates, unclear expectations, lack of structure | | Low morale, unclear leadership, confusing task interpretation, failing to include others or themselves | Create a team contract, encourage communication, weekly high priority meetings that requires all members to share ideas and speak. | | | Project Manager | | Low | High | | |
| 6 | Medium | File loss & Data corruption | | Important data is lost or corrupted due to hardware issues or human error. | | | Data | | Poor backup strategy, carelessness. | | System crashes, storage issues, overwriting files | Implement regular backup on cloud or local drives. Use version control such as Git. | | | Network Engineer | | Low | Medium | | |
| 7 | Medium | Sickness | | Team member, mentor, and client may be unavailable due to sickness, especially going into colder months. | | | Health & Safety | | COVID-19, infection | | COVID and sickness increasing. | Enable remote collaboration is available when needed e.g. Discord, Teams. Flexibility to reschedule to suit client when needed. | | | Project Manager | | High | Medium | | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 8 | High | D-ITG Compatibility Issues | D-ITG may not function correctly or be supported on all selected operating systems. | Technical | Software incompatibility | Errors during installation | Early evaluation on each operating system | Network Engineer | High | Medium |
| 9 | High | Hardware Issues | Failure of critical hardware such as computers and network cards can delay evaluation. | Technical | Improper set up or handling, physical damage, firmware issues, | Loss of connectivity, overheating, inability to boot operating system | Perform early hardware evaluation, document all hardware used for quick replacement, spare equipment, in contact with IT technician | System Architect | Medium | High |
| 10 | Low | Incomplete evaluation due to time constraints | Large number of evaluation combinations may cause the team to run out of time. | Schedule | Time-consuming evaluation and set up | Delay in router configuration, tool issues | Ensure all tools and configurations are fully evaluated during analysis phase and divide responsibilities. | Project Manager | Low | High |
| 11 | Medium | Inconsistent evaluation results | Differing results between operating systems | Technical | Misconfigured routers, hardware issues | Differing results for identical evaluations | Standardise configuration steps, peer review setup. | System Architect | Medium | High |
| 12 | Low | Misinterpretation of data | Team may incorrectly analyse performance data due to unfamiliarity with tools, software, or hardware | Technical | Lack of network experience | Unexpected output formats, inconsistent logs | Research tool output structures, consult mentor if unclear, align on analysis methods as a team | Network Engineer | Low | Medium |

## Appendix K – Risk Management Plan

For our hybrid methodology combining Agile and Waterfall, risk and issue management are handled with both structured planning and continuous review. Risks are identified and documented during the planning phase through past experiences, client input, and technical research.

This involves reviewing:

* The project scope
* Objectives
* Timeline
* Tools
* Costs
* Team structure

However, as conditions change throughout a project, especially during the analysis and development phases, the team must be ready to adapt quicky. To manage this, risks will be monitored regularly and escalated to active issues if they occur.

* + - 1. Identify risk
      2. Access risk
         1. Likelihood
         2. Impact
      3. Record in risk register
      4. Assign owner and plan mitigation
      5. Monitor during weekly meetings

**If a risk has occurred…**

* + - 1. Move to issue log
      2. Assign owner and track resolution
      3. Resolve or escalate as needed
      4. Continue monitoring risks

The team acknowledges that not all problems can be predicted. Unexpected problems that arise will be added directly into the Issue Log. This ensures both anticipated and unexpected challenges are managed effectively.

## Appendix L – Issue Log

**Prepared by: Larissa** **Date: 19/03/2025**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Issue #** | **Issue Description** | **Impact on Project** | **Date Reported** | **Reported By** | **Assigned To** | **Priority M/H/L** | **Due Date** | **Status** | **Comments** |
| 01 | Delayed meeting with client and mentor | Delayed start on entirety of project as we need more information and details from our client. | 17/03/25 | Zafar | Group | H | NA | Closed | First meeting with client has been pushed to 20/03/25 |
| 02 | D-ITG isn’t available in Fedora repos; it won’t be compiled from source | If the source code can’t be fixed, another tool will have to be chosen for evaluation, despite the client’s request. | 25/03/25 | Thomas | Thomas | H | 27/03/25 | Closed | Client isn’t aware of this issue yet – it needs to be raised at 27/03 meeting |
| 03 | Cancelled team meeting | Due to multiple members facing unforeseen transport issues, our team meeting before our meeting with client and mentor had to be cancelled | 27/03/25 | Larissa | Thomas | H | 29/03/25 | Closed | Quick overview over Discord and in-depth debrief after to ensure everyone is on the same page |
| 04 | Management methodology didn't align with our project | Team roles and documentation process needed to be reviewed to indicate which protocol we will be following | 27/03/35 | Zafar | Group | H | 29/03/25 | Closed | A hybrid model approach has been selected to better suit the project |

## Appendix M – Milestone Report

**Author: Larissa Goh**

**Date: 04/04/2025**

**Version: 1.0**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone** | **Date** | **Status** | **Responsible** | **Issues/Comments** |
| Team Portfolio Structure (folder layout) | 19/03/25 | Completed | Thomas | Done on time |
| Project Schedule | 25/03/25 | Completed | Win | Done on time |
| Stakeholder Register V1 | 19/03/25 | Completed | Nathan | Done on time |
| Team Contract | 20/03/25 | Completed | Win | Done on time |
| Upskilling Recommendation List V1 | 20/03/25 | Completed | Thomas | Done on time |
| Skills Analysis | 20/03/25 | Completed | Thomas | Done on time |
| Stakeholder Management Strategy V1 | 19/03/2025 | Completed | Nathan | Done on time |
| Stakeholder Register V1 | 20/03/25 | Completed | Nathan | Done on time |
| Communication Plan V1 | 20/03/25 | Completed | Charmi | Done on time |
| Gantt Chart V1 | 20/03/25 | Completed | Nathan | Done on time |
| Milestone Report V1 | 02/04/25 | Completed | Larissa | Done on time |
| Work Breakdown Structure V1 | 20/03/25 | Completed | Win | Done on time |
| Network Diagram | 31/03/25 | Completed | Nathan | Done on time |
| Critical Path Analysis | 31/03/25 | Completed |  | Done on time |
| Issue Log V1 | 23/03/25 | Completed | Larissa | Done on time |
| Risk Register V1 | 23/03/25 | Completed | Larissa | Done on time |
| Meeting agenda for kick-off meeting | 19/03/25 | Completed | Zafar | Done on time |
| Meeting agenda for meeting #2 | 25/03/25 | Completed | Thomas | Done on time |
| Meeting agenda for meeting #3 | 30/03/25 | Completed | Thomas | Done on time |
| Meeting minutes for kick-off meeting | 19/03/25 | Completed | Win | Done on time |
| Meeting minutes for meeting #2 | 27/03/25 | Completed | Win | Done on time |
| Meeting minutes for meeting #3 | 03/04/25 | Completed | Win | Done on time |
| Estimated Costs Breakdown | 31/03/25 | Completed | Charmi | Done on time |
| Project Proposal Submission | 04/04/25 | Completed | Thomas | Done on time |
| Project Proposal Presentation | 10/04/25 | Pending | Team |  |

## Appendix N – Work Breakdown Structure

**Prepared By:** Win Phyo & Thomas Robinson

**Date:** 02/04/2025

**Project Name:** Linux Network Performance Evaluation

1. Requirement Analysis (03/03/2025 – 04/04/2025)
   1. Entry Criteria
      1. Team Kick-Off Meeting Agenda
      2. Team-Client Kick-Off Meeting Agenda
      3. Team Contract
      4. Meeting Minutes (Kick-Off Meetings)
   2. Develop Project Proposal
      1. Stakeholder Register
      2. Project Charter
      3. Client Contract
      4. Meeting Minutes (Team & Client)
   3. Develop Project Plans
      1. Stakeholder Management Strategy
      2. Risk Register
      3. Issue Log
      4. Communications Management Plan
      5. Communications Management Plan
      6. Scope Statement
      7. Work Breakdown Structure
      8. Project Schedule
         1. Schedule baseline showing originally planned activities with their durations and milestones
         2. Network Diagram
         3. Critical Path Analysis
      9. Upskilling Plan
      10. Estimated Cost
      11. Milestone Report
      12. Project Proposal submission
      13. Project Proposal presentation

**Milestone 1 – Project Proposal**

* 1. Exit Criteria
     1. Client feedback on proposal document
     2. Approval of proposal document from client

1. Evaluation Planning (07/04/2025 – 06/06/2025)
   1. Entry Criteria
      1. Client approval of project proposal
   2. Researching And Upskilling
      1. Team members upskill for necessary skills
      2. Network tools (iPerf & D-ITG)
   3. Define evaluation scenarios
   4. Define evaluation environment
   5. Exit Criteria
      1. Client and supervisor feedback on evaluation planning

**Milestone 2 – Mid-Term Review**

1. Evaluation Case Development (07/06/2025 – 20/07/2025)
   1. Entry Criteria
      1. Client and supervisor feedback on evaluation planning
   2. Define evaluation scenario
      1. Requirement traceability
   3. Exit Criteria
      1. Review and validate evaluation cases
      2. Validate with client
      3. Client feedback

***The Evaluation Environment and Evaluation Execution Phases Combined Will Have Three Sprints, One for Each Linux Operating System.***

1. Evaluation Environment Setup (21/07/2025 – 28/09/2025)
   1. Entry Criteria
      1. Establishment of hardware and software
      2. Prepare evaluation data
      3. Configure evaluation environment
         1. Configure two computers as routers
         2. Configure three separate networks
      4. Write cases for each scenario
   2. Exit Criteria
      1. Fully functional evaluation environment and approved evaluation cases
      2. Client feedback
2. Evaluation Execution Phase (04/08/2025 – 12/10/2025)
   1. Entry Criteria
      1. All exit criteria from previous steps
         1. Evaluation environment is approved and functional
   2. Exit Criteria
      1. All evaluations are performed, and results are documented
         1. Conduct a minimum of 10 evaluation runs for each operating system
            1. Configure environment
            2. Each evaluation consists of 12 packet sizes
            3. Each evaluation will be run on both IPv4 and IPv6 separately
            4. Each evaluation will be run on both TCP and UDP separately
         2. Implement evaluation case failure protocol
         3. Client feedback
         4. Execute evaluation cases, scripts, pings
         5. Record and document results for each evaluation run
         6. Analyse performance metrics to determine operating system performance
         7. Client feedback
         8. Move onto next sprint

Sprint 1: Ubuntu (21/07/2025 – 17/08/2025)

Evaluation Environment Setup (21/07/2025 – 03/08/2025)

Evaluation Execution Phase (04/08/2025 – 17/08/2025)  
**Milestone 3 – Completion of Ubuntu Evaluation**

Sprint 2: Fedora (18/08/2025 – 14/09/2025)

Evaluation Environment Setup (18/08/2025 – 31/08/2025)

Evaluation Execution Phase (01/09/2025 – 14/09/2025)

**Milestone 4 – Completion of Fedora Evaluation**

Sprint 3: Kali (15/09/2025 – 12/10/2025)

Evaluation Environment Setup (15/09/2025 – 28/09/2025)

Evaluation Execution Phase (29/09/2025 – 12/10/2025)

**Milestone 5 – Completion of Kali Evaluation**

1. Evaluation Closure (27/10/2025 - 31/10/2025)
   1. Entry Criteria
      1. Compile and compare results of evaluation
         1. Analyse performance pattern
         2. Summary of the network performance evaluation
         3. Create visualisations and reports
         4. Final poster
   2. Exit Criteria
      1. Document closure report
      2. Evaluation results and analysis
      3. Organise folders for portfolio
      4. Client feedback

**Milestone 6 – Final Poster**

## Appendix O – Gantt Chart

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

### Critical Path Analysis

**Author: Win Phyo**

**Date: 03/04/2025**

**Version: 1.1**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Task name** | **ID** | **Estimated Duration** | **Predecessors** | **Start Date** | **End Date** |
| Evaluation Case Development | **10** | **32 days** | **5** | **07/06/2025** | **20/07/2025** |
| Evaluation Environment Setup and Evaluation Execution Phase | **16** | **60 days** | **10** | **21/07/2025** | **12/10/2025** |
| Sprint 1 – ubuntu | **17** | **21 days** | **10** | **21/07/2025** | **17/08/2025** |
| Sprint 2 – Fedora | **30** | **21 days** | **17** | **18/08/2025** | **14/09/2025** |
| Sprint 3 – Kali | **43** | **21 days** | **30** | **15/09/25** | **12/10/25** |
| Evaluation Closure | **56** | **15 days** | **16** | **12/10/25** | **31/10/25** |
| Milestone 7 - Portfolio | **62** | **0 day** | **60** | **31/10/25** | **31/10/25** |

The table above shows the list of tasks and summary that are indicated as critical to the project, the id, estimated duration, start date and the end date. The critical path analysis has been updated based on the network diagram in Gantt Chart.

## Appendix Q – Labour Breakdown Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Role** | **Cost Breakdown** | **Average pay per hour (inc. GST)** | **Total Hours** | **Total Cost (NZD)** |
| Mentor | Provided by AUT:  $142+GST | $163.30 | 1 hour per week  24 weeks (2 sems)  1 x 24 = **24** hrs | 163.30 x 24  = **$3,919.20** |
| Project Manager | Average yearly salary: $98,461 (PayScale, 2025) | $47.34 | 15 hours per week  24 weeks (2 sems)  15 x 24 = **360** hrs | 360 x 47.24  = **$17,006.40** |
| Network Engineer  (x4) | Average yearly salary: 78,377 (PayScale, 2024) | $37.68 | 15 hours per week  24 weeks (2 sems)  15 x 24 = 360  360 x 4 = **1440** hrs | 1440 x 37.68 = **$54,259.20** |
| System Architect  (x2) | Average yearly salary: $98,895 (PayScale, 2023) | $47.55 | 15 hours per week  24 weeks (2 sems)  15 x 24 = 360hrs  360 x 2 = **720** hrs | 720 x 47.55  = **$34,236.00** |

p

# References

Atlassian. (n.d.). *Waterfall Methodology: A Comprehensive Guide*. <https://www.atlassian.com/agile/project-management/waterfall-methodology>

Chandra, D. G., Kathing, M., & Kumar, D. P. (2013). A Comparative Study on IPv4 and IPv6. *2013 International Conference on Communication Systems and Network Technologies*, 286-289. <http://doi.org/10.1109/CSNT.2013.67>

Gamess, E., & Velásquez, K. (2008). *IPv4 and IPv6 Forwarding Performance Evaluation on Different Operating Systems*. CLEI. <https://clei.org/proceedings_data/CLEI2008/Anales/pdf/CLEI/CLEI_2008_118.pdf>

Katalon. (n.d.). *Software Testing Life Cycle: A Definitive Guide.* Katalon. <https://katalon.com/resources-center/blog/software-testing-life-cycle>

Laoyan, S. (2025, February 20). *What is agile methodology? (A beginner’s guide)*. Asana. <https://asana.com/resources/agile-methodology>

Narayan, S., Sodhi, S. S., Lutui, P. R., & Vijayakumar, K. J. (2010). Network performance evaluation of routers in IPv4/IPv6 environment. *2010 IEEE International Conference on Wireless Communications, Networking and Information Security*, *2*, 707–710. <https://doi.org/10.1109/WCINS.2010.5541871>

PayScale. (2024, May 21). *Average Network Engineer Salary in New Zealand*. <https://www.payscale.com/research/NZ/Job=Network_Engineer/Salary>

PayScale. (2025, January 05). *Average Project Manager, Information Technology (IT) with Project Management Skills Salary in New Zealand*. <https://www.payscale.com/research/NZ/Job=Project_Manager%2C_Information_Technology_(IT)/Salary/ac095581/Project-Management>

PayScale. (2023, October 19). *Average Systems Architect Salary in New Zealand*. <https://www.payscale.com/research/NZ/Job=Systems_Architect/Salary>

PB Technologies. (n.d.). *Supermicro 521R-T Mini Server - 1x Xeon E-2434 4C/8T 3.4GHz, 16GB RAM ... ( SVRSPM34170 )*. <https://www.pbtech.co.nz/product/SVRSPM34170/Supermicro-521R-T-Mini-Server---1x-Xeon-E-2434-4C8>

PB Technologies. (n.d.). *ASUS Vivobook Go 15 L510KA 15.6" FHD Intel Pentium Silver N6000 - 8GB... ( L510KA-EJ599W )*. <https://www.pbtech.co.nz/product/NBKASU510599/ASUS-Vivobook-Go-15-L510KA-156-FHD-Intel-Pentium-S>

PB Technologies. (n.d.). *Philips 243V7QJAB/79 24" FHD Monitor*. <https://www.pbtech.co.nz/product/MONPHS2438/Philips-243V7QJAB79-24-FHD-Monitor-1920x1080---IPS>

PB Technologies. (n.d.). *A4Tech Fstyler F1010 Multimedia Keyboard & Mouse Combo USB Wired -... ( F1010 )*. <https://www.pbtech.co.nz/product/KEYA4T1001/A4Tech-Fstyler-F1010-Multimedia-Keyboard--Mouse-Co>

PB Technologies. (n.d.). *Cruxtec 0.5m Cat7 Ethernet Cable - Black Color -- 10Gb / SFTP Triple... ( RS7-005-BK )*. <https://www.pbtech.co.nz/product/CABCXT920050/Cruxtec-05m-Cat7-Ethernet-Cable---Black-Color----1>

PB Technologies. (n.d.). *TP-Link TG-3468 32-bit Gigabit PCIe Network Adapter, Realtek RTL8168B... ( TG-3468 )*. <https://www.pbtech.co.nz/product/NETTPL3468/TP-Link-TG-3468-32-bit-Gigabit-PCIe-Network-Adapte>

Quintero, A., Sans, F., & Gamess, E. (2016). Performance evaluation of IPv4/IPv6 transition mechanisms. *International Journal of Computer Network and Information Security*, *8*(2), 1–14. <https://doi.org/10.5815/ijcnis.2016.02.01>

Taffer, M. (2024, December 9). *Lean Project Management: Definition, Principles, & Drawbacks*. The Digital Project Manager. <https://thedigitalprojectmanager.com/projects/pm-methodology/lean-project-management/>