Title: “**Enhancing TLS Performance in Kubernetes: Evaluating Cipher Suite Selection Strategies**”

This thesis aims to conduct a comparative analysis of Secure Sockets Layer (SSL) cipher suites to evaluate their security, performance, and suitability within containerized environments. This thesis will also take into account certificate adoption and security implications on its adoption within these environments. SSL cipher suites play a crucial role in establishing secure communication channels over the internet, and their selection can significantly impact the overall security posture of an application or system. By examining a range of popular cipher suites, this research seeks to provide valuable insights into their strengths, weaknesses, and optimal usage.

Chapter 1: Introduction

1. Background and Motivation
   1. Kubernetes is becoming de facto platform for today’s development especially in microservices and the transition from monolithic systems to API based decomposed solutions. Kubernetes provides many benefits due to scalability, performance, failover etc and supporting agile development.
   2. SSL strategy is quite key to secure implementations. TLS offloading - offered by cloud providers, interesting to compare this approach.
   3. Full end to end TLS is the recommended approach with secure DB connection and secure data at rest, will act as the first comparison.
   4. Varying cipher suites and post quantum TLS ciphers, to investigate effect on performance for newer cipher suites
2. Research Objectives
   1. Ensure a solid architecture is in place to have comparative results, with a strong toolset to monitor applications and provide breakdown on performance areas.
   2. Isolate:
      1. app processing
      2. network
      3. SSL handshake
      4. Crypto

to be able to do an effective comparative analysis

1. Research Questions
   1. Post quantum within containerized kubernetes environments is still very new. May be challenging to get this working.
2. Scope and Limitations
   1. Will be based on a simplified custom developed application as this is not the focus of the study area.

Chapter 1b: Technical milestones

1. Choosing the platform - factors came into play - AWS, GCP
2. Platform architecture
   1. GCP cluster with TLS Offloading
   2. Full end to end TLS (on traefik and application)
   3. Other investigations
      1. Lets encrypt doesn’t support google domains by DNS lookup - tried with Google cloud DNS
      2. Lets encrypt cert manager deployment
      3. TLS secrets for certificate inclusion
3. Application architecture for TLS
4. Setup of varying TLS cipher suites on Full E2E TLS

Chapter 2: SSL and Cryptographic Fundamentals

2.1 Overview of SSL/TLS

2.4 Cipher Suites

Chapter 3: SSL Cipher Suite Selection Criteria

3.1 Security Considerations

3.2 Performance Factors

3.3 Compatibility and Interoperability

Chapter 4: Comparative Analysis Methodology

1. Research Methodology
2. Test Environment and Tools
3. Monitoring
4. Test suite investigation
5. Evaluation Metrics

Chapter 5: Comparative Analysis of SSL Cipher Suites

5.1 Cipher Suite 1: Description, Strengths, and Weaknesses

5.2 Cipher Suite 2: Description, Strengths, and Weaknesses

5.3 ...

5.n: Comparative Analysis of Additional Cipher Suites

Chapter 6: Security Evaluation

6.1 Cryptographic Strengths

6.2 Vulnerability Analysis

6.3 Resistance to Attacks

Chapter 7: Performance Evaluation

7.1 Computational Overhead

7.2 Latency and Throughput

7.3 Resource Utilization

Chapter 8: Compatibility and Interoperability Analysis

8.1 Browser Support

8.2 Interoperability with Legacy Systems

8.3 Forward Secrecy and Key Exchange Methods

Chapter 9: Discussion and Findings

9.1 Comparative Analysis Summary

9.2 Key Findings and Observations

Chapter 10: Recommendations and Future Work

10.1 Recommended Cipher Suites for Different Use Cases

10.2 Areas for Further Research

10.3 Future Trends in SSL/TLS Security

Chapter 11: Conclusion

11.1 Summary of Research

11.2 Contributions and Implications

11.3 Final Remarks

Appendices:

A. SSL Cipher Suite Configurations

B. Detailed Test Results

C. Glossary of Terms

D. List of Abbreviations

Ana suggested structure in overleaf:

II. Introduction

II-A. Background

II-B. CLoud environments

SHORT PLEASE

II-C. TLS and company

III. Methodology

III-A. Requirements

ABSTRACT: What do I want to measure/prove/analyze??

III-B. Environment design

III-C. Measurement and monitoring needs

III-D. Technology selection

Which monitoring tool? Libraries? Frameworks?

III-E. Experiment definition

IV. RESULTS

V. DIscussion

VI. Limitations

VII. Conclusions and further work

References:

TLS1.3

<https://www.iana.org/assignments/tls-parameters/tls-parameters.xhtml>

<https://datatracker.ietf.org/doc/html/draft-ietf-tls-rfc8447bis-04>

<https://datatracker.ietf.org/doc/rfc8446/>

<https://en.wikipedia.org/wiki/Transport_Layer_Security#Algorithms>

Prometheus setup:

<https://sysdig.com/blog/kubernetes-monitoring-prometheus/>

<https://traefik.io/blog/capture-traefik-metrics-for-apps-on-kubernetes-with-prometheus/>