**ASSESSMENT 1:**

**Load Balancing**

**and Docker**

**COMP3004**

**Advanced Computing and  
Networking Infrastructures**

**2022/2023**

Table of Contents

[1. Introduction 2](#_Toc130363138)

[2. Experiment and Setup 4](#_Toc130363139)

[3. Automation Process 4](#_Toc130363140)

[4. Results and Discussion 4](#_Toc130363141)

[5. Conclusions 4](#_Toc130363142)

[References 6](#_Toc130363143)

# 1. Introduction

The Internet is one of the fastest growing technologies, having amassed 4.7 billion regular users worldwide (Roser et al., 2015). Providing access to the World Wide Web, people across the world can access websites/platforms that serve a range of purpose (i.e., online shopping, entertainment, online communication).

Chart, line chart

Description automatically generated

**Figure 1:** Number of people who used the Internet in the last three months (Roser et al., 2015)

Media Service Providers are an example of the distribution of entertainment across the Web, wherein digital media (in the forms of video, image, and/or audio) is uploaded to the Internet, providing users global access to digital content. Notable examples of these services include: Netflix, YouTube, and Spotify.

As the Internet grows, so do the media platforms that are hosting content, see Figure 2. With more and more users simultaneously using these platforms, performance becomes a priority, as the load on the backend webservers increases. HTTP requests become slow as response times increase. Errors also become more frequent, wherein HTML content might not be loaded fully. For platforms that are trying to entertain users, having seconds of ‘delay’ results in users moving to competitors’ service.

The reason for surges in user activity could be related to content on the platform going ‘viral’ or because of global events like COVID-19 restricting individuals from in-person socialisation, resulting in them going online and using web platforms to entertain themselves.

Chart, line chart

Description automatically generated

**Figure 2:** Number of people using social media platforms, 2005 to 2019 (Roser et al., 2015)

When servers are under high stress and are performing very inneffectively, or even completely offline, the website will generate less revenue, and profits will decrease. Because of this loss, hackers might be motivated to try to intentionally take down these web servers via brute force DDoS attacks – wherein servers are flooded with ‘botted’ web traffic. Akamai (2018) reported that there was a 16% increase in recorded DDoS attacks between 2017 and 2018. Likewise, within the year 2022 alone, out of 573 UK-based businesses surveyed, 10% identified occurrences of denial-of-service attacks (“*2022 in review: DDoS attack trends and insights”,* 2023).

This threat of downtime incetivises companies to ensure that their websites have built in redundancy and systems to help servers maintain optimal throughput. This study aims to research one such method used to improve throughput – *Load Balancing.* This technology is used to distribute web traffic among hardware resources and the end goal is to reduce the stress on individual nodes (i.e. application servers). The benefit that this provides is that nodes are not overloaded as traffic will be distributed amongst less busy servers, which theoertically reduces response time (Zhang & Fan, 2008).

This report is organised as follows. Section 2 discusses the design and purpose of the experimental system. Section 3 highlights the process of automating *Load Balancing*. Section 4 outlines and justifies the results, explaining their findings. Section 5 concludes the experiment with a summary of the whole process.

# 2. Experiment and Setup

**Docker**

Use of docker within project? What is it?

**Nginx**

Use of Nginx within project? What is it?

**Load Balancing**

Use of load balancing within project? What is it?

**Automation**

Use of load balancing within project? What is it?

375 words. Discuss: (a systematic diagram and descriptions about the system designed, system setup steps to install relevant packages/dependencies (in shell scripts, e.g. install.sh), and some explanations)

* Task: Research and experiment with the possibility of using LOAD BALANCING to balance web traffic.
* Approach: Docker, Ubuntu Linux server > use docker containers as a software load balancer.
* Benefit: Automation methods can be applied to configurations, dockerfile images, shell scripts for initial system setup/install.

# 3. Automation Process

450 words. Discuss: (your written Dockerfiles, Docker Compose file, plus descriptions/explanations)

# 4. Results and Discussion

375 words. Discuss: (evidence to show that the system works as you have designed/configured. This should include screenshots, plus descriptions/explanations, on e.g., load balancing and automation. Discussions should also cover what limitations are for the approaches taken)

# 5. Conclusions

100 words. Conclusions here

# References

“Akamai State of the Internet / Security Summer 2018” (2018) *Akamai* [Preprint]. Available at: https://www.akamai.com/newsroom/press-release/akamai-releases-summer-2018-state-of-the-internet-security-report (Accessed: March 18, 2023).

*“Cyber security breaches survey 2022”* (2022) *GOV.UK*. Available at: https://www.gov.uk/government/statistics/cyber-security-breaches-survey-2022/cyber-security-breaches-survey-2022#chapter-5-incidence-and-impact-of-breaches-or-attacks (Accessed: March 18, 2023).

Roser, M., Ritchie, H. and Ortiz-Ospina, E. (2015) *Internet*, *Our World in Data*. Available at: https://ourworldindata.org/internet (Accessed: March 21, 2023).

Yosifova, V. *et al.* (2019) “Trends review of the Contemporary Security Problems in the Cyberspace,” *Proceedings of the 9th Balkan Conference on Informatics* [Preprint]. Available at: https://doi.org/10.1145/3351556.3351560.

Zhang, Z. and Fan, W. (2008) “Web server load balancing: A queueing analysis,” *European Journal of Operational Research*, 186(2), pp. 681–693. Available at: https://doi.org/10.1016/j.ejor.2007.02.011.