ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 2, pp: (93-96), Month: April - June 2023, Available at: www.researchpublish.com

The Case of Load Balancing a Tier 1 Service

Qasim T. Zaidi, Ahmed S. Ismail[1]

Saudi Arabian Oil Company, Dhahran, Kingdom of Saudi Arabia

DOI: https://doi.org/10.5281/zenodo.7961036

Published Date: 23-May-2023

Abstract: **The research article provides guidelines and a case study to effectively provide resiliency and high availability for a mission critical service which is load balanced in multiple data centers. Most critical services rely on the built-in availability feature but by taking advantage of a hardware or software Load balancer, organizations can disperse the load and provide uninterrupted service while affording downtime for backend applications. By way of these load balancers, requests are automatically made to the least used servers behind them. The same load balancers also function as health checks for the end points on the backend systems while providing continuous and consistent validation of service availability. In the event of any service failure on the backend system, the particular service is not used until it reports healthy.**

Keywords: **Load Balancer, Tier 1 Service Availability, Resiliency, Client Connectivity**.

I. INTRODUCTION

An information technology ("IT") infrastructure implemented in multiple data centers, such as multi-user database and electronic mail ("email") systems, require high availability with zero downtime. Large organizations with many email servers depend on either physical or logical Load Balancers to provide uninterrupted service to their end users. By way of these load balancers, requests are automatically made to the least used servers behind them. The same load balancers also function as health checks for the end points on the backend systems while providing continuous and consistent validation of service availability. In the event of any service failure on the backend system, the particular service is not used until it reports healthy. This basic functionality of a load balancer results in operational success of the organization including times when either the backend systems are in maintenance windows or require troubleshooting for failures etc. Services are labelled different Tiers depending on the criticality to the organization. Tiers decide whether a service is mission critical or whether it useful and helpful but not essential.

Tier 1 Service

"Tier 1 services are the most critical services in your system. A service is considered Tier 1 if a failure of that service will result in a significant impact to customers or to the company's bottom line." [1]

II. CASE STUDY

To understand the importance of Load Balancers and Tier 1 service, the following have been studied with the primary objective of improving availability and higher resiliency of failures. Email service has been selected as a top Tier service for any organization in this case study.

- Physical Load Balancers in multiple Data Centers
- Email Servers in the multiple Data Centers
- DNS (Domain Name Server) for service name resolution, Round Robin, and Netmask ordering

Scenario:

To better illustrate the findings of this study, consider a scenario where a large organization with multiple data centers provide an Information Technology service such as Email to its employees in different regions of a country. In such a case, the organization may rely on a particular load balancer installed and configured in each of the data center. Clients such as

Microsoft Outlook would connect to these load balancers via name resolution by the DNS servers in the closest site. The selected load balancer would then balance the traffic between the backend email servers with health check of the service provided by the email server to which client is connected to. Figure I demonstrates the flow of connectivity between the load balancer and backend email servers.

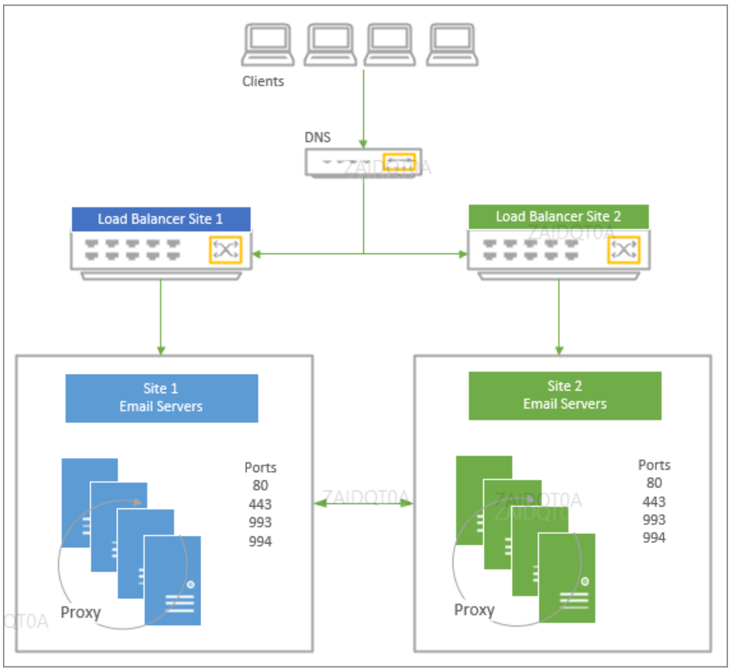

Fig. I

At first glance, and without further tests, this configuration of high availability of email service provides redundancy of not only the email servers but also protection against failures of either load balance. There may be a time when all email servers behind a particular load balancer are unavailable, during which that load balancer continues to provide connectivity to the clients with no backend systems to reply to the clients' requests. In such a scenario, this particular design fails to provide uninterrupted service. Fig II shows email servers in Site 2 as being unavailable while clients continue to connect to the load balancer in Site 2 causing email service outage.

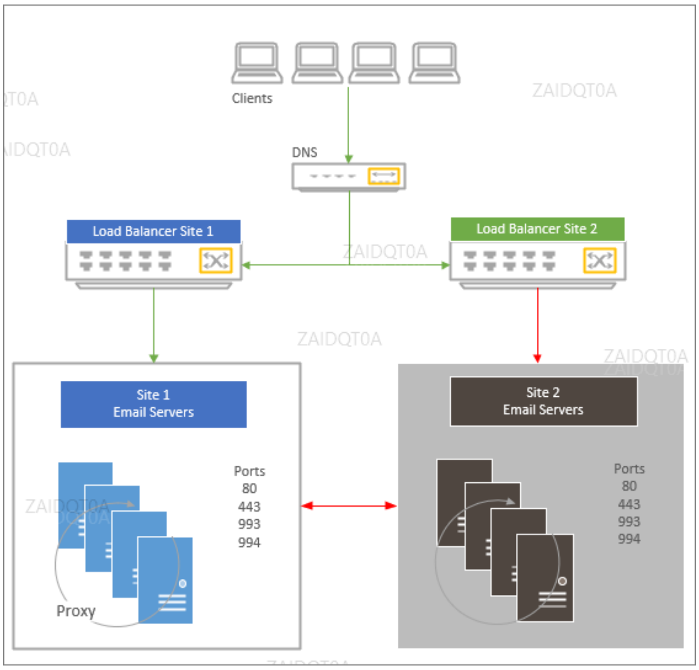

Fig. II

The following table summarizes advantages and disadvantages of the setup:

Table I

Pros	**Cons**
Fault tolerance against single data center failure	Service impact if all backend systems are unavailable
Automatic switch over to either load balancer	In case of no backend systems, load balancer must be disabled
Zero service impact as long as one backend system is available	Maintenance and management of multiple load balancers with updates/configuration etc.
Geo-redundant availability of load balancing	
Uninterrupted client connectivity to either site	

This implementation does provide the best of both worlds that is service availability and resiliency to failures with auto-failover of the load balancers and backend email servers, however, at times there may be optimization needed as presented in the below two options

Option 1:

One of the methods to improve this site-dependent backend service is to distribute them evenly behind each of the load balancer. For instance, organizations could take half of the backend systems in Site 1 and place them behind load balancer in Site 2 and half of the backend systems in Site 2 behind load balancer in Site 1. With this, the number of backend systems behind each load balancer remain the same, while providing an extra layer of protection if one of the two data centers is inaccessible. Availability and failover of the backend service remains the same while load is distributed by the two load balancers to the servers in the available data center. This method provides a much higher resilience to failures of either site. Fig III illustrates this option.

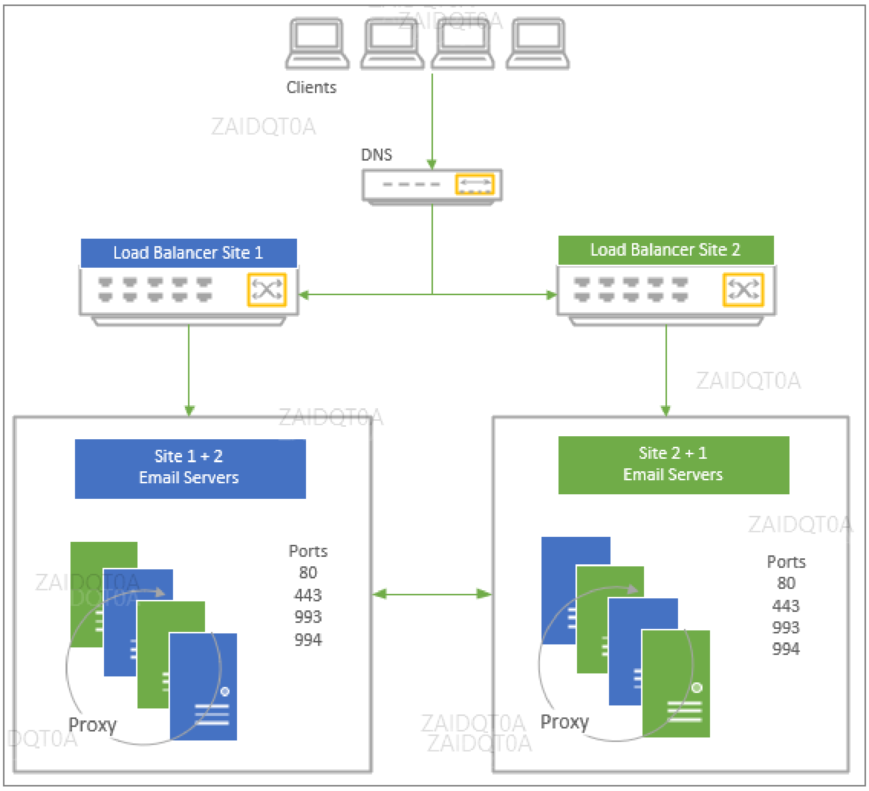

Fig. III

Option 2:

Second method would be to remove DNS entry for one of the load balancer IP address and use the automatic failover of the load balancer to another site while keeping all backend servers behind it. In this method, clients will use a single highly-available load balancer which is geo-redundant in the two sites. All backend servers are enabled behind the single load balancer. This method provides resiliency of failures against either the load balancer of the site-specific backend system. A setup such as this is depicted in Figure IV and organizations may find this as the most suitable option to provide uninterrupted access to their Tier 1 critical service such as email.

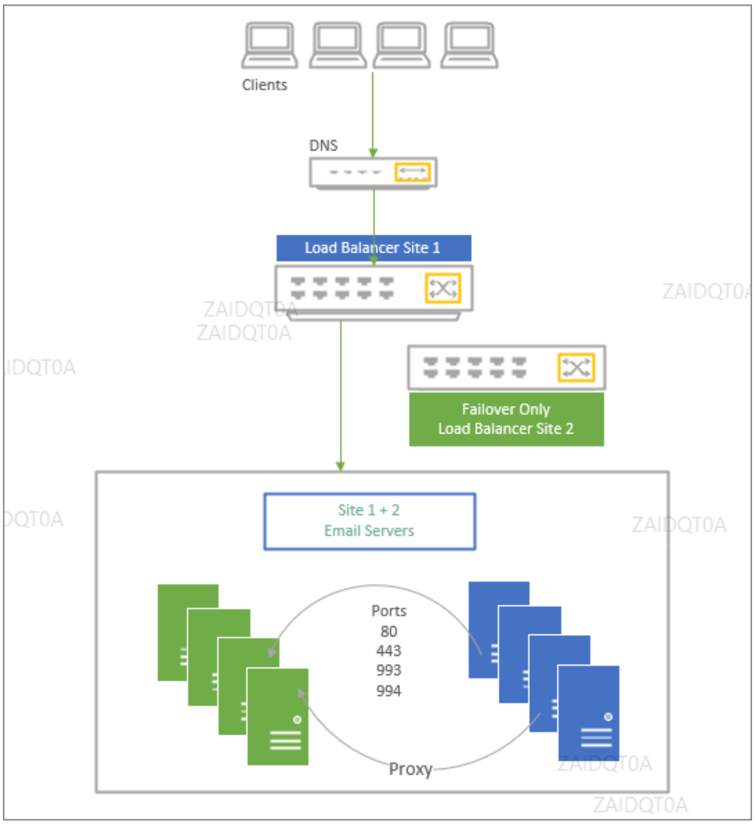

Fig. IV

III. CONCLUSION

This research paper concludes the different approach to re-evaluate load balancing for a Tier 1 application such as email service. Some load balancing products may offer more advanced features such as rejecting requests from clients if none of the backend systems respond. As for the scope of this research paper, such advanced features are not considered. Features which are common to all load balancers with mission critical service's high availability and resiliency are researched to provide options to be considered by large organizations. Business critical applications are expected to be available at all times hence improvements must be carefully studied to enhance the Information Technology's infrastructure.

REFERENCES

[1] https://thenewstack.io/how-service-tiers-can-help-to-avoid-microservices-disasters/#:~:text=Tier%201%20Tier%201%20services%20are%20the%20most,to%20customers%20or%20to%20the%20company%E2%80%99s%20bottom%20line.

[2] Load Balancing in Exchange Server https://learn.microsoft.com/en-us/exchange/architecture/client-access/load-balancing?view=exchserver-2019