Lab 9 Configuring Backup and Recovery Functions

Osamudiamen Eweka

Cyb-605-Z2 Principles of Cybersecurity

Utica University

Introduction

In this lab, we will explore essential strategies for ensuring business continuity, disaster recovery, and high availability within IT infrastructures. Starting with the setup of a daily System State backup using the Windows Server Backup feature, we aim to prepare for quick recovery in case of system failures. We'll then configure high availability for web services by establishing a shared Network File System (NFS) and implementing load balancing with a pfSense firewall-router. This hands-on experience is designed to equip participants with practical skills in configuring backup and recovery functions, vital for maintaining operations and swiftly recovering from disruptions. Through these exercises, we'll underscore the importance of proactive planning and execution in safeguarding business processes against unforeseen events.

Objective

The objective of this lab is to provide participants with a comprehensive understanding of the integral processes involved in ensuring business resilience, including business impact analysis, risk analysis, risk assessment, business continuity planning, and disaster recovery planning. Participants will gain hands-on experience in installing and configuring Windows Server Backup for System State backups, restoring a Domain Controller from such backups, setting up a Linux NFS server with clients, and implementing load balancing across redundant web servers. By the end of this lab, participants will be equipped with the necessary skills to effectively safeguard IT infrastructures against disruptions, ensuring the continuity of business operations and the rapid recovery from unforeseen events.

Lab setup

To successfully complete this lab and achieve its objectives, participants will need access to and familiarity with a specific set of tools and software. These utilities are crucial for performing tasks related to backup configuration, system restoration, network file sharing, and load balancing. Students are encouraged to research these tools to gain a deeper understanding of their functionalities and applications within the lab. Below is a detailed list of the required software and utilities

1. **Windows Server Backup**: Integrated backup and recovery for Windows servers.
2. **Wbadmin**: Command-line utility for managing Windows backups and recoveries.
3. **NFS (Network File System)**: Protocol for accessing files over a network as if they were on the user's local storage, requiring setup of an NFS server and clients on Linux.
4. **Vi Editor**: Text editor for Unix and Linux systems, essential for editing configuration files.
5. **pfSense**: Open-source firewall and router software for network security, including load balancing across web servers.

Section 1

Part 1: Install Windows Server Backup

In this section of the lab, participants will engage in the process of installing the Windows Server Backup feature on a Windows 2019 Server using the Server Manager. This step is crucial for enabling the ability to perform backups of specified files, directories, or entire drives on the server, and it allows for the designation of a backup location, which could be on a network drive. The installation process begins with accessing the DomainController01(DC01) system through the Lab View toolbar, using a specific command sequence to log in as the domain administrator with a predefined password.

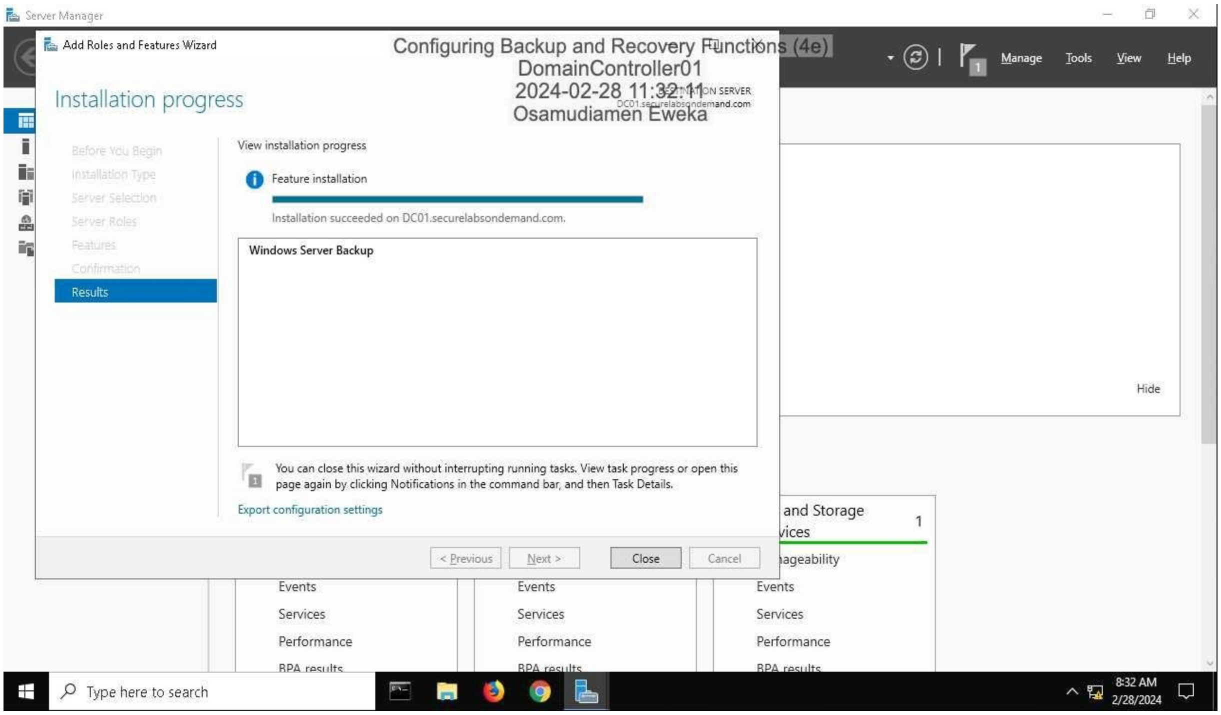
Once logged in, participants will navigate to the Server Manager application from the DomainController01 taskbar. The Server Manager is an essential tool for administrators, as it provides a consolidated console for managing roles and features across local and remote servers, facilitating the implementation of recovery plans. It displays the current roles (e.g., AD DS, Domain Name System (DNS), File and Storage Services) and server groups, which are instrumental in defining the functions a server performs within a network and managing multiple servers simultaneously.

The next step involves selecting the "Manage" option from the Server Manager menu bar, followed by "Add Roles and Features" to initiate the Add Roles and Features Wizard. This wizard guides users through the installation process, beginning with the selection of the installation type (Role-based or feature-based installation) and the server on which the feature will be installed (in this case, DC01, the DomainController01 server). Upon reaching the Features page, participants will select the Windows Server Backup checkbox to include this feature for installation.

The procedure culminates with the Confirmation page, where participants confirm their selections and proceed with the installation of the Windows Server Backup feature. This lab section is designed to familiarize participants with the practical aspects of setting up backup capabilities on a Windows server, an essential component of any disaster recovery and business continuity strategy. The ability to efficiently install and configure backup features prepares participants to effectively manage data protection and recovery in real-world scenarios.

Figure 1

Make a screen capture showing the completed Windows Server Backup feature installation.



Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 11:32:11
Osamudiamen Eweka

Add Roles and Features Wizard

Installation progress

DESTINATION SERVER
DC01.securelabsondemand.com

View installation progress

Feature installation

Installation succeeded on DC01.securelabsondemand.com.

Windows Server Backup

You can close this wizard without interrupting running tasks. View task progress or open this page again by clicking Notifications in the command bar, and then Task Details.

Export configuration settings

Note. Figure 1 confirms the successful completion of the Windows Server Backup feature installation, a crucial tool for administrators in Windows Server environments to execute backup and recovery tasks (Jones & Bartlett, 2024). This milestone equips the system with essential capabilities for data protection, enabling server state restoration in case of data loss or failure, thereby enhancing the server's data management resilience and reliability.

Section 1

Part 2: Configure a System State Backup

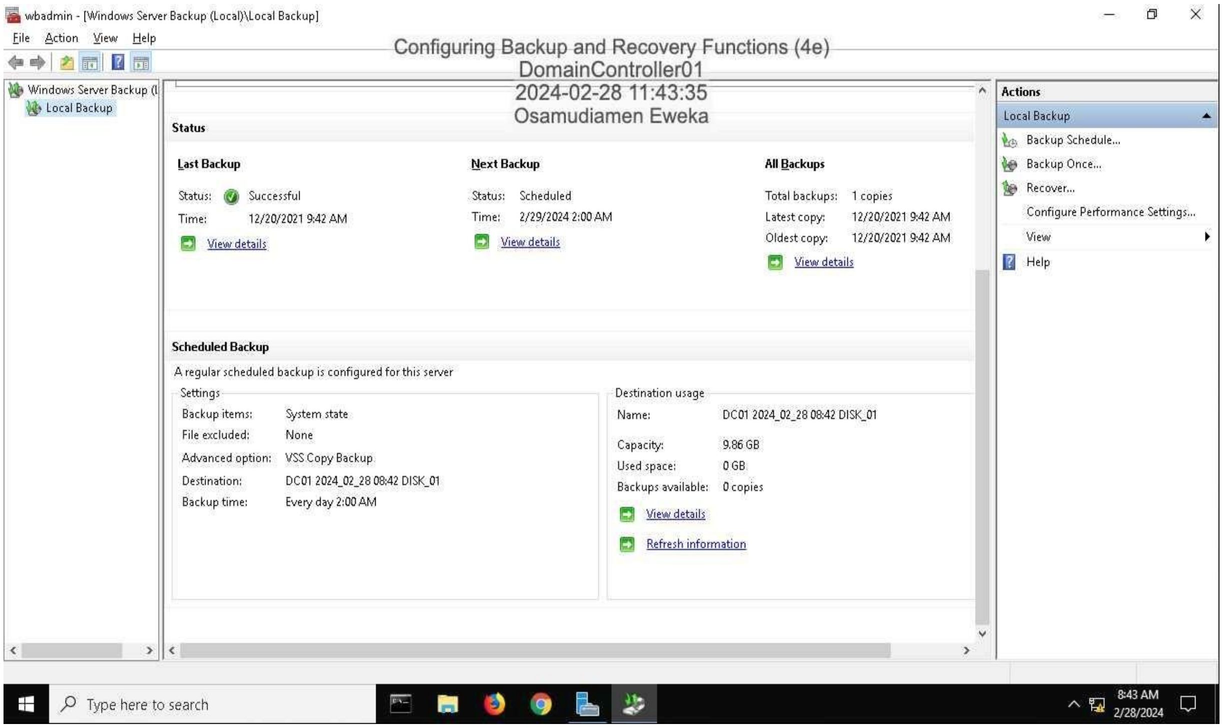
In this lab section, participants learn to set up a daily System State backup for the DomainController01 server using the Windows Backup Admin (wbadmin) console, highlighting the importance of System State backups for disaster recovery. The process involves launching the Backup Schedule wizard from the wbadmin console and choosing a custom backup configuration to specifically target the System State. This includes crucial system configurations and data necessary for a server's recovery.

Participants select a backup time of 2:00 AM to minimize server load impact and choose a dedicated 10GB hard disk as the backup destination, following best practices for backup storage. The disk is formatted to prepare it for storing backup files, ensuring the backup's integrity and availability.

The lab concludes with the successful scheduling of the System State backup, demonstrating the implementation of key disaster recovery planning aspects. This includes selecting the appropriate backup type, timing, and storage location to facilitate efficient and effective recovery in case of system disruptions.

Figure 2

Make a screen capture showing the Scheduled Backup settings, including the destination and backup time.



Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 11:43:35
Osamudiamen Eweka

Status

Last Backup
Status: Successful
Time: 12/20/2021 9:42 AM
View details

Next Backup
Status: Scheduled
Time: 2/29/2024 2:00 AM
View details

All Backups
Total backups: 1 copies
Latest copy: 12/20/2021 9:42 AM
Oldest copy: 12/20/2021 9:42 AM
View details

Scheduled Backup

A regular scheduled backup is configured for this server

Settings
Backup items: System state
File excluded: None
Advanced option: VSS Copy Backup
Destination: DC01 2024_02_28 08:42 DISK_01
Backup time: Every day 2:00 AM

Destination usage
Name: DC01 2024_02_28 08:42 DISK_01
Capacity: 9.86 GB
Used space: 0 GB
Backups available: 0 copies
View details
Refresh information

Note. Figure 2 presents a screenshot detailing the Scheduled Backup settings, which encompass both the chosen destination for the backups and the timing for these operations (Jones & Bartlett, 2024). By specifying where the backups are stored and when they are performed, administrators can tailor the backup process to meet the specific needs and schedules of their server environment, thus enhancing data protection and operational efficiency.

Section 1

Part 3: Restore from a System State Backup

In this section of the lab, participants go through the steps necessary to restore the DomainController01 server from an existing System State backup. This process is vital for quickly returning a server to its operational state following a failure, minimizing the Mean Time to Repair/Restore (MTTR) and ensuring continuity in critical services like domain controllers. Regular System State backups, as mandated by a Business Continuity Plan (BCP) or Disaster Recovery Plan (DRP), are key to this recovery process.

The restoration process starts with configuring the server to boot in safe mode specifically for Active Directory repair. This is done through the System Configuration application, where participants select the Safe boot option and the Active Directory repair mode under the Boot tab. This ensures the server restarts in Directory Services Restore Mode (DSRM), a prerequisite for performing a System State recovery since Active Directory cannot be online during the restore.

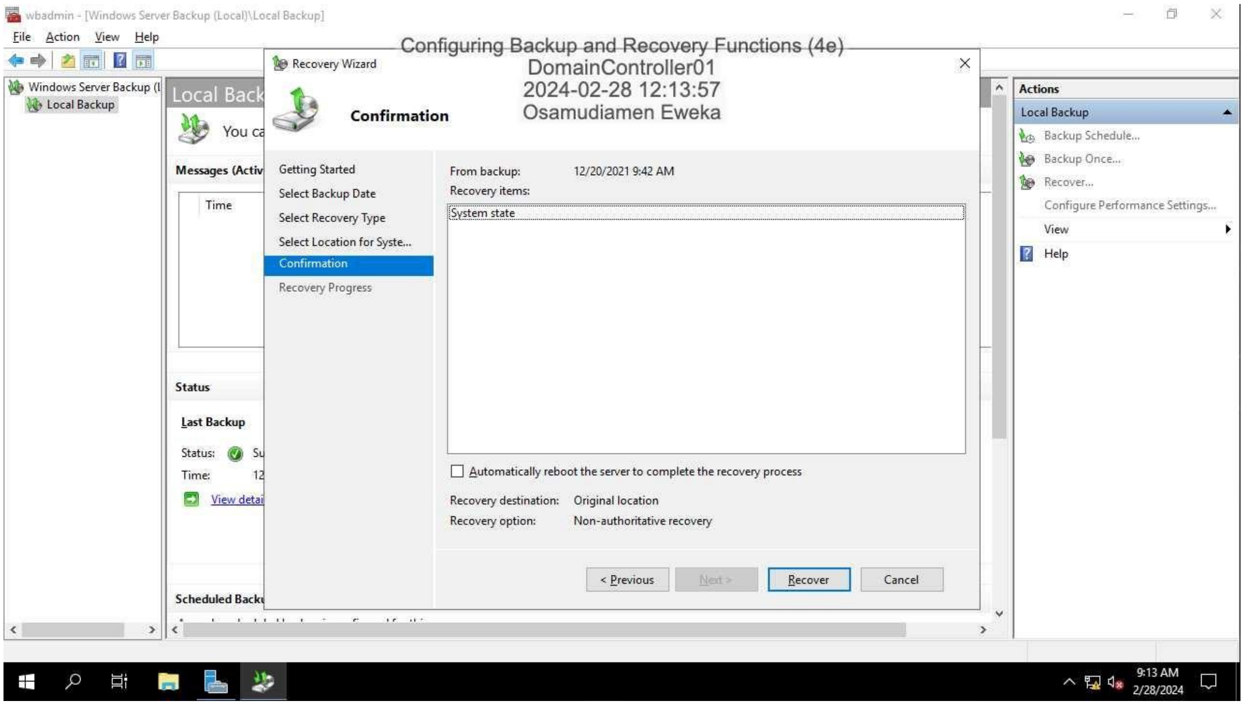
After restarting the server and logging in as the local administrator, participants are directed to open the Windows Server Backup (wbadmin) console from the Server Manager. From here, the Recover action is initiated, guiding the user through a series of selections that lead up to the recovery process. This includes choosing the server (DC01) as the recovery source, selecting a specific backup date (in this lab, a pre-existing backup stored on the server's D: drive), and specifying the System State as the recovery type. The recovery is intended to be performed in the original location of the System State data.

However, for the purposes of this lab, the actual restoration process is not executed to save time, given that a full System State recovery can take upwards of an hour to complete. Participants are instructed to reach the Recovery Wizard Confirmation page, where they would theoretically proceed with the restore, and then to cancel the operation.

This exercise demonstrates the critical steps involved in preparing for and initiating a System State recovery, underscoring the importance of such backups in an organization's disaster recovery strategy. Through this process, participants gain practical experience in using Windows Server Backup for disaster recovery purposes, preparing them to apply these skills in real-world scenarios to enhance system resilience and recovery capabilities.

Figure 3

Make a screen capture showing the Recovery Wizard Confirmation page.



wbadmin - [Windows Server Backup (Local)\Local Backup]

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 12:13:57
Osamudiamen Eweka

Recovery Wizard

Confirmation

Getting Started
Select Backup Date
Select Recovery Type
Select Location for Syste...
Confirmation
Recovery Progress

From backup: 12/20/2021 9:42 AM
Recovery items:
System state

☐ Automatically reboot the server to complete the recovery process

Recovery destination: Original location
Recovery option: Non-authoritative recovery

Note. Figure 3 showcases the Recovery Wizard Confirmation page, marking the last step before starting data recovery (Jones & Bartlett, 2024). It outlines recovery details, such as the data source and restoration target, allowing administrators to verify and confirm the operation's specifics, ensuring accurate data restoration to the correct location.

Section 2

Part 1: Configure an NFS Share

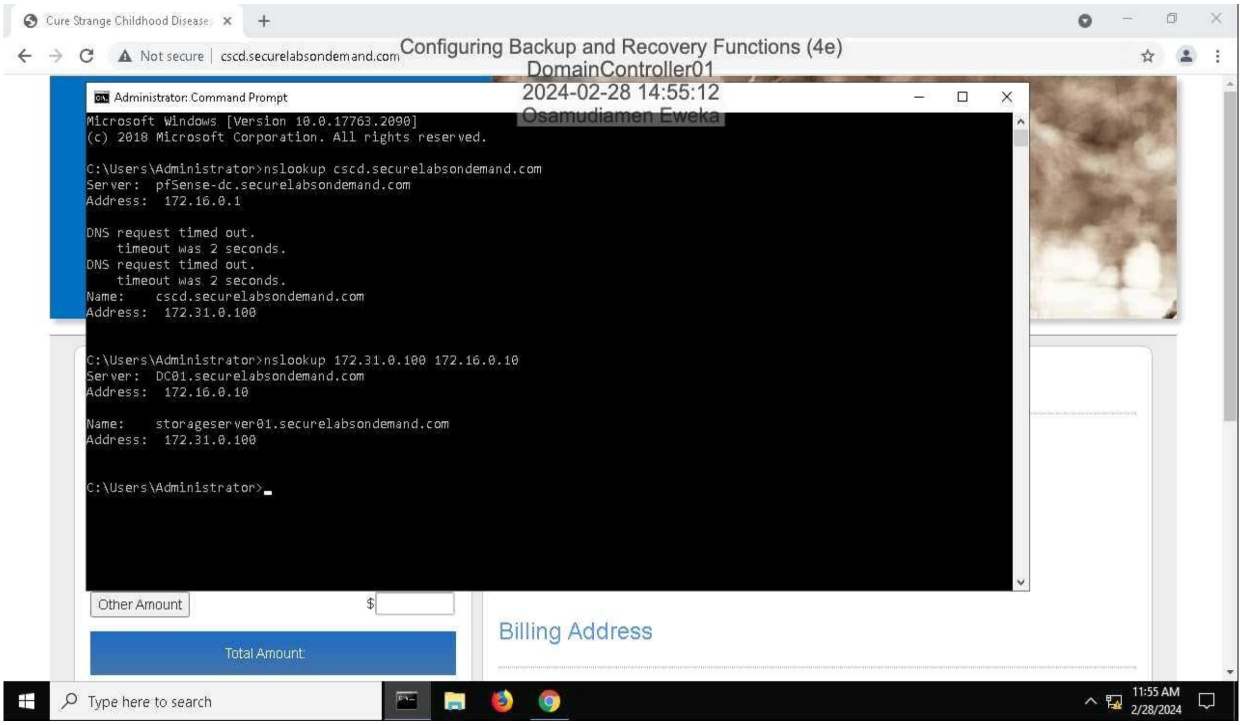
In this lab section, the task involves transitioning from a single-point-of-failure web server to a distributed system utilizing an NFS. The objective is to separate data storage from transport services, enhancing scalability and redundancy. Two web servers, webserver01 and webserver02, are configured to access web content from an NFS on a dedicated storage server. This setup ensures consistent data delivery and introduces redundancy, making the potential unavailability of one server transparent to end users.

The process begins with resetting the lab environment, followed by logging into the DomainController01 system to review the current web service configuration. The Community Supervision and Corrections Departments (CSCD) Society website, initially served by a single server, is accessed to evaluate its functionality before the transition. Subsequent steps involve executing DNS lookup commands to map out the network configuration and identifying the roles and IP addresses of the website and the storage server. This groundwork is crucial for a smooth transition to the NFS-based architecture, ensuring uninterrupted access to web content.

This lab segment emphasizes the practical application of NFS in creating a robust, scalable web server infrastructure. It highlights the importance of redundancy and scalability in web services, offering insights into fault-tolerant system design. Through this exercise, the concept of separating storage from service delivery is explored, underscoring the benefits of such an architecture in enhancing the reliability and scalability of web services.

Figure 4

Make a screen capture showing the results of the reverse DNS query.



Cure Strange Childhood Disease
Not secure | cscd.securelabsondemand.com
Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 14:55:12
Osamudiamen Eweka
Administrator: Command Prompt
Other Amount
$
Total Amount
Billing Address
Type here to search
11:55 AM
2/28/2024

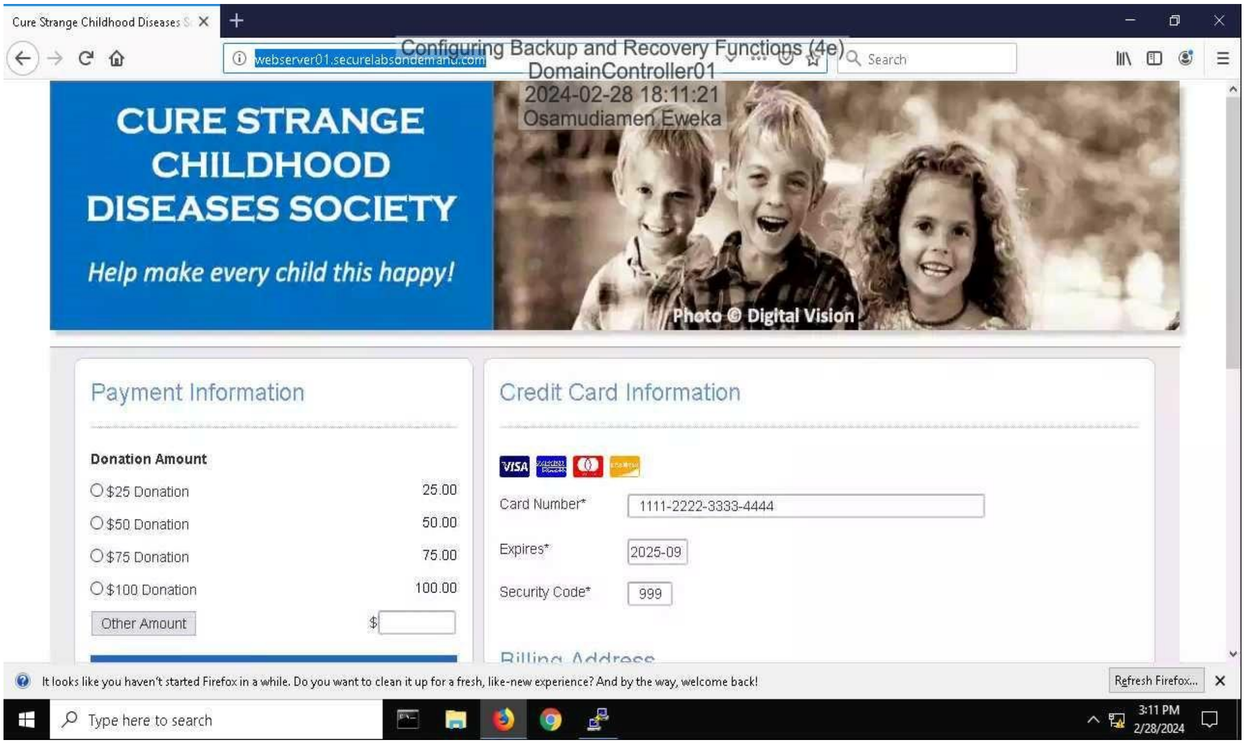
```
Microsoft Windows [Version 10.0.17763.2090]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>nslookup cscd.securelabsondemand.com
Server:  pfSense-dc.securelabsondemand.com
Address:  172.16.0.1

DNS request timed out.
    timeout was 2 seconds.
DNS request timed out.
    timeout was 2 seconds.
Name:    cscd.securelabsondemand.com
Address:  172.31.0.100


C:\Users\Administrator>nslookup 172.31.0.100 172.16.0.10
Server:  DC01.securelabsondemand.com
Address:  172.16.0.10

Name:    storageserver01.securelabsondemand.com
Address:  172.31.0.100


C:\Users\Administrator>_
```

Note. The screen capture illustrates the outcomes of a reverse DNS query (Jones & Bartlett, 2024). This process involves mapping an IP address back to its associated domain name, contrary to the more common forward DNS lookup, which translates domain names into IP addresses. The results displayed in the image provide insights into the domain name or names associated with a specific IP address, offering valuable information for network diagnostics, security analyses, and understanding the configuration of internet services.

After NFS server setup was completed to share the /var/www directory. The web servers, webserver01 and webserver02, were configured as NFS clients to mount this shared directory, ensuring both served the same web content for high availability. The configuration involved editing the /etc/fstab file on both web servers to include the NFS share, followed by immediate mounting with the mount -av command. Verification was done using the df -h command to confirm successful mounting. This setup aimed to enhance website reliability by enabling redundant web servers to serve identical content, demonstrating the effectiveness of distributed systems in improving web service scalability and redundancy.

Figure 5

Make a screen capture showing the updated webserver01 home page.



Cure Strange Childhood Diseases S

webserver01.securelabsondemand.com

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 18:11:21
Osamudiamen Eweka

CURE STRANGE CHILDHOOD DISEASES SOCIETY

Help make every child this happy!

Photo © Digital Vision

Payment Information

Donation Amount

○ $25 Donation	25.00
○ $50 Donation	50.00
○ $75 Donation	75.00
○ $100 Donation	100.00
Other Amount	$

Credit Card Information

Card Number* 1111-2222-3333-4444

Expires* 2025-09

Security Code* 999

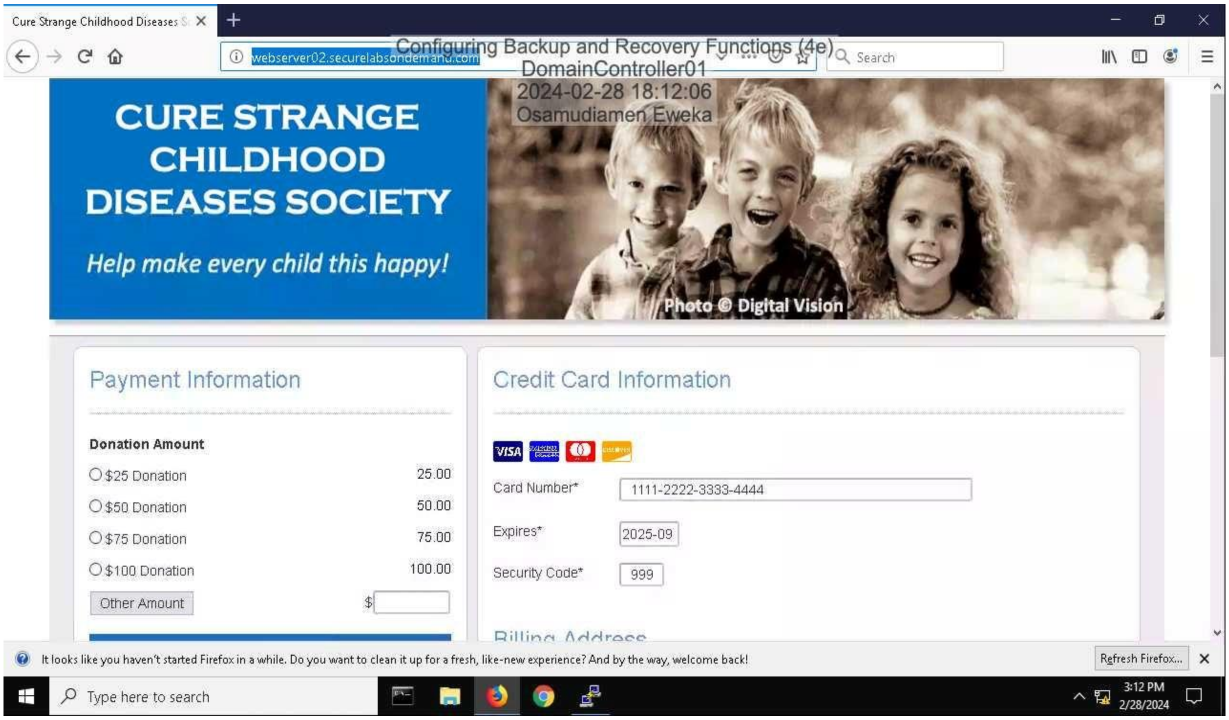
Billing Address

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Note. The image above displays the refreshed homepage of webserver01, the web servers now have files to serve when requested (Jones & Bartlett, 2024).

Figure 6

Make a screen capture showing the updated webserver02 home page.



Cure Strange Childhood Diseases S
webserver02.securelabsondemand.com
Search
Photo © Digital Vision
Refresh Firefox...
Type here to search

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 18:12:06
Osamudiamen Eweka

CURE STRANGE CHILDHOOD DISEASES SOCIETY

Help make every child this happy!

Payment Information

Donation Amount

○ $25 Donation	25.00
○ $50 Donation	50.00
○ $75 Donation	75.00
○ $100 Donation	100.00
Other Amount	$

Credit Card Information

Card Number* 1111-2222-3333-4444

Expires* 2025-09

Security Code* 999

Billing Address

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

3:12 PM
2/28/2024

Note. The image displays the refreshed homepage of webserver02, the web servers now have files to serve when requested (Jones & Bartlett, 2024).

Section 2

Part 2: Configure Load Balancing

In this part of the lab, the primary goal was to configure the pfSense firewall/router for load balancing across two redundant web servers, employing HAProxy. This setup is vital for distributing incoming web traffic evenly across servers, ensuring the website's high availability and reliability. By navigating to the pfSense webGUI, the process commenced with configuring HAProxy, a critical step towards achieving a balanced load distribution between webserver01 and webserver02.

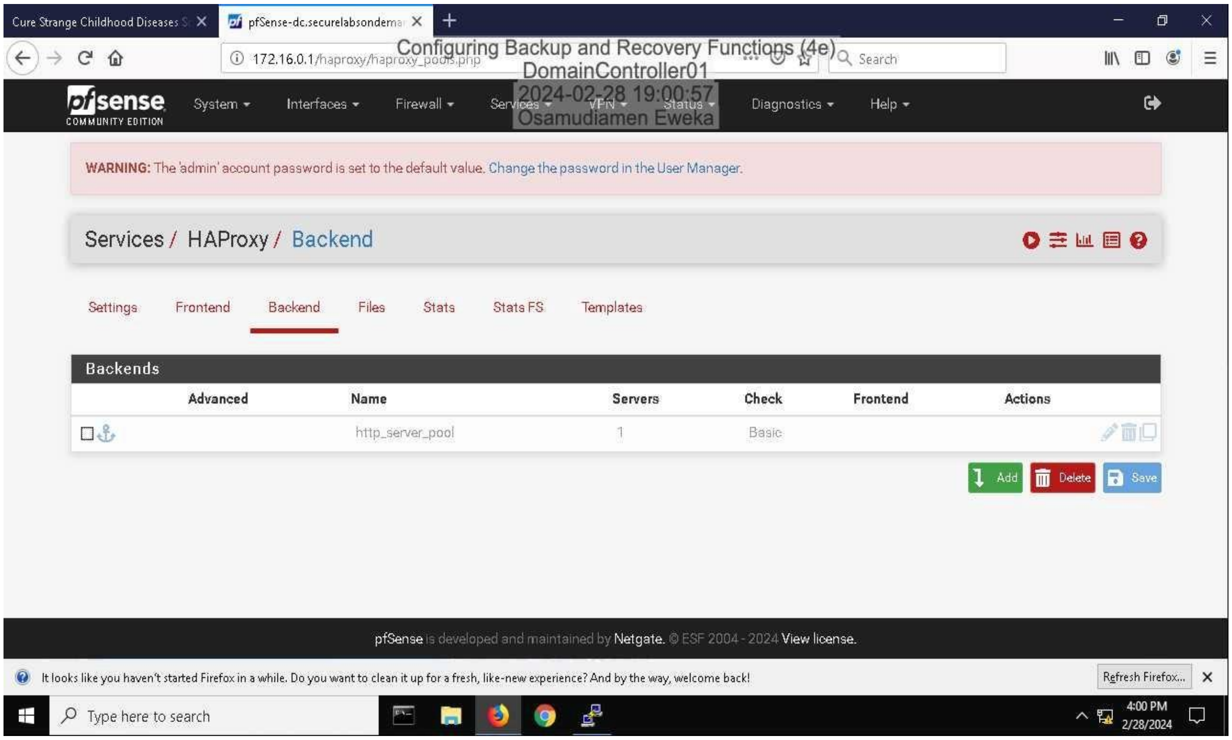
The configuration entailed defining a backend server pool within HAProxy, aptly named "http_server_pool," which included both web servers. These servers were set to listen on port 80, ready to handle Hypertext Transfer Protocol HTTP traffic. This backend setup is crucial for load balancing, as it determines how traffic is distributed among the servers. By employing the Static Round Robin algorithm, the system was configured to alternate traffic equally between webserver01 and webserver02, a method that ensures no single server bears too much load at any given time.

An essential part of the setup was the health checking mechanism, set to Basic for simplicity in this initial configuration. Although this means no active health checks were performed, it simplifies the lab's scope. However, in a real-world scenario, more sophisticated health checks would be necessary to ensure that traffic is only directed to servers that are fully operational, enhancing the website's overall availability and resilience.

After configuring the backend with the server details, the changes were saved and applied to the pfSense configuration, effectively enabling it to act as a load balancer. This setup ensures that the CSCD website can handle incoming traffic efficiently, maintaining performance and availability even under increased loads. This approach aligns with best practices for achieving high availability and reliability in critical web applications, ensuring that the website always remains accessible to users.

Figure 7

Make a screen capture showing the http_server_pool backend.



Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 19:00:57
Osamudiamen Eweka

172.16.0.1/haproxy/haproxy_pools.php

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Services / HAProxy / Backend

Backends

Advanced	Name	Servers	Check	Frontend	Actions
□ ⚓	http_server_pool	1	Basic		

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

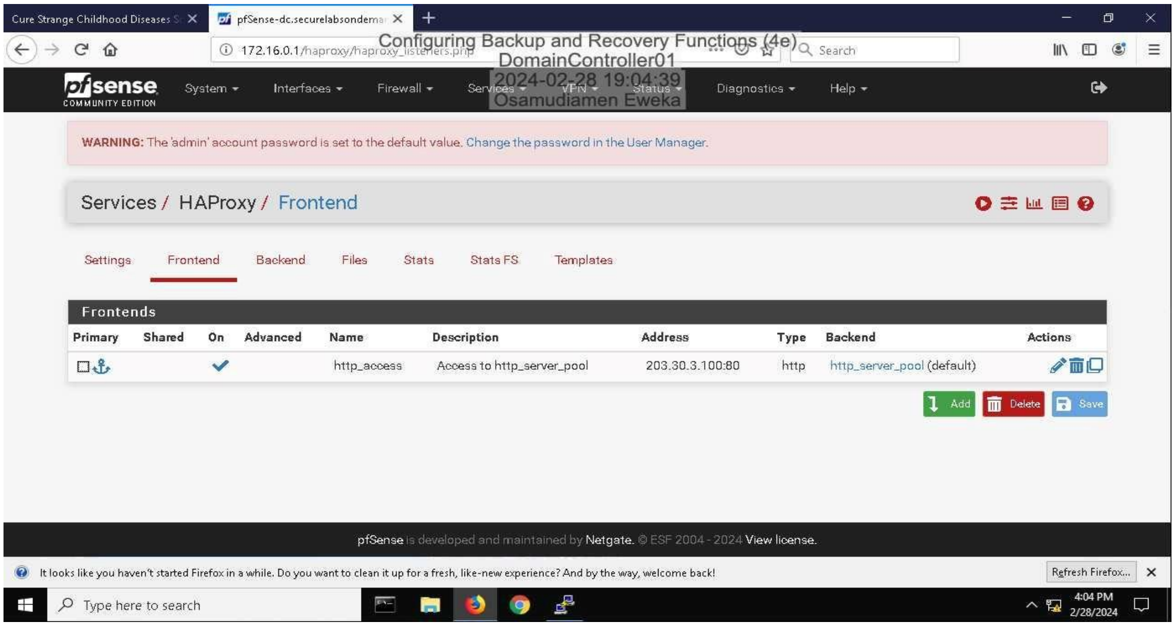
It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Note. Figure 7 shows the configuration of the "http_server_pool" backend, highlighting the setup details for a group of servers designated to handle HTTP requests (Jones & Bartlett, 2024). This backend configuration is crucial for distributing incoming web traffic across multiple servers, enhancing the web application's availability, load handling, and response time.

Moving on Participant configured the Frontend of the HAProxy on the pfSense firewall to manage and distribute incoming traffic to the CSCD website across two web servers, achieving load balancing. This was accomplished by adding a new Frontend named `http_access` with a specified public IP address (`203.30.3.100`) for the CSCD Society. This setup listens on the pfSense device's WAN interface to receive internet traffic. The Frontend was then linked to the `http_server_pool` backend, directing it to distribute incoming requests between the web servers in the backend pool. By applying these configurations, the pfSense firewall, equipped with HAProxy, now efficiently routes web traffic, ensuring high availability and performance of the CSCD website. This demonstrates the practical application of load balancing within a network infrastructure to maintain service accessibility and prevent server overloads.

Figure 8

Make a screen capture showing the http_access frontend.



Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 19:04:39
Osamudiamen Eweka

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Services / HAProxy / Frontend

Frontends

Primary	Shared	On	Advanced	Name	Description	Address	Type	Backend	Actions
☐ ⚓		✓		http_access	Access to http_server_pool	203.30.3.100:80	http	http_server_pool (default)	

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Note. The screen capture illustrates the "http_access" frontend configuration, detailing the setup through which incoming HTTP requests are received and managed (Jones & Bartlett, 2024). This configuration is pivotal for defining how web traffic is initially handled, including the criteria for request routing, security measures like SSL termination, and the initial point of entry for users accessing the system.

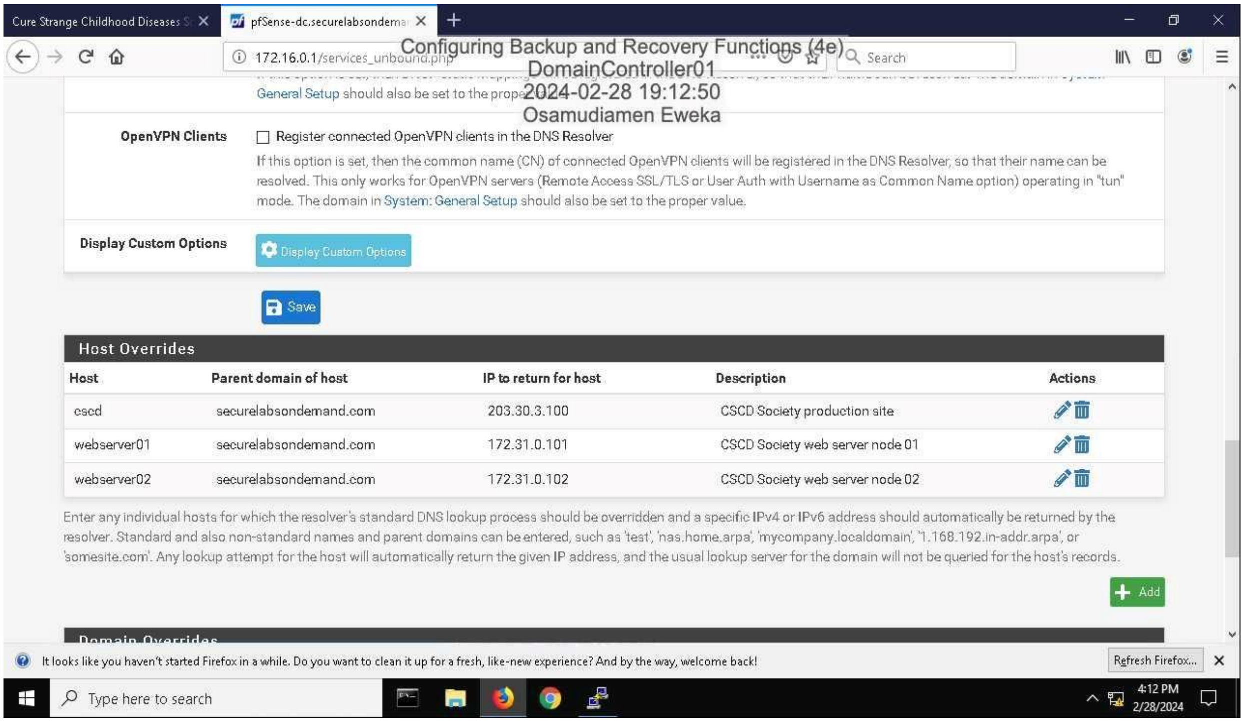
The next step involves, enabling and configuring HAProxy on the pfSense firewall to ensure the load balancing setup was operational for the CSCD website traffic. This involved enabling HAProxy through the General settings, where we also set a limit of 1000 maximum connections per process to manage the load efficiently. Additionally, we configured an internal stats port (2200) to monitor HAProxy's performance directly from the firewall.

A crucial update was made to the firewall's DNS Resolver settings, specifically within the Host Overrides section. We altered the DNS record for cscd.securelabsondemand.com, which initially pointed to storageserver01's IP in the DMZ, to now direct to the public IP (203.30.3.100) assigned to our HAProxy frontend. This change ensures that all traffic intended for the CSCD website is correctly routed through HAProxy, enabling effective load balancing across the redundant web servers.

By applying these configurations and updating the DNS record, we've completed the setup for a high-availability web service. This ensures that the CSCD website can handle traffic efficiently, maintain performance under load, and provide resilience against server failures, aligning with best practices for deploying mission-critical services**.**

Figure 9

Make a screen capture showing the new Host Overrides entry for cscd.securelabsondemand.com.



Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 19:12:50
Osamudiamen Eweka

172.16.0.1/services_unbound.php

General Setup should also be set to the prope

OpenVPN Clients □ Register connected OpenVPN clients in the DNS Resolver

If this option is set, then the common name (CN) of connected OpenVPN clients will be registered in the DNS Resolver, so that their name can be resolved. This only works for OpenVPN servers (Remote Access SSL/TLS or User Auth with Username as Common Name option) operating in "tun" mode. The domain in System: General Setup should also be set to the proper value.

Display Custom Options

Host Overrides

Host	Parent domain of host	IP to return for host	Description	Actions
cscd	securelabsondemand.com	203.30.3.100	CSCD Society production site	
webserver01	securelabsondemand.com	172.31.0.101	CSCD Society web server node 01	
webserver02	securelabsondemand.com	172.31.0.102	CSCD Society web server node 02	

Enter any individual hosts for which the resolver's standard DNS lookup process should be overridden and a specific IPv4 or IPv6 address should automatically be returned by the resolver. Standard and also non-standard names and parent domains can be entered, such as 'test', 'nas.home.arpa', 'mycompany.localdomain', '1.168.192.in-addr.arpa', or 'somesite.com'. Any lookup attempt for the host will automatically return the given IP address, and the usual lookup server for the domain will not be queried for the host's records.

Domain Overrides

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!

Note. The screen capture displays the addition of a new Host Overrides entry for the domain "cscd.securelabsondemand.com." (Jones & Bartlett, 2024).

Section 2

Part 3: Verify Load Balancing

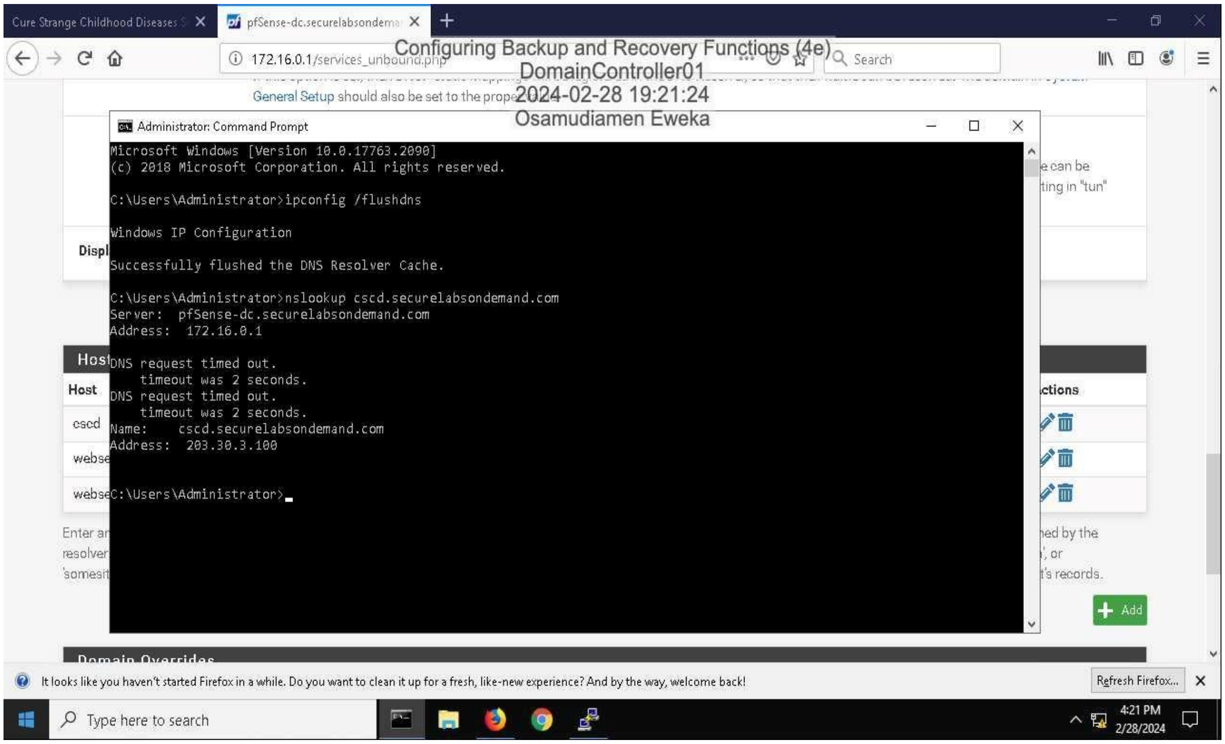
In this lab segment, participants performed a critical verification step for the load-balancing configuration by ensuring that the DNS changes were properly applied and recognized across the network. Load balancer acts as an intermediary for data traffic between clients and your application servers. Clients send requests to your load balancer and the load balancer distributes the requests to your backend servers according to rules you establish (Load Balancer Metrics, n.d.). This involved clearing the DNS cache on your system with the `ipconfig /flushdns` command, a necessary step to remove any old DNS records that could interfere with accessing the updated DNS information. By doing so, you ensured that any subsequent DNS queries would fetch the most recent DNS information, rather than relying on potentially outdated cached data.

Following the DNS cache flush, participants executed an `nslookup` command for `cscd.securelabsondemand.com`. This step was pivotal in confirming that the DNS changes were effective, as it verified that the domain now resolved to the new IP address of the load balancer frontend (203.30.3.100), as intended in the configuration. Previously, this DNS record pointed to the internal IP address of storageserver01.securelabsondemand.com, which was part of the initial setup before the load balancing configuration.

The successful resolution of `cscd.securelabsondemand.com` to the new IP address of the load balancer signifies that the DNS changes have propagated as expected. This confirms that the pfSense firewall, acting as the DNS resolver, is correctly directing traffic intended for the CSCD website to the HAProxy load balancer. Consequently, this setup ensures that incoming traffic to the CSCD website will be efficiently distributed across the redundant web servers, aligning with the high-availability and load balancing goals set for this lab exercise.

Figure 10

Make a screen capture showing the result of your DNS query for cscd.securelabsondemand.com.


Cure Strange Childhood Diseases
pfSense-dc.securelabsondema
172.16.0.1/services_unbound.php
Search
Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 19:21:24
Osamudiamen Eweka
General Setup should also be set to the prope
Administrator: Command Prompt
Host
cscd
Add
Domain Overrides
It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back!
Refresh Firefox...
Type here to search
4:21 PM
2/28/2024

```
Microsoft Windows [Version 10.0.17763.2090]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig /flushdns

Windows IP Configuration

Successfully flushed the DNS Resolver Cache.

C:\Users\Administrator>nslookup cscd.securelabsondemand.com
Server:  pfSense-dc.securelabsondemand.com
Address:  172.16.0.1

DNS request timed out.
    timeout was 2 seconds.
DNS request timed out.
    timeout was 2 seconds.
Name:    cscd.securelabsondemand.com
Address:  203.30.3.100


C:\Users\Administrator>_
```

Note. The screen capture reveals the outcome of a DNS query for the domain "cscd.securelabsondemand.com." (Jones & Bartlett, 2024).

Moving on the functionality of the load-balancing configuration was tested to ensure it operates as intended. The process began with clearing the DNS cache on the local system to ensure the updated DNS record for the CSCD website would be retrieved, reflecting the new load balancer frontend IP address. An nslookup command confirmed that the DNS record for cscd.securelabsondemand.com now resolves to 203.30.3.100, the IP assigned to the HAProxy frontend, signifying a successful redirection of the domain to the new load-balanced setup.

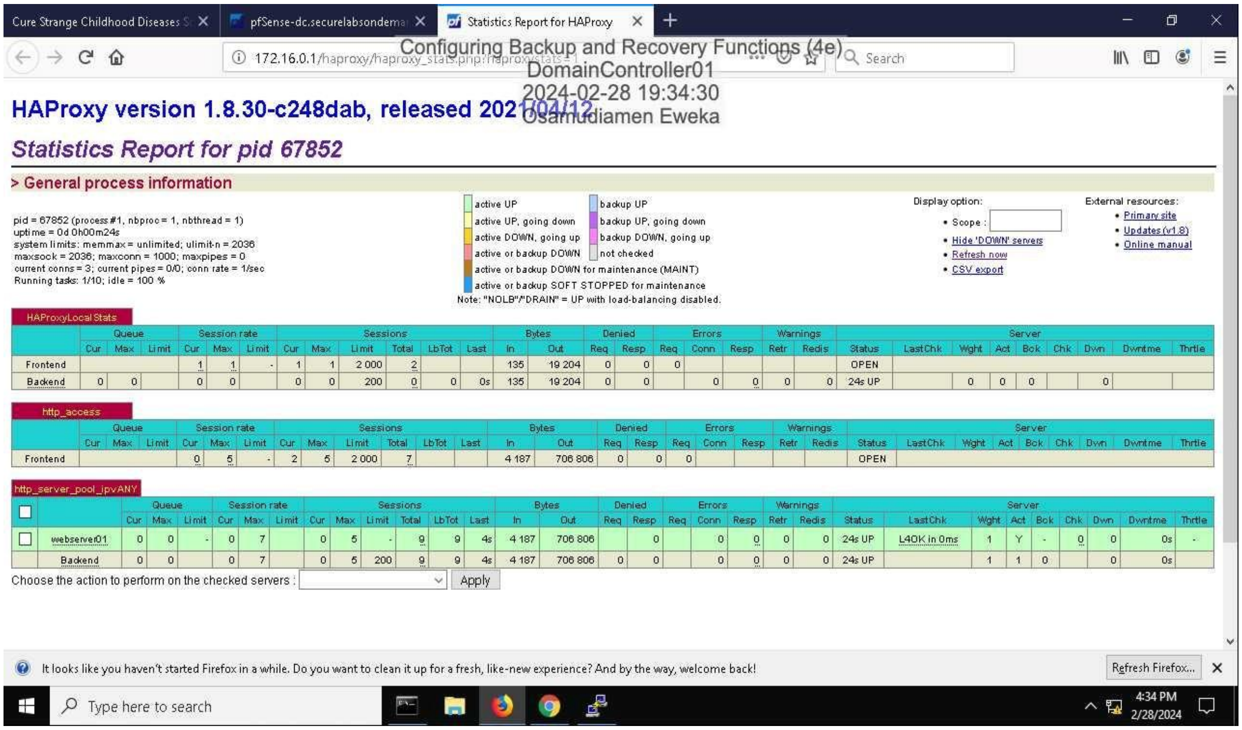
Next, to assess the effectiveness of the load balancing, the HAProxy statistics report was accessed via the pfSense webGUI. This report provides detailed insights into the traffic handled by HAProxy, including the number of sessions and the distribution of traffic across the server pool. Initially, the report showed zero sessions for both webserver01 and webserver02, indicating no traffic had yet been processed through the load balancer.

To generate traffic, the CSCD website was accessed multiple times from a browser. Refreshing the HAProxy statistics report afterward showed an increase in the "Sessions > Total" column for both webserver01 and webserver02. This change confirmed that the Static Round Robin method of load balancing was functioning correctly, distributing incoming HTTP requests evenly between the two servers as intended.

This practical exercise demonstrated the load balancer's ability to evenly distribute web traffic across multiple servers, enhancing the website's availability and reliability. The observation of sessions being split between webserver01 and webserver02 in the HAProxy statistics report served as evidence that the load-balancing setup was operational and performing as expected, marking a successful implementation of the load-balancing configuration in the lab environment.

Figure 11

Make a screen capture showing the Statistics Report with a value of at least 1 in the Sessions > Total column of the http_server_pool_ipvANY box, for both webserver01 and webserver02.



Cure Strange Childhood Diseases S
pfSense-dc.securelabsondema
Statistics Report for HAProxy
172.16.0.1/haproxy/haproxy_stats.php?haproxystats=1
Search
Type here to search
4:34 PM
2/28/2024

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 19:34:30
Osamudiamen Eweka

HAProxy version 1.8.30-c248dab, released 2021/04/12

Statistics Report for pid 67852

> General process information

pid = 67852 (process #1, nbproc = 1, nbthread = 1)
uptime = 0d 0h00m24s
system limits: memmax = unlimited; ulimit-n = 2036
maxsock = 2036; maxconn = 1000; maxpipes = 0
current conns = 3; current pipes = 0/0; conn rate = 1/sec
Running tasks: 1/10; idle = 100 %

🟩	active UP	▨	backup UP
🟨	active UP, going down	▨	backup UP, going down
🟧	active DOWN, going up	▨	backup DOWN, going up
🟥	active or backup DOWN	▨	not checked
🟫	active or backup DOWN for maintenance (MAINT)		
🟦	active or backup SOFT STOPPED for maintenance		

Note: "NOLB"/"DRAIN" = UP with load-balancing disabled.

Display option:
- Scope :
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:
- Primary site
- Updates (v1.8)
- Online manual

HAProxyLocalStats

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	1	-	1	1	2 000	2			135	19 204	0	0	0					OPEN								
Backend	0	0		0	0		0	0	200	0	0	0s	135	19 204	0	0		0	0	0	0	24s UP		0	0	0		0		

http_access

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	5	-	2	5	2 000	7			4 187	706 806	0	0	0					OPEN								

http_server_pool_ipvANY

☐	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
☐ webserver01	0	0	-	0	7		0	5	-	9	9	4s	4 187	706 806		0		0	0	0	0	24s UP	L4OK in 0ms	1	Y	-	0	0	0s	-
Backend	0	0		0	7		0	5	200	9	9	4s	4 187	706 806	0	0		0	0	0	0	24s UP		1	1	0		0	0s	

Choose the action to perform on the checked servers : Apply

It looks like you haven't started Firefox in a while. Do you want to clean it up for a fresh, like-new experience? And by the way, welcome back! Refresh Firefox...

Note. The screen capture presents the Statistics Report, specifically highlighting the "http_server_pool_ipvANY" section, where both "webserver01" and "webserver02" show a value of at least 1 in the "Sessions > Total" column (Jones & Bartlett, 2024). This data indicates active or completed sessions handled by these web servers, signifying that both servers are participating in the load balancing scheme and are actively processing HTTP requests.

Section 3: Challenge and Analysis

Part 1: Add Failover Functionality

To elevate the CSCD website deployment to a highly-available configuration, you were tasked with implementing health checks on the load-balanced web servers. This step is crucial for ensuring that traffic is only directed to operational servers, enhancing the reliability and uptime of the website. Health checks are automated tests performed by load balancers to verify the status of backend servers, ensuring they're ready and available to handle requests.

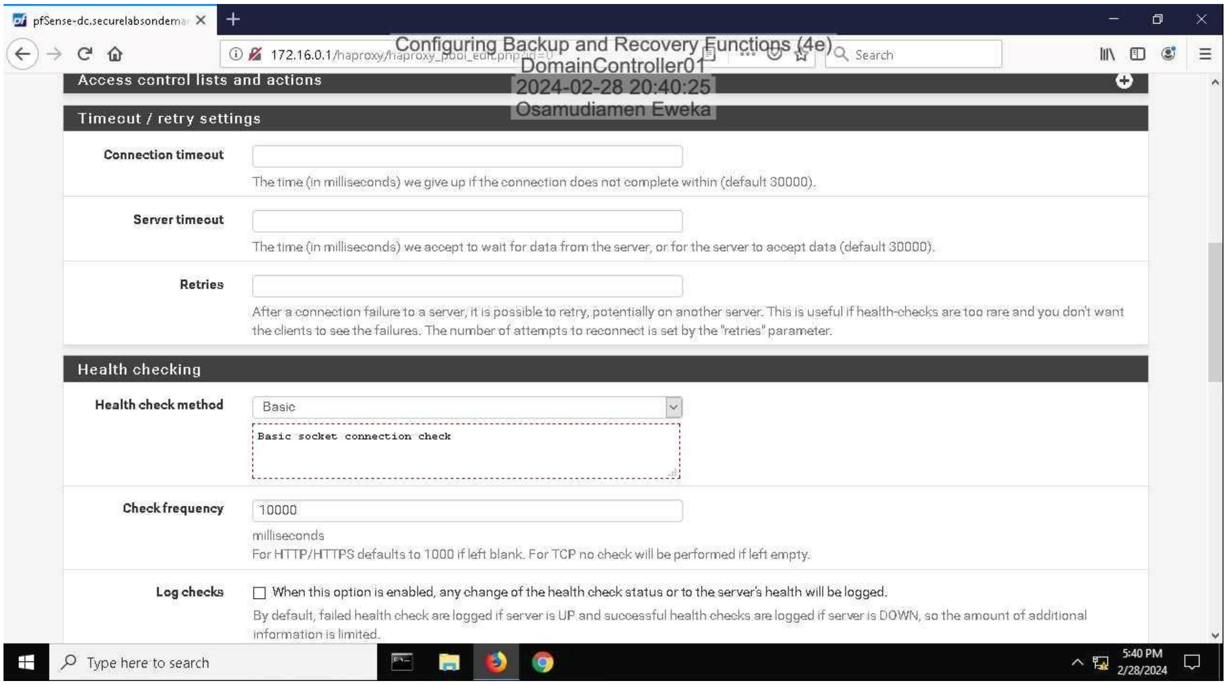
In the pfSense webGUI, you navigated to the HAProxy package settings to adjust the backend configuration, specifically focusing on the health checking mechanism for the web server nodes. The goal was to configure HAProxy to perform a Basic TCP health check on the servers every 10 seconds. This type of check attempts to establish a TCP connection with the server; if successful, the server is considered healthy and capable of serving traffic. If the check fails, the server is marked as down, and the load balancer stops directing traffic to it until it passes a health check again.

The Check frequency setting was crucial in this process. Initially set to perform no checks, you changed this setting to ensure a health check is conducted every 10 seconds. This frequency strikes a balance between being responsive to server outages and not overloading the servers with health check traffic. The configuration change was made by accessing the health checking module within the HAProxy backend settings and specifying the check interval.

By implementing these health checks, you've added a layer of fault tolerance to the CSCD website's infrastructure. Now, in the event a web server becomes unresponsive, HAProxy will automatically reroute traffic to the remaining healthy server(s), minimizing potential downtime and providing a seamless experience to end-users. This adjustment is a significant step towards achieving a high-availability deployment, as it ensures that the system can automatically detect and isolate failures, maintaining service availability even in the face of server issues.

Figure 12

Make a screen capture showing the updated Check Frequency value in the Health checking module.



pfSense-dc.securelabsondemar
172.16.0.1/haproxy/haproxy_pool_edit.php?id=0
Search
Type here to search
5:40 PM
2/28/2024

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 20:40:25
Osamudiamen Eweka

Access control lists and actions

Timeout / retry settings

Connection timeout

The time (in milliseconds) we give up if the connection does not complete within (default 30000).

Server timeout

The time (in milliseconds) we accept to wait for data from the server, or for the server to accept data (default 30000).

Retries

After a connection failure to a server, it is possible to retry, potentially on another server. This is useful if health-checks are too rare and you don't want the clients to see the failures. The number of attempts to reconnect is set by the "retries" parameter.

Health checking

Health check method: Basic

```
Basic socket connection check
```

Check frequency: 10000

milliseconds
For HTTP/HTTPS defaults to 1000 if left blank. For TCP no check will be performed if left empty.

Log checks: □ When this option is enabled, any change of the health check status or to the server's health will be logged.

By default, failed health check are logged if server is UP and successful health checks are logged if server is DOWN, so the amount of additional information is limited.

Note. the above screen capture of the updated Check Frequency value in the Health checking module shows the new 10-second interval setting (10 seconds equals 10000 milliseconds), reflecting the successful configuration of health checks for the CSCD website deployment (Jones & Bartlett, 2024).

Section 3: Challenge and Analysis

Part 2: Validate Failover Functionality

To test the effectiveness of the newly configured health checks within the HAProxy setup, an intentional failure was induced on one of the web servers. By accessing either WebServer01 or WebServer02 via PuTTY and executing the command `service nginx stop`, the web service on the selected server was shut down. This action mimics a server failure, triggering HAProxy's health check mechanism to identify the server as non-operational.

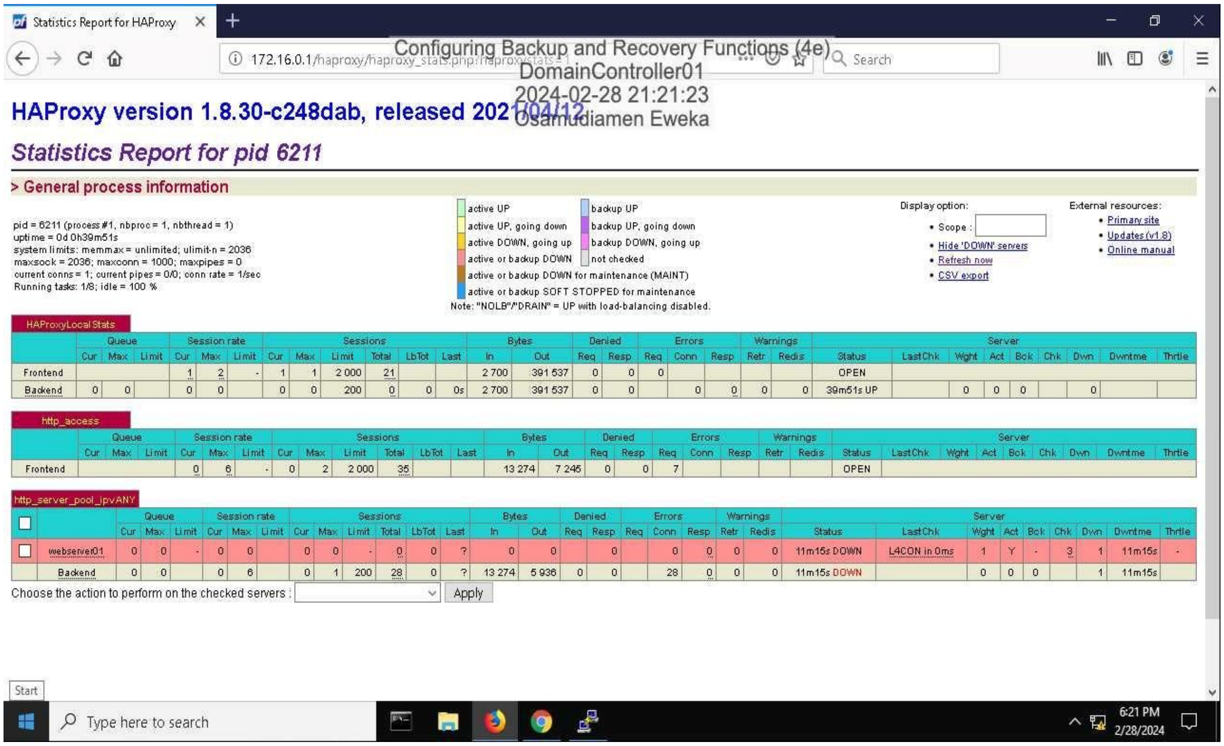
Upon visiting the HAProxy Statistics Report after a brief period, at least 10 seconds to allow for the health check cycle, one of the servers within the `http_server_pool` was indeed reported as down. This confirms the health check's functionality, as it successfully detected the lack of response from the stopped nginx service, marking the server as down in the HAProxy stats.

Subsequent attempts to access the CSCD website via a new Chrome browser tab resulted in the load balancer diverting all traffic away from the failed server to the remaining operational server. This behavior was evidenced in the HAProxy Statistics Report, where the session count for the downed server remained constant, indicating no new traffic was being directed to it. In contrast, the session count for the operational server continued to increase with each page reload, demonstrating the load balancer's effective rerouting of traffic to ensure uninterrupted service.

This exercise validates the high-availability configuration's resilience, ensuring that even in the event of a server failure, the CSCD website remains accessible to its users, with HAProxy seamlessly redirecting traffic to the healthy server.

Figure 13

Make a screen capture showing the HAProxy Statistics Report with a host in a DOWN state, as well as the UP host having more total sessions (http_server_pool_ipvANY, Sessions > Total) than the DOWN host.


Statistics Report for HAProxy
172.16.0.1/haproxy/haproxy_stats.php?haproxystats=1
Search
Type here to search
6:21 PM
2/28/2024
Start

Configuring Backup and Recovery Functions (4e)
DomainController01
2024-02-28 21:21:23
Osamudiamen Eweka

HAProxy version 1.8.30-c248dab, released 2021/04/12

Statistics Report for pid 6211

> General process information

pid = 6211 (process #1, nbproc = 1, nbthread = 1)
uptime = 0d 0h39m51s
system limits: memmax = unlimited; ulimit-n = 2036
maxsock = 2036; maxconn = 1000; maxpipes = 0
current conns = 1; current pipes = 0/0; conn rate = 1/sec
Running tasks: 1/8; idle = 100 %

active UP	backup UP
active UP, going down	backup UP, going down
active DOWN, going up	backup DOWN, going up
active or backup DOWN	not checked
active or backup DOWN for maintenance (MAINT)	
active or backup SOFT STOPPED for maintenance	

Note: "NOLB"/"DRAIN" = UP with load-balancing disabled.

Display option:

- Scope :
- Hide 'DOWN' servers
- Refresh now
- CSV export

External resources:

- Primary site
- Updates (v1.8)
- Online manual

HAProxyLocalStats

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				1	2	-	1	1	2 000	21			2 700	391 537	0	0	0					OPEN								
Backend	0	0		0	0		0	0	200	0	0	0s	2 700	391 537	0	0		0	0	0	0	39m51s UP		0	0	0		0		

http_access

	Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
	Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
Frontend				0	6	-	0	2	2 000	35			13 274	7 245	0	0	7					OPEN								

http_server_pool_ipvANY

□		Queue			Session rate			Sessions						Bytes		Denied		Errors			Warnings		Server								
		Cur	Max	Limit	Cur	Max	Limit	Cur	Max	Limit	Total	LbTot	Last	In	Out	Req	Resp	Req	Conn	Resp	Retr	Redis	Status	LastChk	Wght	Act	Bck	Chk	Dwn	Dwntme	Thrtle
□	webserver01	0	0	-	0	0		0	0	-	0	0	?	0	0		0		0	0	0	0	11m15s DOWN	L4CON in 0ms	1	Y	-	3	1	11m15s	-
	Backend	0	0		0	6		0	1	200	28	0	?	13 274	5 936	0	0		28	0	0	0	11m15s DOWN		0	0	0		1	11m15s	

Choose the action to perform on the checked servers : Apply

Note. The above screen capture of the HAProxy Statistics Report shows one server marked as DOWN and the other handling the increased load, visually confirming the successful failover mechanism in action (Jones & Bartlett, 2024).

Conclusion

In conclusion, the lab work provided a comprehensive understanding and hands-on experience with configuring backup and recovery functions critical for maintaining the availability component of the CIA triad within an organization's security program. Participants successfully installed the Windows Server Backup feature, configured daily System State backups, and restored a Domain Controller from a System State backup. Furthermore, the lab extended into configuring a Network File System (NFS) share, setting up load balancing for redundant web servers, and applying these configurations to ensure business continuity, disaster recovery, and high availability of services. Through these exercises, participants deepened their understanding of the relationship between business impact analysis, risk assessment, business continuity planning, and disaster recovery planning. The skills acquired in this lab are vital for securing information systems against disruptions, ensuring the resilience of critical operations against various risks.

Reference

Load Balancer Metrics. (n.d.). (C) Copyright 2024. https://docs.oracle.com/en-us/iaas/Content/Balance/Reference/loadbalancermetrics.htm

Jones & Bartlett (2024). Implementing Security Monitoring and Logging (Figures). *Jones and BartlettLearning Virtual Lab*. URL: https://jbl-lti.hatsize.com/startlab