



**IE4030**  
**Virtualization & Cloud Computing**  
**Technologies**  
**4<sup>th</sup> Year, 1<sup>st</sup> Semester**

Assignment

**Implementation of DNS, DHCP, and Proxy  
Servers using Docker Containers**

Submitted to  
Sri Lanka Institute of Information Technology

In partial fulfillment of the requirements for the  
Bachelor of Science Special Honors Degree in Information Technology

2025-10-05

## **Declaration**

I certify that this report does not incorporate without acknowledgement, any material previously submitted for a degree or diploma in any university, and to the best of my knowledge and belief it does not contain any material previously published or written by another person, except where due reference is made in text.

Registration Number: IT22218926

Name: B.S.Premathilaka

## Table of Contents

<b>1. Initial Setup .....</b>	5
1.1 Update the System Packages .....	5
1.2 Install Required Dependencies .....	5
1.3 Add Docker's Official GPG Key.....	6
1.4 Set Up the Docker Repository .....	6
1.5 Update the Package Index.....	6
1.6 Install Docker Engine .....	7
1.7 Add Current User to Docker Group.....	7
1.8 Verify Docker Installation .....	8
1.9 Login to Docker Hub .....	8
<b>2. DNS Server .....</b>	10
2.1 Create the DNS Project Directory.....	10
2.2 Create the Dockerfile .....	10
2.3 Configure the DNS Files.....	11
2.4 Build the DNS Docker Image.....	14
2.5 Run the DNS Container .....	14
2.6 Test the DNS Server .....	15
<b>3. DHCP Server .....</b>	16
3.1 Create the DHCP Project Directory .....	16
3.2 Create the Dockerfile .....	16
3.3 Create the DHCP Configuration File .....	17
3.4 Build the DHCP Docker Image .....	18
3.5 Run the DHCP Container.....	19
3.6 Test the DHCP Server from Client VM (Docker-Client) .....	20
<b>4. Proxy Server (Squid) .....</b>	21
4.1 Creating the Project Directory .....	21
4.2 Create the Dockerfile .....	21
4.3 Create the Squid Configuration File .....	22
4.4 Create the Entrypoint Script.....	23

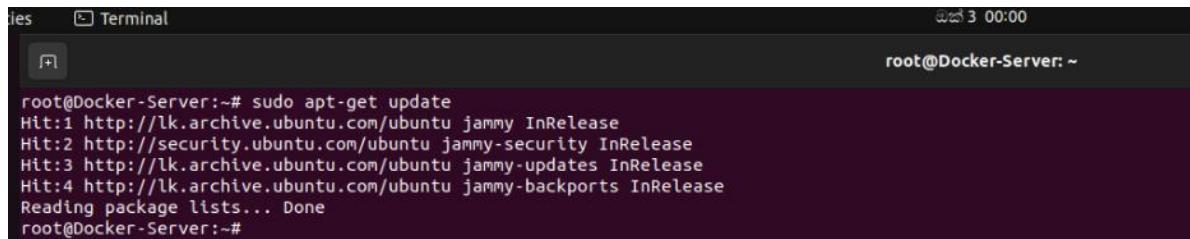
4.5 Build the Proxy Server Image.....	24
4.6 Run the Proxy Server Container .....	25
4.7 Verify Container Status and Logs.....	25
4.8 Test Proxy Functionality.....	25
<b>5. Docker Hub Commands .....</b>	<b>27</b>
5.1 Login to Docker Hub .....	27
5.2 Tag Images for Docker Hub .....	27
5.3 Push Images to Docker Hub .....	27
5.4 Remove Local Images.....	28
5.5 Pull Images from Docker Hub .....	28
5.6 Deploys all three services (DNS, DHCP, Proxy) from images stored on Docker Hub.....	29
<b>6. Verification Commands.....</b>	<b>30</b>
6.1 List All Running Containers .....	30
6.2 View Logs of All Services .....	30
6.3 Functional Testing from Client.....	31
<b>7. Bonus Task – Docker Compose .....</b>	<b>33</b>
7.1 Create docker-compose.yml .....	33
7.2 Deploy All Services Using Docker Compose.....	34
7.3 Check Status of All Services.....	34
7.4 Stop and Remove Services.....	35

# 1. Initial Setup

## 1.1 Update the System Packages

Before installing Docker, the system package list was updated to ensure all existing packages are current.

*sudo apt-get update*

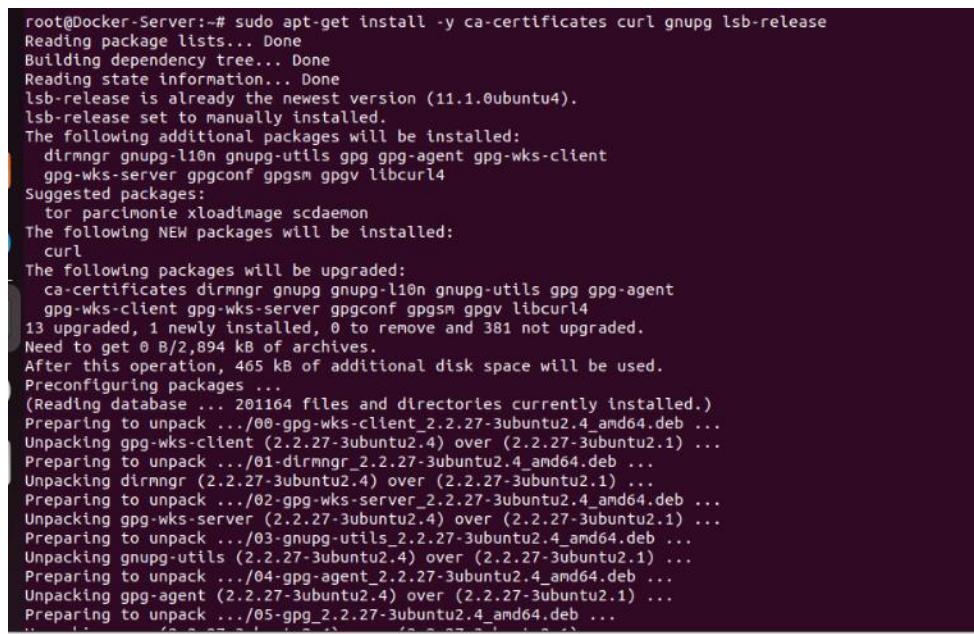


```
root@Docker-Server:~# sudo apt-get update
Hit:1 http://lk.archive.ubuntu.com/ubuntu jammy InRelease
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://lk.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:4 http://lk.archive.ubuntu.com/ubuntu jammy-backports InRelease
Reading package lists... Done
root@Docker-Server:~#
```

## 1.2 Install Required Dependencies

The following dependencies were installed to enable secure communication and repository management.

*sudo apt-get install -y ca-certificates curl gnupg lsb-release*



```
root@Docker-Server:~# sudo apt-get install -y ca-certificates curl gnupg lsb-release
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
lsb-release is already the newest version (11.1.0ubuntu4).
lsb-release set to manually installed.
The following additional packages will be installed:
  dirmngr gnupg-l10n gnupg-utils gpg gpg-agent gpg-wks-client
  gpg-wks-server gpgconf gpgsm gpgv libcurl4
Suggested packages:
  tor parcellite xloadimage scdaemon
The following NEW packages will be installed:
  curl
The following packages will be upgraded:
  ca-certificates dirmngr gnupg gnupg-l10n gnupg-utils gpg gpg-agent
  gpg-wks-client gpg-wks-server gpgconf gpgsm gpgv libcurl4
13 upgraded, 1 newly installed, 0 to remove and 381 not upgraded.
Need to get 0 B/2,894 kB of archives.
After this operation, 465 kB of additional disk space will be used.
Preconfiguring packages ...
(Reading database ... 20164 files and directories currently installed.)
Preparing to unpack .../00-gpg-wks-client_2.2.27-3ubuntu2.4_amd64.deb ...
Unpacking gpg-wks-client (2.2.27-3ubuntu2.4) over (2.2.27-3ubuntu2.1) ...
Preparing to unpack .../01-dirmngr_2.2.27-3ubuntu2.4_amd64.deb ...
Unpacking dirmngr (2.2.27-3ubuntu2.4) over (2.2.27-3ubuntu2.1) ...
Preparing to unpack .../02-gpg-wks-server_2.2.27-3ubuntu2.4_amd64.deb ...
Unpacking gpg-wks-server (2.2.27-3ubuntu2.4) over (2.2.27-3ubuntu2.1) ...
Preparing to unpack .../03-gnupg-utils_2.2.27-3ubuntu2.4_amd64.deb ...
Unpacking gnupg-utils (2.2.27-3ubuntu2.4) over (2.2.27-3ubuntu2.1) ...
Preparing to unpack .../04-gpg-agent_2.2.27-3ubuntu2.4_amd64.deb ...
Unpacking gpg-agent (2.2.27-3ubuntu2.4) over (2.2.27-3ubuntu2.1) ...
Preparing to unpack .../05-gpg_2.2.27-3ubuntu2.4_amd64.deb ...
```

### 1.3 Add Docker's Official GPG Key

A secure directory was created for storing the GPG key, and the Docker GPG key was downloaded and saved in that directory.

```
sudo install -m 0755 -d /etc/apt/keyrings
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

```
root@Docker-Server:~# 
root@Docker-Server:~# 
root@Docker-Server:~# sudo install -m 0755 -d /etc/apt/keyrings
root@Docker-Server:~# 
root@Docker-Server:~# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
root@Docker-Server:~# 
root@Docker-Server:~#
```

### 1.4 Set Up the Docker Repository

The official Docker repository was added to the system's APT sources list.

```
echo \ 
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] \
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | \
sudo tee /etc/apt/sources.list.d/docker.list >/dev/null
```

```
root@Docker-Server:~# 
root@Docker-Server:~# 
root@Docker-Server:~# echo \ 
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] \
https://download.docker.com/linux/ubuntu $(lsb_release -cs) stable" | \
> 
> sudo tee /etc/apt/sources.list.d/docker.list >/dev/null
root@Docker-Server:~#
```

### 1.5 Update the Package Index

After adding the Docker repository, the system package index was updated again to include the new Docker packages.

```
sudo apt-get update
```

```
root@Docker-Server:~# sudo apt-get update
Get:1 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Hit:2 http://security.ubuntu.com/ubuntu jammy-security InRelease
Hit:3 http://lk.archive.ubuntu.com/ubuntu jammy InRelease
Get:4 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [56.1 kB]
Hit:5 http://lk.archive.ubuntu.com/ubuntu jammy-updates InRelease
Hit:6 http://lk.archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 105 kB in 2s (50.1 kB/s)
Reading package lists... Done
root@Docker-Server:~# sudo apt-get install -y docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-buildx-plugin docker-ce-rootless-extras docker-compose-plugin
  git git-man liberror-perl libslirp0 pigz slirp4netns
Suggested packages:
  cgroups-mount | cgroup-lite docker-model-plugin git-daemon-run
  | git-daemon-sysvinit git-doc git-email git-gui gitkit gitweb git-cvs
  git-mediawiki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin git git-man
  ...
```

## 1.6 Install Docker Engine

Docker Engine, the Docker CLI, and the container runtime (Containerd) were installed using the following command:

```
sudo apt-get install -y docker-ce docker-ce-cli containerd.io
```

```
Reading package lists... Done
root@Docker-Server:~# sudo apt-get install -y docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  docker-buildx-plugin docker-ce-rootless-extras docker-compose-plugin
  git git-man liberror-perl libslirp0 pigz slirp4netns
Suggested packages:
  cgroupfs-mount | cgroup-lite docker-model-plugin git-daemon-run
  | git-daemon-sysvinit git-doc git-email git-gui gitk gitweb git-cvs
  git-mediatlki git-svn
The following NEW packages will be installed:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli
  docker-ce-rootless-extras docker-compose-plugin git git-man
  liberror-perl libslirp0 pigz slirp4netns
0 upgraded, 12 newly installed, 0 to remove and 381 not upgraded.
Need to get 109 MB of archives.
After this operation, 457 MB of additional disk space will be used.
Get:1 https://download.docker.com/linux/ubuntu jammy/stable amd64 containerd.io amd64 1.7.28-0~ubuntu.22.04-jammy [31.9 MB]
Get:2 http://lk.archive.ubuntu.com/ubuntu jammy/universe amd64 pigz amd64 2.6.1 [63.6 kB]
Get:3 http://lk.archive.ubuntu.com/ubuntu jammy/main amd64 liberror-perl all 0.17029-1 [26.5 kB]
Get:4 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git-man all 1:2.34.1-1ubuntu1.15 [955 kB]
Get:5 http://lk.archive.ubuntu.com/ubuntu jammy-updates/main amd64 git amd64 1:2.34.1-1ubuntu1.15 [3,166 kB]
Get:6 http://lk.archive.ubuntu.com/ubuntu jammy/main amd64 libslirp0 amd64 4.6.1-1build1 [61.5 kB]
Get:7 http://lk.archive.ubuntu.com/ubuntu jammy/universe amd64 slirp4netns amd64 1.0.1-2 [28.2 kB]
Get:8 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-cli amd64 5:28.4.0-1ubuntu.22.04-jammy [16.5 MB]
Get:9 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce amd64 5:28.4.0-1ubuntu.22.04-jammy [19.7 MB]
Get:10 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-buildx-plugin amd64 0.29.0-0~ubuntu.22.04-jammy [15.9 MB]
Get:11 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-ce-rootless-extras amd64 5:28.4.0-1ubuntu.22.04-jammy [6,479 kB]
Get:12 https://download.docker.com/linux/ubuntu jammy/stable amd64 docker-compose-plugin amd64 2.39.4-0~ubuntu.22.04-jammy [14.2 MB]
Fetched 109 MB in 1min 4s (1.702 kB/s)
Selecting previously unselected package containerd.io.
(Reading database ... 201180 files and directories currently installed.)
Preparing to unpack .../00-containerd.io_1.7.28-0~ubuntu.22.04-jammy_amd64.deb ...
..
```

## 1.7 Add Current User to Docker Group

To allow Docker commands to be run without using sudo, the current user was added to the Docker group.

```
sudo usermod -aG docker $USER
newgrp docker
```

```
root@Docker-Server:~#
root@Docker-Server:~# sudo usermod -aG docker $USER
root@Docker-Server:~# newgrp docker
root@Docker-Server:~#
```

## 1.8 Verify Docker Installation

The Docker installation was verified by checking the installed version and confirming the successful setup of Docker Engine.

*docker version*

*systemctl status docker*

```
root@Docker-Server:~# docker version
Client: Docker Engine - Community
  Version:           28.4.0
  API version:      1.51
  Go version:       go1.24.7
  Git commit:       d8eb465
  Built:            Wed Sep  3 20:57:05 2025
  OS/Arch:          linux/amd64
  Context:          default

Server: Docker Engine - Community
  Engine:
    Version:          28.4.0
    API version:     1.51 (minimum version 1.24)
    Go version:      go1.24.7
    Git commit:      249d679
    Built:           Wed Sep  3 20:57:05 2025
    OS/Arch:         linux/amd64
    Experimental:   false
  containerd:
    Version:          v1.7.28
    GitCommit:        b98a3aace656320842a23f4a392a33f46af97866
  runc:
    Version:          1.3.0
    GitCommit:        v1.3.0-0-g4ca628d1
  docker-init:
    Version:          0.19.0
    GitCommit:        de40ad0
root@Docker-Server:~# systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor >
   Active: active (running) since Thu 2025-10-02 23:55:42 +0530; 1min 5s
     Docs: https://docs.docker.com
TriggeredBy: ● docker.socket
             Main PID: 10369 (dockerd)
```

## 1.9 Login to Docker Hub

A Docker Hub account was created, and the login was performed through the terminal using the following command:

*docker login*

```
root@bhagya-virtual-machine:~#
root@bhagya-virtual-machine:~# docker login -u bhugz
Password:
WARNING! Your password will be stored unencrypted in /root/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credential-stores

Login Succeeded
root@bhagya-virtual-machine:~#
```

The screenshot shows the Docker Home interface for the user 'bhugz'. The top navigation bar includes links for Home, Hub, Build Cloud, Scout, Testcontainers Cloud, Docker Desktop, Settings, and Billing. A prominent blue banner at the top right says 'Welcome to Docker Home, bhugz' and provides links to 'Get started with Docker guidance' and 'Learn about Docker concepts'. Below the banner, there's a section titled 'Docker products' featuring three cards: 'docker.desktop' (Innovate with Docker Desktop), 'buildcloud' (Build with Docker Build Cloud), and 'scout' (Secure with Docker Scout). Each card has a 'Go to download' button. On the far right of the 'scout' card, there's a link to 'Give feedback'.

## 2. DNS Server

### 2.1 Create the DNS Project Directory

A separate directory was created to maintain the DNS configuration files and Docker setup in an organized manner. This helps to manage and isolate DNS-related files from other service configurations such as DHCP and Squid.

```
mkdir -p ~/vcct/dns  
cd ~/vcct/dns
```

```
root@Docker-Server:~#  
root@Docker-Server:~# mkdir -p ~/vcct/dns  
root@Docker-Server:~# cd ~/vcct/dns  
root@Docker-Server:~/vcct/dns#
```

### 2.2 Create the Dockerfile

A new Dockerfile was created to define the environment for the DNS server container.

*nano Dockerfile*

The following configuration was added:

```
FROM ubuntu:22.04  
RUN apt-get update && apt-get install -y bind9 bind9utils dnsutils && rm -rf  
/var/lib/apt/lists/*  
COPY named.conf.options /etc/bind/named.conf.options  
COPY named.conf.local /etc/bind/named.conf.local  
COPY db.csne.vcct.com /etc/bind/db.csne.vcct.com  
EXPOSE 53/udp 53/tcp  
CMD ["named", "-g", "-c", "/etc/bind/named.conf"]
```

FROM ubuntu:22.04: Uses the official Ubuntu base image.

RUN: Installs bind9, bind9utils, and dnsutils.

COPY: Copies custom DNS configuration files into the container.

EXPOSE: Opens ports 53/UDP and 53/TCP for DNS communication.

CMD: Launches the BIND9 service in the foreground.

```
root@Docker-Server:~/vcct/dns#
root@Docker-Server:~/vcct/dns# nano Dockerfile
root@Docker-Server:~/vcct/dns#
```

```
GNU nano 6.2                               Dockerfile
FROM ubuntu:22.04
RUN apt-get update && apt-get install -y bind9 bindutils dnsutils && rm -rf /var/lib/apt/lists/*
COPY named.conf.options /etc/bind/named.conf.options
COPY named.conf.local /etc/bind/named.conf.local
COPY named.ca /etc/bind/named.ca
COPY db.csne.vcct.com /etc/bind/db.csne.vcct.com
EXPOSE 53/udp 53/tcp
CMD ["named", "-g", "-c", "/etc/bind/named.conf"]
```

The terminal window shows the command `nano Dockerfile` being run. The Dockerfile content is displayed, including the `FROM`, `RUN`, `COPY`, `EXPOSE`, and `CMD` instructions. The nano editor interface is visible at the bottom of the screen.

## 2.3 Configure the DNS Files

### A) *named.conf.options*

This file defines the DNS server's global options and access permissions.

#### *nano named.conf.options*

Add the following configuration:

```
options {
    directory "/var/cache/bind";
    recursion yes;
    allow-query { 192.168.8.0/24; 127.0.0.1; };
    listen-on { any; };
    listen-on-v6 { none; };
    dnssec-validation no;
    auth-nxdomain no;
};
```

Allows DNS queries only from the local subnet (192.168.8.0/24) and localhost.

Disables IPv6 and DNSSEC for simplicity.

```
root@Docker-Server:~/vcct/dns#  
root@Docker-Server:~/vcct/dns# nano named.conf.options  
root@Docker-Server:~/vcct/dns#
```

```
GNU nano 6.2                                         named.conf.options

options {
    directory '/var/cache/bind';
    recursion yes;
    allow-query { 192.168.8.0/24; 127.0.0.1; };
    listen-on { any; };
    listen-on-v6 { none; };
    dnssec-validation no;
    auth-nxdomain no;
};

[Read 10 lines]  Location  M-U Undo  M-A Set Mark  M-] To Bracket  M-O Previous  M-B Back
^C Help        M-W Next      M-C Copy     M-[ Where Was  M-W Forward  M-F Forward
^X Exit        M-R Read File  M-I Replace  M-K Cut      M-N Execute  M-J Justify  M-Z Go To Line  M-E Redo
```

### B) *named.conf.local*

This file defines the local DNS zone for the custom domain.

#### *nano named.conf.local*

Add the following:

```
zone "csne.vcct.com" {  
    type master;  
    file "/etc/bind/db.csne.vcct.com";  
};
```

Instructs the BIND9 server to load the custom zone file db.csne.vcct.com for the domain csne.vcct.com.

```
root@Docker-Server:~/vcct/dns#  
root@Docker-Server:~/vcct/dns# nano named.conf.local  
root@Docker-Server:~/vcct/dns#
```

```
GNU nano 6.2                                         named.conf.local
zone "csne.vcct.com" {
    type master;
    file "/etc/bind/db.csne.vcct.com";
};
```

File menu    Edit menu    View menu    Insert menu    Format menu    Search menu    Help menu

Help    Exit    Write Out    Where Is    Cut    Execute    Read 5 lines ]    Location    Undo    Set Mark    To Bracket    Previous    Back

Read File    Replace    Paste    Justify    Go To Line    Redo    Copy    Where Was    Next    Forward

### C) db.csne.vcct.com

This file defines the actual DNS records for the domain.

**nano db.csne.vcct.com**

Add the following content:

```
$TTL 1h
@ IN SOA ns1.csne.vcct.com. admin.csne.vcct.com. (
    2025092301 ; serial
    1h      ; refresh
    15m    ; retry
    1w      ; expire
    1h )   ; minimum
```

**IN NS ns1.csne.vcct.com.**

**ns1 IN A 192.168.8.2**

**@ IN A 192.168.8.132**

The domain csne.vcct.com resolves to 192.168.8.132.

The nameserver (ns1.csne.vcct.com) is defined as 192.168.8.2.

```

root@Docker-Server:~/vcct/dns# nano db.csne.vcct.com
root@Docker-Server:~/vcct/dns# 
root@Docker-Server:~/vcct/dns# 

GNU nano 6.2                               db.csne.vcct.com
$TTL 1h
@    IN SOA ns1.csne.vcct.com. admin.csne.vcct.com. (
        2025092301 ; serial
        1h         ; refresh
        15m       ; retry
        1w         ; expire
        1h )       ; minimum
IN NS    ns1.csne.vcct.com.
ns1   IN A     192.168.8.2
@    IN A     192.168.8.132

```

Floppy Disk

File Edit View Insert Format Tools Help

Help Write Out Read File Replace Cut Paste Execute Justify Location Go To Line Undo Redo Set Mark Copy To Bracket Where Was Previous Next Back Forward

## 2.4 Build the DNS Docker Image

The Docker image was built using the custom Dockerfile.

*docker build -t bhugz/dns-bind:1.0 .*

```

root@Docker-Server:~/vcct/dns# docker build -t bhugz/dns-bind:1.0 .
[+] Building 2.2s (10/10) FINISHED
--> transferring dockerfile: 57B
--> [internal] load metadata for docker.io/library/ubuntu:22.04
--> [internal] load .dockerrunignore
--> [internal] load build context: 2B
--> [internal] load build context: 11B
--> [internal] transfer context: 11B
--> [1/3] FROM docker.io/library/ubuntu:22.04@sha256:c35ed21f8af2d6f79e671ab55318d9bc6ff966d5d827fb852181f1fb15b64567e37
--> CACHED 1/3: RUN apt-get update && apt-get install vbind9-dbg=9.10.6-0.1+deb10u10
--> CACHED 1/3: COPY named.conf.options /etc/bind/named.conf.options
--> CACHED 1/3: COPY named.conf.local /etc/bind/named.conf.local
--> CACHED 1/3: COPY db.csne.vcct.com /etc/bind/db.csne.vcct.com
--> CACHED 1/3: chown -R bind:bind .
--> CACHED 1/3: chmod 755 db.csne.vcct.com
--> writing Tarfile sha256:29c7ca8dc3939a7648bf79a371b709f247e926c04455b263647eabc0ce384
--> naming to docker.io/bhugz/dns-bind:1.0

```

## 2.5 Run the DNS Container

The DNS server container was launched using the following command:

*docker run -d --name dns \
-p 53:53/udp -p 53:53/tcp \
--restart unless-stopped \
bhugz/dns-bind:1.0*

Runs the container in detached mode (-d).

Maps both TCP and UDP port 53 to the host.

Configures automatic restart policy for reliability.

DNS uses UDP 53 for standard queries and TCP 53 for large responses or zone transfers.

```

dns
root@Docker-Server:~/vcct/dns# docker run -d --name dns \
-p 53:53/udp -p 53:53/tcp \
--restart unless-stopped \
bhugz/dns-bind:1.0
7ccccdeee7ed6403c9c545b1e1950d2d4e5b138f95357be43e44ebc74ad2449b

```

## 2.6 Test the DNS Server

From the Docker Server;

*docker ps*  
*dig @192.168.8.2 csne.vcct.com*

```

root@Docker-Server:~/vcct/dns# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
7ccccdeee7ed bhugz/dns-bind:1.0 "named -g -c /etc/bl..." 6 seconds ago Up 6 seconds 0.0.0.0:53->53/tcp, 0.0.0.0:53->53/udp, [::]:53->53/tcp, [::]:53->53/udp dns

; <>> DIG 9.18.18-0ubuntu0.22.04.2-Ubuntu <>> @192.168.8.2 vcct.csne.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 53665
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: udp: 1232
; COOKIE: 3549da7c7d9ec360100000068dec9268798e3f738a5b212 (good)
;; QUESTION SECTION:
;vcct.csne.com. IN A

;; ANSWER SECTION:
vcct.csne.com. 3600 IN A 13.248.169.48
vcct.csne.com. 3600 IN A 76.223.54.146

;; Query time: 552 msec
;; SERVER: 192.168.8.2#53(192.168.8.2) (UDP)
;; WHEN: Fri Oct 03 00:19:10 +0530 2025
;; MSG SIZE rcvd: 102

```

### 3. DHCP Server

#### 3.1 Create the DHCP Project Directory

A dedicated directory was created to store all files related to the DHCP configuration. Organizes DHCP configuration files separately, similar to the structure used for DNS and Squid server configurations.

```
mkdir -p ~/vcct/dhcp  
cd ~/vcct/dhcp
```

```
root@Docker-Server:~#  
root@Docker-Server:~# mkdir -p ~/vcct/dhcp  
root@Docker-Server:~# cd ~/vcct/dhcp  
root@Docker-Server:~/vcct/dhcp#
```

#### 3.2 Create the Dockerfile

A Dockerfile was created to build a containerized version of the ISC DHCP server.

```
nano Dockerfile
```

The following content was added:

```
FROM ubuntu:22.04  
RUN apt-get update && apt-get install -y isc-dhcp-server iproute2 && rm -rf  
/var/lib/apt/lists/*  
RUN mkdir -p /var/lib/dhcp && touch /var/lib/dhcp/dhcpd.leases  
COPY dhcpd.conf /etc/dhcp/dhcpd.conf  
EXPOSE 67/udp  
CMD ["sh", "-c", "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf"]
```

FROM ubuntu:22.04: Uses Ubuntu 22.04 as the base image.

RUN: Installs isc-dhcp-server and iproute2, creates the required leases directory and file.

COPY: Adds the DHCP configuration file into the container.

EXPOSE: Indicates UDP port 67 for DHCP, though host networking bypasses this.

CMD: Runs the DHCP daemon in the foreground for logging and debugging.

```

root@Docker-Server:~/vcct/dhcp#
root@Docker-Server:~/vcct/dhcp# nano Dockerfile
root@Docker-Server:~/vcct/dhcp#



GNU nano 6.2
FROM ubuntu:22.04
RUN apt-get update && apt-get install -y tsc-dhcp-server iproute2 && rm -rf /var/lib/apt/lists/*
# dhclient requires a leases file present
RUN mkdir -p /var/lib/dhcp && touch /var/lib/dhcp/dhcpd.leases
COPY dhcpd.conf /etc/dhcp/dhcpd.conf
# Expose ix_informational; host mode will bypass it
EXPOSE 67/udp
# Default: foreground, debug, IPv4, use our config
CMD ["sh", "-c", "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf"]

```

File menu: Dockerfile

Help menu:

- PG Help
- PF Exit
- PO Write Out
- AM Where Is
- AN Replace
- AK Cut
- AL Paste
- MT Execute
- AJ Justify
- AC Location [Read 10 lines]
- CU Undo
- HA Set Mark
- H-U To Bracket
- CO Copy
- HE Redo
- HO Previous
- HK Next
- AB Back
- AF Forward

### 3.3 Create the DHCP Configuration File

The DHCP configuration file defines the IP address pool and network parameters for the subnet.

#### *nano dhcpd.conf*

The following configuration was added:

```

default-lease-time 43200; # 12 hours
max-lease-time 86400; # 24 hours
authoritative;

```

```

subnet 192.168.8.0 netmask 255.255.255.0 {
    range 192.168.8.130 192.168.8.160;
    option routers 192.168.8.2;
    option subnet-mask 255.255.255.0;
    option domain-name "csne.vcct.com";
    option domain-name-servers 192.168.8.2;
}

```

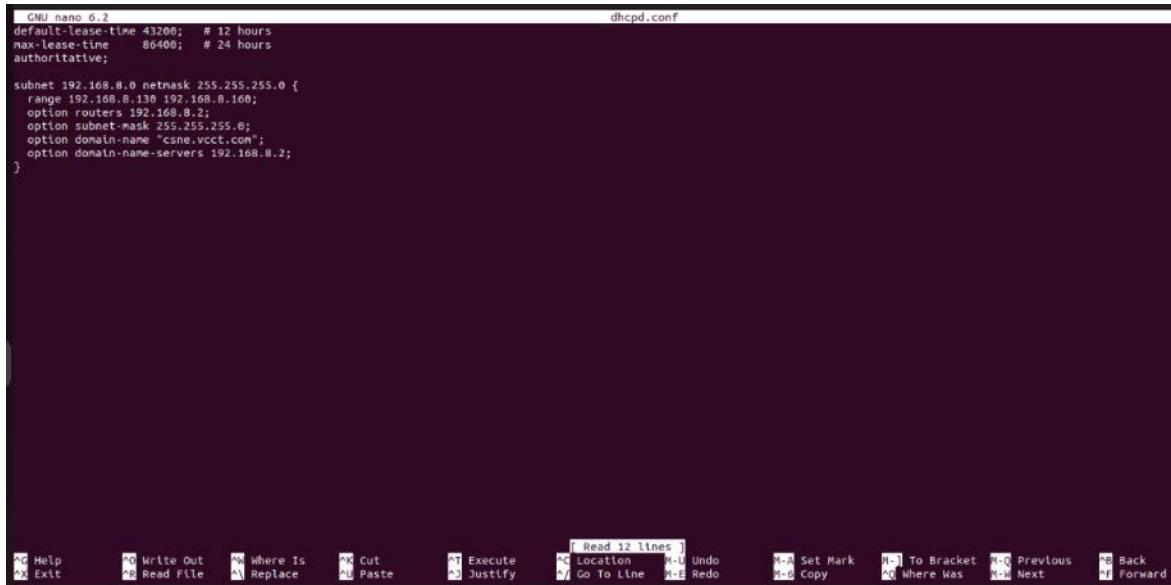
Defines a subnet of 192.168.8.0/24.

Assigns dynamic IP addresses in the range 192.168.8.130 – 192.168.8.160.

Sets 192.168.8.2 as the default gateway and DNS server.

Marks this DHCP server as authoritative for the network.

```
root@Docker-Server:~/vcct/dhcp#  
root@Docker-Server:~/vcct/dhcp# nano dhcpcd.conf  
root@Docker-Server:~/vcct/dhcp#
```



```
GNU nano 6.2                                         dhcpcd.conf  
default-lease-time 43200; # 12 hours  
max-lease-time 86400; # 24 hours  
authoritative;  
  
subnet 192.168.8.0 netmask 255.255.255.0 {  
    range 192.168.8.100 192.168.8.160;  
    option routers 192.168.8.2;  
    option subnet-mask 255.255.255.0;  
    option domain-name "csne.vcct.com";  
    option domain-name-servers 192.168.8.2;  
}  
  
[Read 12 lines]
```

The terminal window shows the command `nano dhcpcd.conf` being run. The file content is displayed in the main pane, and the bottom pane shows the nano editor's status bar with various keyboard shortcut keys.

### 3.4 Build the DHCP Docker Image

The Docker image was built using the following command:

***docker build -t bhugz/dhcp-isc:1.0 .***

```
root@Docker-Server:~/vcct/dhcp#  
root@Docker-Server:~/vcct/dhcp# docker build -t bhugz/dhcp-isc:1.0 .  
[+] Building 39.3s (9/9) FINISHED                               docker:default  
=> [internal] load build definition from Dockerfile           0.0s  
=> => transferring dockerfile: 471B                           0.0s  
=> [internal] load metadata for docker.io/library/ubuntu:22.04 3.6s  
=> [internal] load .dockerignore                            0.0s  
=> => transferring context: 2B                           0.0s  
=> CACHED [1/4] FROM docker.io/library/ubuntu:22.04@sha256:c35ed11 0.0s  
=> [internal] load build context                          0.1s  
=> => transferring context: 362B                         0.0s  
=> [2/4] RUN apt-get update && apt-get install -y isc-dhcp-server 34.5s  
=> [3/4] RUN mkdir -p /var/lib/dhcp && touch /var/lib/dhcp/dhcpd.l 0.7s  
=> [4/4] COPY dhcpcd.conf /etc/dhcp/dhcpcd.conf          0.1s  
=> exporting to image                                     0.2s  
=> => exporting layers                                    0.2s  
=> => writing image sha256:45112d8ceb695adf9b5e1009305ecf098e487af 0.0s  
=> => naming to docker.io/bhugz/dhcp-isc:1.0            0.0s  
root@Docker-Server:~/vcct/dhcp#  
root@Docker-Server:~/vcct/dhcp#
```

### 3.5 Run the DHCP Container

Before running a new container, any previously running DHCP container was removed:

```
docker rm -f dhcp 2>/dev/null || true
```

```
root@Docker-Server:~/vcct/dhcp#  
root@Docker-Server:~/vcct/dhcp# docker rm -f dhcp 2>/dev/null || true  
root@Docker-Server:~/vcct/dhcp#
```

Then, the new container was started in host networking mode to allow it to handle broadcast requests:

```
sudo docker run -d --name dhcp  
--net=host  
--cap-add=NET_ADMIN  
--restart unless-stopped  
bhugz/dhcp-isc:1.0 sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"
```

--net=host: Allows direct access to host network interfaces.

--cap-add=NET\_ADMIN: Grants administrative privileges to manage network settings.

--restart unless-stopped: Ensures the container restarts automatically after reboots.

Ens37: Interface name

```
root@Docker-Server:~/vcct/dhcp# sudo docker run -d --name dhcp --net=host --cap-add=NET_ADMIN --restart unless-stopped bhugz/dhcp-isc:1.0 sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"  
193434d2daacc4a5cbd34329a8fd735bd746cd3d81847ed7c9ce79bd111e42f6  
root@Docker-Server:~/vcct/dhcp#
```

To verify that the container is running:

```
docker ps
```

```
root@Docker-Server:~/vcct/dhcp#  
root@Docker-Server:~/vcct/dhcp# docker ps  
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES  
193434d2daac bhugz/dhcp-lsc:1.0 "sh -c 'dhcpd -4 -f ..'" 14 seconds ago Up 13 seconds  
7cccddeee7ed bhugz/dns-bind:1.0 "named -g -c /etc/bi.." 19 minutes ago Up 19 minutes 0.0.0.0:53->53/tcp, 0.0.0.0:53->53/udp, [::]:53->53/tcp, [::]:53->53/udp dns  
root@Docker-Server:~/vcct/dhcp#
```

To check DHCP logs:

```
docker logs dhcp --tail 50
```

```
docker: command not found  
root@Docker-Server:~/vcct/dhcp# docker logs dhcp --tail 50  
Internet Systems Consortium DHCP Server 4.4.1  
Copyright 2004-2018 Internet Systems Consortium.  
All rights reserved.  
For info, please visit https://www.isc.org/software/dhcp/  
Config file: /etc/dhcp/dhcpd.conf  
Database file: /var/lib/dhcp/dhcpd.leases  
PID file: /var/run/dhcpd.pid  
Wrote 0 leases to leases file.  
Multiple interfaces match the same subnet: ens33 ens37  
Multiple interfaces match the same shared network: ens33 ens37  
Multiple interfaces match the same subnet: ens33 ens37  
Multiple interfaces match the same shared network: ens33 ens37  
Multiple interfaces match the same subnet: ens33 ens37  
Multiple interfaces match the same shared network: ens33 ens37  
Listening on LPF/ens37/00:0c:29:03:b2:68/192.168.8.0/24  
Sending on  LPF/ens37/00:0c:29:03:b2:68/192.168.8.0/24  
Sending on  Socket/fallback/fallback-net  
Server starting service.  
root@Docker-Server:~/vcct/dhcp#
```

### 3.6 Test the DHCP Server from Client VM (Docker-Client)

On the client virtual machine connected to vmnet2, network settings were configured for Automatic (DHCP).

Once the DHCP lease was obtained, connectivity to the server and DNS functionality were verified using:

```
dig @192.168.8.2 csne.vcct.com  
sudo dhclient -v ens37
```

```
bhagya@Docker-Client:~$  
bhagya@Docker-Client:~$ sudo su  
[sudo] password for bhagya:  
root@Docker-Client:/home/bhagya#  
root@Docker-Client:/home/bhagya#  
root@Docker-Client:/home/bhagya# dig @192.168.8.2 csne.vcct.com  
  
; <>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <>> @192.168.8.2 csne.vcct.com  
; (1 server found)  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50973  
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1  
  
;; OPT PSEUDOSECTION:  
; EDNS: version: 0, flags:; udp: 1232  
; COOKIE: c56947ce73b3dc140100000068ded7a9db77217c885c7864 (good)  
; QUESTION SECTION:  
;csne.vcct.com. IN A  
  
;; ANSWER SECTION:  
csne.vcct.com. 3600 IN A 192.168.8.132  
  
;; Query time: 0 msec  
;; SERVER: 192.168.8.2#53(192.168.8.2) (UDP)  
;; WHEN: Fri Oct 03 01:21:06 +0530 2025  
;; MSG SIZE rcvd: 86  
  
root@Docker-Client:/home/bhagya#  
root@Docker-Client:/home/bhagya#
```

```
root@Docker-Client:/home/bhagya# sudo dhclient -v ens37  
Internet Systems Consortium DHCP Client 4.4.1  
Copyright 2004-2018 Internet Systems Consortium.  
All rights reserved.  
For info, please visit https://www.isc.org/software/dhcp/  
  
Listening on LPF/ens37/00:0c:29:4d:a3:28  
Sending on LPF/ens37/00:0c:29:4d:a3:28  
Sending on Socket/fallback  
DHCPDISCOVER on ens37 to 255.255.255.255 port 67 interval 3 (xid=0x3aca1d3a)  
DHCPOffer of 192.168.8.131 from 192.168.8.2  
DHCPREQUEST for 192.168.8.131 on ens37 to 255.255.255.255 port 67 (xid=0x3a1dca3a)  
DHCPACK of 192.168.8.131 from 192.168.8.2 (xid=0x3aca1d3a)  
bound to 192.168.8.131 -- renewal in 20056 seconds.  
root@Docker-Client:/home/bhagya#  
root@Docker-Client:/home/bhagya#
```

## 4. Proxy Server (Squid)

### 4.1 Creating the Project Directory

Creates a separate directory to store all proxy-related configuration and build files.

```
mkdir -p ~/vcct/proxy-server  
cd ~/vcct/proxy-server
```

```
root@Docker-Server:~/vcct# mkdir proxy-server  
root@Docker-Server:~/vcct# ls  
dhcp dns proxy-server  
root@Docker-Server:~/vcct# cd proxy-server  
root@Docker-Server:~/vcct/proxy-server#  
root@Docker-Server:~/vcct/proxy-server#
```

### 4.2 Create the Dockerfile

Defines how the Squid image is built and installs required packages, copies configuration files, initializes cache directories, and sets /entrypoint.sh as the startup script.

```
nano Dockerfile
```

The following configuration was added:

```
FROM debian:bullseye-slim  
ENV DEBIAN_FRONTEND=noninteractive  
RUN apt-get update \  
    && apt-get install -y --no-install-recommends squid ca-certificates curl \  
    && rm -rf /var/lib/apt/lists/*  
COPY squid.conf /etc/squid/squid.conf  
COPY entrypoint.sh /entrypoint.sh  
RUN mkdir -p /var/log/squid /run/squid /var/spool/squid \  
    && chown -R proxy:proxy /var/log/squid /run/squid /var/spool/squid  
EXPOSE 3128  
RUN mkdir -p /var/spool/squid && chown -R proxy:proxy /var/spool/squid  
RUN squid -Nz -f /etc/squid/squid.conf  
ENTRYPOINT ["/entrypoint.sh"]
```

```
root@Docker-Server:~/vcct/proxy-server#  
root@Docker-Server:~/vcct/proxy-server# nano Dockerfile  
root@Docker-Server:~/vcct/proxy-server#  
root@Docker-Server:~/vcct/proxy-server#
```

```
GNU nano 6.2  
FROM debian:bullseye-slim  
ENV DEBIAN_FRONTEND=noninteractive  
  
RUN apt-get update \  
    && apt-get install -y --no-install-recommends squid ca-certificates curl \  
    && rm -rf /var/lib/apt/lists/*  
  
COPY squid.conf /etc/squid/squid.conf  
COPY entrypoint.sh /entrypoint.sh  
  
RUN mkdir -p /var/log/squid /run/squid /var/spool/squid \  
    && chown -R proxy:proxy /var/log/squid /run/squid /var/spool/squid  
  
EXPOSE 3128  
  
# Create the cache directory and set correct ownership  
RUN mkdir -p /var/spool/squid && chown -R proxy:proxy /var/spool/squid  
  
# Initialize the cache directory  
RUN squid -Nz -f /etc/squid/squid.conf  
  
ENTRYPOINT ["/entrypoint.sh"]
```

### 4.3 Create the Squid Configuration File

Configures Squid to listen on port 3128, allow requests from the local network, cache up to 100 MB of content, and log activities.

*nano squid.conf*

The following configuration was added:

```
http_port 0.0.0.0:3128  
acl localnet src 192.168.8.0/24  
acl localnet src 172.18.0.0/16  
acl localhost src 127.0.0.1/32 ::1  
acl SSL_ports port 443  
acl Safe_ports port 80 443  
acl CONNECT method CONNECT  
http_access deny !Safe_ports  
http_access deny CONNECT !SSL_ports  
http_access allow localhost  
http_access allow localnet  
http_access deny all  
cache_mem 128 MB  
maximum_object_size_in_memory 512 KB  
cache_dir ufs /var/spool/squid 100 16 256  
maximum_object_size 4 MB  
refresh_pattern . 0 20% 4320
```

```

access_log stdio:/var/log/squid/access.log
cache_log stdio:/var/log/squid/cache.log
dns_nameservers 8.8.8.8 8.8.4.4
max_filedescriptors 65536
coredump_dir /var/spool/squid

```

```

root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# nano squid.conf
root@Docker-Server:~/vcct/proxy-server#

```

```

GNU nano 6.2
http_port 0.0.0.0:3128

# ACLs
acl localnet src 192.168.8.0/24
acl localnet src 172.18.0.0/16    # add Docker internal subnet
acl localhost src 127.0.0.1/32 ::1
acl SSL_ports port 443
acl Safe_ports port 80 443
acl CONNECT method CONNECT

http_access deny !safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localhost
http_access allow localnet
http_access deny all

# Memory cache
cache_mem 128 MB
maximum_object_size_in_memory 512 KB

# Disk cache (100MB)
cache_dir ufs /var/spool/squid 100 16 256
maximum_object_size 4 MB

# Refresh patterns
refresh_pattern . 0 20% 4320

# Logs
access_log stdio:/var/log/squid/access.log
cache_log stdio:/var/log/squid/cache.log

# Stability / DNS
dns_nameservers 8.8.8.8 8.8.4.4
max_filedescriptors 65536
coredump_dir /var/spool/squid

```

#### 4.4 Create the Entrypoint Script

Ensures required directories exist with correct permissions before starting Squid in foreground mode inside the container.

##### *nano entrypoint.sh*

The following configuration was added:

```

#!/bin/sh
set -e
mkdir -p /run/squid /var/log/squid /var/spool/squid
chown -R proxy:proxy /run/squid /var/log/squid /var/spool/squid
rm -f /run/squid.pid
exec /usr/sbin/squid -N -f /etc/squid/squid.conf -d 1

```

```
root@Docker-Server:~/vcct/proxy-server# nano entrypoint.sh
root@Docker-Server:~/vcct/proxy-server#
```

```
GNU nano 6.2
#!/bin/sh
set -e
mkdir -p /run/squid /var/log/squid /var/spool/squid
chown -R proxy:proxy /run/squid /var/log/squid /var/spool/squid
rm -f /run/squid.pid
exec /usr/sbin/squid -N -f /etc/squid/squid.conf -d 1
```

Make the script executable:

```
chmod +x entrypoint.sh
```

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# chmod +x entrypoint.sh
root@Docker-Server:~/vcct/proxy-server#
```

## 4.5 Build the Proxy Server Image

Builds the Docker image tagged proxy-squid:1.0. Replace yourhub with your Docker Hub username.

```
docker build -t yourhub/proxy-squid:1.0 .
```

```
root@Docker-Server:~/vcct/proxy-server# docker build -t bhugz/proxy-squid:1.0 .
[+] Building 6.6s (12/12) FINISHED
          docker:default
=> [internal] load build definition from Dockerfile          0.0s
=> => transferring dockerfile: 674B                         0.0s
=> [internal] load metadata for docker.io/library/debian:bullseye- 2.8s
=> [internal] load .dockerignore                           0.0s
=> => transferring context: 2B                            0.0s
=> [1/7] FROM docker.io/library/debian:bullseye-slim@sha256:f807f4 0.0s
=> [internal] load build context                         0.0s
=> => transferring context: 280B                         0.0s
=> CACHED [2/7] RUN apt-get update && apt-get install -y --no-ins 0.0s
=> CACHED [3/7] COPY squid.conf /etc/squid/squid.conf      0.0s
=> [4/7] COPY entrypoint.sh /entrypoint.sh                 0.3s
=> [5/7] RUN mkdir -p /var/log/squid /run/squid /var/spool/squid 0.5s
=> [6/7] RUN mkdir -p /var/spool/squid && chown -R proxy:proxy /va 0.7s
=> [7/7] RUN squid -Nz -f /etc/squid/squid.conf           1.5s
=> exporting to image                                     0.7s
=> => exporting layers                                    0.7s
=> => writing image sha256:5053ceea3036be927b9336ab985a4c683c0d77e 0.0s
=> => naming to docker.io/bhugz/proxy-squid:1.0          0.0s
```

## 4.6 Run the Proxy Server Container

Launches the Squid container in host networking mode so it can listen directly on the host interface. The NET\_ADMIN capability permits network-related operations.

```
sudo docker run -d --name proxy \
--net=host \
--cap-add=NET_ADMIN \
--restart unless-stopped \
bhugz/proxy-squid:1.0 sh -c "squid -N -f/etc/squid/squid.conf -d 1"
```

```
root@Docker-Server:~/vcct/proxy-server# docker run -d --name proxy -p 3128:3128 --restart unless-stopped bhugz/proxy-squid:1.0
2048b22085cd02b484d42534664407501318e0c6becc804c0d549c53bd0cf62c
root@Docker-Server:~/vcct/proxy-server#
```

## 4.7 Verify Container Status and Logs

Confirms that the proxy container is running and shows recent Squid log output.

```
docker ps
```

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
2048b22085cd bhugz/proxy-squid:1.0 "/entrypoint.sh" 23 seconds ago Up 22 seconds 0.0.0.0:3128->3128/tcp, [::]:3128->3128/tcp
193434d2daac bhugz/dhcp-lsc:1.0 "sh -c 'dhcpd -4 -f ..." 2 days ago Up 32 minutes dhcp
7ccccdeee7ed bhugz/dns-bind:1.0 "named -g -c /etc/bi..." 2 days ago Up 32 minutes 0.0.0.0:53->53/tcp, 0.0.0.0:53->53/udp, [::]:53->53/tcp, [::]:53->53/udp dns
root@Docker-Server:~/vcct/proxy-server#
```

## 4.8 Test Proxy Functionality

Tests the proxy by sending an HTTP request through it.

```
curl -I --proxy 192.168.8.2:3128 http://example.com
```

From Server:

```
root@Docker-Server:~/vcct/proxy-server# curl -I --proxy 192.168.8.2:3128 http://google.com
HTTP/1.1 301 Moved Permanently
Location: http://www.google.com/
Content-Type: text/html; charset=UTF-8
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-5ElhemuGn9X9RhaFqKVmyA' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https: http;;report-uri https://csp.withgoogle.com/csp/gws/other-hp
Date: Sun, 05 Oct 2025 11:30:14 GMT
Expires: Tue, 04 Nov 2025 11:30:14 GMT
Cache-Control: public, max-age=2592000
Server: gws
Content-Length: 219
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Age: 4
X-Cache: HIT from Docker-Server
X-Cache-Lookup: HIT from Docker-Server:3128
Via: 1.1 Docker-Server (squid/4.13)
Connection: keep-alive
Show Applications
root@Docker-Server:~/vcct/proxy-server#
```

## From Client:

```
root@Docker-Client:/home/bhagya# ping 192.168.8.2
PING 192.168.8.2 (192.168.8.2) 56(84) bytes of data.
64 bytes from 192.168.8.2: icmp_seq=1 ttl=64 time=1.13 ns
64 bytes from 192.168.8.2: icmp_seq=2 ttl=64 time=2.14 ns
^C
-- 192.168.8.2 ping statistics --
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.128/1.634/2.141/0.506 ms
root@Docker-Client:/home/bhagya#
root@Docker-Client:/home/bhagya#
root@Docker-Client:/home/bhagya# curl -I --proxy 192.168.8.2:3128 http://google.com
HTTP/1.1 301 Moved Permanently
Location: http://www.google.com/
Content-Type: text/html; charset=UTF-8
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-SELhemuGn9X9RhaFqKVnyA' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https
: http;report-uri https://csp.wthg.google.com/csp/gws/other-hp
Date: Sun, 05 Oct 2025 11:30:14 GMT
Expires: Tue, 04 Nov 2025 11:30:14 GMT
Cache-Control: public, max-age=2592000
Server: gws
Content-Length: 219
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Age: 1108
X-Cache: HIT from Docker-Server
X-Cache-Lookup: HIT from Docker-Server:3128
VIA: 1.1 Docker-Server (squid/4.13)
Connection: keep-alive
root@Docker-Client:/home/bhagya#
```

## 5. Docker Hub Commands

### 5.1 Login to Docker Hub

Authenticates your local Docker client with your Docker Hub account.

***docker login***

```
root@Docker-Server:~/vcct/proxy-server# 
root@Docker-Server:~/vcct/proxy-server# docker login -u bhugz

[ Info → A Personal Access Token (PAT) can be used instead.  
To create a PAT, visit https://app.docker.com/settings

Password:
Login Succeeded
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
```

### 5.2 Tag Images for Docker Hub

Prepares local images with the correct repository namespace for uploading.

***docker tag dns-bind:1.0 bhugz/dns-bind:v1***

***docker tag dhcp-isc:1.0 bhugz/dhcp-isc:v1***

***docker tag proxy-squid:1.0 bhugz/proxy-squid:v1***

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker tag bhugz/dns-bind:1.0 bhugz/dns-bind:v1
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker tag bhugz/dhcp-isc:1.0 bhugz/dhcp-isc:v1
root@Docker-Server:~/vcct/proxy-server# docker tag bhugz/proxy-squid:1.0 bhugz/proxy-squid:v1
root@Docker-Server:~/vcct/proxy-server#
```

### 5.3 Push Images to Docker Hub

Uploads the DNS, DHCP, and Proxy images to Docker Hub repositories.

***docker push yourhub/dns-bind:v1***

***docker push yourhub/dhcp-isc:v1***

***docker push yourhub/proxy-squid:v1***

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker push bhugz/dns-bind:v1
The push refers to repository [docker.io/bhugz/dns-bind]
e22263de5530: Pushed
f93336bd2276: Pushed
95326fc46b9e: Pushed
c357d25c3c20: Pushed
767e56ba346a: Pushed
v1: digest: sha256:449f6db93087a16832cb40be9db1272b99b62b1a9e09759bc1375af8d0e9b524 size: 1362
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker push bhugz/dhcp-isc:v1
The push refers to repository [docker.io/bhugz/dhcp-isc]
a4e0ce0fecf7: Pushed
53cff010fe72: Pushed
8ed4512e479d: Pushed
767e56ba346a: Mounted from bhugz/dns-bind
v1: digest: sha256:47253cee3010747d23a3441ca0ee762ffa4e6f6157413a00e9e3477232010769 size: 1154
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker push bhugz/proxy-squid:v1
The push refers to repository [docker.io/bhugz/proxy-squid]
2027ff5608cd: Pushed
5e796fd44810: Pushed
ae2884ba4c0c: Pushed
58452f4d2169: Pushed
303dbeee10f1: Pushed
bab48d7f14ac: Pushed
a90cccd363f64: Mounted from library/debian
v1: digest: sha256:ab2cfeaa3d11632ec7d1df1428e96fdf4995b37ec64aac9270c31bf426c2293 size: 1778
root@Docker-Server:~/vcct/proxy-server#
```

Displays all locally stored images with their tags and sizes.

### *docker images*

```
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
7beb94f38f99 bhugz/proxy-squid:1.0 "/entrypoint.sh sh -" 5 hours ago Up 54 minutes
193434d2daac bhugz/dhcp-isc:1.0 "sh -c 'dhcpd -4 -f ..." 2 days ago Up 44 minutes
7ccccdeee7ed bhugz/dns-bind:1.0 "named -g -c /etc/bi..." 2 days ago Up 54 minutes 0.0.0.0:53->53/tcp, 0.0.0.0:53->53/udp, [::]:53->53/tcp, [::]:53->53/udp dns
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
```

## 5.4 Remove Local Images

Deletes local image copies to ensure future pulls come from Docker Hub.

*docker rm -f bhugz/dns-bind:v1*

*docker rm -f bhugz/dhcp-isc:v1*

*docker rm -f bhugz/proxy-squid:v1*

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker rm -f 7ccccdeee7ed
7ccccdeee7ed
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
7beb94f38f99 bhugz/proxy-squid:1.0 "/entrypoint.sh sh -" 5 hours ago Up 56 minutes proxy
193434d2daac bhugz/dhcp-isc:1.0 "sh -c 'dhcpd -4 -f ..." 2 days ago Up 46 minutes dhcp
root@Docker-Server:~/vcct/proxy-server# docker rm -f 193434d2daac
193434d2daac
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
7beb94f38f99 bhugz/proxy-squid:1.0 "/entrypoint.sh sh -" 5 hours ago Up 57 minutes proxy
root@Docker-Server:~/vcct/proxy-server# docker rm -f 7beb94f38f99
7beb94f38f99
```

## 5.5 Pull Images from Docker Hub

Downloads the images back from Docker Hub to verify successful upload.

*docker pull bhugz/dns-bind:v1*

*docker pull bhugz/dhcp-isc:v1*

*docker pull bhugz/proxy-squid:v1*

```
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker pull bhugz/dns-bind:v1
v1: Pulling from bhugz/dns-bind
Digest: sha256:449f6db93087a16832cb40be9db1272b99b62b1a9e09759bc1375af8d0e9b524
Status: Image is up to date for bhugz/dns-bind:v1
docker.io/bhugz/dns-bind:v1
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker pull bhugz/dhcp-isc:v1
v1: Pulling from bhugz/dhcp-isc
Digest: sha256:47253cee3010747d23a3441ca0ee762ffa4e6f6157413a00e9e3477232010769
Status: Image is up to date for bhugz/dhcp-isc:v1
docker.io/bhugz/dhcp-isc:v1
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker pull bhugz/proxy-squid:v1
v1: Pulling from bhugz/proxy-squid
Digest: sha256:ab2cfeaa36d11632ec7d1df1428e96dfd4995b37ec64aac9270c31bf426c2293
Status: Image is up to date for bhugz/proxy-squid:v1
docker.io/bhugz/proxy-squid:v1
root@Docker-Server:~/vcct/proxy-server#
```

## 5.6 Deploys all three services (DNS, DHCP, Proxy) from images stored on Docker Hub.

```
sudo docker run -d --name dns \
```

```
--net=host \
```

```
--cap-add=NET_ADMIN \
```

```
--restart unless-stopped \
```

```
bhugz /dns-bind:v1
```

```
sudo docker run -d --name dhcp \
```

```
--net=host \
```

```
--cap-add=NET_ADMIN \
```

```
--restart unless-stopped \
```

```
bhugz /dhcp-isc:v1 sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"
```

```
sudo docker run -d --name proxy \
```

```
--net=host \
```

```
--cap-add=NET_ADMIN \
```

```
--restart unless-stopped \
```

```
bhugz /proxy-squid:v1 sh -c "squid -N -f /etc/squid/squid.conf -d 1"
```

```
root@Docker-Server:~/vcct/proxy-server# 
root@Docker-Server:~/vcct/proxy-server# sudo docker run -d --name dns \
--net=host \
--cap-add=NET_ADMIN \
--restart unless-stopped \
bhugz/dns-bind:v1
863a56ccdd69943e40868b162c3f1890c7d330778ebe559977411cb083127837
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# sudo docker run -d --name dhcp \
--net=host \
--cap-add=NET_ADMIN \
--restart unless-stopped \
bhugz/dhcp-isc:v1 sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"
6403e35ce37bac939edb39da1edb949bb393cfbc87f89899f32d4acedacf1b
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# sudo docker run -d --name proxy \
--net=host \
--cap-add=NET_ADMIN \
--restart unless-stopped \
bhugz/proxy-squid:v1 sh -c "squid -N -f /etc/squid/squid.conf -d 1"
b8e28a6a628f30ed234d4127b2f3dfcb150a77876f19c0d60dfffb261a52f2d27
root@Docker-Server:~/vcct/proxy-server#
```

## 6. Verification Commands

## 6.1 List All Running Containers

Confirms that all service containers are running.

*docker ps*

```
root@Docker-Server:~/vcct/proxy-server# docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
b8e28a6a28f bhugz/proxy-squid:v1 "/entrypoint.sh sh -" 17 seconds ago Up 17 seconds proxy
6403e35c37b bhugz/dhcp-isc:v1 "sh -c 'dhcpd -4 -f ..." About a minute ago Up About a minute dhcp
863a56ccdd69 bhugz/dns-bind:v1 "named -g -c /etc/bi..." 2 minutes ago Up 2 minutes dns
root@Docker-Server:~/vcct/proxy-server#
```

## 6.2 View Logs of All Services

Displays recent log entries for each container to ensure successful startup.

*docker logs dns --tail 30*

*docker logs dhcp --tail 30*

```
docker logs proxy --tail 30
```

```

root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker logs dhcp --tail 30
Database file: /var/lib/dhcp/dhcpd.leases
PID file: /var/run/dhcpd.pid
Wrote 0 leases to leases file.
Listening on LPF/ens37/00:0c:29:03:b2:68/192.168.8.0/24
Sending on  LPF/ens37/00:0c:29:03:b2:68/192.168.8.0/24
Sending on  Socket/fallback/fallback-net
Server starting service.
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 11 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 31 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 52 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 68 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 86 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 106 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
reuse_lease: lease age 124 (secs) under 25% threshold, reply with unaltered, existing lease for 192.168.8.131
DHCPREQUEST for 192.168.8.131 from 00:0c:29:4d:a3:28 (Docker-Client) via ens37
DHCPACK on 192.168.8.131 to 00:0c:29:4d:a3:28 (Docker-Client) via ens37
root@Docker-Server:~/vcct/proxy-server#

```

```

root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# docker logs proxy --tail 30
2025/10/05 16:27:44] Swap maxSize 102400 + 131072 KB, estimated 17959 objects
2025/10/05 16:27:44] Target number of buckets: 897
2025/10/05 16:27:44] Using 8192 Store buckets
2025/10/05 16:27:44] Max Mem size: 131072 KB
2025/10/05 16:27:44] Max Swap size: 102400 KB
2025/10/05 16:27:44] Rebuilding storage in /var/spool/squid (no log)
2025/10/05 16:27:44] Using Least Load store dir selection
2025/10/05 16:27:44] Set Current Directory to /var/spool/squid
2025/10/05 16:27:44] Finished loading MIME types and icons.
2025/10/05 16:27:44] HTTP Disabled.
2025/10/05 16:27:44] Pinger socket opened on FD 18
2025/10/05 16:27:44] Squid plugin modules loaded: 0
2025/10/05 16:27:44] Adaptation support is off.
2025/10/05 16:27:44] Accepting HTTP Socket connections at local=0.0.0.0:3128 remote=[::] FD 16 flags=9
2025/10/05 16:27:44] Done scanning /var/spool/squid dir (0 entries)
2025/10/05 16:27:44] Finished rebuilding storage from disk.
2025/10/05 16:27:44]          0 Entries scanned
2025/10/05 16:27:44]          0 Invalid entries.
2025/10/05 16:27:44]          0 With invalid flags.
2025/10/05 16:27:44]          0 Objects loaded.
2025/10/05 16:27:44]          0 Objects expired.
2025/10/05 16:27:44]          0 Objects cancelled.
2025/10/05 16:27:44]          0 Duplicate URLs purged.
2025/10/05 16:27:44]          0 Swapfile clashes avoided.
2025/10/05 16:27:44] Took 0.10 seconds ( 0.00 objects/sec).
2025/10/05 16:27:44] Beginning Validation Procedure
2025/10/05 16:27:44] Completed Validation Procedure
2025/10/05 16:27:44] Validated 0 Entries
2025/10/05 16:27:44] store_swap_size = 0.00 KB
2025/10/05 16:27:45] storeLateRelease: released 0 objects
root@Docker-Server:~/vcct/proxy-server#

```

### 6.3 Functional Testing from Client

Tests connectivity (ICMP), DNS resolution, and proxy access from a client system.

*ping -c 3 192.168.8.2  
dig @192.168.8.2 csne.vcct.com  
curl -I --proxy 192.168.8.2:3128 http://google.com*

```

root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# ping -c 3 192.168.8.2
PING 192.168.8.2 (192.168.8.2) 56(84) bytes of data.
64 bytes from 192.168.8.2: icmp_seq=1 ttl=64 time=0.164 ms
64 bytes from 192.168.8.2: icmp_seq=2 ttl=64 time=0.121 ms
64 bytes from 192.168.8.2: icmp_seq=3 ttl=64 time=0.126 ms

--- 192.168.8.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2081ms
rtt min/avg/max/mdev = 0.121/0.137/0.164/0.019 ms
root@Docker-Server:~/vcct/proxy-server# ^C
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server# dig @192.168.8.2 csne.vcct.com

; <>> DiG 9.18.30-0ubuntu0.22.04.2-Ubuntu <>> @192.168.8.2 csne.vcct.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 11892
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 1232
; COOKIE: 5845e7e151752e4e0100000068e29d271785f7df59d132b8 (good)
;; QUESTION SECTION:
;csne.vcct.com.           IN      A

;; ANSWER SECTION:
csne.vcct.com.       3600    IN      A       192.168.8.132

;; Query time: 1 msec
;; SERVER: 192.168.8.2#53(192.168.8.2) (UDP)
;; WHEN: Sun Oct 05 22:00:31 +0530 2025
;; MSG SIZE rcvd: 86

root@Docker-Server:~/vcct/proxy-server#

```

```

root@Docker-Server:~/vcct/proxy-server# curl -I --proxy 192.168.8.2:3128 http://google.com
HTTP/1.1 301 Moved Permanently
Location: http://www.google.com/
Content-Type: text/html; charset=UTF-8
Content-Security-Policy-Report-Only: object-src 'none';base-uri 'self';script-src 'nonce-AUjdwOX_SQLvKdINzoSew' 'strict-dynamic' 'report-sample' 'unsafe-eval' 'unsafe-inline' https : http;report-uri https://csp.withgoogle.com/csp/gws/other-hp
Date: Sun, 05 Oct 2025 16:31:53 GMT
Expires: Tue, 04 Nov 2025 16:31:53 GMT
Cache-Control: public, max-age=2592000
Server: gws
Content-Length: 219
X-XSS-Protection: 0
X-Frame-Options: SAMEORIGIN
Age: 3
X-Cache: HIT from Docker-Server
X-Cache-Lookup: HIT from Docker-Server:3128
Via: 1.1 Docker-Server (squid/4.13)
Connection: keep-alive

root@Docker-Server:~/vcct/proxy-server#
root@Docker-Server:~/vcct/proxy-server#

```

## 7. Bonus Task – Docker Compose

### 7.1 Create docker-compose.yml

Defines all three services and their configurations for simultaneous deployment.

*nano docker-compose.yml*

```
root@Docker-Server:~/vcct# nano docker-compose.yml
root@Docker-Server:~/vcct#
```

The following configuration was added:

*version: '3.8'*

*services:*

*dns:*

```
image: bhugz/dns-bind:v1
network_mode: host
cap_add:
- NET_ADMIN
restart: unless-stopped
```

*dhcp:*

```
image: bhugz/dhcp-isc:v1
network_mode: host
cap_add:
- NET_ADMIN
restart: unless-stopped
command: sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"
```

*proxy:*

```
image: bhugz/proxy-squid:v1
network_mode: host
cap_add:
- NET_ADMIN
restart: unless-stopped
command: sh -c "squid -N -f /etc/squid/squid.conf -d 1"
```

```

GNU nano 6.2
Version: '3.8'

services:
  dns:
    image: bhugz/dns-bind:v1
    network_mode: host
    cap_add:
      - NET_ADMIN
    restart: unless-stopped

  dhcp:
    image: bhugz/dhcp-isc:v1
    network_mode: host
    cap_add:
      - NET_ADMIN
    restart: unless-stopped
    command: sh -c "dhcpd -4 -f -d -cf /etc/dhcp/dhcpd.conf ens37"

  proxy:
    image: bhugz/proxy-squid:v1
    network_mode: host
    cap_add:
      - NET_ADMIN
    restart: unless-stopped
    command: sh -c "squid -N -f /etc/squid/squid.conf -d 1"

```

## 7.2 Deploy All Services Using Docker Compose

Starts all containers defined in the Compose file in detached mode.

*sudo docker compose up -d*

```

root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct# sudo docker compose up -d
WARN[0000] /root/vcct/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 3/3
  ✓ Container vcct-dns-1    Started          0.45s
  ✓ Container vcct-proxy-1   Started          0.45s
  ✓ Container vcct-dhcp-1   Started          0.35s
root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct#

```

## 7.3 Check Status of All Services

Displays the current running state of each service container.

*sudo docker compose ps*

```

root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct# sudo docker compose ps
WARN[0000] /root/vcct/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
NAME           IMAGE             COMMAND            SERVICE          CREATED         STATUS          PORTS
vcct-dhcp-1   bhugz/dhcp-isc:v1 "sh -c 'dhcpd -4 -f ..."  dhcp            28 seconds ago Up 27 seconds
vcct-dns-1    bhugz/dns-bind:v1 "named -g -c /etc/bu..." dns             28 seconds ago Up 27 seconds
vcct-proxy-1   bhugz/proxy-squid:v1 "/entrypoint.sh ..." proxy           28 seconds ago Restarting (1) 6 seconds ago
root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct# sudo docker compose logs --tail 50
WARN[0000] /root/vcct/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
proxy-1 | 2025/10/05 16:35:12| Closing Pinger socket on FD 18
proxy-1 | 2025/10/05 16:35:38| WARNING: (B) '127.0.0.1' is a subnetwork of (A) '127.0.0.1'
proxy-1 | 2025/10/05 16:35:38| WARNING: because of this '127.0.0.1' is ignored to keep splay tree searching predictable
proxy-1 | 2025/10/05 16:35:38| WARNING: You should probably remove '127.0.0.1' from the ACL named 'localhost'
proxy-1 | 2025/10/05 16:35:38| WARNING: (B) '127.0.0.1' is a subnetwork of (A) '127.0.0.1'
proxy-1 | 2025/10/05 16:35:38| WARNING: because of this '127.0.0.1' is ignored to keep splay tree searching predictable
proxy-1 | 2025/10/05 16:35:38| WARNING: You should probably remove '127.0.0.1' from the ACL named 'localhost'
proxy-1 | 2025/10/05 16:35:38| WARNING: (B) '::1' is a subnetwork of (A) '::1'
proxy-1 | 2025/10/05 16:35:38| WARNING: because of this '::1' is ignored to keep splay tree searching predictable
proxy-1 | 2025/10/05 16:35:38| WARNING: You should probably remove '::1' from the ACL named 'localhost'
proxy-1 | 2025/10/05 16:35:38| WARNING: (B) '::1' is a subnetwork of (A) '::1'
proxy-1 | 2025/10/05 16:35:38| WARNING: because of this '::1' is ignored to keep splay tree searching predictable
proxy-1 | 2025/10/05 16:35:38| WARNING: You should probably remove '::1' from the ACL named 'localhost'
proxy-1 | 2025/10/05 16:35:38| Created PID file (/run/squid.pid)
proxy-1 | 2025/10/05 16:35:38| Set Current Directory to /var/spool/squid
proxy-1 | 2025/10/05 16:35:38| Starting Squid Cache version 4.13 for x86_64-pc-linux-gnu...
proxy-1 | 2025/10/05 16:35:38| Service Name: squid
proxy-1 | 2025/10/05 16:35:38| Process ID 1
proxy-1 | 2025/10/05 16:35:38| Process Roles: master worker
proxy-1 | 2025/10/05 16:35:38| With 65536 file descriptors available

```

## 7.4 Stop and Remove Services

Gracefully stops and removes all containers created by the Compose deployment.

***sudo docker compose down***

```
root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct# sudo docker compose down
WARN[0000] /root/vcct/docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion
[+] Running 3/3
✓ Container vcct-dhcp-1  Removed          10.1s
✓ Container vcct-dns-1   Removed          0.1s
✓ Container vcct-proxy-1 Removed          0.2s
root@Docker-Server:~/vcct#
root@Docker-Server:~/vcct#
```