**Cloud Server Project: dptech.online**

**Project Name:** dptech.online  
**Student:** Diego Pedraza (35549445)  
**Unit:** ICT171 - Server Environments and Architectures  
**Project Type:** Cloud-hosted website with secure access, TCO analysis, and automation

**1. Project Purpose and Scope**

This cloud infrastructure project aimed to deploy a fully operational IT services website, **dptech.online**, using Amazon Web Services (AWS) EC2 to simulate a real-world business scenario. DPTech provides diverse services, such as hardware repairs, virus removals, data recovery, networking assistance, and cloud IT consulting. The solution was designed to be secure, cost-effective, and scalable with automated tasks to simulate a professional-grade cloud deployment. The scope involved registering a domain, configuring DNS records, launching a Linux EC2 instance, installing and configuring a web server, implementing TLS encryption, scripting backup procedures, automating tasks with cron, and documenting total cost of ownership over three years. Each step was carried out with a focus on reproducibility, allowing the server to be redeployed independently in the future.

**2. Hosting Platform: Amazon EC2**

Amazon EC2 (Elastic Compute Cloud) was selected for this project due to its flexibility, affordability, and industry relevance.

**Step-by-Step Configuration:**

* Logged into AWS Console and selected EC2 from the Services menu.
* Launched a new instance using **Ubuntu Server 24.04 LTS (64-bit ARM)**.
* Chose **t3.micro** instance type under the free tier.
* Created a new key pair (diegokey.pem) and downloaded it securely.
* Set up the security group to allow incoming traffic on TCP ports:
  + 22 (SSH) for secure shell access
  + 80 (HTTP) for web access
  + 443 (HTTPS) for encrypted connections
* Allocated and associated a static public IP (Elastic IP) to the instance: 3.107.180.255.
* Used Terminal Software from Windows to SSH into the server:
* ssh -i diegokey.pem ubuntu@3.107.180.255
* Updated the package list and installed Apache web server:
* sudo apt update
* sudo apt install apache2 -y
* sudo systemctl enable apache2
* sudo systemctl start apache2
* Verified service with systemctl status apache2 and accessed the default Apache landing page from a browser.

**3. Domain Registration and DNS Setup**

* Purchased the domain dptech.online from Namecheap.
* Logged into Namecheap Dashboard > Domain List > Manage > Advanced DNS.
* Added two A records:
  + @ pointing to 3.107.180.255 (main domain)
  + www pointing to the same IP (subdomain)
* Set TTL to Automatic.
* Confirmed DNS propagation using external DNS checkers and the following terminal commands:
* dig dptech.online
* nslookup dptech.online
* ping dptech.online
* wget http://dptech.online
* curl -Iv https://dptech.online
* Full propagation occurred within approximately 15–20 minutes.

**4. TLS Certificate Setup (Let's Encrypt)**

To enable secure connections:

* Ensured port 443 was open in the EC2 instance’s security group.
* Installed Snap and Certbot tools:
* sudo snap install core
* sudo snap refresh core
* sudo snap install --classic certbot
* sudo ln -s /snap/bin/certbot /usr/bin/certbot
* Installed the certificate with Apache integration:
* sudo certbot --apache
* Selected both domains during the prompt (dptech.online and www.dptech.online).
* Auto-redirect from HTTP to HTTPS was enabled.
* Verified certificate with browser (lock icon) and CLI tools:
* curl -Iv https://dptech.online
* Confirmed certificate was issued by Let's Encrypt and set to renew automatically.

**5. Website Deployment**

* Cleaned the default contents of /var/www/html:
* sudo rm /var/www/html/index.html
* Uploaded custom HTML, CSS, and images using terminal software:
* scp -i diegokey.pem index.html ubuntu@3.107.180.255:/var/www/html/
* Verified file permissions and ownership:
* sudo chown www-data:www-data /var/www/html/index.html
* Reloaded Apache to confirm new content:
* sudo systemctl reload apache2
* Website loaded properly at [https://dptech.online](https://dptech.online/).

**6. Backup Automation Script**

* Created structure:
* mkdir -p /home/ubuntu/Documents
* mkdir -p /home/ubuntu/backup
* Added sample files using touch command:
* touch /home/ubuntu/Documents/file1.txt
* touch /home/ubuntu/Documents/file2.txt
* Created the backup script /usr/bin/testscript:

#!/bin/bash

now=$(date +"%d\_%m\_%y")

mkdir -p /home/ubuntu/backup

cp -R /home/ubuntu/Documents/\* /home/ubuntu/backup/

zip -r /home/ubuntu/backup/$now.zip /home/ubuntu/backup/\*

scp -i /home/ubuntu/diegokey.pem /home/ubuntu/backup/$now.zip ubuntu@3.107.180.255:/home/ubuntu/

* Made it executable:
* chmod +x /usr/bin/testscript
* Executed manually to validate zip output:
* /usr/bin/testscript

**7. Cron Job Setup**

* Opened crontab:
* sudo nano /etc/crontab
* Appended:
* 0 \* \* \* \* ubuntu /usr/bin/testscript
* Checked cron logs and /home/ubuntu/backup to verify automatic execution
* Confirmed zip files were created with date-based filenames every hour

**8. Total Cost of Ownership (TCO) Analysis**

* Estimated 3-year operating costs for On-Prem, AWS EC2 (IaaS), and WP.com (SaaS)
* Considered hardware, energy, staff labor, domain, cloud charges, and software fees
* AWS EC2 provided flexibility and root control while costing far less than On-Prem

**9. Justification for AWS EC2**

* Greater customization with root access
* Allows implementation of automation, security measures, and Linux tools
* Industry-aligned environment for real DevOps practices
* Supports future scaling or transition to containers or CI/CD pipelines

**10. GitHub Repository**

* Pending publication at GitHub
* Will include:
  + Complete script with comments
  + HTML/CSS source code
  + Deployment instructions in README.md
  + DNS and TLS testing outputs

**11. Final Testing Checklist**

✅ DNS resolves and A records confirmed  
✅ Apache Web Server fully operational  
✅ HTTPS enabled with valid certificate  
✅ Website content deployed successfully  
✅ Backup script tested manually and via cron  
✅ Files timestamped correctly with zip format  
✅ Server secure, ports configured properly  
✅ Platform aligns with real-world IT business simulation

**Prepared by:** Diego Pedraza  
**Date:**