Varun L M

[Company name]  [Company address]

CTF Project documentation

Contents

[Getting Started 2](#_Toc171530801)

[Creating resources using Cloudformation 2](#_Toc171530802)

[Template for creating a new VPC and resources 2](#_Toc171530803)

[Parameters 2](#_Toc171530804)

[Steps 2](#_Toc171530805)

[OpenVPN server Configuration 3](#_Toc171530806)

[Install PuTTY 3](#_Toc171530807)

[Steps 3](#_Toc171530808)

[Connecting to VPN 9](#_Toc171530809)

[Important note 9](#_Toc171530810)

[Steps 9](#_Toc171530811)

[CTFd Configuration 13](#_Toc171530812)

[Setup 13](#_Toc171530813)

[Importing Challenges from Juice shop 16](#_Toc171530814)

[Multi-Juicer 17](#_Toc171530815)

[Prerequisites 17](#_Toc171530816)

[Setup 17](#_Toc171530817)

[Deinstallation 18](#_Toc171530818)

# Getting Started

There are 3 components in this project:

* OpenVPN server – Participants connect to VPN to play CTF
* CTFd – A hosting platform for CTFs
* OWASP Multi Juicer – A Kubernetes cluster deploying instances of OWASP Juice shop instances (Vulnerable website)

# Creating resources using Cloudformation

Before setting up, it is expected that you have created a keypair in AWS console and have a copy of .ppk file in your local computer.

## Template for creating a new VPC and resources



Use the above script for creating the resources

## Parameters

|  |  |
| --- | --- |
| KeyPair | Select your Key Pair Name |
| WhiteListedIPs | Enter the IP addresses allowed to access OpenVPN (default 0.0.0.0/0) |
| VPCId  PublicSubnet1Id  PublicSubnet2Id  PrivateSubnet1Id  PrivateSubnet2Id | These are optional parameters. Select VPC and subnets for resource creation.  **Before using an existing VPC for this project, make sure of the following:**   * The VPC has 2 public and 2 private subnets. * Has Internet Gateway and NAT Gateway attached. * Has proper route tables and subnet associations.   If no value is given, new VPC and subnets will be created. |
| OpenVPNAMI  CTFDAMI | Select the appropriate AMI value according to the region used  The default AMIs are for the us-east-2 region. |

## Steps

1. Log in to your AWS account.
2. Search “Cloudformation” and create a new stack.
3. Upload your template.yaml file, leave everything default and create the stack.
4. Wait till you get the confirmation like the one shown below.

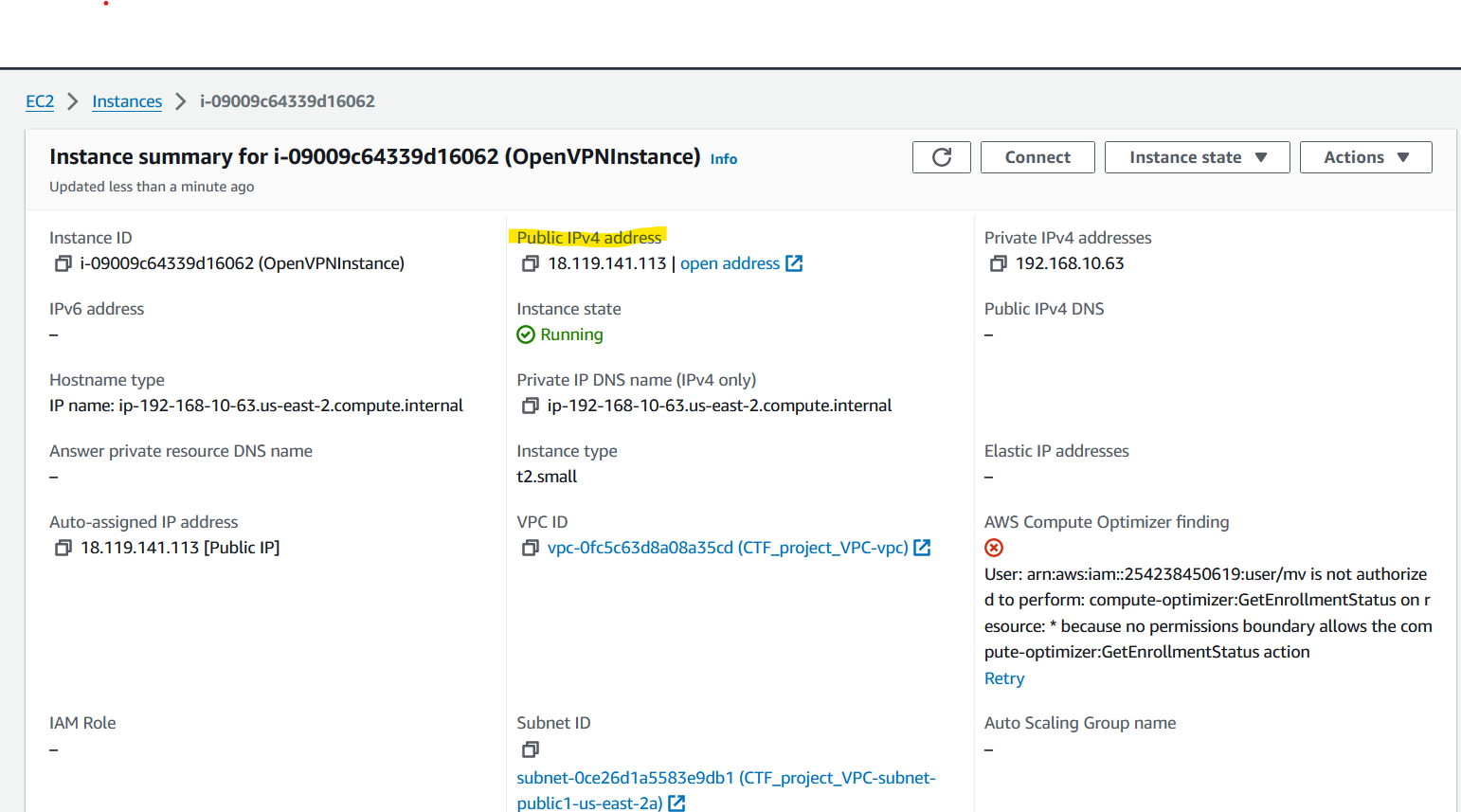
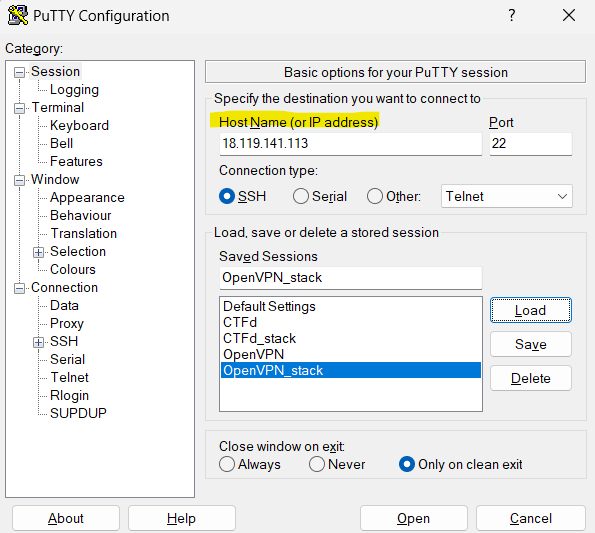
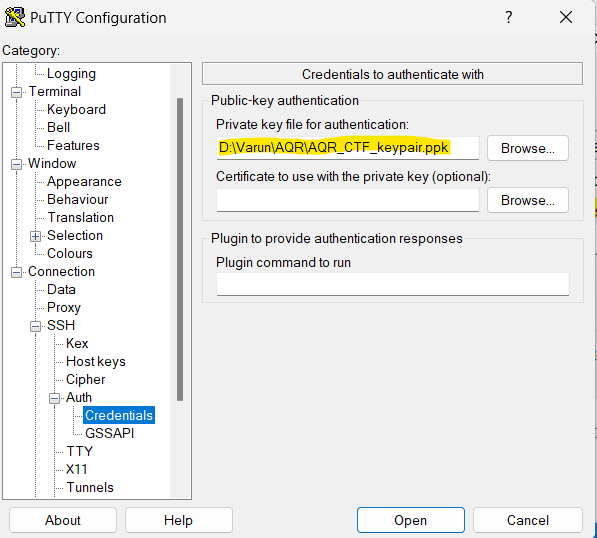
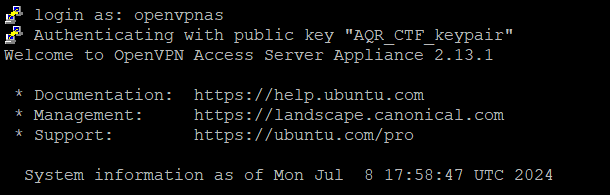
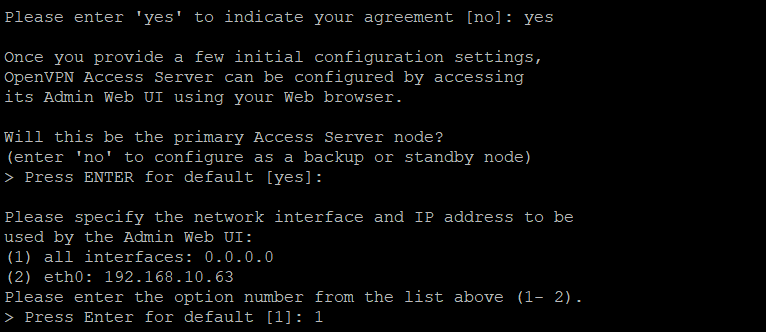
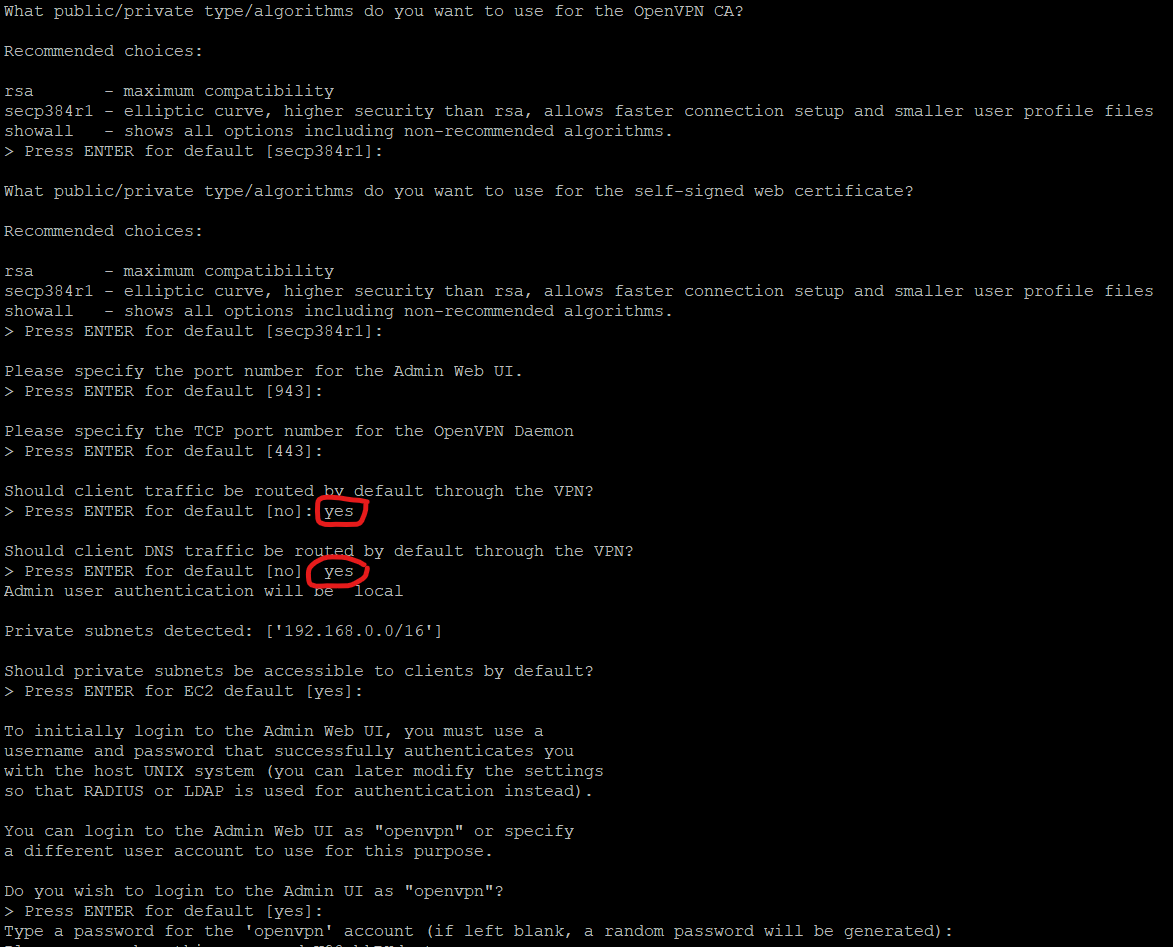
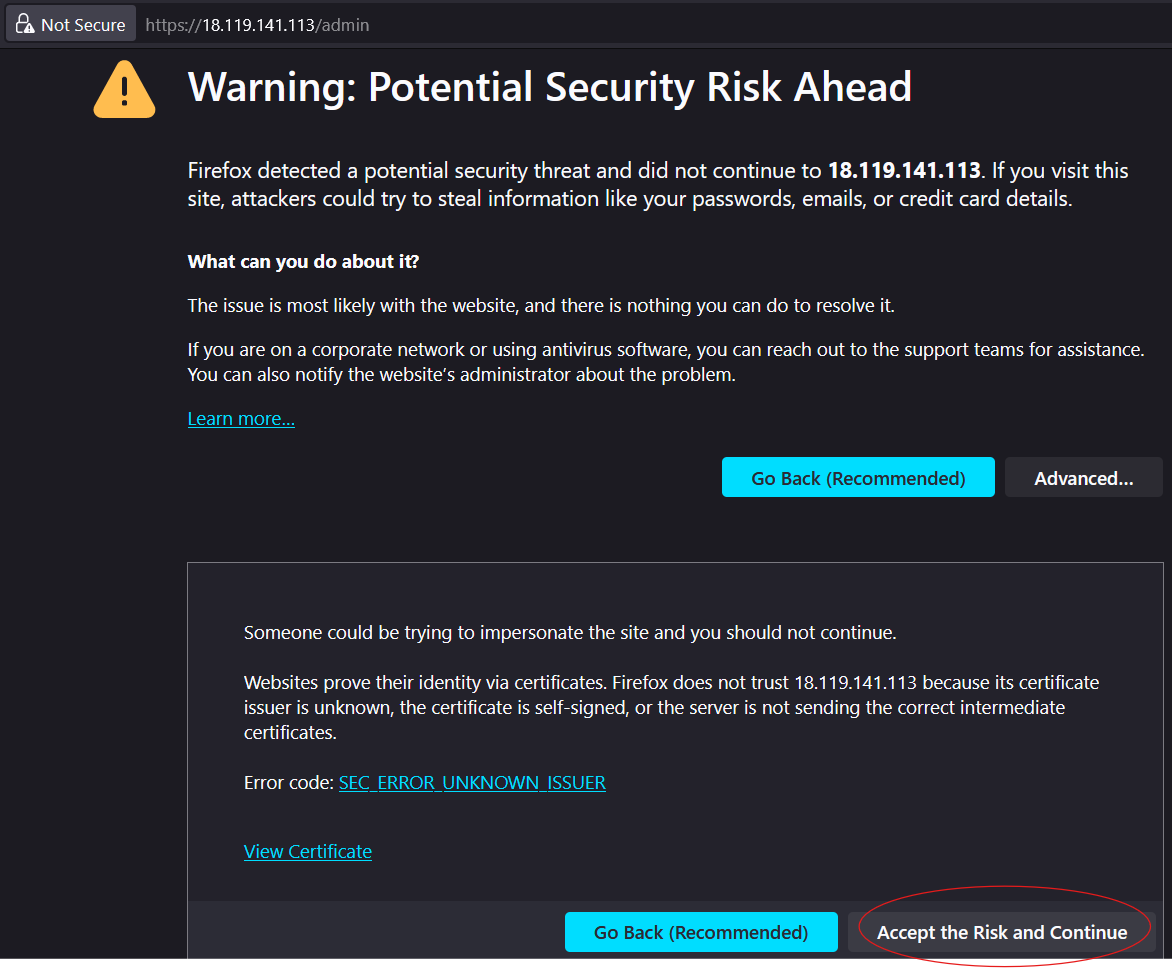
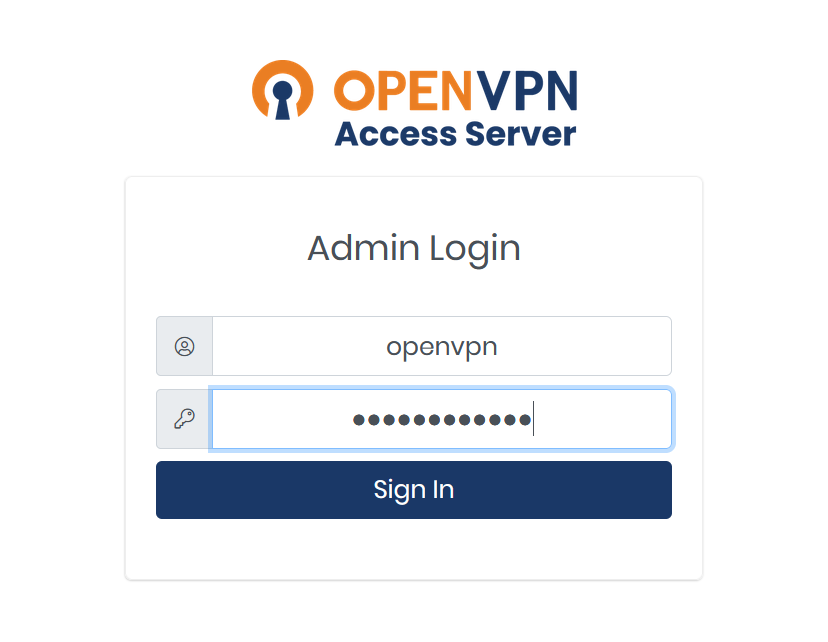
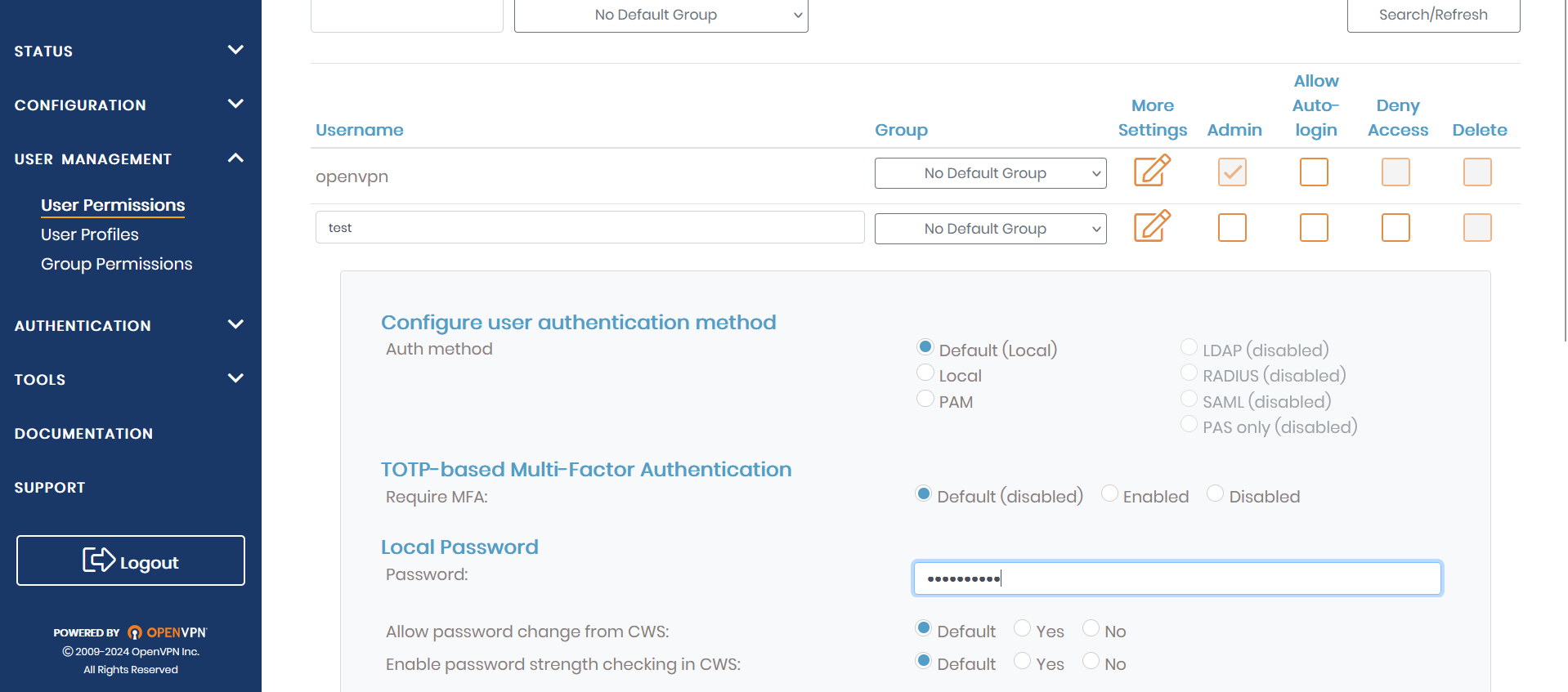
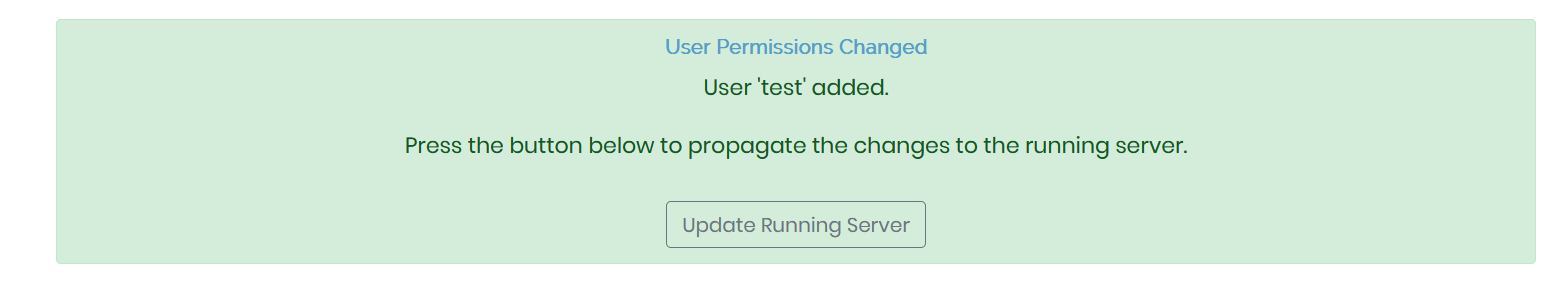
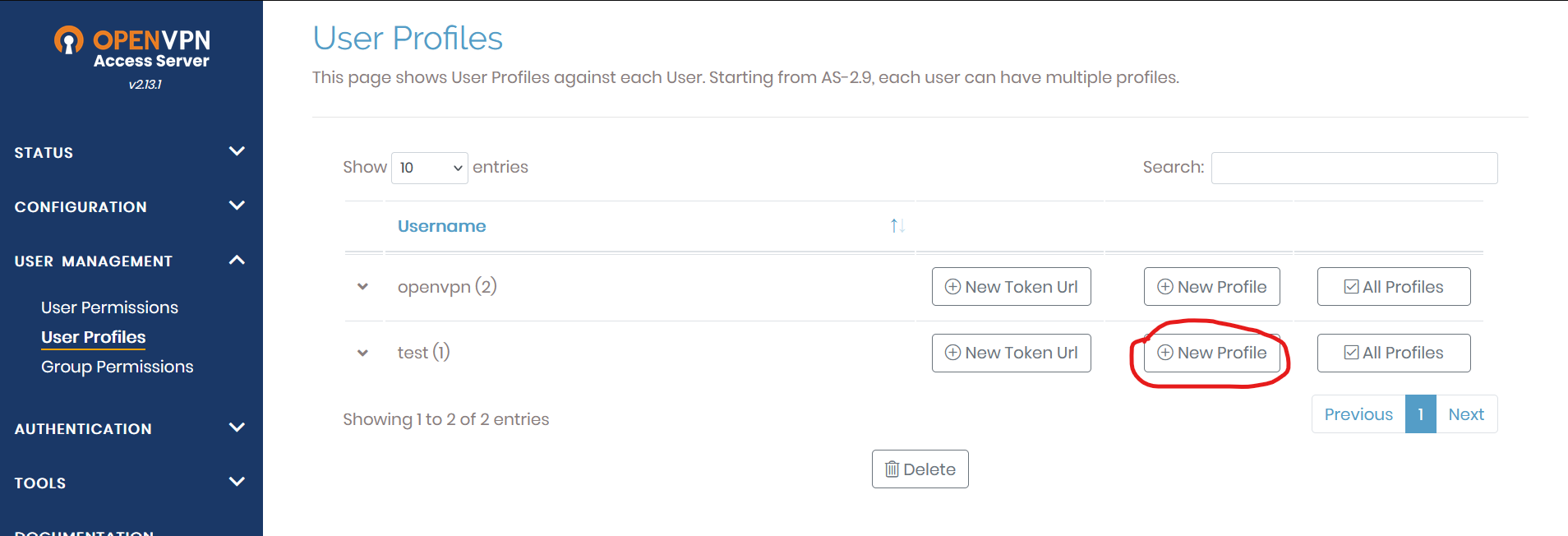


# OpenVPN server Configuration

## Install PuTTY

<https://www.putty.org/>

## Steps

1. Copy the public IP of the OpenVPNInstance. 
2. Open PuTTY and paste it in Host Name (or IP Address) field
3. Navigate to Connection -> SSH -> Auth ->Credentials and upload your private key (.ppk file) 
4. Navigate back to Session, save the session and click “Open”.
5. A PuTTY security alert will be shown when logging in for the first time. Click “Accept” 
6. Log in as “openvpnas”
7. Agree to the User License Agreement
8. Leave everything default except the ones highlighted
9. Type a password for your OpenVPN admin account or get the randomly generated password. **Do not lose the password.**
10. Take note of the links for Admin UI and Client UI
11. Go to admin UI. Create users and their VPN credentials in the website. Click “Accept the risk and continue” and login as “openvpn” 
12. Create users and passwords for them as shown below
13. Click on “Update running server” after adding user
14. OpenVPN username and password can be shared with the participants. Alternatively, a new profile can be created as shown below and .ovpn file can be shared. 

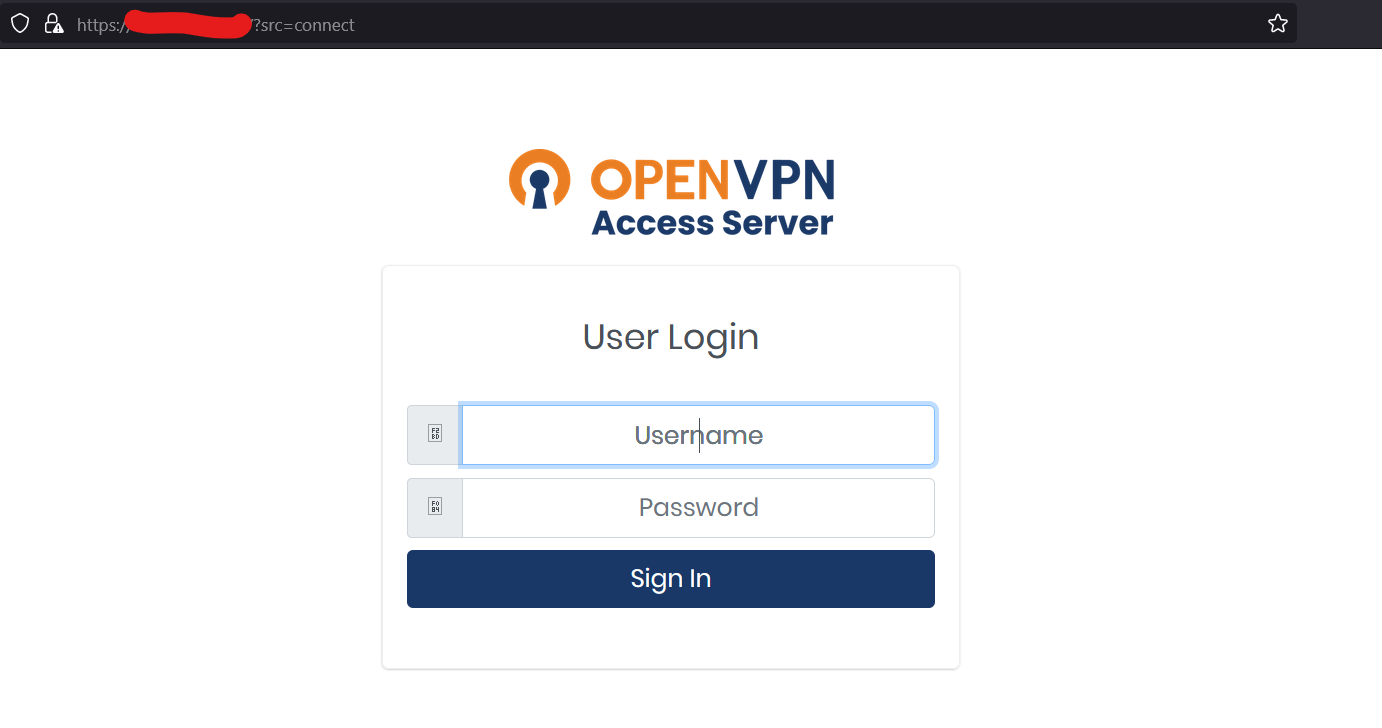
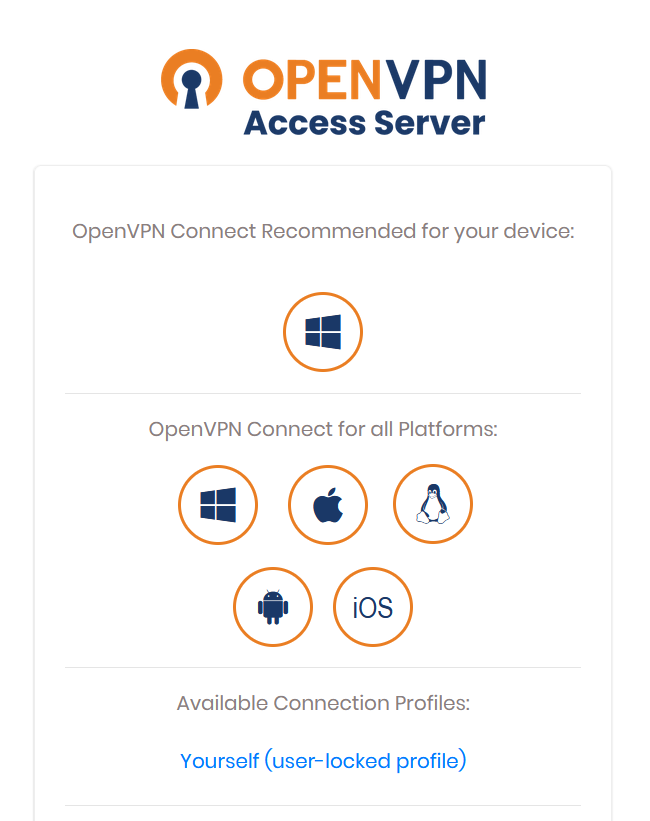
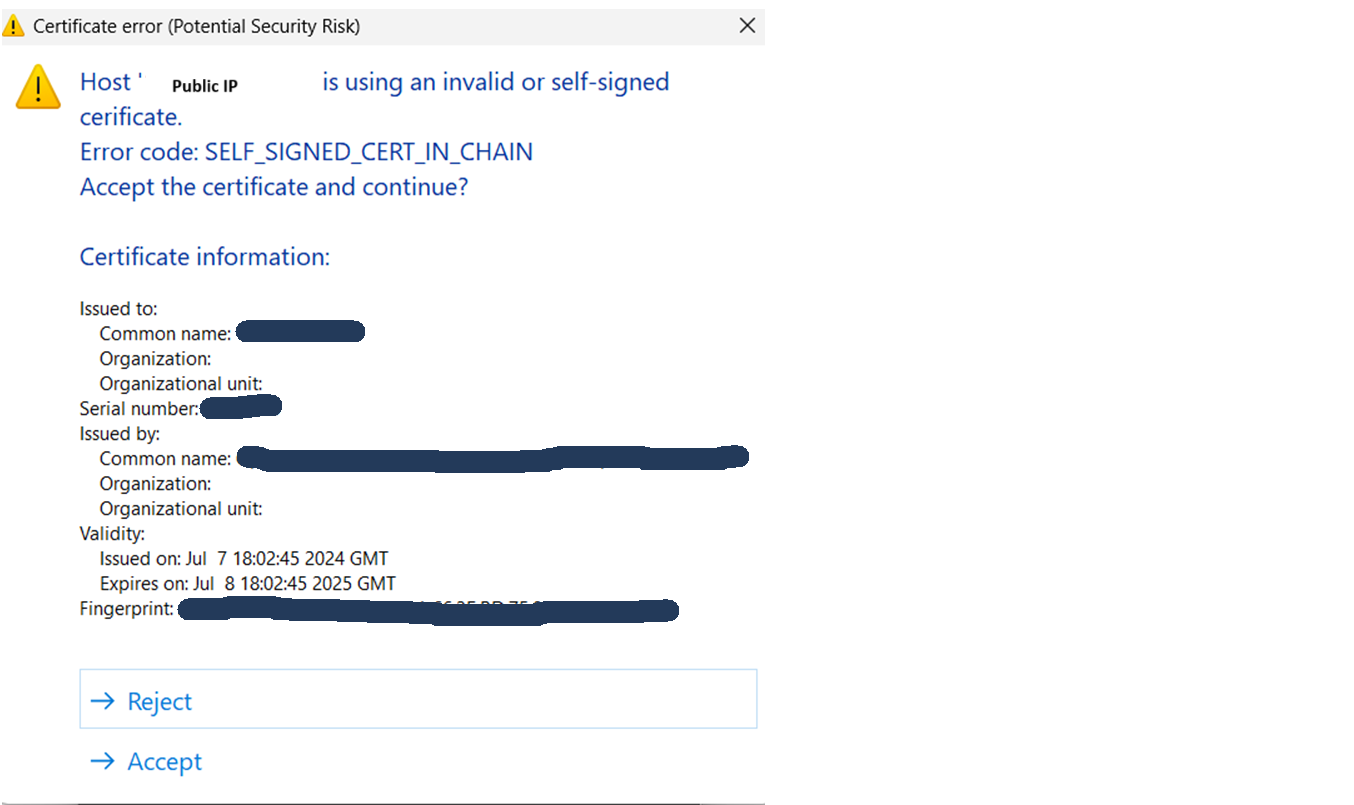
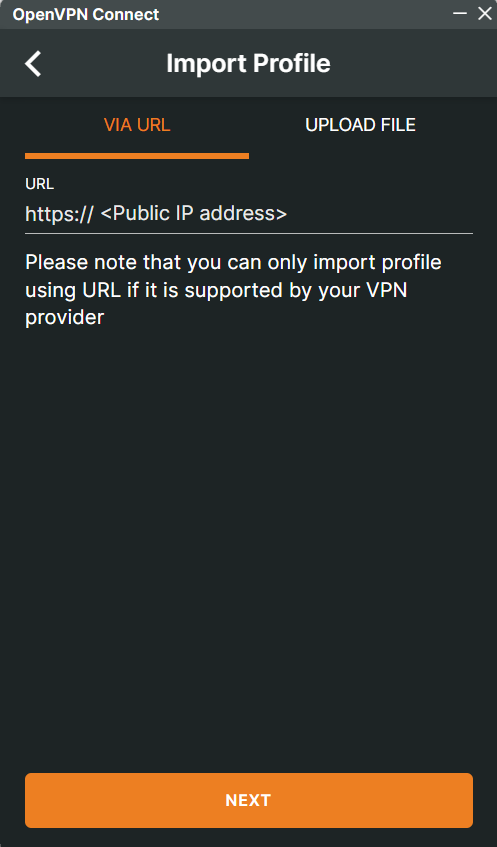
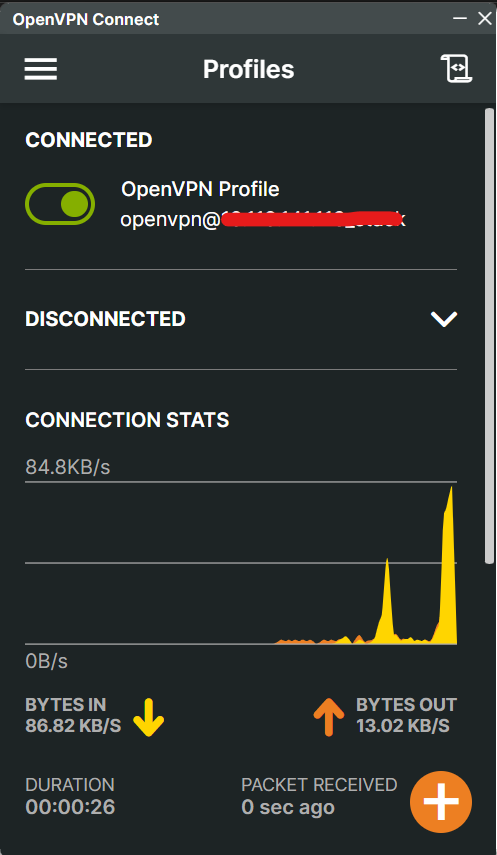
# Connecting to VPN

Important note

**This step is common to both the admin and the participants. Admin must share this part of the documentation to the users.**

**Before sharing the steps to users, ensure you have completed this step,** [**configured CTFd**](#_CTFd_Configuration)

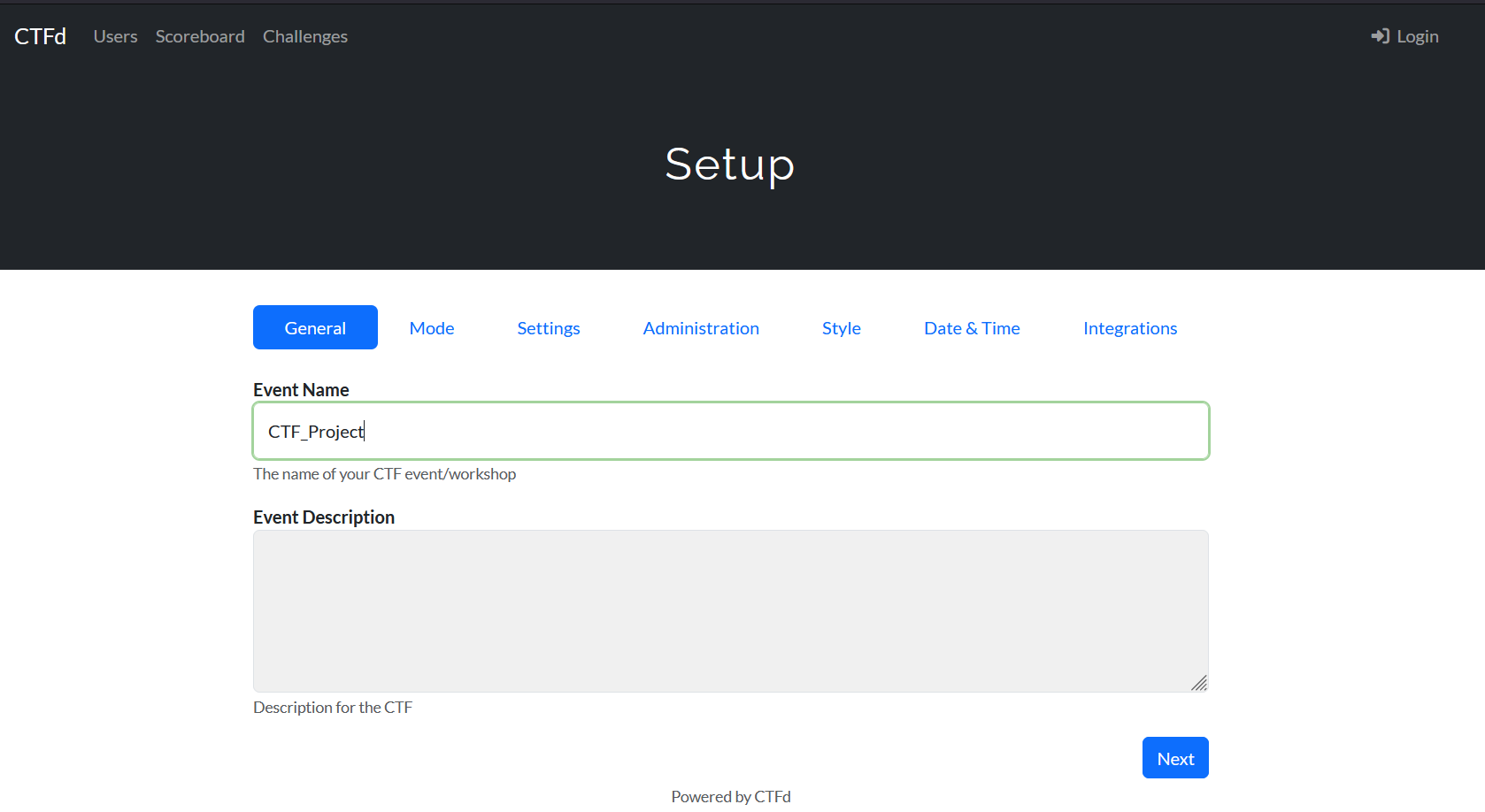
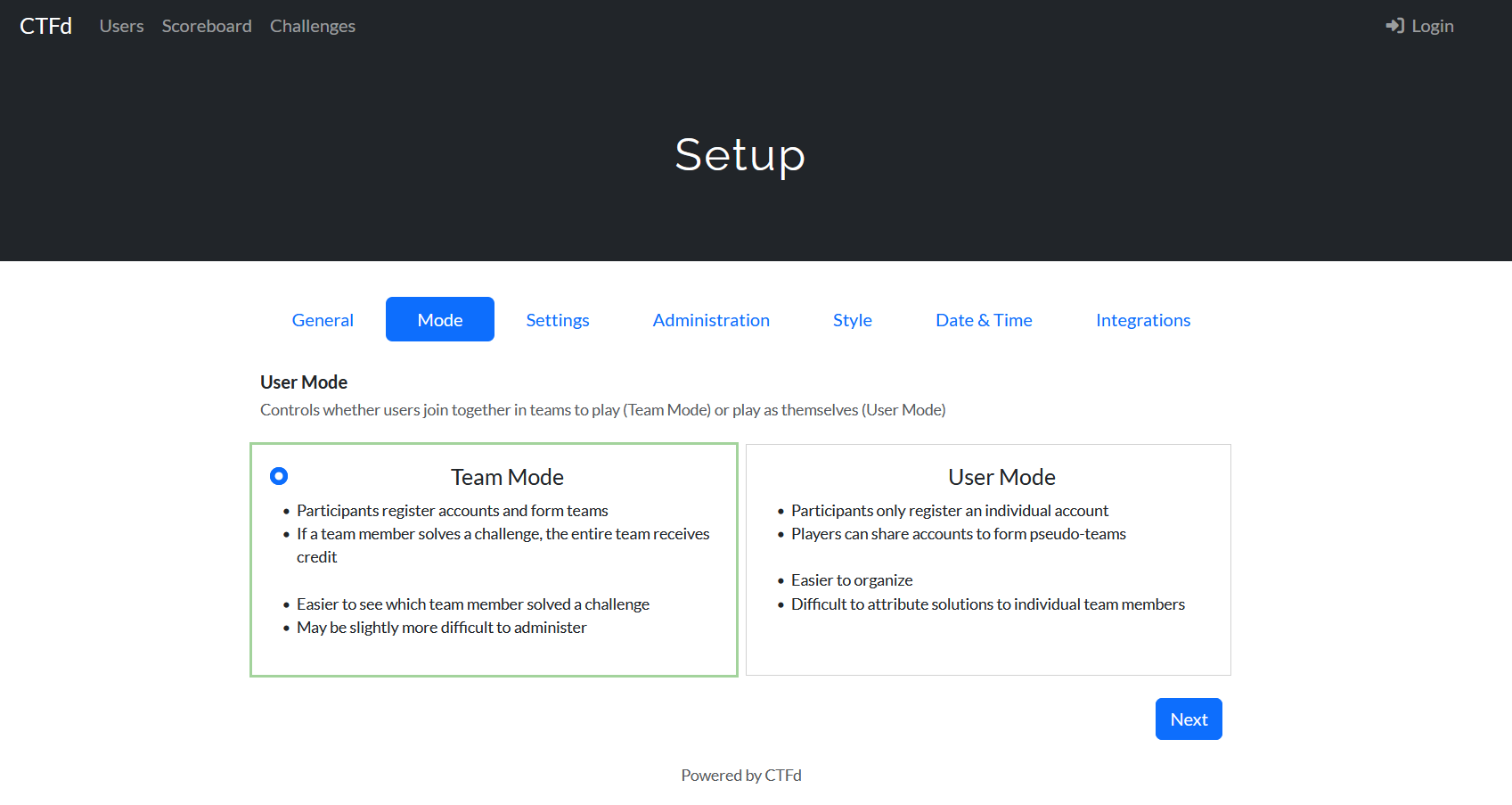
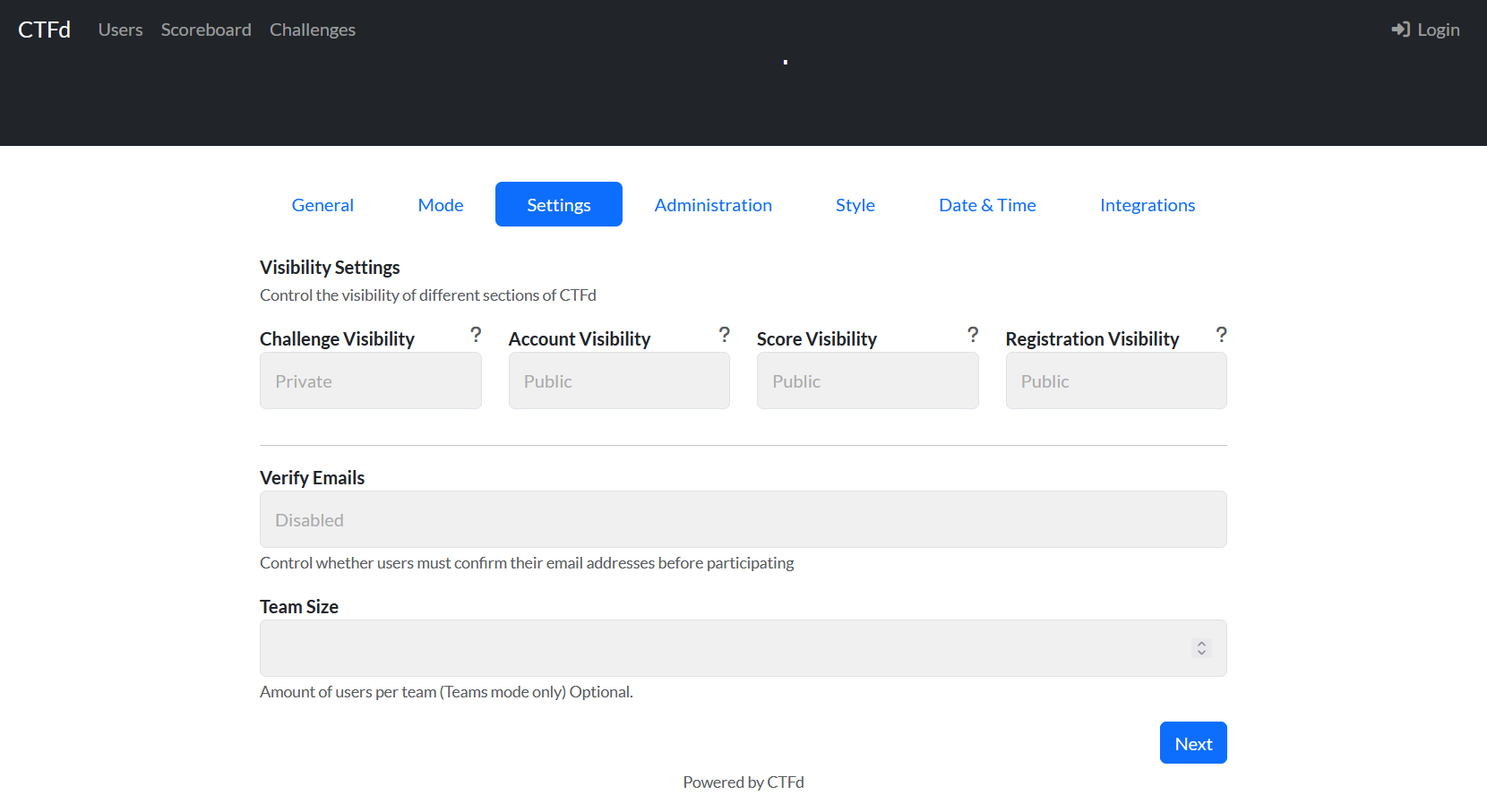
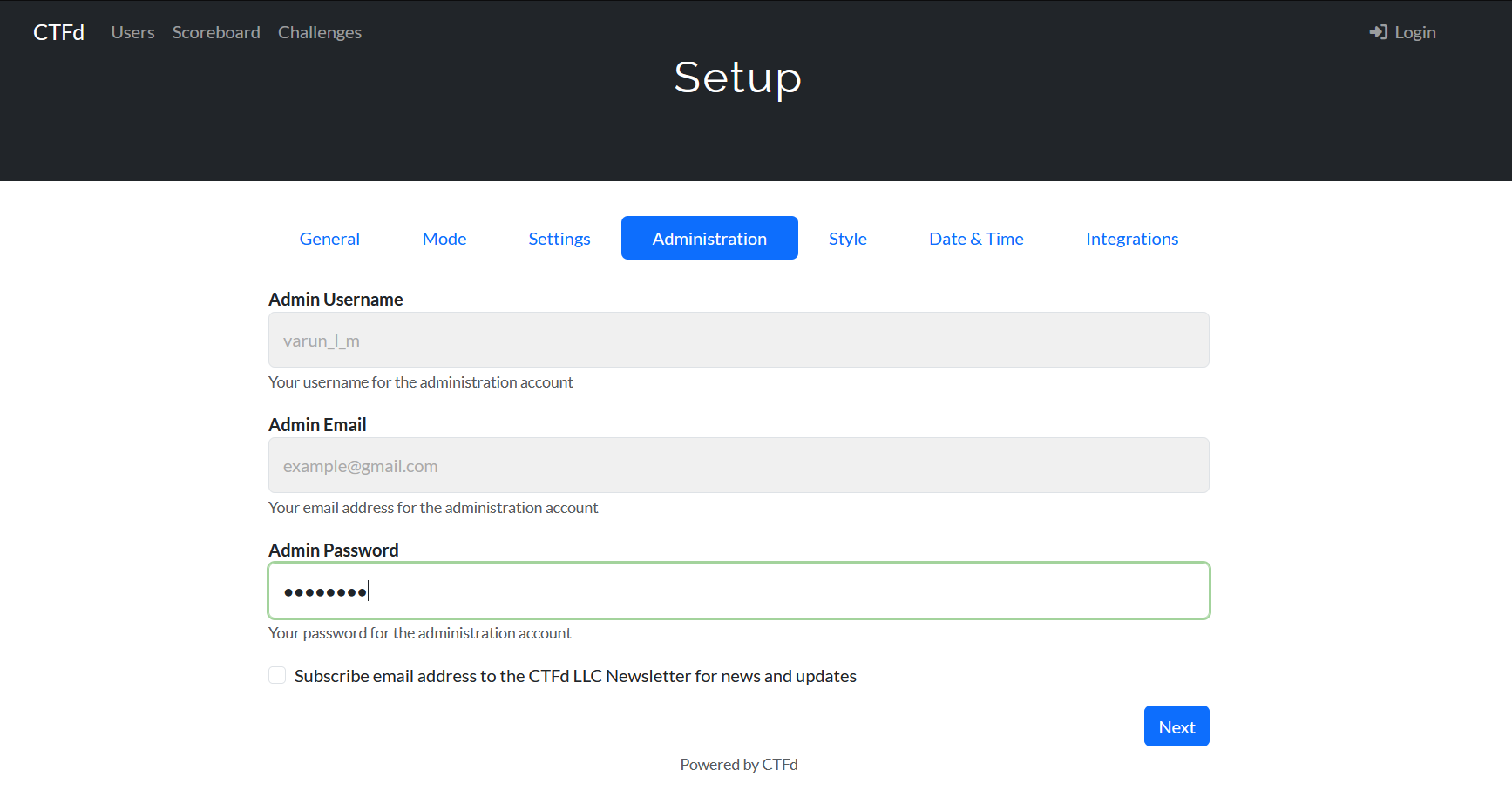
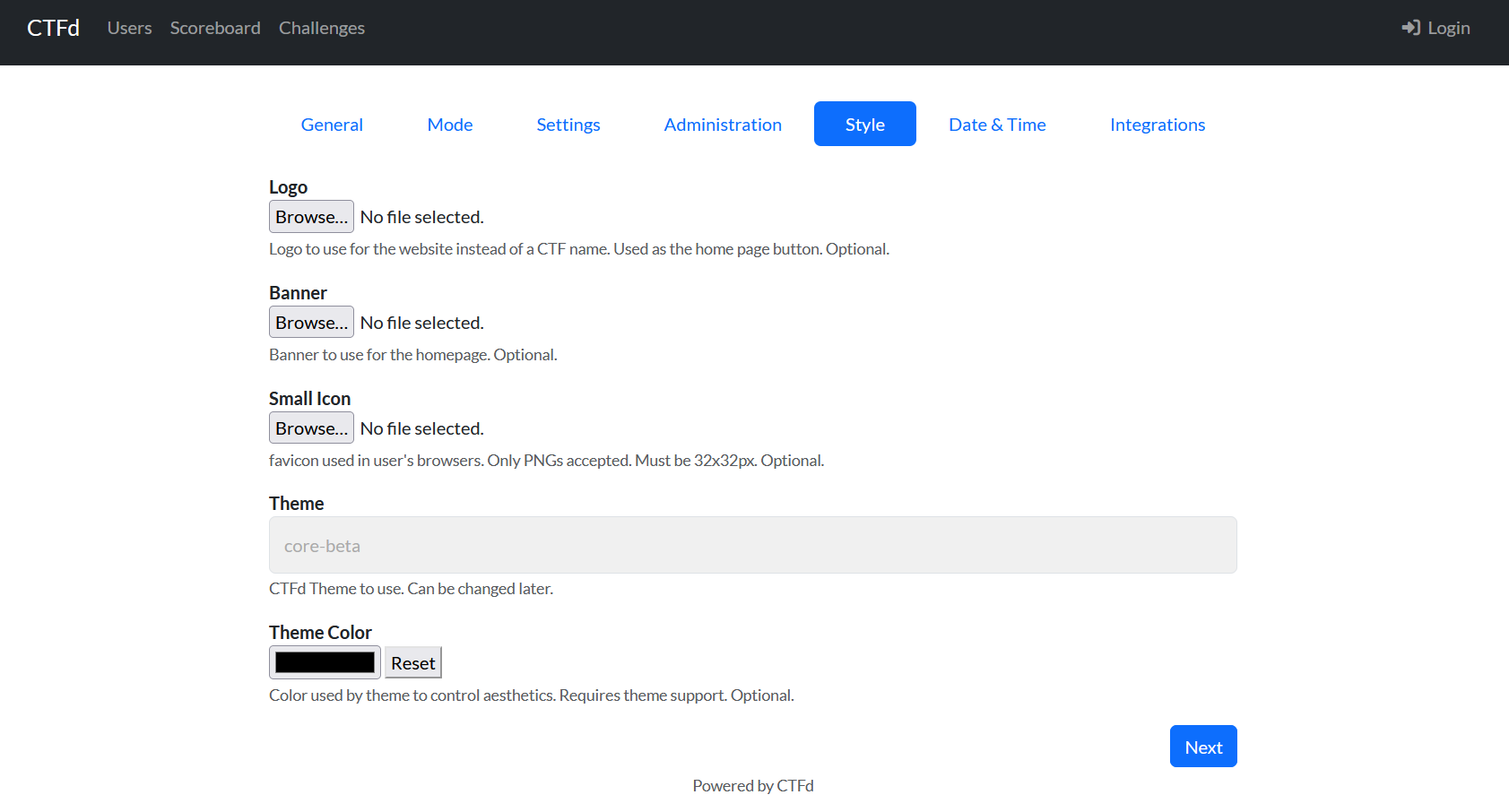
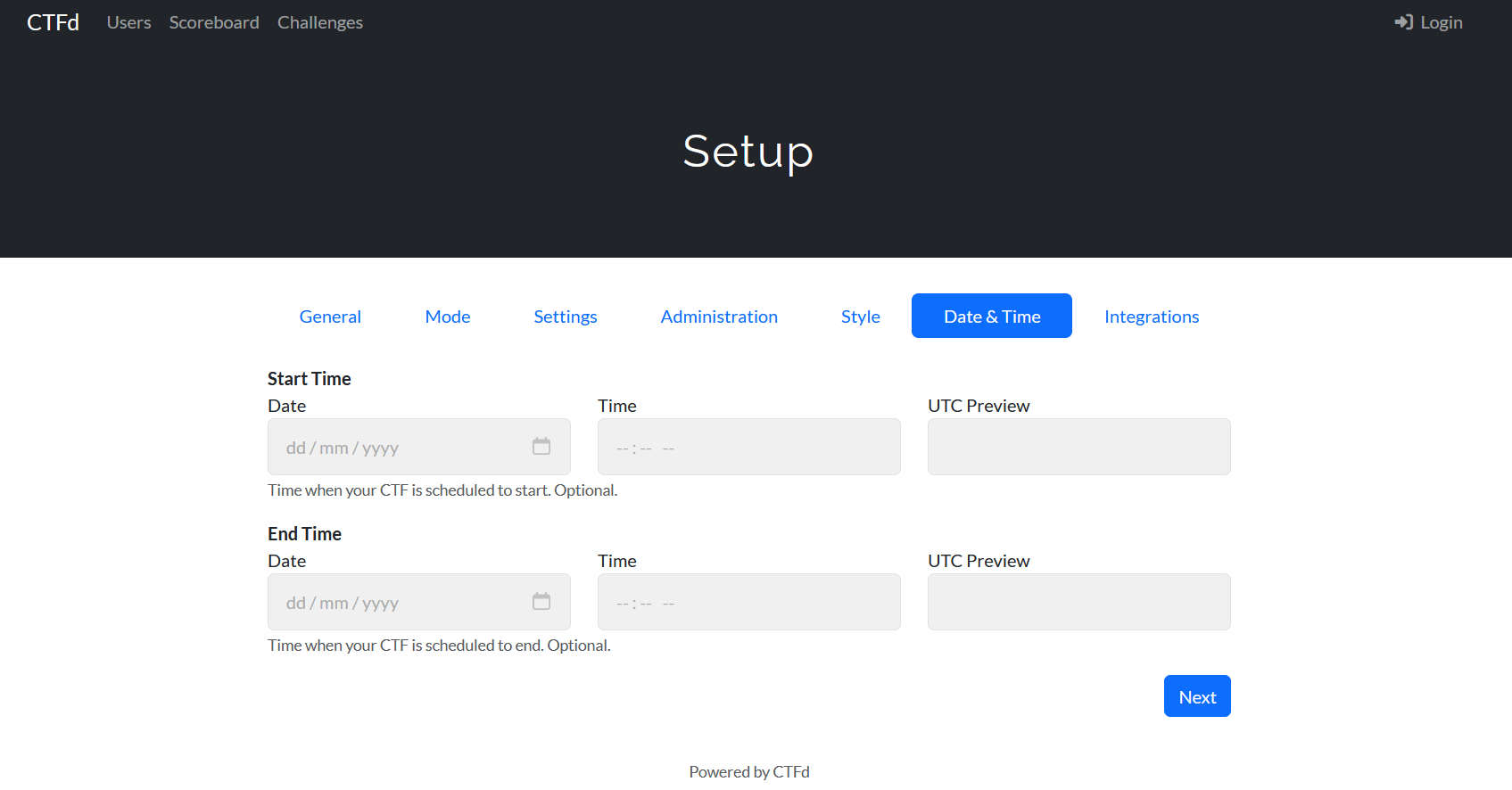
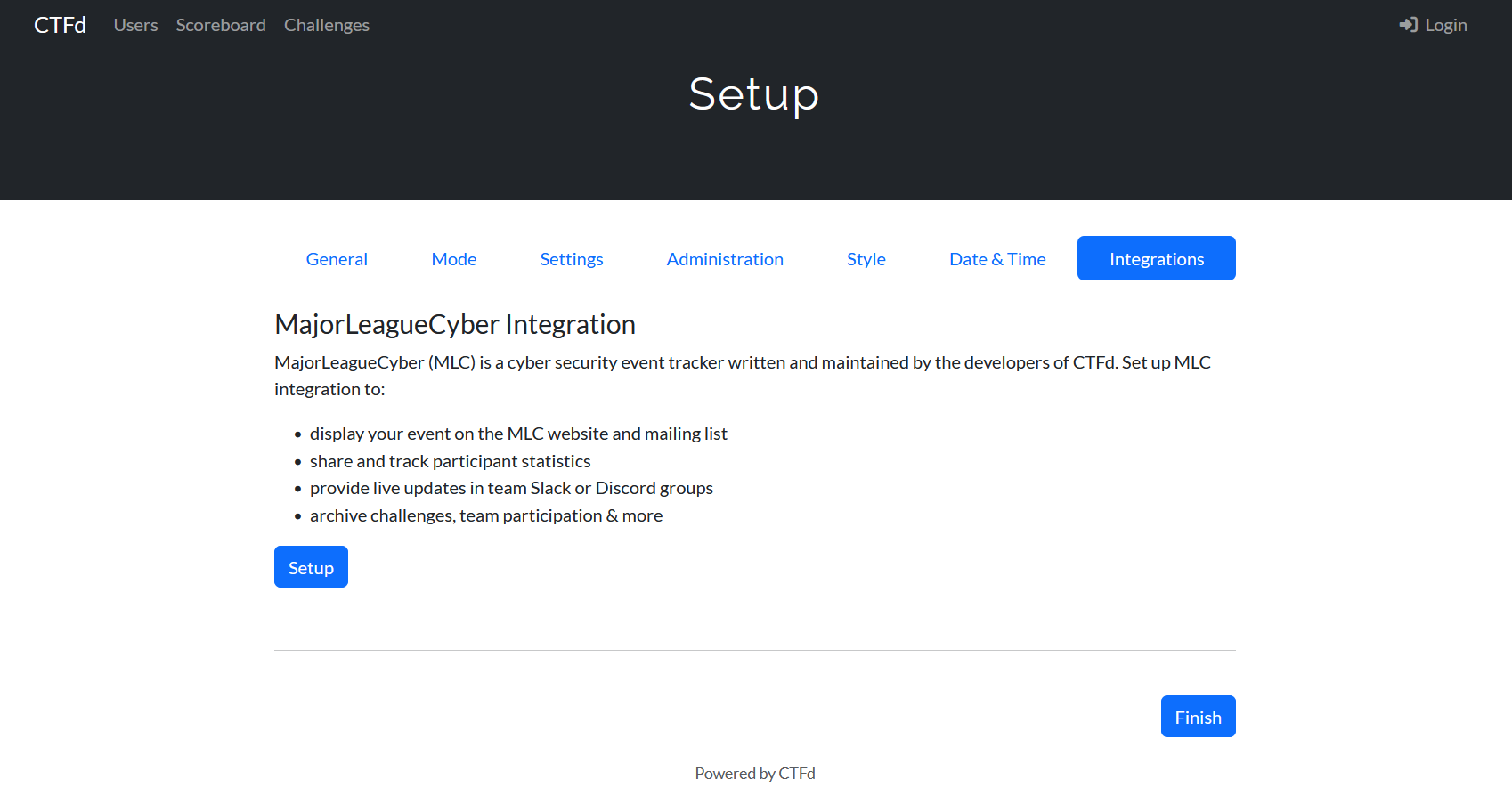
## Steps

1. Open the public IP address of the OpenVPN access server shared by the admin and login with the credentials given to you
2. Download the OpenVPN client app for your OS. 
3. Import the OpenVPN profile either by .ovpn file shared to you by the admin or by entering the Public IP address of the OpenVPN server shared by the admin in place of url. Accept the certificate and continue.  
   
4. Connect to the VPN using your credentials. Once you are connected to the VPN, you will get something similar to the image shown below. 
5. Get the IP address of CTFd portal, register for the CTF.
6. Get the IP address of OWASP Multi-Juicer, create your instance and start hacking.

# CTFd Configuration

CTFd docs for reference : <https://docs.ctfd.io/docs/overview>

## Setup

Connect to the VPN and search the private IP address of CTFdInstance and complete the setup as per your requirement.  
  
  
  
  
  
   
  


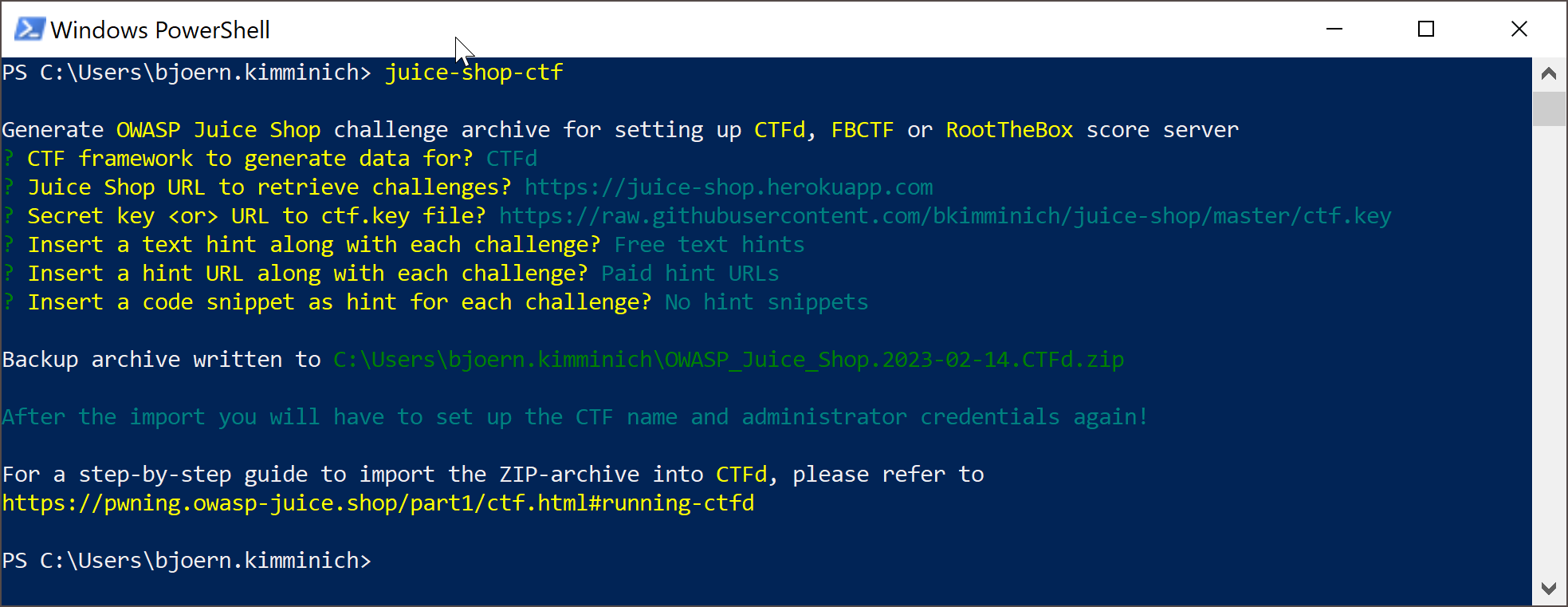
## Importing Challenges from Juice shop

Juice shop ctf cli docs for reference : https://github.com/juice-shop/juice-shop-ctf

Install juice shop ctf cli as shown below.  
  
npm install -g juice-shop-ctf-cli

Open a command line and run:  
  
juice-shop-ctf

Follow the below image to import challenges.  
**Important note : Give a secret key(string) of your choice instead of the default URL and store the secret key in a safe place, which will be used in the future.**



* After this step, a CSV file containing challenges will be downloaded.
* In the CTFd portal, login as admin and navigate to Admin Panel -> Config -> Backup -> Import and Export.
* Click the Import CSV option, CSV Type -> Challenges and upload the CSV file.
* **Recommended : After this step or at any point of time, you may export your current CTF state. This zip file may be used to recreate the same CTF environment with the same challenges, users, teams in the future.**

# Multi-Juicer

Multi-Juicer documentation for reference : <https://github.com/juice-shop/multi-juicer/blob/main/guides/aws/aws.md>

## Prerequisites

Open Cloud shell in your AWS console and follow the [Setup](#_Setup) section

In case of permission error, follow this [article](https://medium.com/@prateek.malhotra004/mastering-aws-cli-a-comprehensive-guide-to-command-line-power-ca2260167839) to install aws cli in your command line and connect to your AWS account using access token.

## Setup

This example expects you to have the following cli tools setup.

1. [awscli](https://aws.amazon.com/cli/)
2. [eksctl](https://eksctl.io/installation/)
3. [helm](https://helm.sh)
4. [kubectl](https://kubernetes.io/docs/tasks/tools/install-kubectl/#install-kubectl-on-macos)

Run the following commands in your command line.

# Creating a cluster

# First, we will need a cluster, you can create one using the eksctl cli.

**# Replace the private and public subnets with the ones in your VPC**

# This will take a couple of minutes

eksctl create cluster \

--name multi-juicer \

--vpc-private-subnets=subnet-0b00e94052cfbcc85,subnet-0405b87f32e6ef9b3 \

--vpc-public-subnets=subnet-0ce26d1a5583e9db1,subnet-04dc418fd47c7950d \

--nodegroup-name standard-workers \

--node-type t3.medium \

--nodes 2 \

--nodes-min 1 \

--nodes-max 10

# After completion verify that your kubectl context has been updated:

# Should print something like: Administrator@multi-juicer.eu-central-1.eksctl.io

kubectl config current-context

# Installing MultiJuicer via helm

# **Replace <Your CTF Key> with the secret key you used while importing Juice shop challenges**

helm install multi-juicer oci://ghcr.io/juice-shop/multi-juicer/helm/multi-juicer --set juiceShop.ctfKey= <Your CTF Key>,juiceShop.nodeEnv=ctf

# kubernetes will now spin up the pods

# to verify everything is starting up, run:

kubectl get pods

# This should show you two pods a juice-balancer pod and a progress-watchdog pod

# Wait until both pods are ready

Optional step

# Let us test out if the app is working correctly before proceeding

# for that we can port forward the JuiceBalancer service to your local machine

kubectl port-forward service/juice-balancer 3000:3000

# Open your browser for localhost:3000

# You should be able to see the MultiJuicer Balancer UI

# Try to create a team and see if everything works correctly

# You should be able to access a JuiceShop instances after a few seconds after creating a team,

# and after clicking the "Start Hacking" Button

Create a loadbalancer which is exposed is achieved by running the following command:

kubectl create -f https://raw.githubusercontent.com/juice-shop/multi-juicer/main/guides/aws/loadbalancer.yaml

You can get the LoadBalancer's DNS record either from the AWS console, or by running:

kubectl get services

# Check the external IP of the Loadbalancer to get the DNS Record

Change the security group of the Load balancer to allow traffic from the Public IP of the OpenVPNInstance.

**Before the start of the CTF, make sure participants are connected to VPN. Provide them the IP address of CTFdInstance and DNS record of the Multi-Juicer Loadbalancer.**

## Deinstallation

helm delete multi-juicer

# Delete the loadbalancer setup

kubectl delete -f https://raw.githubusercontent.com/juice-shop/multi-juicer/main/guides/aws/loadbalancer.yaml

# Delete the kubernetes cluster

eksctl delete cluster multi-juicer