

Versioning	3
1. Development Server	4
1.1 Bentray.work	4
1.2 Test.bentraytech.com	6
2. Production Server	7
2.1 Shared Server (WHM/CPANEL)	7
2.2 VPC	8
2.2.1 AWS VPC	8
2.2.1.1 TCA Portal (Proadvance)	8
2.2.1.2 Chat Server	9
2.2.2 Digital Ocean	9
2.2.2.1 Foton Nepal VPC	9
2.2.2.2 KshamadeviGroup VPC	10
2.2.2.3 GFST Mail	10
2.2.2.4 Demo.bentray.work	11
2.2.2.5 Cloud2.tukihost.com	11
2.2.2.6 App.Flexyear.com	12
APPENDIX	13

Versioning

Date	Description	Created By	Version
15 March, 2022	DevOps Documentation	Mohamod Sahil Ansari	0.0.0
	Development Server		0.1.0
16 March, 2022	Bentray.work	Mohamod Sahil Ansari	0.2.0
	Test.bentraytech.com		0.3.0
	Production Server		0.4.0
	Shared Server		0.5.0
	VPC - TCA, Chat, Foton, Kshamadevi, GFST		0.6.0
18 March, 2022	VPC - demo, cloud2, app	Mohamod Sahil Ansari	0.7.0
	APPENDIX		0.8.0

DevOps Documentations

This is a document on the information of the server along with the information of the server. Follow along this document to get a better understanding of the servers.

This document describes the servers used in the organization.

1. Development Server

This server is used for the development phase and for the testing phase as well. All the changes are made in the development server and are tested accordingly.

There are two development servers used for developing and testing.

- Bentray.work (Hosted in AWS)
- Test.bentraytech.com (Hosted in Digital Ocean)

1.1 Bentray.work

This server is hosted in AWS and can be accessed through Webmin which is similar to Cpanel. In order to login to the server, you need to have a key from the supervisor. This key will be stored in a google drive of the company. This link will be linked below.

- **Webmin:**

URL: <https://bentray.work:10000/>

Username: root

Password: La7EbsawG3

- **Terminal/Termius:**

You need the key to login through the terminal. Get the key from google drive and login using terminal command or use the GUI of termius to login into the server.

Use the following Command in order to login on the server.

```
$ chmod 400 <path_to_key>
```

```
$ ssh -i <path_to_key> user@ip
```

To access the server using termius use the following credentials

Username: centos

Password: <key_ending_with_sujit_ppk> - Get it from Google Drive

IP: get it by using ping command

Aside from accessing the server, this AWS server is used to route the domain to different servers using Route53.

In order to login into AWS, you need an IAM user login credential. After you have access to AWS, follow the steps in order to route the domain to a different server.

- Search for the service Route53
- Go to hosted zone which is in the left side of your screen
- You will see different hosted zones but we need to work on bentray.work
- Click on the bentray.work hosted zone and get the list of domains of our dev server.
- Select the domain whose route you want to redirect and put the IP of the destination server.

- Use ping from local to check the redirection of the domain to the desired server.

NOTE:

Use the Terminal to access the server and pull the changes from the git rather than using Webmin and all the projects are placed in */home* directory. This server consists of a production server as well so be careful of what you are doing in AWS.

1.2 Test.bentraytech.com

This server consists of the development server needed for the developer to test the code. In this server, projects are containerized and run through reverse proxy along with SSL. So in case you need to make a new container for a project, you need to run it through a reverse proxy with SSL. (Reference at the end of this document)

Reverse proxy Link: <https://hub.docker.com/r/jwilder/nginx-proxy>

Use this link to have a better understanding of the reverse proxy.

This server can be accessible through terminal or termius using the credentials below.

IP: 165.22.209.140

Username: root

Password: 5.zMfqN@Msy9ysA

All the projects are placed in the */home* directory and it's common practice to use home directories for multiple projects rather than in the root directory.

2. Production Server

This server is used to roll out the projects to the client and host them as per client requirements. Mostly the websites are hosted in a shared server using WHM and Cpane whereas others are hosted on the VPC(Virtual Private Cloud).

2.1 Shared Server (WHM/CPANEL)

This server hosts multiple websites of a client and restores the website as per client requirement. Mostly you will be working on the UI using WHM login and configure as per the requirement.

You can access the server by two ways either using WHM login or login to the server. Recommended to use WHM login unless you need to login to the server.

- **WHM Login Credentials**

URL: <http://bentraytech.com/whm>

Username: root

Password: agRKg[^0ju3H

- **Server Login Credentials**

IP: 209.159.155.2

Username: root

Password: agRKg[^0ju3H

Port: 1157

2.2 VPC

Most of the clients have VPC and need to be managed and provide the support to the client as per need. You are responsible for troubleshooting the server in case of any issue. Some VPC are hosted in AWS and some in Digital Ocean.

2.2.1 AWS VPC

Some VPC are hosted in the AWS and need to access the server in case of any issue to troubleshoot the server. So as mentioned earlier, you need to have IAM login credentials to access these servers.

2.2.1.1 TCA Portal (Proadvance)

This server is an AWS VPC and can be accessed using the following credentials. You can even access through AWS Console to see the servers. To access the server, you can use the webmin or directly login to AWS Console using IAM Credentials.

- **Webmin**

URL: <https://server.tcaportal.net:10000/>

Username: root

Password: O98hbQREX52UDnD@

- **Terminal/Termius**

IP: 54.69.176.174

Username: centos

Password: <Key_starting_with_Bents> # Look in Google Drive

2.2.1.2 Chat Server

This is another server hosted in AWS. You can view this server through AWS Console but need to have a login key to access the server.

- **Terminal/Termius**

IP: 44.236.37.115

Username: ec2-user

Password: <Key_starting_with_chat> #Look in Google Drive

2.2.2 Digital Ocean

Beside from AWS, we have hosting on Digital Ocean. You don't need any IAM user to login to the server. You just need to have IP and root login credentials.

2.2.2.1 Foton Nepal VPC

This server is a Foton Nepal VPC where the website is hosted. You can use webmin or login to the server to troubleshoot the issue.

- **Webmin**

URL: <http://fotonnepal.com:10000/>

Username: root

Password: 5.zMfqN@Msy9ysA

- **Terminal/Termius**

IP: 139.59.95.116

Username: root

Password: 5.zMfqN@Msy9ysA

2.2.2.2 KshamadeviGroup VPC

This server is used to host the website of KshamadeviGroup. You can access the server through Webmin or terminal to troubleshoot the issue.

- **Webmin**

URL: `http://kshamadevigroup.com:10000/`

Username: root

Password: uTh37(iQp20!Uzs9

- **Terminal/Termius**

IP: 159.89.166.116

Username: root

Password: uTh37(iQp20!Uzs9

2.2.2.3 GFST Mail

This server is used to host mail from GFST. It is running in a container environment. You can only access this server using terminal or termius.

- **Terminal/Termius**

IP: 165.232.181.61

Username: root

Password: 5.zMfqN@Msy9ysA

2.2.2.4 Demo.bentray.work

This server is used to host communication tools i.e mattermost. Along with the tool, there's a password manager BitWarden. On this server, we have hosted websites as well i.e MeroReport, MySirani.

- **Terminal/Termius**

IP: 139.59.80.80

Username: root

Password: 5.zMfqN@Msy9ysA

2.2.2.5 Cloud2.tukihost.com

This server is used to host the attendance system of the company. You can see the details of this server using Webmin as well

- **Webmin**

URL: <https://cloud2.tukihost.com:10000/>

Username: root

Password: 6ezRd~zFn^mJL

- **Terminal/Termius**

IP: 68.183.237.66

Username: root

Password: 5.zMfqN@Msy9ysA

2.2.2.6 App.Flexyear.com

This server hosts the new version of attendance system and is used to run frontend only.

- **Terminal/Termius**

IP: 165.22.208.109

Username: root

Password: 5.zMfqN@Msy9ysA

APPENDIX

- **Google Drive Link:**

https://drive.google.com/drive/folders/11dulpUQyqNpV1aJidVWP6IjMcqQA_rPRL

- **List of Projects in the Servers:**

- **TCA Portal Server**

```
activepro    caringpro    complify     eceministry  jitsi-docker  onestop      proworks2    rpspro       stable-6865.tar.gz  syncs        tcaporta     ultrapro
activepro.tar.gz  centos      covidsafe4all  excellpro    jsplatinums  options      regispro     saasrest     staffforce         sync-simple  tcaunderwriting  variablem
agnatrixtcaporta  changinlivespro  diversity    growthtcaportal  jsplatinumpro  premierpartners  restore      samuelpro    staffinghub         syncup       test          villageofhope
alliedpro         choosejoy     downloads    integritypro  leading       proadvance    righthire   solvis       staffmastergroup   syncup-dev   toprank       workhorse
armada            citywaypro    drepro       istaffing    memcached     proworks     riyanev     southeast    summit             syncuponlinetest  trcportal      xlstaffingtcapor
```

- **Chat Server**

```
chat ec2-user jitsi
```

- **demo.bentray.work**

```
bitwarden  common_db_meroreport.sql  demo  focalboard  mattermost  meroreport  mysirani  rocketchat  scripts  test_meroreport
```

- **Cloud2.tukihost.com**

```
accswift  bentrayervices  changannepal  fbinary  foton  ghcc  iglobefdn  madhyanha  meroreport  nepalhosting  sunmedical  transiti  tukidomain  whmcs
balkhuve  budsufutu  dolphinm  flexyear  gfst  hotelmoonlight  lifeactive  mazdanepal  mysqlbkup  newsajako  taecons  tukiblogs  tukihost
```

- **Test.bentraytech.com**

```
HLH  anshuinvest  bentraytech  cert  chatmodule  common  fotonnepal  jenkins  mgmotors  mysirani_dev  neef  sfcl  syncup_test  tukiblog
accswift  ayurveda_dev  bluelotus  chabdi  cms  crm  gct  kshamadevi  multiHTTPS  nagarjuna  scrapping  stepbystep  tewa  tukihost
```

- **Bentray.Work**

```
accswift  brt  chroot  crmfrontend  getyourtrip  globalrecruitment  manpower  nml  otsbackend  reoccurrence  stepbystep  testfront
authentication  centos  cms  dmsfrontend  hotelmoonlight  icleanlink  mgmotors  onlinetest  otsfrontend  saasbackend  sync-up  testsaas
bentray  chat  configuration  downloads  gforce  kshamadevigroup  nagarjuna  onlinetest.bentray.work  phpmyadmin  simplifyfrontend  syncup-onlinetest  tewa
bluelotus  chatmodule  crm  drm  ghcc  proadvancerabin  simplifyfrontend  syncupfront  user-authentication
```

REVERSE PROXY WITH SSL SAMPLE:

The reverse proxy that we use is jwilder reverse proxy. Along with reverse proxy, we have set up nginx letsencrypt companion that will generate SSL for the domains. Before running the compose file, first you need to create a network “rev-proxy” as shown below.

```
version: "3"
services:
  rev-proxy:
    image: jwilder/nginx-proxy
    container_name: rev-proxy
    ports:
      - "80:80"
      - "443:443"
    volumes:
      - "/var/run/docker.sock:/tmp/docker.sock:ro"
      - "./certs-volume:/etc/nginx/certs:rw"
      - "./nginx-host:/etc/nginx/vhost.d"
      - "./nginx-html:/usr/share/nginx/html"
      - "./nginx-conf:/etc/nginx/conf.d"

  nginx-letsencrypt-companion:
    restart: always
    container_name: nginx-letsencrypt-companion
    image: jracs/letsencrypt-nginx-proxy-companion
    volumes:
      - "./certs-volume:/etc/nginx/certs"
      - "/var/run/docker.sock:/var/run/docker.sock:ro"
    volumes_from:
      - "rev-proxy"

volumes:
  certs-volume:
  nginx-host:
  nginx-html:
  nginx-conf:

networks:
  default:
    external:
      name: rev-proxy
```

To use this reverse proxy, you need to define network mode in your project compose file along with the environment as shown below.

```
network_mode: "rev-proxy"
environment:
  - VIRTUAL_HOST=backend.ansutest.bentraytech.com
  - LETSENCRYPT_HOST=backend.ansutest.bentraytech.com
```

GITLAB CI/CD

CI/CD is implemented using Gitlab. For that, you need to have access to gitlab projects. Follow along to set up CI/CD.

1. Access the project where you want to set up the CI/CD.
2. Go to the CI/CD section shown on the left hand side of the project.
3. Select Editor to configure the CI/CD.
4. Select the respective branch for Development and Production Server.

```
1 image: php:latest
2
3 before_script:
4   - apt-get update -qq
5   - apt-get install -qq git
6   - "which ssh-agent || ( apt-get install -qq openssh-client )"
7   - eval $(ssh-agent -s)
8   - ssh-add <(echo "$SSH_PRIVATE_KEY")
9   - mkdir -p ~/.ssh
10  - touch ~/.ssh/known_hosts
11  - ssh-keyscan -H 165.22.209.140 >> ~/.ssh/known_hosts
12  - '[[ -f /.dockerenv ]] && echo -e "Host *\n\tStrictHostKeyChecking no\n\n" > ~/.ssh/config'
13  #- chmod 600
14 deploy_stage:
15   stage: deploy
16   environment: Staging
17   only:
18     - dev
19   script:
20     - ls
21     - scp -o StrictHostKeyChecking=no -r * root@165.22.209.140:/home/anshuinvest/anshuinvest/ansu
```

5. You need to define **SSH_PRIVATE_KEY** in CI/CD variables.
6. Go to Setting-> CI/CD just below the CI/CD to configure the variables.




Variables

Variables store information, like passwords and secret keys, that you can use in job scripts. [Learn more.](#)

Variables can be:

- **Protected:** Only exposed to protected branches or tags.
- **Masked:** Hidden in job logs. Must match masking requirements. [Learn more.](#)

Environment variables are configured by your administrator to be **protected** by default.

Type	↑ Key	Value	Protected	Masked	Environments	
Variable	SSH_PRIVATE_KEY 	***** 	×	×	All (default)	

[Add variable](#) [Reveal values](#)

7. SSH_PRIVATE_KEY will be the remote server ssh private key. (id_rsa)
8. Login into the remote server and copy the ssh public key to authorized_keys.
Use the command below: `cat id_rsa.pub >> ~/.ssh/authorized_keys`
9. Commit the CI/CD file.
10. You have set up the CI/CD using Gitlab.

TOOLS USED FOR TROUBLESHOOT

- <https://www.pingdom.com/>
- <https://mxtoolbox.com/>
- <https://hide.me/en/proxy>
- <https://intodns.com>
- Terminal