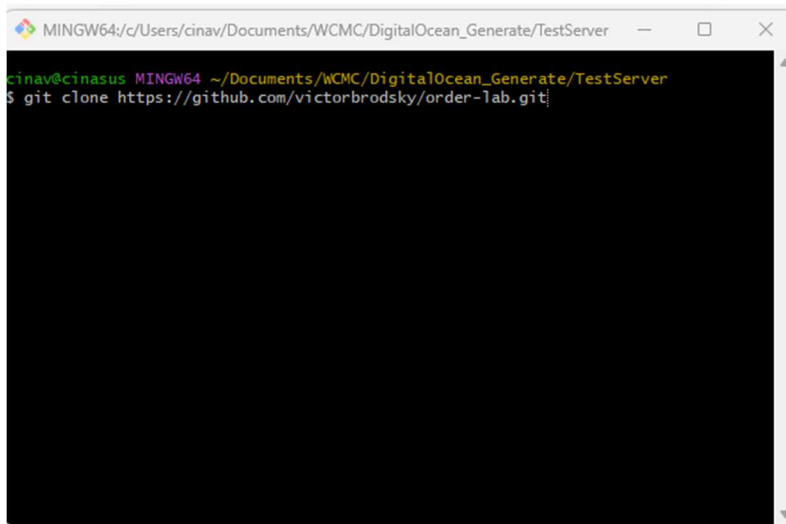


Installation Instructions

I. Digital Ocean via Packer

- create new folder "New Server" and open a command console from this folder (i.e. 'Git Bash Here')



```
MINGW64~/c/Users/cinav/Documents/WCMC/DigitalOcean_Generate/TestServer
cinav@cinavus MINGW64 ~/Documents/WCMC/DigitalOcean_Generate/TestServer
$ git clone https://github.com/victorbrodsky/order-lab.git
```

- Clone ORDER repository from git:
`git clone https://github.com/victorbrodsky/order-lab.git`
- Go to packer directory:
`cd order-lab/packer/`
- Run packer installation script:
`bash deploy-order-digital-ocean.sh --token mytoken -os alma9 --protocol https`

Example to install Alma9 server with multi-tenancy for “https://view.online” domain:

```
bash deploy-order-digital-ocean.sh --token mytoken --os alma9 --protocol https --domainname
view.online --sslcertificate installcertbot --email oli2002@med.cornell.edu --sshfingerprint
1a:55:66:2b:11:3c:11:b4:33:d5:99:44:f6:33:22:11 --multitenant haproxy | tee deploypacker.log
```

Notes:

--sshfingerprint option is required for certbot and certificate installation.

To use ssh, ssh key should be added to DigitalOcean: Settings->Security->Add SSH Key

If certbot installation will fail during automated installation, certbot can be installed later manually via Droplet Console or by running `doctl`.

Via Droplet Console:

Go to DigitalOcean: Droplets -> click on newly generated droplet -> click Access -> Launch Droplet Console

The screenshot shows the DigitalOcean Droplet Console interface. At the top, the droplet name is **packer-1745532836**, located in **NYC3** with **AlmaLinux** OS. It has **2 GB Memory** and **40 GB Disk**. A blue **Upsize Droplet** button and a green **ON** toggle are visible. Below this, network details are shown: **ipv4: 68.183.51.53**, **ipv6: Enable now**, **Private IP: 10.132.0.9**, and **Reserved IP: Enable now**. A **Console** link with a terminal icon is on the right. On the left, a sidebar lists management options: **Graphs**, **Access** (highlighted), **Power**, **Volumes**, **Resize**, **Networking**, **Backups**, **Snapshots**, **Kernel**, **History**, **Destroy**, **Tags**, and **Recovery**. The main content area has two sections: **Droplet Console** and **Recovery Console**. The **Droplet Console** section includes a text box with **Log in as... root** and a blue **Launch Droplet Console** button. The **Recovery Console** section has a grey **Launch Recovery Console** button.

In console, run these command:

```
cd /srv/order-lab-homepagemanager/packer/
```

```
bash /srv/order-lab-homepagemanager/packer/install-certbot.sh view.online installcerbot  
oli2002@med.cornell.edu haproxy
```

The screenshot shows a terminal window titled "packer-1745532836 - DigitalOcean Droplet Web Console — Mozilla Firefox". The URL bar shows https://cloud.digitalocean.com/droplets/491394663/terminal/ui?os_user=root. The terminal output is as follows:
Activate the web console with: `systemctl enable --now cockpit.socket`
Last login: Fri Apr 25 13:56:23 2025 from 162.243.190.66
[root@packer-1745532836 ~]# cd /srv/order-lab-homepagemanager/packer/
[root@packer-1745532836 packer]#
[root@packer-1745532836 packer]# bash /srv/order-lab-homepagemanager/packer/install-certbot.sh view.online installcerbot
oli2002@med.cornell.edu haproxy

This script will install Certbot, set up the certificate, and configure haproxy.cfg to apply it.

By running doctl from local PC:

```
doctl compute ssh "packer-1745521900" --ssh-key-path ./sshkey --ssh-command 'bash /srv/order-lab-homepagemanager/packer/install-certbot.sh view.online installcertbot oli2002@med.cornell.edu haproxy'
```

It might be required to resize the droplet to 4 GB Memory (php-fpm is not working properly if the memory size is 2 GB):

The screenshot shows the DigitalOcean Droplet management interface. At the top, there is a navigation bar with a "Back to Droplets" link. Below this, the droplet's name "packer-1745532836" is displayed, along with its specifications: "in Effort / 4 GB Memory / 40 GB Disk / NYC3 - AlmaLinux packer-1745532836". A blue "Upsize Droplet" button and a toggle switch labeled "OFF" are visible. Below the header, the droplet's network details are shown: "ipv4: 68.183.51.53", "ipv6: Enable now", "Private IP: 10.132.0.9", "Reserved IP: Enable now", and "Console: [icon]". On the left side, there is a sidebar menu with options: "Graphs", "Access", "Power", "Volumes", "Resize", "Networking", "Backups", "Snapshots", "Kernel", "History", "Destroy", "Tags", and "Recovery". The main content area is divided into two sections. The top section, titled "Droplet Console", contains the message "Your Droplet must be powered on to access the Recovery Console." and a blue "Power On" button. The bottom section, titled "Recovery Console", also contains the same message and a blue "Power On" button. A "Learn" link with a book icon is located in the top right corner of the "Droplet Console" section.

To initialize tenant run in browser:

<http://view.online/directory/admin/first-time-login-generation-init>

II. Installation ORDER on the provided server with already installed Alma9 or RHEL9

If the server is already provided with Alma9 or RHEL9, go to the server console and run:

```
sudo yum install -y git
cd /srv
git clone https://github.com/victorbrodsky/order-lab.git
cd order-lab/packer/
```

Run `alma9_install.sh` to install Apache, Postgresql, PHP, required utilities:
`bash alma9_install.sh dbusername dbpassword protocol domain sslcertificate email multitenant`

Run `install-multitenancy.sh` to install multi-tenant ORDER system with HaProxy for `view.online` domain, the installation log will be stored in `multitenancy.log` file:
`bash install-multitenancy.sh -u dbusername -t dbpassword -m haproxy -p /srv -s none -d none -e none -l none | tee multitenancy.log`

Example to install Apache, Postgresql, PHP, required utilities, multi-tenant ORDER system with HaProxy for `view.online` domain:

```
bash alma9_install.sh symfony symfony https view.online installcertbot oli2002@med.cornell.edu haproxy
bash install-multitenancy.sh -u symfony -t symfony -m haproxy -p /srv -s none -d none -e none -l none |
tee multitenancy.log
```

To initialize tenant run in browser:

`http://view.online/directory/admin/first-time-login-generation-init`

In all cases: if the web site is not opening, check haproxy and php-fpm status on the server:

```
sudo systemctl status haproxy
sudo systemctl status php-fpm
sudo systemctl status httpdhomepagemanager
sudo systemctl status httpdtenantmanager
sudo systemctl status httpdtenantapp1
sudo systemctl status httpdtenantapp2
sudo systemctl status httpdtenantappdemo
```

III. Initial Configuration

By default, '/' points to the home page manager tenant. First, initialize this tenant by this url:

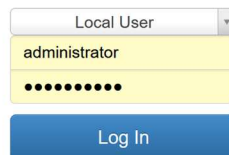
<https://view.online/directory/admin/first-time-login-generation-init/>

This will initialize the home page manager and create administrator user. Default password is '1234567890', and you will be redirected to the login page:

Employee Directory

Congratulations! You have successfully installed the system. Please select 'Local User' from the menu below and enter the user name 'administrator' and the password '1234567890' to log in. Then visit [/directory/settings/] and run the initialization scripts 1 through 7 in the listed order (skipping 4a and 4b) in the Miscellaneous section. After that, change the administrator password!

Deploy script run successfully: Cache cleared, Assets dumped



A login form with a dropdown menu set to 'Local User'. Below the dropdown, the username 'administrator' is entered in a yellow field, and the password is represented by ten dots in another yellow field. A blue 'Log In' button is at the bottom.

Please use your [CWID](#) to log in.
[Request a new account](#) if you can't log in.

Select 'Local User', enter the default username 'administrator' and password '1234567890' and click 'Log In'.

On the first login, you will be redirected to the initializing page.

The screenshot shows a web browser window with the URL <https://view.online/tenant-manager/directory/settings/initial-configuration>. The page has a navigation bar with links: Home, List Current, List Previous, Add, Employment dates, Download, Search, and Admin. The main content area is divided into two sections: 'Server Instance' and 'Institution Settings'.

Server Instance

- Environment:** dev
- Internal Connection Channel (http or https; Clearing Cache is required):** http
- External Connection Channel (if using HaProxy, internal connection channel should be 'http' and external connection channel should be 'https'; http or https; Clearing Cache is required):** http
- Live Site Root URL (such as "http://my.server.com/order"):** (empty field)
- Server Network Accessibility and Role:** Intranet (Solo)
- User Group:** WCM Department of Pathology and Laboratory Medicine

Institution Settings

- Institution Name (Copyright Link in Footer):** Cornell University
- Institution URL (Copyright Link in Footer):** http://www.cornell.edu/
- Institution Name (Instance Owner in Footer):** Weill Cornell Medicine

Set all required initializing fields:

For multi-tenancy with HaProxy set the 'Internal Connection Channel' to 'https' and 'External Connection Channel' to 'https'.

Set the 'Environment' to 'live, If it is a live server.

Set 'E-Mail address for the Administrator account'.

It is also recommended to change administrator's password and in case of 'test' or 'dev' environment, set the 'Reroute all outgoing emails...' field to your email to prevent sending automatic emails to other users.

Click 'Update'.

You will be redirected to the home page. On the top 'Admin' menu, click 'Site Settings' and click 'Miscellaneous'. For homepage and tenant manager, pre-populate only lists 1), 2), 3).

Go to the cron jobs (Admin -> Cron Jobs) and review all cron jobs, for example, remove all cron jobs for homepage and tenant manager tenants, except 'Status cron'.

Cron Jobs Management

Email spooling settings

| Parameter | Value | Action |
|--|-------|--------|
| Use email spooling (Instead of sending every email directly to the SMTP server individually, add outgoing emails to a queue and then periodically send the queued emails. This makes form submission appear faster.) | No | Edit |
| Frequency of sending emails in the queue (in minutes between eruptions). cron:swift cron job status: not found | 15 | Edit |

Create cron jobs (Email spooling, Fellowship Import, Fellowship Verification, Unpaid Invoices, Project Expiration, Project Sync)

| Parameter | Value | Action |
|--|---|--------|
| Email spooling cron | cron:swift cron job status: not found | Remove |
| Fellowship Applications Import cron | cron:importfellapp cron job status: not found | Remove |
| Fellowship Applications Verification Import cron | cron:verifyimport cron job status: not found | Remove |
| Unpaid Invoices Reminder Email cron | cron:invoice-reminder-emails cron job status: not found | Remove |
| Project Expiration Reminder Email cron | cron:expiration-reminder-emails cron job status: not found | Remove |
| Project Sync with public server | cron:project-sync cron job status: not found | Remove |

Create status cron job (check if the system in the maintenance mode)

| Parameter | Value | Action |
|-------------------------------------|---|--------|
| Status cron (check for Maintenance) | cron:status cron job status: */30 * * * * /bin/php /srv/order-lab-homepagemanager/orderflex/bin/console cron:status --env=prod | Remove |

To apply the update, click on Admin -> Run deploy script deploy.sh.

Similarly, initialize the tenant manager:

<https://view.online/tenant-manager/directory/admin/first-time-login-generation-init/>

All tenants are already pre-installed on the server. Utility script `sync_tenants.sh` allows to synchronize the source code across all tenants and perform some other tasks. This script can be run from any tenant's orderflex folder.

For example, after installation, we need to add all versions to the symfony's database migration: go to `/srv/order-lab-tenantapp1/orderflex` and run: `bash sync_tenants.sh /srv addallversions`

Plus, we need to create db.config for backup/restore: `bash sync_tenants.sh /srv dbconfig`

Some other options:

basic - sync the source code, run deploy script, check migration status, install python requirements

full – all from basic, plus sync database, sync yarn for react js

yarn – only sync yarn for react js

sync – only sync the source code

dbstatus – only check database sync status

dbmigrate – only sync the database schemas. Symfony calls it 'migrate'

composer – only install all symphony updates

python – only install python requirements

dbconfig – only create db.config for postgres management by python (backup/restore)

addallversions – only add all versions to the symfony's database migration, so the command php bin/console doctrine:migrations:status will not show any new migrations

IV. Multi-Tenancy Configuration

Go to the 'Tenancy Configuration' page:

<https://view.online/tenant-manager/directory/settings/tenant-manager/configure/>

Click on 'Update DB Configuration from the server'. This will populate multi-tenant settings in Database from the server configurations: haproxy.cfg and tenant's httpd (httpdhomepagemanager, httpdtenantmanager, httpdtenantapp1 ...). You will be redirected to the Tenancy Configuration page with all tenant's settings:

The screenshot shows a web browser window with the address bar displaying <https://view.online/tenant-manager/directory/settings/tenant-manager/configure/>. The page title is "Tenancy Configuration". Below the title, there is a list of tenants with links to "Initialize Tenant". The main content area is titled "Tenant(s)" and contains a form for configuring a tenant. The form includes the following fields:

- Tenant's data source:
- Tenant name (without spaces and special characters):
- Display Order:
- URL Slug:
- Tenant Port:
- Database Host:
- Database Name:
- Database User:
- Database Password:
- Platform Administrator Account User Name:
- Tenant Institution Title:
- Tenant Department Title:
- Billing Tenant Administrator Contact Name:
- Billing Tenant Administrator Contact Email:
- Operational Tenant Administrator Contact Name:
- Operational Tenant Administrator Contact Email:
- Show on Homepage: ☒
- Enabled: ☒
- Primary Tenant: ☐

On this page, click 'Initialize Tenant' to make initial initialization of the tenant.

The tenant's parameters can be edited on the edit page: click 'Edit'.

To make a tenant to be accessible by /, check 'Primary Tenant' check box. For example, check 'Primary Tenant' for 'c/wcm/pathology' tenant and this tenant can be accessed by / path.

After making changes in Database, click 'Update Server Configuration from DB' button.

If the apache user does not have a permission to update haproxy.cfg file, then haproxy.cfg must be changed manually:

Replace

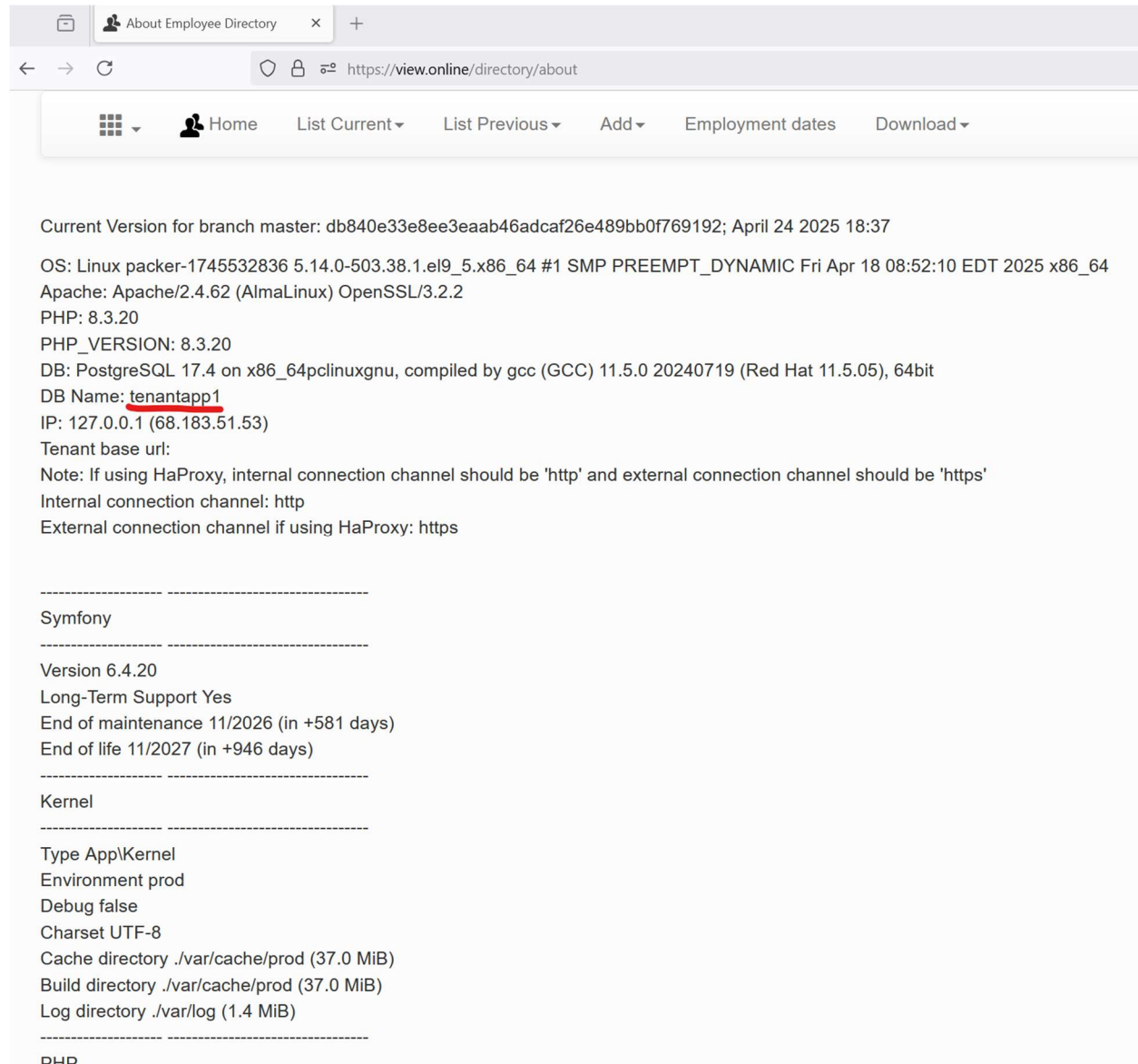
'use_backend homedirectorymanager_backend if homedirectorymanager_url'

by

'use_backend tenantapp1_backend if homedirectorymanager_url'.

Run systemctl restart haproxy.

After that, both url / and https://view.online/c/wcm/pathology/ will lead to tenantapp1, it is possible to verify that on 'about' page <https://view.online/directory/about>:



Current Version for branch master: db840e33e8ee3eaab46adcaf26e489bb0f769192; April 24 2025 18:37

OS: Linux packer-1745532836 5.14.0-503.38.1.el9_5.x86_64 #1 SMP PREEMPT_DYNAMIC Fri Apr 18 08:52:10 EDT 2025 x86_64

Apache: Apache/2.4.62 (AlmaLinux) OpenSSL/3.2.2

PHP: 8.3.20

PHP_VERSION: 8.3.20

DB: PostgreSQL 17.4 on x86_64pclinuxgnu, compiled by gcc (GCC) 11.5.0 20240719 (Red Hat 11.5.05), 64bit

DB Name: tenantapp1

IP: 127.0.0.1 (68.183.51.53)

Tenant base url:

Note: If using HaProxy, internal connection channel should be 'http' and external connection channel should be 'https'

Internal connection channel: http

External connection channel if using HaProxy: https

Symfony

Version 6.4.20

Long-Term Support Yes

End of maintenance 11/2026 (in +581 days)

End of life 11/2027 (in +946 days)

Kernel

Type App\Kernel

Environment prod

Debug false

Charset UTF-8

Cache directory ./var/cache/prod (37.0 MiB)

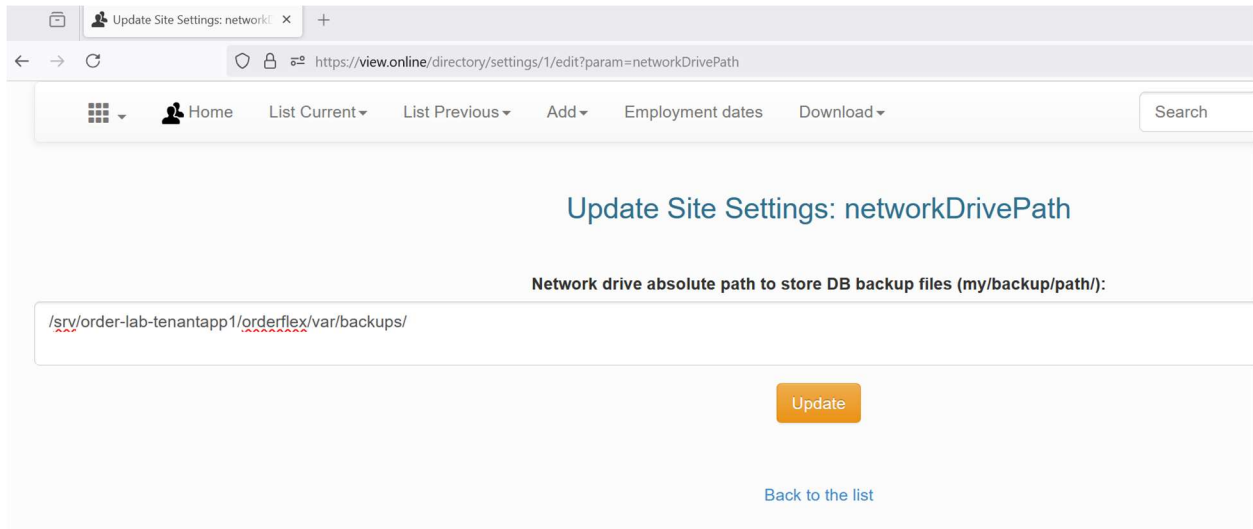
Build directory ./var/cache/prod (37.0 MiB)

Log directory ./var/log (1.4 MiB)

PHP

V. Restore Database and Uploaded files

First set the 'Network drive absolute path to store DB backup files' on the Site Settings page, for example to `/srv/order-lab-tenantapp1/orderflex/var/backups/`



Update Site Settings: networkDrivePath

Network drive absolute path to store DB backup files (my/backup/path/):

`/srv/order-lab-tenantapp1/orderflex/var/backups/`

Update

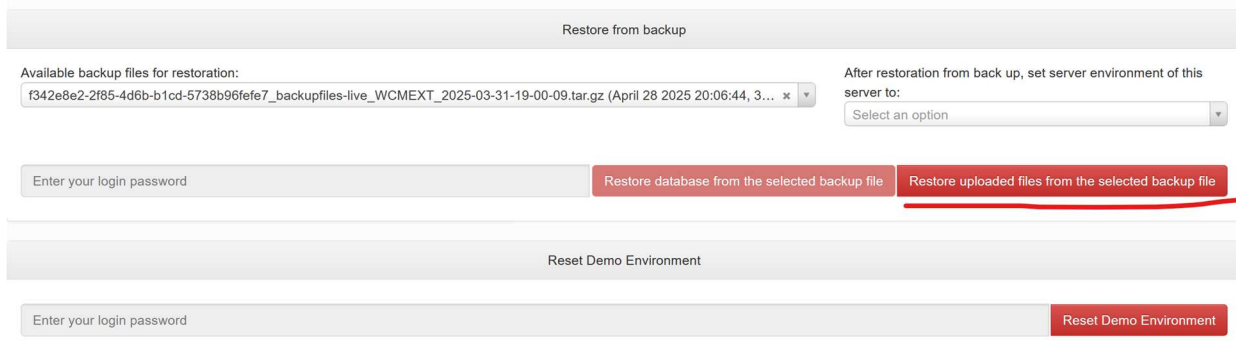
[Back to the list](#)

Before, restoration in the Site Setting, set the environment to 'dev', to prevent pulling the fellowship applications from Google drive.

Then go to the 'Manual Backup/Restore' page <https://view.online/directory/manual-backup-restore/>.

Upload file upload archived file using 'Upload backup files' section. The database backup file should include 'backupdb' in its name, while uploaded files should contain 'backupfiles'.

In 'Restore from backup' section, choose upload backup file, confirm password and click 'Restore uploaded files...' button.



Restore from backup

Available backup files for restoration:

`f342e8e2-2f85-4d6b-b1cd-5738b96fefe7_backupfiles-live_WCMEXT_2025-03-31-19-00-09.tar.gz` (April 28 2025 20:06:44, 3... x)

After restoration from back up, set server environment of this server to:

Select an option

Enter your login password

Restore database from the selected backup file

Restore uploaded files from the selected backup file

Reset Demo Environment

Enter your login password

Reset Demo Environment

Restore database backup: choose the backup, set environment to 'dev', confirm password and click 'Restore database ...'. After restoration, verification and testing, the environment should be changed to 'live' if it is a live server.

VI. Backup/Restore via PgAdmin

For large database size and live server, it is recommended to shut down haproxy service 'systemctl stop haproxy' and use pgAdmin.

To access database postgres config file pg_hba.conf might need to be modified to have online access:

```
host all all 0.0.0.0/0 md5
```

```
listen_addresses='*'
```

After PgAdmin use, change back trust to md5

replace 'md5' with 'trust' in /var/lib/pgsql/17/data/pg_hba.conf

To take effect changes run: systemctl restart postgresql-17

pgAdmin access: sudo systemctl stop firewallld

To Backup a database using pgAdmin4 (To get a data dump using a pgAdmin4):

Open pgAdmin 4 and connect to your server.

Select the database you wish to back up from the left sidebar.

Right-click on the database and choose Backup.

Specify the file path and name (i.e. backupdb-live-12March2025.sql) for the backup file in the Filename field.

Choose the format you want (e.g., Custom, Tar, Plain, Directory) from the format dropdown.

Click Backup

To restore database backup:

"Create" > "Database..." > Save

Right-click on the 'recoverdb' database you just created and select "Restore."

In the dialog box that appears, navigate to the location of your backup file ('backup.sql') and select it.

Click "Restore" to initiate the restoration process.

After DB restore check (reference function restoreDBWrapper):

env

connectionChannel (set http for HaProxy)

urlConnectionChannel (set https for HaProxy if using ssl certificate)

networkDrivePath

monitorScript
mailerdeliveryaddresses

Optional:
to test the functionality set mailerDeliveryAddresses to oli2002
to avoid stealing fellapp, turn off imports