Single Node Installations

Installing a Standalone Server

Overview

Our installer can handle the installation of environments in which only one server is available. This environment consists of a single server with a microk8s deployment in which we deploy our Element Server Suite to, resulting in a fully functioning version of our platform.

To get started with a standalone installation, there are several things that need to be considered and this guide will work through them:

- Operating System
- Postgresql Database
- TURN Server
- SSL Certificates
- Extra configuration items

Once these areas have been covered, you'll be ready to install your standalone server!

Server minimum requirements

CPU and Memory

The installer binary requires support for the x86_64 architecture. The Standalone deployment will need 2 GiB of memory to run properly the OS and microk8s. The ESS deployment resource usage is described in ESS Sizing.

Disk size

It is **crucial** that your storage provider supports `fsync` for data integrity.

- `/var` : 50Gb
- `/data/element-deployment` : It will contain your Synapse medias. The path can be adjusted in the UI. Please refer to ESS Sizing page to find an estimation of the expecting size growth.
- `/data/postgres` : It will contain your postgres servers data. The path can be adjusted in the U. Please refer to ESS Sizing page to find an estimation of the expecting size growth.

Operating System

We provide support for Ubuntu 20.04 and Red Hat Enterprise Linux (RHEL) versions 8 and 9 and suggest that you start there as well. Please note that the installer binary requires support for the x86_64 architecture.

You can grab an Ubuntu iso here:

https://releases.ubuntu.com/20.04.3/ubuntu-20.04.3-live-server-amd64.iso

You can get Red Hat Enterprise Linux 8 with a Developer Subscription at:

https://access.redhat.com/downloads/content/479/ver=/rhel---8/8.7/x86_64/product-software

Ubuntu Specific instructions

Make sure to select docker as a package option. Do set up ssh.

Once you log in, please run:

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install git
```

The installer requires that you run it as a non-root user who has sudo permissions. Please make sure that you have a user who can use `sudo`. If you wanted to make a user called `element-demo` that can use `sudo`, the following commands (run as root) would achieve that:

```
useradd element-demo
gpasswd -a element-demo sudo
```

The installer also requires that your non-root user has a home directory in /home.

RHEL Specific instructions

Make sure to select "Container Management" in the "Additional Software" section.

Once you log in, please run:

```
sudo yum update -y
sudo yum install python39-pip python39-devel make gcc git -y
sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm -y
sudo update-alternatives --config python3
```

You should also follow the steps linked here to Install microk8s on RHEL, or included below, if you run into `Error: System does not fully support snapd: cannot mount squashfs image using "squashfs"`:

1. Install the EPEL repository

- **RHEL9**:

  ```
  sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-
  9.noarch.rpm
  sudo dnf upgrade
  ```

- **RHEL8**:

  ```
  sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-
  8.noarch.rpm
  sudo dnf upgrade
  ```

1. Install Snap, enable main snap communication socket and enable classic snap support

```
sudo yum install snapd
sudo systemctl enable --now snapd.socket
sudo ln -s /var/lib/snapd/snap /snap
```

1. Reboot
2. (Optional) Install `microk8s` using `sudo snap install microk8s --classic`, the installer will do this for you otherwise.

On the `update-alternatives` command, if you see more than one option, select the option with a command string of `/usr/bin/python3.9`.

The installer requires that you run it as a non-root user who has sudo permissions. Please make sure that you have a user who can use `sudo`. If you wanted to make a user called `element-demo` that can use `sudo`, the following commands (run as root) would achieve that:

```
useradd element-demo
gpasswd -a element-demo wheel
```

The installer also requires that your non-root user has a home directory in /home.

Kernel modules

microk8s rquires the kernel module `nf_conntrack` to be enabled.

```
if ! grep nf_conntrack /proc/modules; then
    echo "nf_conntrack" | sudo tee --append /etc/modules
    sudo modprobe nf_conntrack
fi
```

Migrating from our older installer

If you have previously used installer versions 2023-03.01 and earlier, you will need to run our migration script to convert your previous configuration to the new format that is used with our UI based installer. This script became available in 2023-03.02, so you must have at least that version or higher of the graphical installer for this to work.

NOTE: Before running the migration script, we highly recommend that you take a backup or snapshot of your working environment. While we have tested the migration script against several configurations at this point, we have not tested for all of the combinations of configuration that the previous installer allowed. We expect that migration will be a quick process for most customers, but in the event that something goes wrong, you'll want to be able to get back to a known good state through a backup or snapshot.

NB: If you are using group sync, you cannot presently migrate to the graphical installer. We are working to address the issues with migrating group sync and will remove this note once we have those addressed.

If you have not used our installer before, you may safely ignore this section.

To run the migration script, please do the following:

```
chmod +x ./element-enterprise-graphical-installer-YYYY-MM.VERSION-gui.bin
./element-enterprise-graphical-installer-YYYY-MM.VERSION-gui.bin --import ~/.element-
onpremise-config
```

Make sure to replace `~/.element-onpremise-config` with the path that your actual configuration exists in. Further, replace `YYYY-MM.VERSION` with the appropriate tag for the installer you downloaded.

Once the import has finished, the GUI will start and you will be able to browse to the installer at one of the provided URLs, much as if you had started the installer without doing a migration as detailed in the following section.

Network Specifics

Element Enterprise On-Premise needs to bind and serve content over:

- Port 80 TCP
- Port 443 TCP
- Port 8443 TCP (Installer GUI)

microk8s needs internally to bind and serve content over:

- Port 16443 TCP
- Port 10250 TCP
- Port 10255 TCP
- Port 25000 TCP
- Port 12379 TCP
- Port 10257 TCP
- Port 10259 TCP
- Port 19001 TCP

For more information, see https://microk8s.io/docs/ports.

In a default Ubuntu installation, these ports are allowed through the firewall. You will need to ensure that these ports are passed through your firewall.

For RHEL instances with firewalld enabled, the installer will take care of opening these ports for you.

Further, you need to make sure that your host is able to access the following hosts on the internet:

- api.snapcraft.io
- *.snapcraftcontent.com
- gitlab.matrix.org
- gitlab-registry.matrix.org
- pypi.org
- docker.io
- *.docker.com
- get.helm.sh
- k8s.gcr.io
- cloud.google.com
- storage.googleapis.com
- registry.k8s.io
- fastly.net
- GitHub.com

In addition, you will also need to make sure that your host can access your distributions' package repositories. As these hostnames can vary, it is beyond the scope of this documentation to enumerate them.

Network Proxies

We also cover the case where you need to use a proxy to access the internet. Please see this article for more information: Configuring a microk8s Single Node Instance to Use a Network Proxy

Postgresql Database

The installation requires that you have a postgresql database with a locale of C and UTF8 encoding set up. See https://github.com/element-hq/synapse/blob/develop/docs/postgres.md#set-up-database for further details.

If you have this already, please make note of the database name, user, and password as you will need these to begin the installation.

If you do not already have a database, then the single node installer will set up PostgreSQL on your behalf.

Beginning the Installation

Head to https://ems.element.io/on-premise/download and download the latest installer. The installer will be called `element-enterprise-graphical-installer-YYYY-MM.VERSION-gui.bin`. You will take this file and copy it to the machine where you will be installing the Element Server Suite. Once you have this file on the machine in a directory accessible to your sudo-enabled user, you will run:

```
chmod +x ./element-enterprise-graphical-installer-YYYY-MM.VERSION-gui.bin
```

replacing the `YYYY-MM.VERSION` with the appropriate tag for the installer you downloaded.

Once you have done this, you will run:

```
./element-enterprise-graphical-installer-YYYY-MM.VERSION-gui.bin
```

replacing the `YYYY-MM.VERSION` with the appropriate tag for the installer you downloaded, and this will start a web server with the installer loaded.

You will see a message similar to:

```
[user@element-demo ~]$ ./element-enterprise-graphical-installer-2023-02.02-gui.bin
Testing network...

Using self-signed certificate with SHA-256 fingerprint:

F3:76:B3:2E:1B:B3:D2:20:3C:CD:D0:72:A3:5E:EC:4F:BC:3E:F5:71:37:0B:D7:68:36:2E:2C:AA:7A:F2:83:9
4

To start configuration open:
        https://192.168.122.47:8443 or https://10.1.185.64:8443 or https://127.0.0.1:8443
```

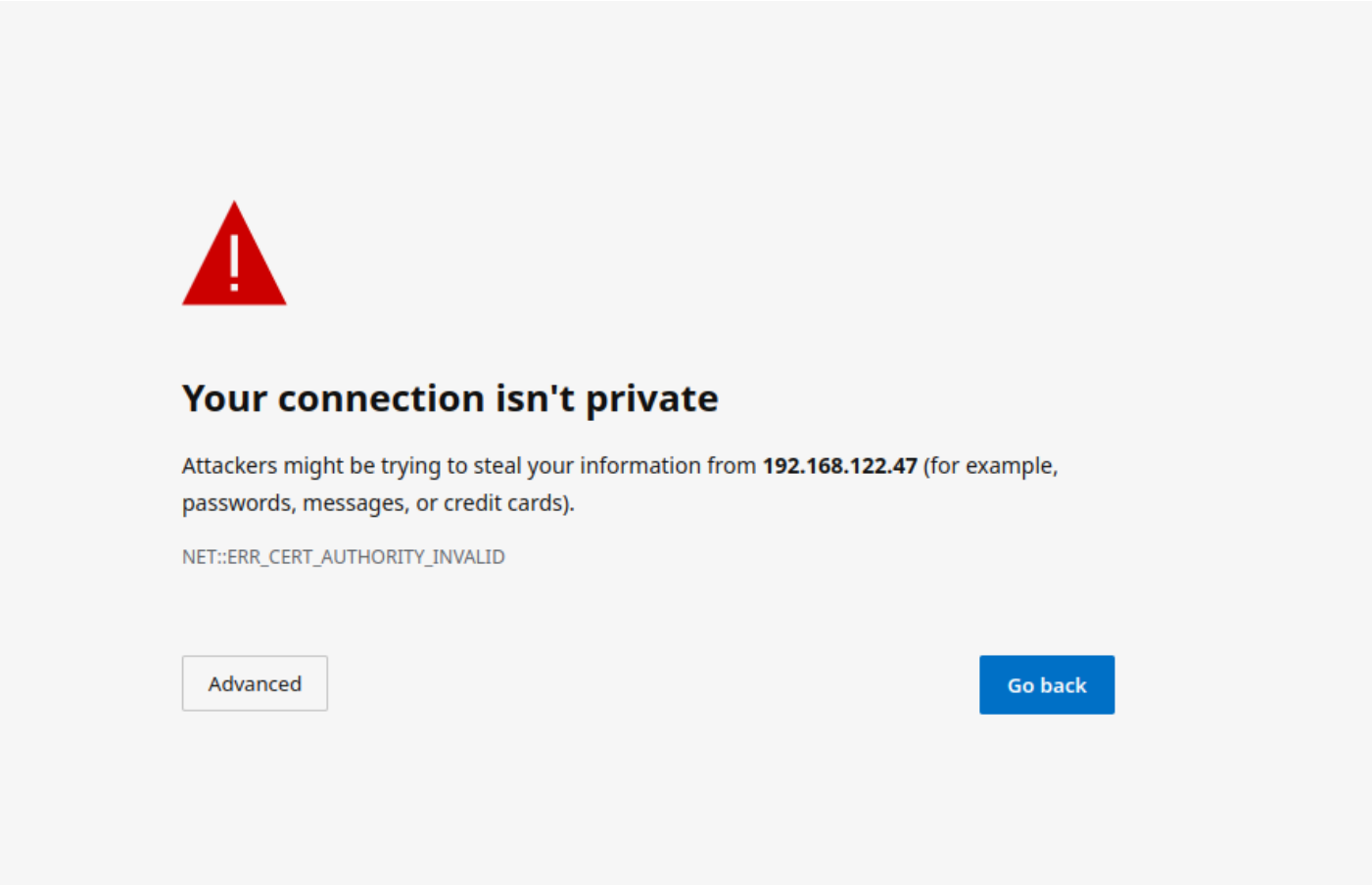
At this point, you will need to open a web browser and browse to one of these IPs. You may need to open port 8443 in your firewall to be able to access this address from a different machine.

If you are unable to open port 8443 or you are having difficulty connecting from a different machine, you may want to try ssh port forwarding in which you would run:

```
ssh <host> -L 8443:127.0.0.1:8443
```

replacing host with the IP address or hostname of the machine that is running the installer. At this point, with ssh connected in this manner, you should be able to use the https://127.0.0.1:8443 link as this will then forward that request to the installer box via ssh.

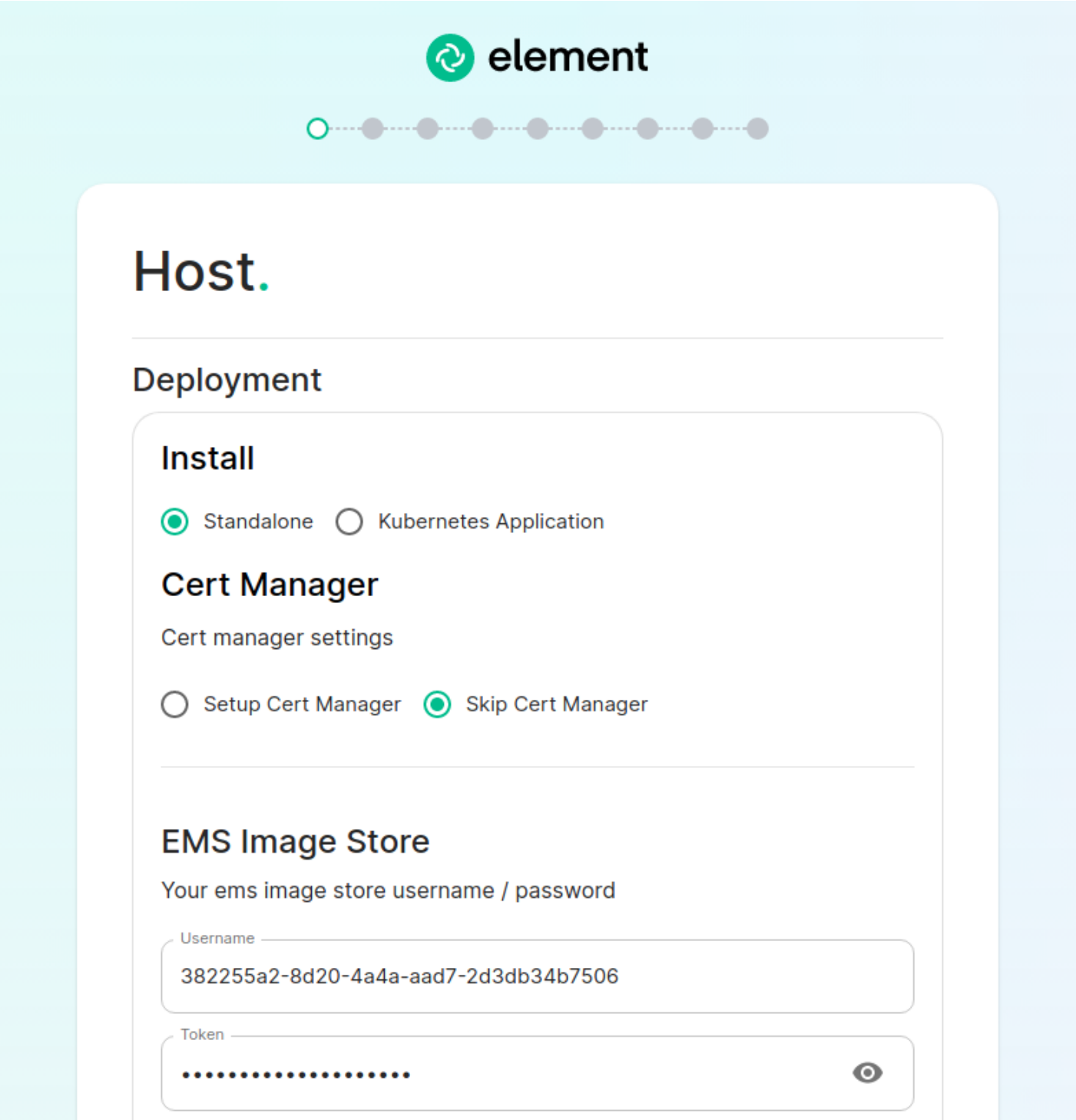
Upon loading this address for the first time, you may be greeted with a message informing you that your connection isn't private such as this:



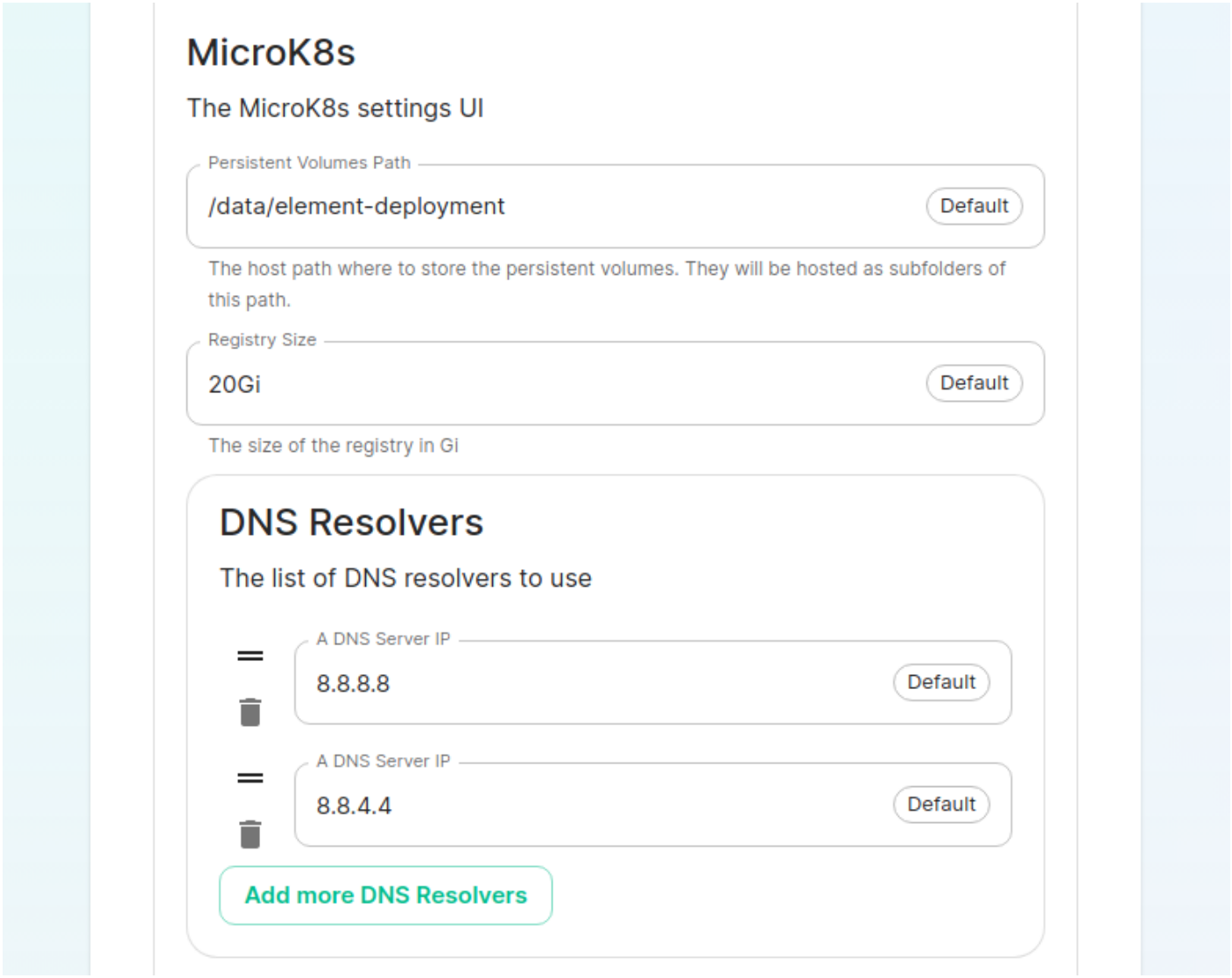
In this case, you'll need to click "Advanced" and then "Continue to <IP> (unsafe)" in order to view the installer. As the exact button names and links can vary between browsers, it would be hard for us to document them all, so you may have slightly different wording depending on your browser.

The Hosts Screen

The very first page that you come to is the host screen.

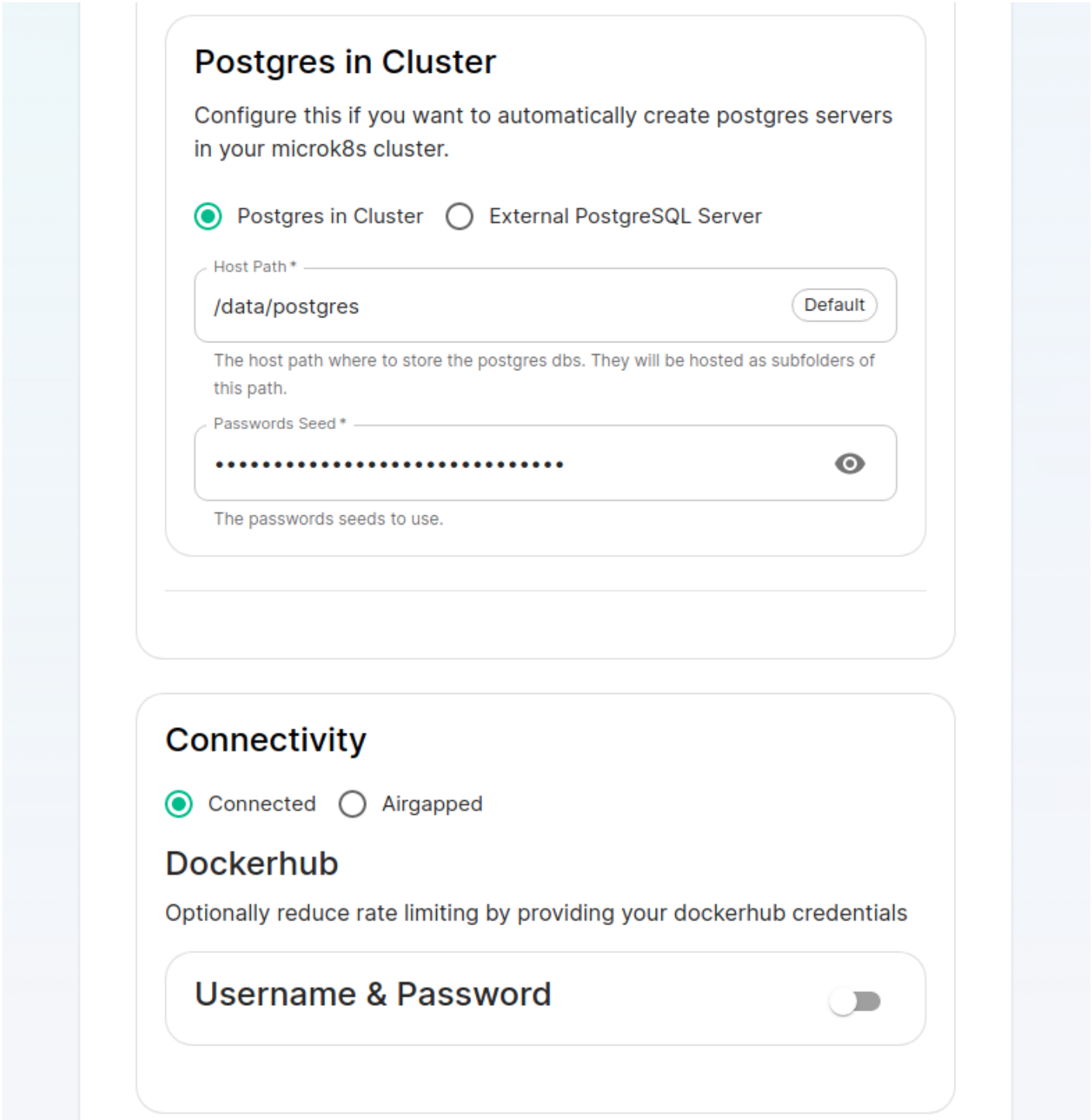


You will want to make sure that "Standalone" is selected. If you are using LetsEncrypt for your certificates, you will want to make sure that you select "Setup Cert Manager" and enter an email address for LetsEncrypt to associate with your certificates. If you are using custom certificates or electing to manage SSL certificates yourself, then you will want to select "Skip Cert Manager".
The very next prompt that you come to is for an EMS Image Store Username and Token. These are provided to you by element as access tokens for our enterprise container registries. If you have lost your token, you can always generate a new token at https://ems.element.io/on-premise/subscriptions.



The next option that you have is for microk8s. By default, microk8s will set up persistent volumes in /data/element-deployment and will allow 20GB of space to do this. For most installations, this is fine and can be left alone, but if you'd like to customize those options, you can do that here.

Next, we have DNS resolvers. The default DNS resolvers are Google (8.8.8.8 and 8.8.4.4). If you need to use your company's DNS servers, please change these values appropriately.

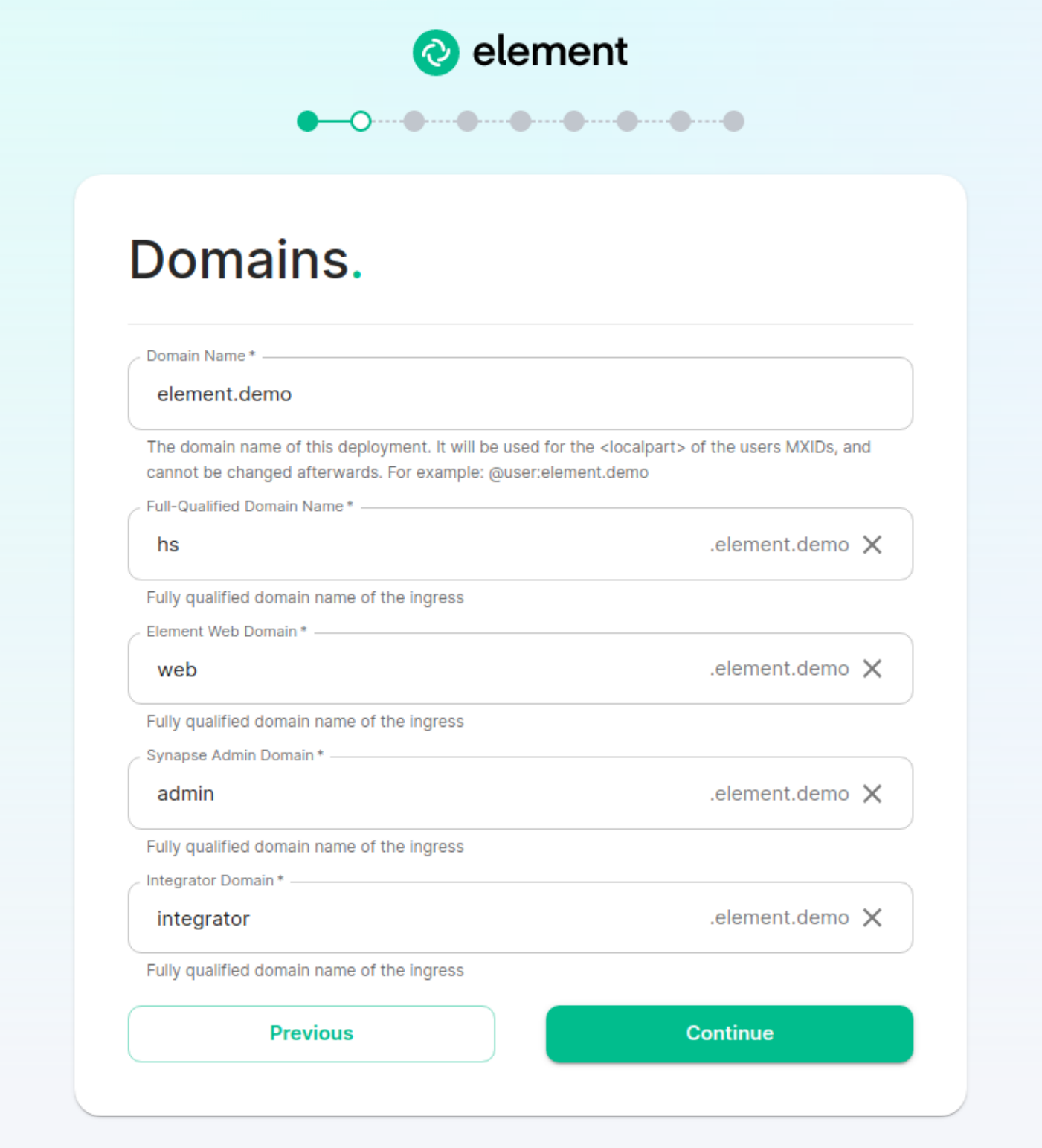


Next, we get the option to either have the installer install Postgres in your cluster or to use an external postgresql server. The Postgres in cluster option is only supported for our standalone installation and you should read our storage and backup guidelines for this configuration. At any rate, if you use the in cluster postgres, you will see that the installer defaults to /data/postgres and has generated a random password for your postgresql admin account. You can use the eye to see the password and you can certainly change this to whatever you'd like.

The final options on the host page are related to connectivity. For this guide, we are assuming "Connected" and you can leave that be. If you are doing "Airgapped", you would pick airgapped at this point and then please see the section on airgapped installations.

You are presented with the option to provide docker hub credentials. These are optional, but if you do not provide them, you may be rate limited by Docker and this could cause issues pulling container images.

The Domains Screen



On this page, we get to specify the domains for our installation. In this example, we have a domain name of airgap.local and this would mean our MXIDs would look like @kabbott:airgap.local.

Our domain page has checking to ensure that the host names resolve. Once you get green checks across the board, you can click continue.

The Certificates Screen

On the Certificates screen, you will provide SSL certificate information for well-known delegation, Synapse, Element Web, Synapse Admin, and Integrator.

2 options

Option 1: You already host a base domain `example.com` on a web server, then Well-Known Delegation should be set to `Externally Managed`.

Element clients need to be able to request `https://example.com/.well-known/matrix/client` to work properly.

The web server hosting the domain name should forward the requests to `.well-known/matrix/client` to the element enterprise server so that the wellKnownPod can serve it to the clients.

If that's not possible, the alternative is to copy the well known file directly on the example.com web server. The wellKnownPod will still be present but wont be used by any system.

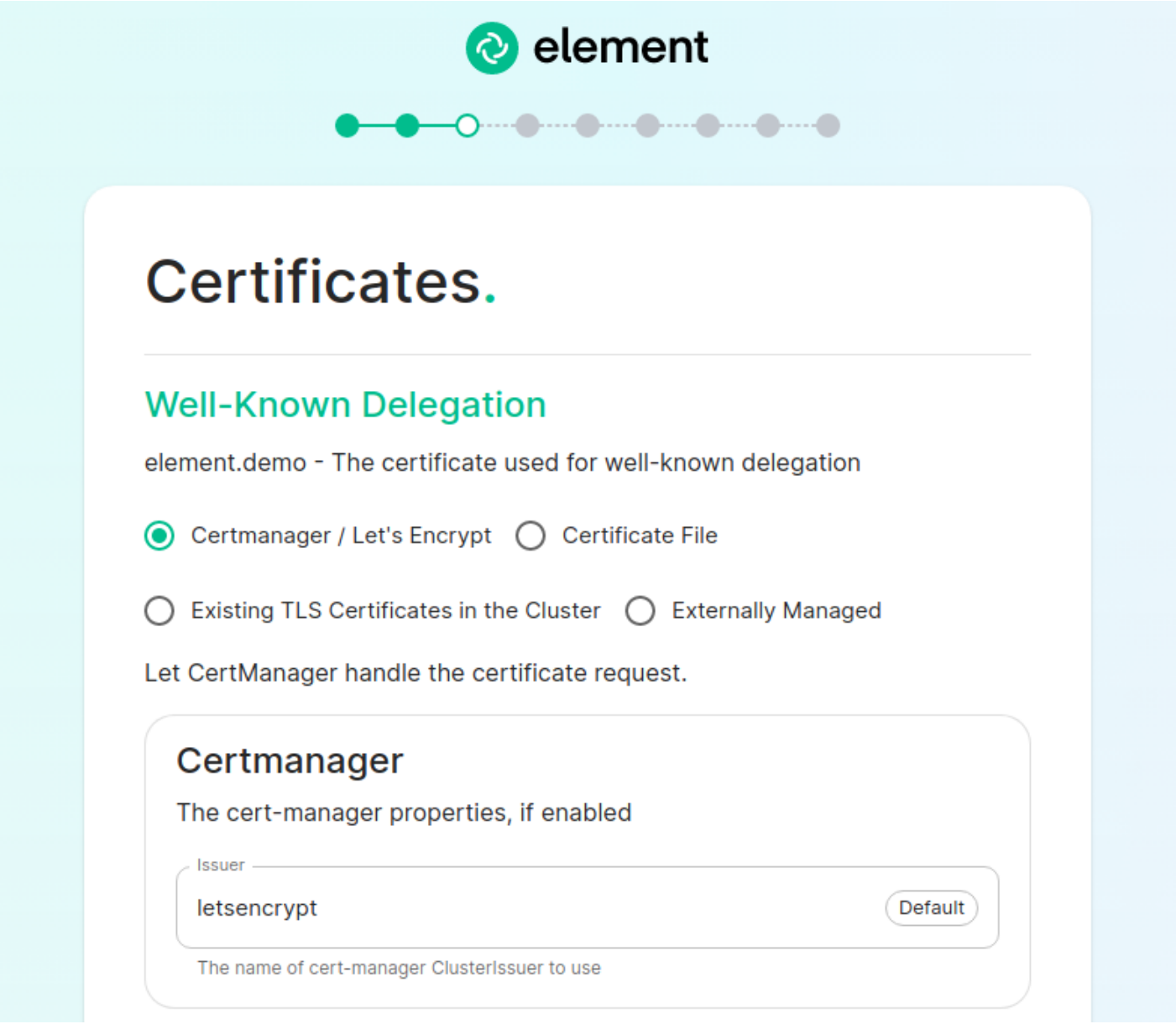
It cannot be set to `Certmanager / Let's Encrypt`.

Option 2: You don't already host a base domain `example.com`, then the wellKnownPod hosts the well-known file and serves the base domain `example.com`

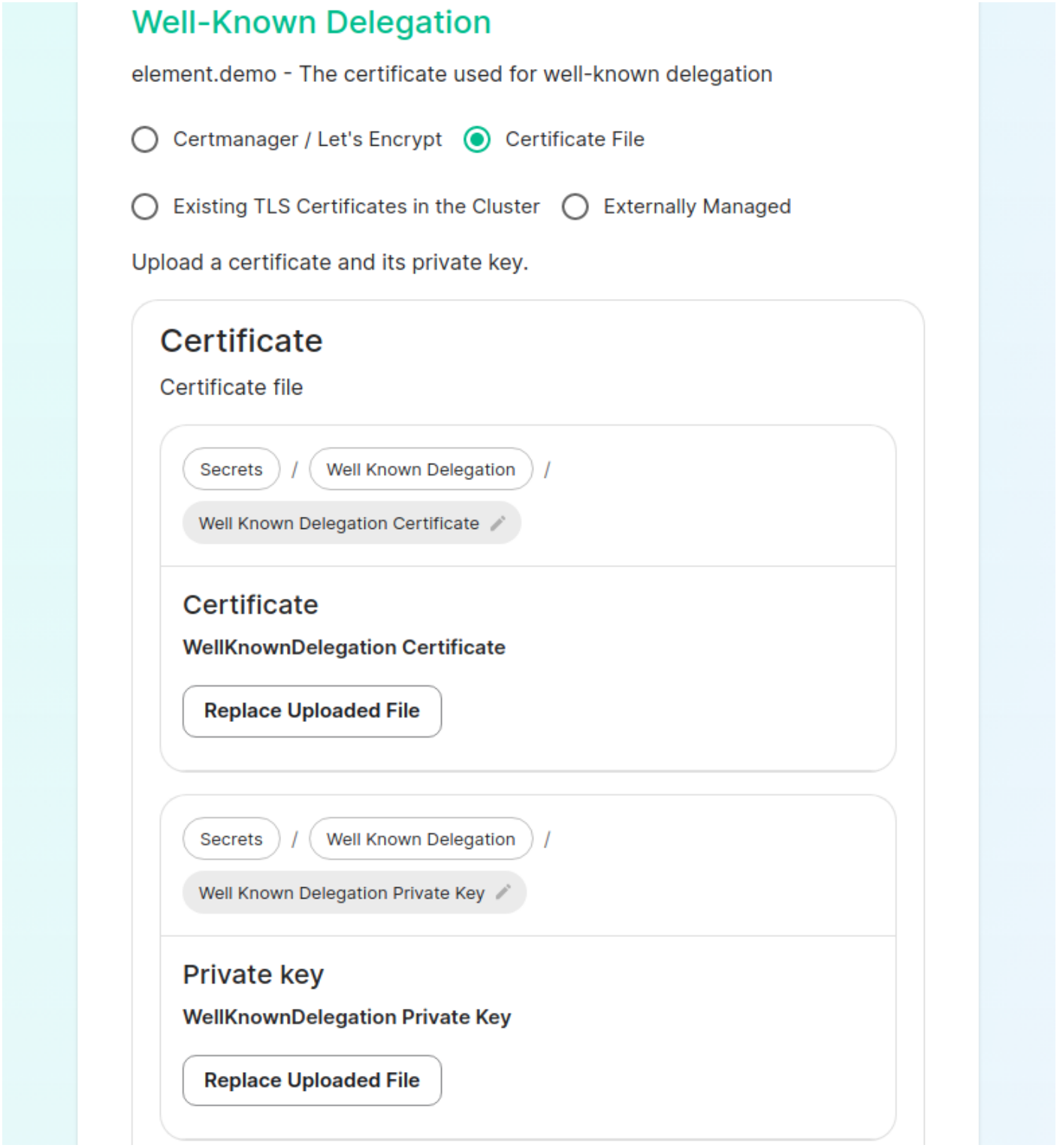
You can choose those 3 different settings:

- `Certmanager / Let's Encrypt`: the certificate for the base domain is signed by Let's Encrypt
- `Certificate File`: the certificate is signed by your own CA or by a public CA (Verisign, Sectigo,..)
- `Existing TLS Certificate in the Cluster`: certificate already uploaded in a secret

If you are using Let's Encrypt, then each of the sections should look like:



If you are using certificate files, then you will see a screen like:

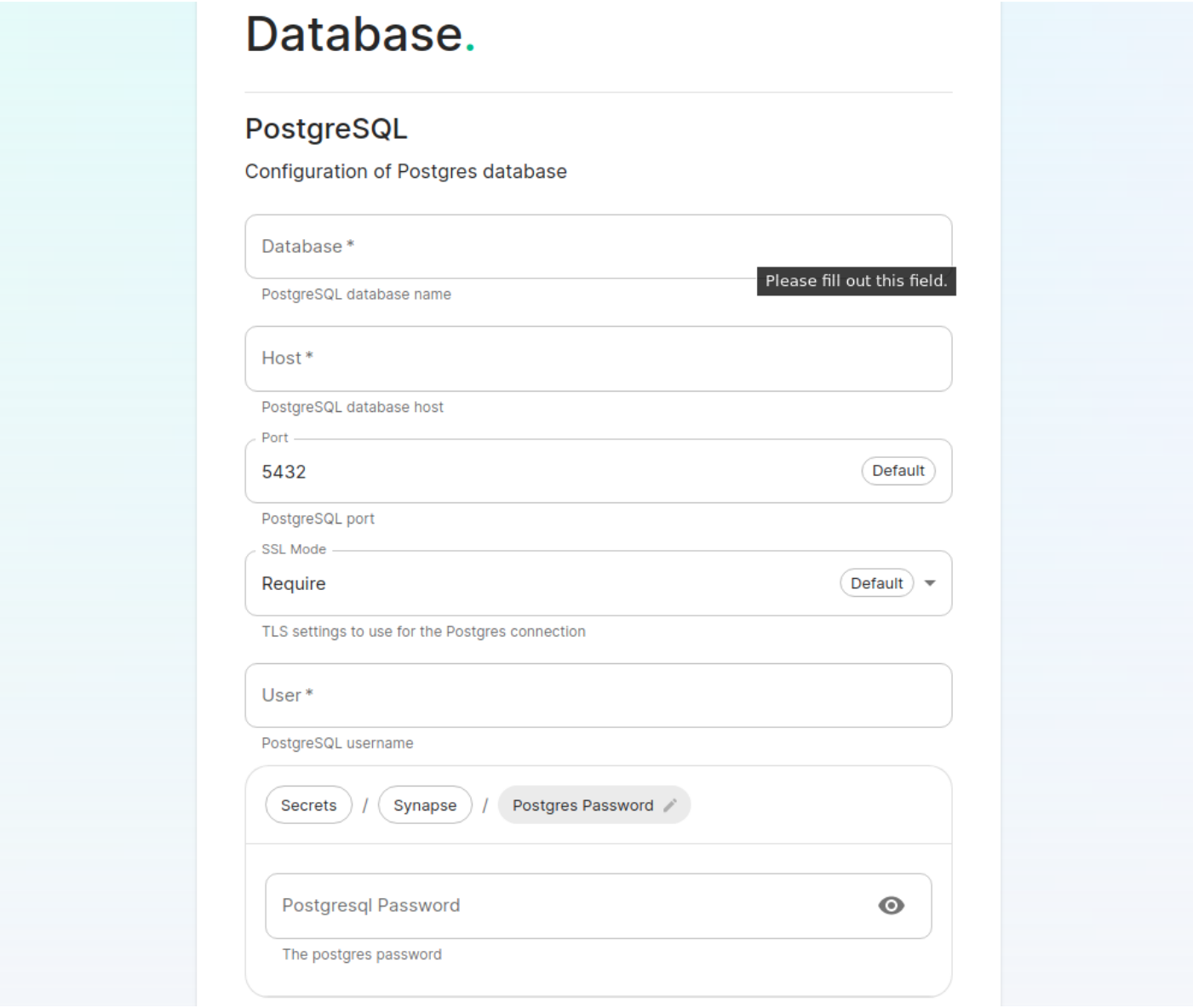


which allows you to upload a .crt and .key file for each host. These files must be in PEM encoding. Our installer does accept wildcard certificates.

Once you have completed the certificate section for each host on the page, you may click continue.

The Database Screen

If you have elected to have the installer configure PostgreSQL for you, then you will not see this screen and can skip this section.



If you are using an external database, then you will see this page, where we provide the option to specify the database name, the database host name, the port to connect to, the SSL mode to use, and finally, the username and password to connect with.

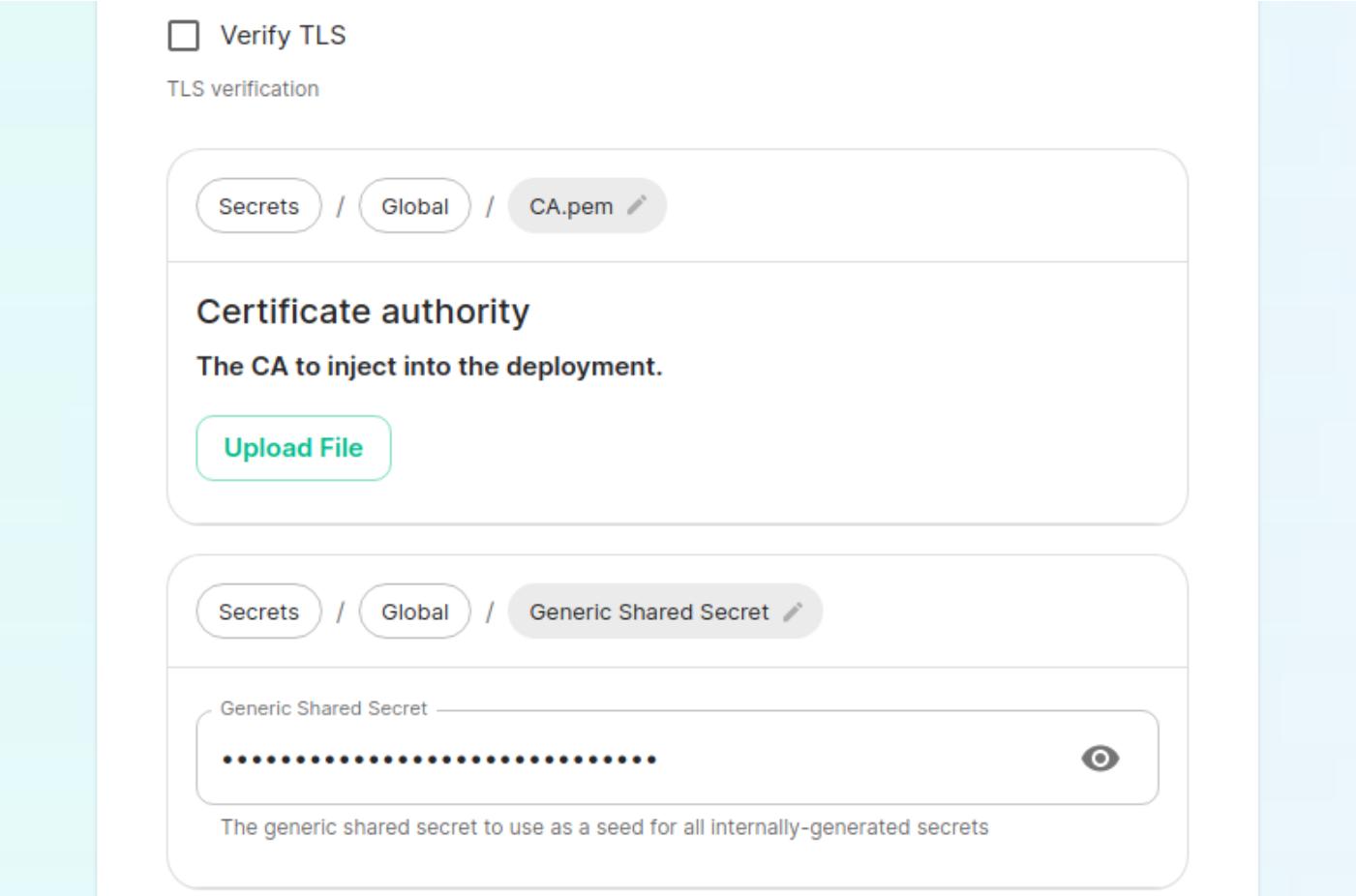
If your database is installed on the same server where Element is installed, you have to enter the server public IP address since the container is not sharing the host network namespace. Entering 127.0.0.1 will resolve to the container itself and cause the installation failing.

Once you have completed this section, you may click continue.

The Cluster Screen

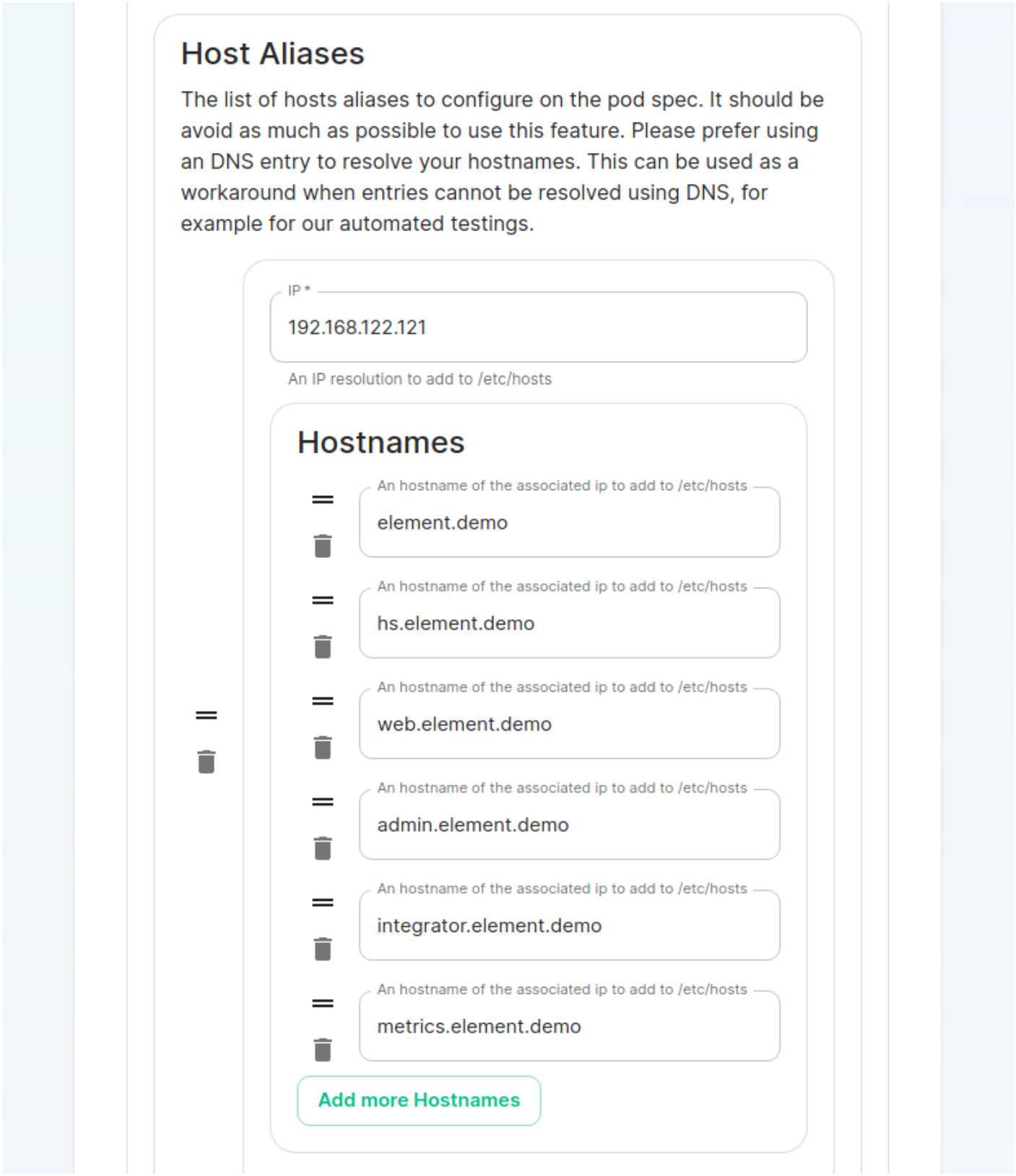
Most deployments can ignore this, however, if you want to change any microk8s cluster parameters, this is where to do it.

If you are in an environment where you have self-signed certificates, you will want to disable TLS verification, by clicking "Advanced" and then scrolling down and unchecking `Verify TLS`:



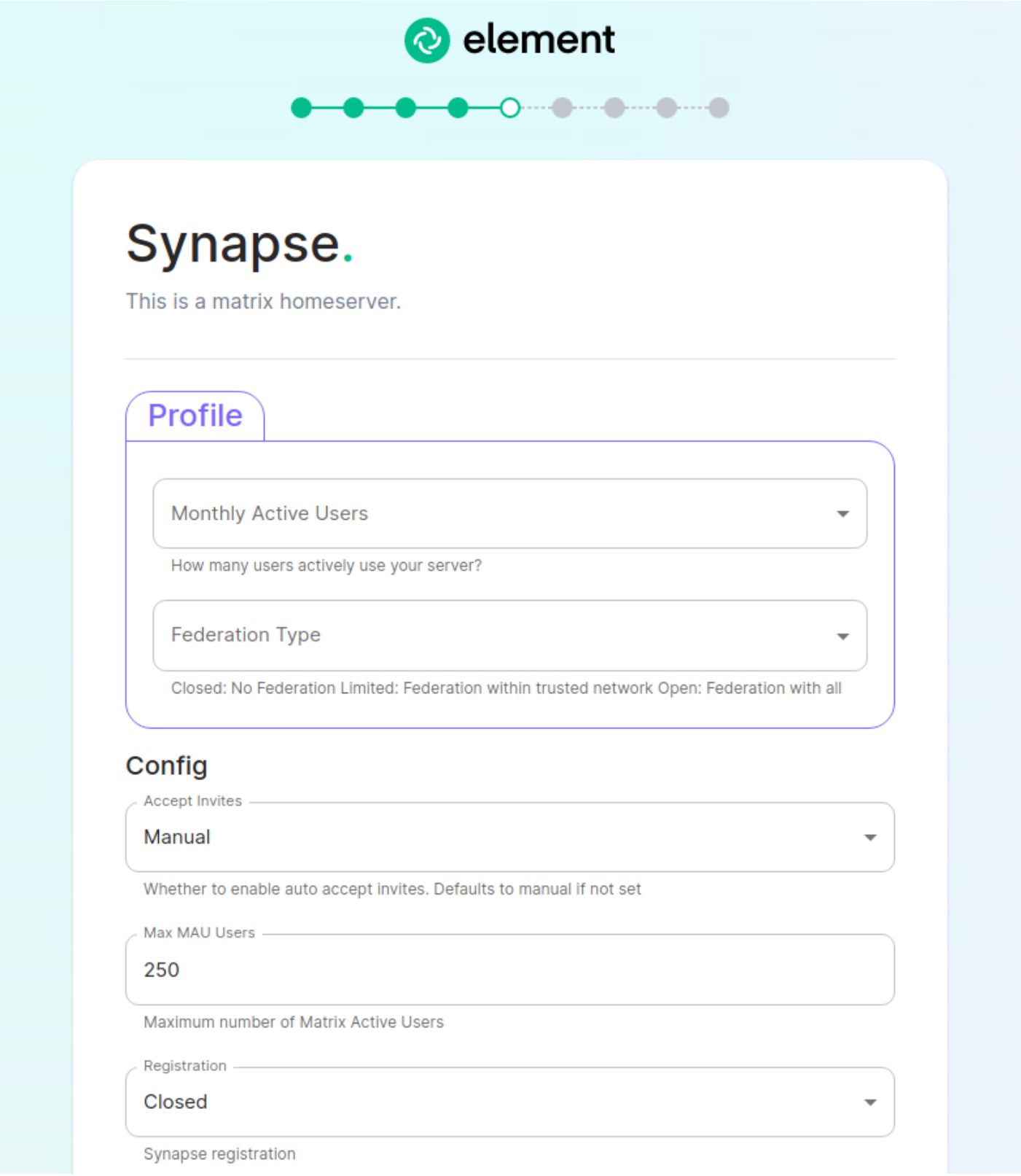
Please bear in mind that disabling TLS verification and using self-signed certificates is not recommended for production deployments.

If your host names are not DNS resolvable, you need to use host aliases and this can be set up here. You will also click "Advanced" and scroll down to the "Host Aliases" section in "k8s". In here, you will click "Add Host Aliases" and then you will specify an IP and host names that resolve to that IP as such:



When you are finished with this page, you can click continue.

The Synapse Screen



The first setting that you will come to is our built in performance profiles. Select the appropriate answers for "Monthly Active Users" and "Federation Type" to apply our best practices based on years of running Matrix homeservers.

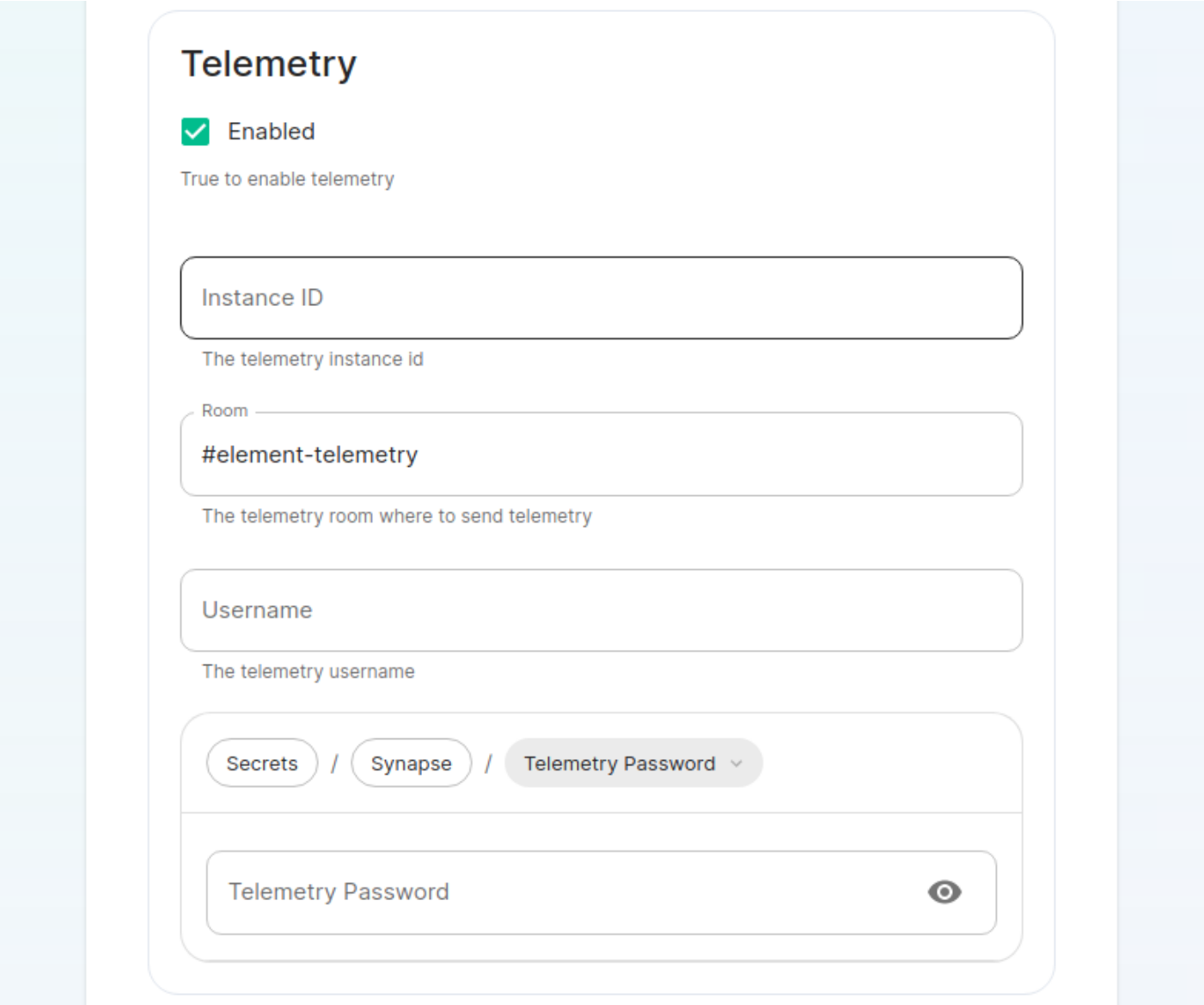
The next setting that you will see is whether you want to auto accept invites. The default of "Manual" will fit most use cases, but you are welcome to change this value.

The next setting is the maximum number of monthly active users (MAU) that you have purchased for your server. Your server will not allow you to go past this value. If you set this higher than your purchased MAU and you go

over your purchased MAU, you will need to true up with Element to cover the cost of the unpaid users.

The next setting concerns registration. A server with open registration on the open internet can become a target, so we default to closed registration. You will notice that there is a setting called "Custom" and this requires explicit custom settings in the additional configuration section. Unless instructed by Element, you will not need the "Custom" option and should instead pick "Closed" or "Open" depending on your needs.

After this, you will see that the installer has picked an admin password for you. You will want to use the eye icon to view the password and copy this down as you will use this with the user `onprem-admin-donotdelete` to log into the admin panel after installation.

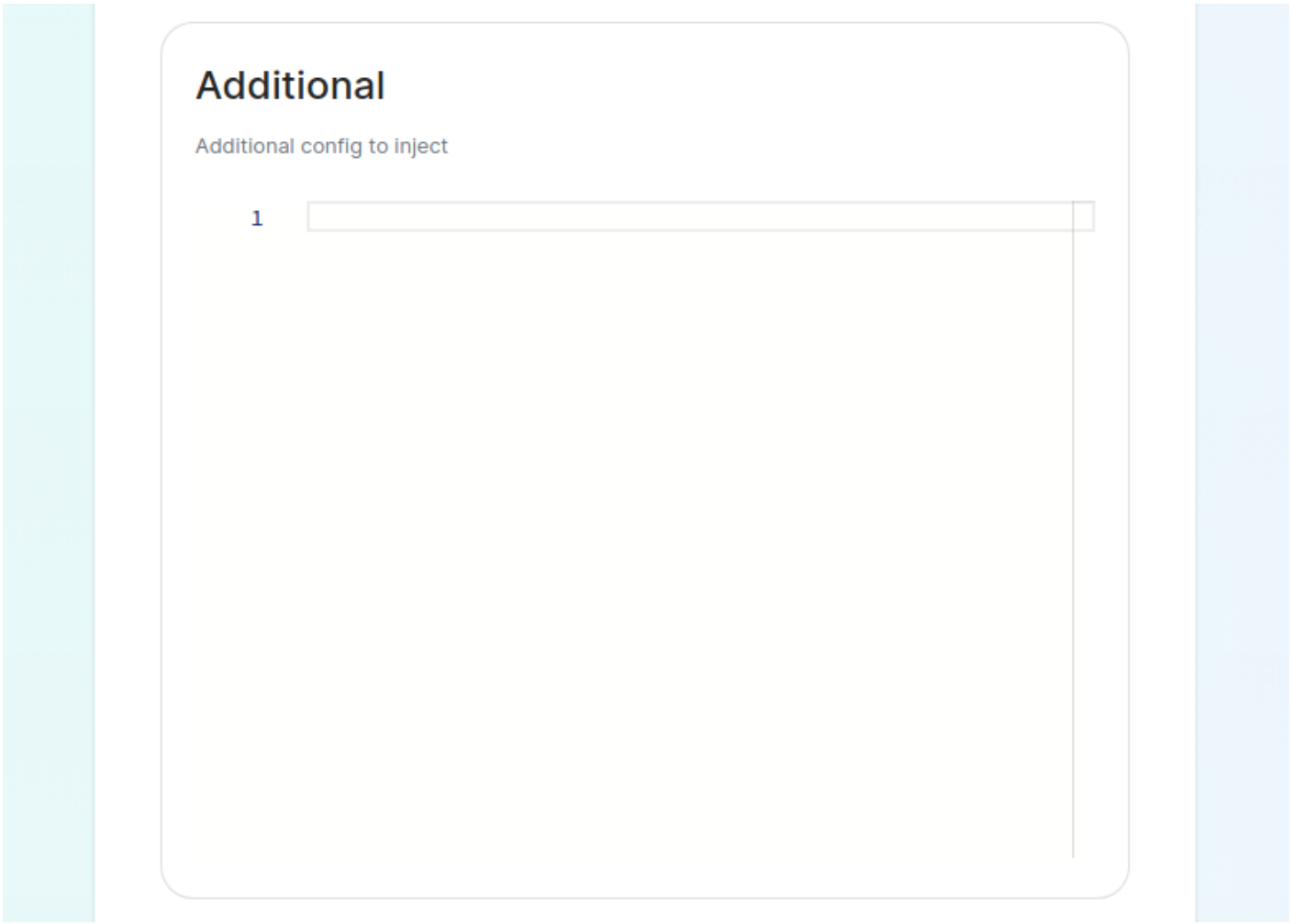


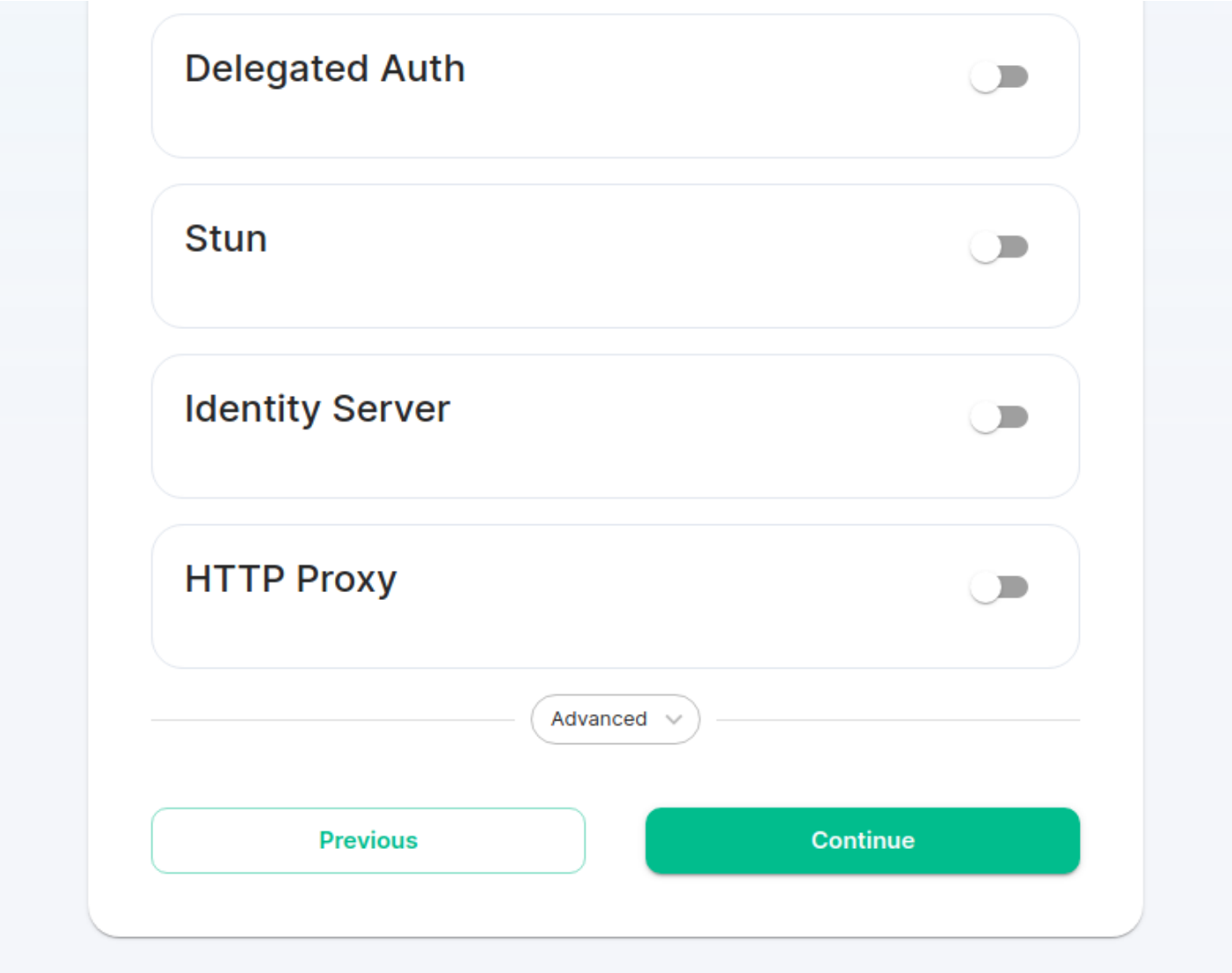
Continuing, we see telemetry. You should leave this enabled as you are required to report MAU to Element. In the event that you are installing into an environment without internet access, you may disable this so that it does not continue to try talking to Element. That said, you are still required to generate an MAU report at regular intervals and share that with Element.

For more information on the data that Element collects, please see: What Telemetry Data is Collected by Element?

Next, we have an advanced button, which allows you to configure further settings about synapse and the kubernetes environment in which it runs. The additional configuration text box allows you to inject additional

synapse configs (See homeserver.yaml and the Synapse Configuration Manual).



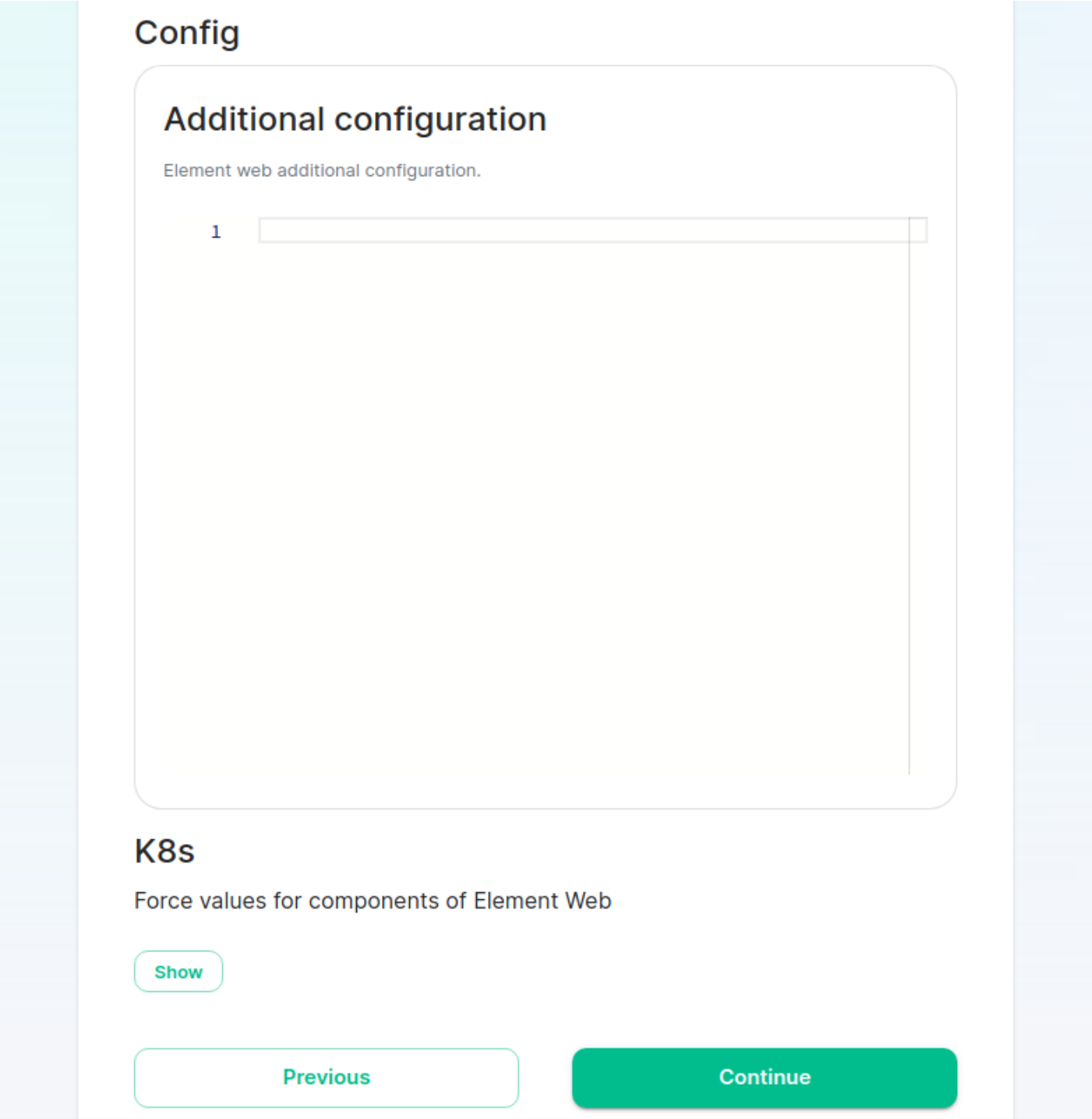


You can hit continue to go to the next screen.

The Element Web Screen

Most users will be able to simply click "Continue" here.

The Advanced section allows you to set any custom element web configurations you would like (config.json).



A common custom configuration would be configuring permalinks for Element, which we have documented here: Setting up Permalinks With the Installer

Further, it provides access to the k8s section, allowing you to explicitly set any microk8s cluster settings that you would like just for the element-web pod.

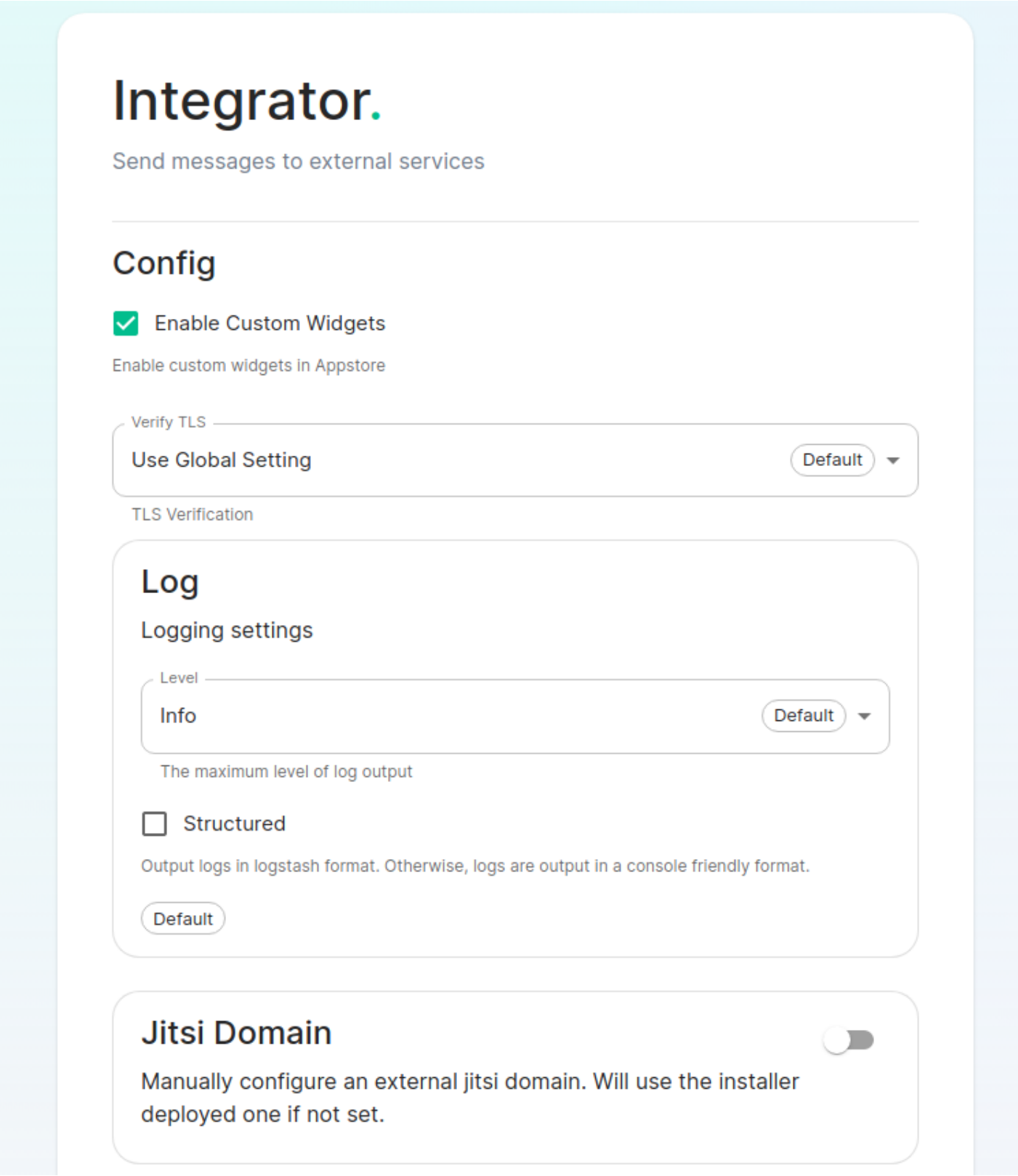
The Homeserver Admin

Most users will be able to simply click "Continue" here. The Advanced section allows you to explicitly set any microk8s cluster settings that you would like just for the synapse-admin-ui pod.

One word to note here is that if you are not using delegated authentication, then the initial username that an administrator will use to log into this dashboard post-installation is `onprem-admin-donotdelete`. You can find the password for this user on the Synapse page in the "Admin Password" field.

If you are using delegated authentication, you will need to assign a user admin rights as detailed in this article: How do I give a user admin rights when I am using delegated authentication and cannot log into the admin console?

The Integrator Screen



On this page, you can set up Integrator, the integrations manager.

The first option allows you to choose whether users can add custom widgets to their rooms with the integrator or not.

The next option allows you to specify which Jitsi instance the Jitsi widget will create conferences on.

The verify TLS option allows you to set this specifically for Integrator, regardless of what you set on the cluster screen.

The logging section allows you to set the log level and whether the output should be structured or not.

The Advanced section allows you to explicitly set any microk8s cluster settings that you would like just for the integrator pods.

Click "Continue to go to the next screen".

The Integrations Screen

This screen is where you can install any available integrations.

Some of these integrations will have "YAML" next to them. When you see this designation, this integration requires making settings in YAML, much like the old installer. However, with this installer, these YAML files are pre-populated and often only involve a few changes.

If you do not see a "YAML" designation next to the integration then this means that will use regular GUI elements to configure this integration.

Over time, we will do the work required to move the integrations with "YAML" next to them to the new GUI format.

For specifics on configuring well known delegation, please see Setting Up Well Known Delegation

For specifics on setting up Delegated Authentication, please see Setting up Delegated Authentication With the Installer

For specifics on setting up Group Sync, please see Setting up Group Sync with the Installer

For specifics on setting up GitLab, GitHub, and JIRA integrations, please see Setting up GitLab, GitHub, and JIRA Integrations With the Installer

For specifics on setting up Adminbot and Auditbot, please see: Setting up Adminbot and Auditbot

For specifics on setting up Hydrogen, please see: Setting Up Hydrogen

For specifics on pointing your installation at an existing Jitsi instance, please see Setting Up Jitsi and TURN With the Installer

If you do not have an existing TURN server or Jitsi server, our installer can configure these for you by following the extra steps in Setting Up Jitsi and TURN With the Installer

For specifics on configuring the Teams Bridge, please see Setting Up the Teams Bridge

For specifics on configuring the Telegram Bridge, please see Setting Up the Telegram Bridge

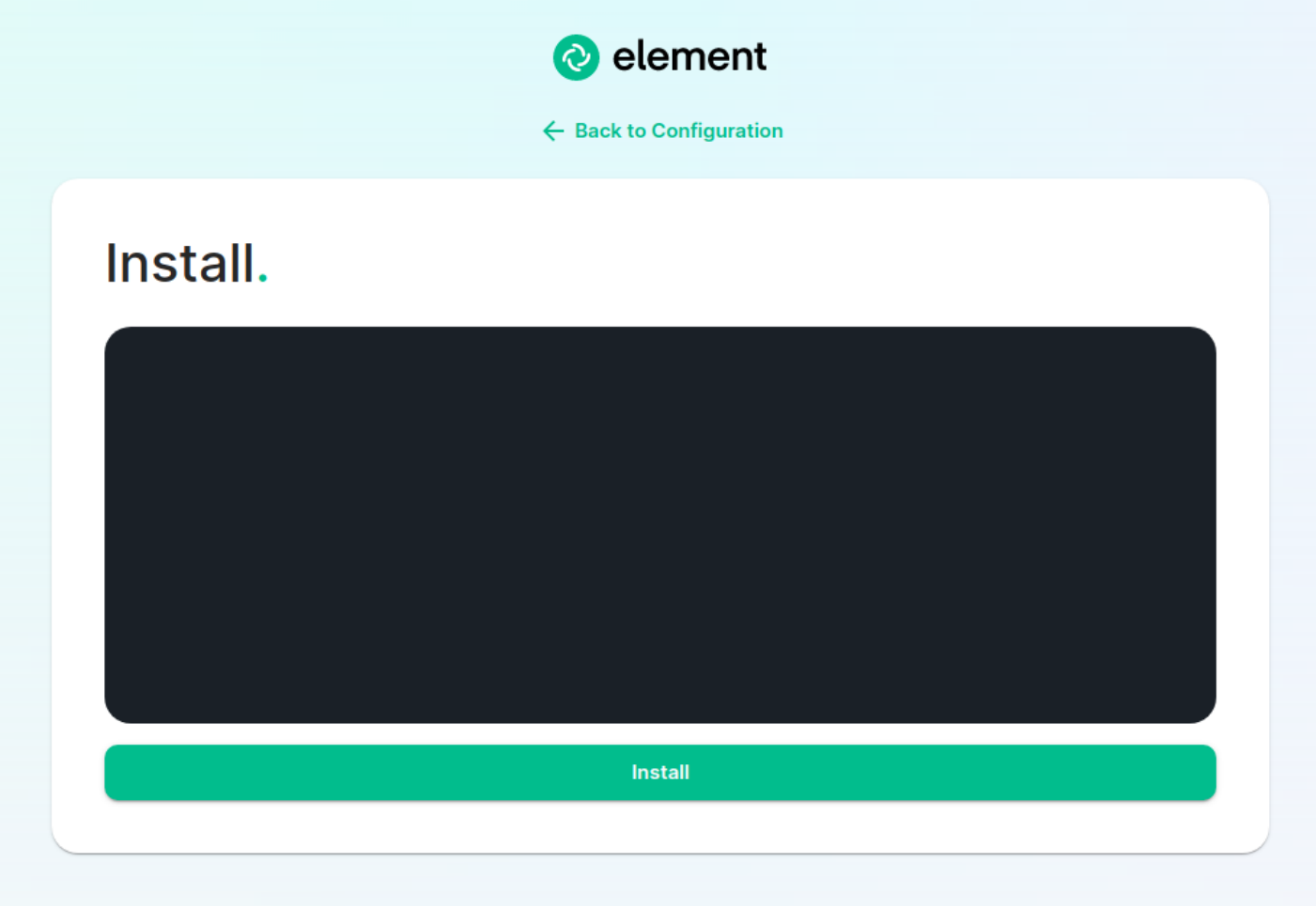
For specifics on configuring the IRC Bridge, please see Setting Up the IRC Bridge

For specifics on configuring the XMPP Bridge, please see Setting Up the XMPP Bridge

Once you have configured all of the integrations that you would like to configure, you can click "Continue" to head to the installation screen.

The Installation Screen

On the installation screen, you should see a blank console and a start button:



Click Start.

After a moment, you will notice the installer hang. If you go back to the prompt where you are running the installer, you will see that you are being asked for the sudo password:


```
[karl1@airgap ~]$ ./element-enterprise-graphical-installer-2023-02.02-gui-rc1.bin
Testing network...

Using self-signed certificate with SHA-256 fingerprint:
         CB:8E:4A:75:80:32:D5:E0:A0:2C:90:A7:DF:F9:2F:9F:6D:14:F7:18:53:D0:C5:6C:20:D5:95:A8:1A:57:67:21

To start configuration open:
        https://192.168.122.47:8443 or https://10.1.185.64:8443 or https://127.0.0.1:8443
API resolved without sending a response for /api/logs, this may result in stalled requests.
[sudo] password for karl1:
```

Go ahead and enter the sudo password and the installation will continue.

On the very first time that you run the installer, you will be prompted to log out and back in again to allow Linux group membership changes to be refreshed. This means that you will need to issue a ctrl-C in the terminal running your installer and actually log all the way out of your Linux session, log back in, restart the installer, navigate back to the installer screen, click start again, and then re-enter your sudo password. You will only have

to perform this step once per server.

Verifying Your Installation

Once the installation has finished, it can take as much as 15 minutes on a first run for everything to be configured and set up. If you use:

```
kubectl get pods -n element-onprem
```

You should see similar output to:

```
NAME                                        READY   STATUS    RESTARTS   AGE
app-element-web-c5bd87777-rqr6s             1/1     Running   1          29m
server-well-known-8c6bd8447-wddtm           1/1     Running   1          29m
postgres-0                                  1/1     Running   1          40m
instance-synapse-main-0                     1/1     Running   2          29m
instance-synapse-haproxy-5b4b55fc9c-hnlmp   1/1     Running   0          20m
```

Once the admin console is up and running:

```
first-element-deployment-synapse-admin-ui-564cbf5665-dn8nv   1/1     Running
1 (4h4m ago)   3d1h
```

and synapse:

```
first-element-deployment-synapse-redis-59548698df-gqkcq      1/1     Running
1 (4h4m ago)   3d2h
first-element-deployment-synapse-haproxy-7587dfd6f7-gp6wh    1/1     Running
2 (4h3m ago)   2d23h
first-element-deployment-synapse-appservice-0                1/1     Running
3 (4h3m ago)   3d
first-element-deployment-synapse-main-0                      1/1     Running
0              3h19m
```

then you should be able to log in at your admin panel (in our case https://admin.airgap.local/) with the `onprem-admin-donotdelete` user and the password that was specified on the "Synapse" screen.

Manually create your first user

If you wish to create users from your terminal, run the following command:

```
$ kubectl --namespace element-onprem exec --stdin --tty \
    first-element-deployment-synapse-main-0 \
    -- register_new_matrix_user --config /config/homeserver.yaml

New user localpart: your_username
Password:
Confirm password:
Make admin [no]: yes
Sending registration request...
Success!
```

Make sure to enter `yes` on `Make admin` if you wish to use this user on the installer or standalone Admin page.

Please note, you should be using the Admin page or the Synapse Admin API instead of `kubectl` / `register_new_matrix_user` to create subsequent users.

A word about Configuration Files

In the new installer, all configuration files are placed in the directory `.element-enterprise-server`. This can be found in your user's home directory. In this directory, you will find a subdirectory called `config` that contains the actual configurations.

Running the Installer without the GUI

It is possible to run the installer without using the GUI provided that you have a valid set of configuration files in the `.element-enterprise-server/config` directory. Directions on how to do this are available at: https://ems-docs.element.io/books/ems-knowledge-base/page/how-do-i-run-the-installer-without-using-the-gui. Using this

method, you could use the GUI as a configuration editor and then take the resulting configuration and modify it as needed for further installations.

This method also makes it possible to set things up once and then run future updates without having to use the GUI.

Cleaning up images cache

The installer, from version 24.02, comes with the tool `crictl` which lets you interact with microk8s containerd daemon.

After upgrading, once all pods are running, you might want to run the following command to clean-up old images :

```
~/.element-enterprise-server/installer/.install-env/bin/crictl -r
unix:///var/snap/microk8s/common/run/containerd.sock rmi --prune
```

Upgrading microk8s

Prior to versions 23.10.35 and 24.04.05

Using the installer to upgrade

Upgrading microk8s rely on uninstalling, rebooting the machine, and reinstalling ESS on the new version. It thus involves a downtime.

To upgrade microk8s, please run the installer with : `./<installer>.bin --upgrade-cluster`.

The machine will reboot during the process. Once it has rebooted, log in as the same user, and run : `./<installer>.bin unattended`. ESS will be reinstalled on the upgraded microk8s cluster.

Manually upgrading microk8s

Upgrading microk8s

The first step in upgrading microk8s to the latest version deployed by the installer is to remove the existing microk8s installation. Given that all of microk8s is managed by a snap, we can do this without worrying about our Element Enterprise On-Premise installation. The important data for your installation is all stored outside of the snap space and will not be impacted by removing microk8s. Start by running:

```
sudo snap list
```

and just determine that microk8s is installed:

```
[user@element2 element-enterprise-installer-2022-05.06]$ sudo snap list
Name      Version    Rev    Tracking     Publisher   Notes
core      16-2.55.5  13250  -            canonical✓  core
core18    20220428   2409   -            canonical✓  base
microk8s  v1.21.13    3410   1.21/stable  canonical✓  classic
```

Once you've made sure that microk8s is installed, remove it by running:

```
sudo snap remove microk8s
```

Now at this point, you should be able to verify that microk8s is no longer installed by running:

```
sudo snap list
```

and getting output similar to:

```
[user@element2 element-enterprise-installer-2022-05.06]$ sudo snap list
Name      Version    Rev    Tracking     Publisher   Notes
core      16-2.55.5  13250  -            canonical✓  core
core18    20220428   2409   -            canonical✓  base
```

Now that you no longer have microk8s installed, you are ready to run the latest installer. Once you run the latest installer, it will install the latest version of microk8s.

When the installer finishes, you should see an upgraded version of microk8s installed if you run `sudo snap list` similar to:

```
Name      Version   Rev    Tracking       Publisher   Notes
core18    20220706  2538   latest/stable  canonical✓  base
microk8s  v1.24.3   3597   1.24/stable    canonical✓  classic
snapd     2.56.2    16292  latest/stable  canonical✓  snapd
```

At this point, you will need to reboot the server to restore proper networking into the microk8s cluster. After a reboot, wait for your pods to start and your Element Enterprise On-Premise installation is now running a later version of microk8s.

After versions 23.10.35 and 24.04.05 and 24.05.01

Microk8s will be upgraded gracefully automatically when the new installer is used. The upgrade involves upgrading the addons, and might involve a downtime of a couple of minutes while it runs.

End-User Documentation

After completing the installation you can share our User Guide to help orient and onboard your users to Element!

Revision #92
Created 2022-07-28 18:53:51 UTC by Karl Abbott
Updated 2025-06-04 09:34:21 UTC by Kieran Mitchell Lane