

Ubersmith Installation and Upgrade Utility

- [Installation](#)
 - [Prepare the Host](#)
 - [Install Dependencies](#)
 - [CentOS](#)
 - [Debian / Ubuntu LTS](#)
 - [Install Docker](#)
 - [Clone the Repository or Download a Release](#)
 - [Install](#)
 - [Configure](#)
 - [Search](#)
 - [License](#)
- [Upgrade](#)
 - [Clone the Repository or Download a Release](#)



This utility only supports upgrading Ubersmith installations running version 4.0 or newer. It does not support directly upgrading older versions of Ubersmith.

Summary

The Ubersmith Installer is a command line utility that allows a system administrator to install (or upgrade) Ubersmith on a [supported platform](#) without assistance from Ubersmith Support. This can be useful when access to the Ubersmith environment is restricted, or for provisioning development/testing environments. The installer can be used to install both the Ubersmith Core (frontend) software as well as the Ubersmith Appliance.

Installation

Prepare the Host

Ensure that your host has been updated to the latest available packages as well as the latest Linux kernel. If necessary, reboot to ensure that the system is running the latest kernel.

Install Dependencies

The installation process will fail without these dependencies:

CentOS

CentOS 7

```
sudo yum install gcc libffi-devel python-devel openssl-devel epel-release git
```

Once EPEL is configured, run:

CentOS 7

```
sudo yum install python-pip
```

CentOS 8

```
sudo dnf remove buildah podman
sudo dnf install gcc libffi-devel python3-devel openssl-devel git
sudo alternatives --set python /usr/bin/python3
sudo alternatives --install /usr/bin/pip pip /usr/bin/pip3.6 1
```

Debian / Ubuntu LTS

Debian 10

```
sudo apt install build-essential libssl-dev libffi-dev python3-dev python3-setuptools python3-pip git
sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.7 1
sudo update-alternatives --install /usr/bin/pip pip /usr/bin/pip3 1
```

Ubuntu 18.04 LTS

```
sudo apt install build-essential libssl-dev libffi-dev python3-dev python3-setuptools python3-pip git
sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.6 1
sudo update-alternatives --install /usr/bin/pip pip /usr/bin/pip3 1
```

Ubuntu 20.04 LTS

```
sudo apt install build-essential libssl-dev libffi-dev python3-dev python3-setuptools python3-pip git
sudo update-alternatives --install /usr/bin/python python /usr/bin/python3.8 1
sudo update-alternatives --install /usr/bin/pip pip /usr/bin/pip3 1
```

Install Docker

The Ubersmith Installer requires that [Docker](#), as well as some additional dependencies be installed prior to installing Ubersmith. The script located at:

<https://get.docker.com/>

can be helpful when installing Docker. Ensure that Docker is configured to start on boot; in many cases it may not be. Depending on the operating system used, you may be able to enable the `docker` service with:

```
systemctl enable docker
```

Ensure that Docker is up and running:

```
service docker start
```



If `iptables` is not installed, Docker will fail to start.

Clone the Repository or Download a Release

On the host Ubersmith is to be installed on, clone the `ubersmith_installer` repository.

```
git clone https://github.com/TeamUbersmith/ubersmith_installer.git
```

Alternatively, you can download a tarball release of the installer from [here](#).

Install

As `root`, run `./install_ubersmith.sh` to install Ubersmith Core or `./install_appliance.sh` to install the Ubersmith Appliance. Ensure the path you choose can be written to by the user running the installation script, and that the user is a member of the `docker` group.



Ubersmith Core and Ubersmith Appliance should not be deployed to the same host.

Follow the prompts to complete the installation.

```
Checking for pip...
Installing Ansible...
Installing Ubersmith...
Choose an installation directory for Ubersmith [/usr/local/ubersmith]:
Enter the hostname where you will be hosting Ubersmith [ubersmith.example.com]:
Enter the email address of the Ubersmith administrator [admin@example.org]:
```

At this point the installer will begin, and will create the directory structure and supporting files to bring Ubersmith online.

Configure

Ubersmith is now listening for HTTPS connections; to load the Ubersmith installation wizard go to the address of IP of your Ubersmith host in your web browser, for example:

```
https://10.0.0.1
```

or, if you have a DNS record created, use that name directly:

```
https://ubersmith.example.com
```

Ensure you are connecting via HTTPS, and walk through the steps of the Ubersmith Setup Wizard.

Search

When configuring the Solr search engine, use the values:

- Host: solr
- Port: 8983
- Path: /solr/collection1

License

When prompted for license details, enter the licensing username and password provided to you by Ubersmith Support. If you do not have licensing details, please contact Ubersmith Support for assistance.



During the installation process, the Python package manager `pip` will install the needed binaries for Ansible and Docker to `~/.local/bin`. You should run:

```
export PATH="$HOME/.local/bin:$PATH"
```

to ensure these binaries are readily available to you from the command line. Failure to do so may cause the following commands to fail, indicating that the command cannot be found.

At the end of the Setup Wizard, you will be prompted to delete the `setup` directory. Change to the directory you deployed Ubersmith to (by default `/usr/local/ubersmith`) and run:

```
docker-compose exec web rm -rf /var/www/ubersmith_root/app/www/setup
```

With the `/setup` directory removed, the Ubersmith web user interface will now load, and the login page will appear. Log in with the username and password created during the installation process.

Upgrade

Before upgrading, be sure to back up your Ubersmith database.

Before upgrading, make sure you have the latest version of the installation / upgrade utility.

Clone the Repository or Download a Release

On the host Ubersmith is to be installed on, clone the `ubersmith_installer` repository.

```
git clone https://github.com/TeamUbersmith/ubersmith_installer.git
```

Alternatively, you can download a tarball release of the installer from [here](#).



If you have never used the Ubersmith installation / upgrade utility before, configuration is necessary. Without running the configuration step, the upgrade utility will fail.

As root, run `configure.sh` to configure the location of your existing Ubersmith installation, as well as your Ubersmith URL/address, and an email address associated with the Ubersmith administrator. Follow the steps to configure your environment:

```
./configure.sh
Configuring the Ubersmith installer for an existing installation...
Current path in use by Ubersmith / Appliance [/usr/local/ubersmith]:
Enter the address in use by Ubersmith / Appliance [ubersmith.example.com]: ubersmith.mydomain.com
Enter the email address of the Ubersmith administrator [admin@example.org]: jsmith@mydomain.com

[ output follows ]

localhost                : ok=3    changed=1    unreachable=0    failed=0
```

This step creates the file `.ubersmith_installer.ini` in the home directory of the user running the configuration utility.



If upgrading from a version of Ubersmith prior to 4.2.0, make sure to add volume entries in `docker-compose.override.yml` to include mount points for the appropriate time zone files in each service (web, mail, php, etc.). For example:

```
php:

  volumes:

    - "/etc/localtime:/etc/localtime"
```

Failing to make this update will cause those containers not updated to default to using UTC as a timezone.



When upgrading from versions prior to Ubersmith 4.3.0, a change is being made to the naming convention for the database host. Prior versions had the database host defined as `ubersmith_db_1`, newer versions define it as simply `db`. Please contact support at ubersmith.com to ensure your license record is updated if you encounter issues post-upgrade.

To upgrade Ubersmith, run the `./upgrade_ubersmith.sh` script. The upgrade process will complete automatically.

Troubleshooting

- When running CentOS, if the installer is unable to install `pip`, you may need to enable the [EPEL yum repository](#).
- Do not install the OS provided `pip`` packages for Ubuntu and Debian as the version of `pip`` provided is older and has difficulty installing the dependencies for the Ubersmith installer. `python-setuptools` will install the `easy_install` utility, which will allow the installer to install a more contemporary version of `pip`.
- Dependencies installed by `pip` will be installed using the `--user` option, which will install to the Python user install directory for your platform; typically `~/.local/`. (See the Python documentation for `site.USER_BASE` for full details.) This allows for the installer to be executed as a non-root user. You may want to add this directory to your `PATH` shell variable so that the supporting utilities (`docker-compose`, for example) can be run without having to specify the full path to the utility. To do this, run:

```
export PATH="$HOME/local/.bin/:$PATH"
```