# Passbolt

# Use cases

- Share passwords with staff/family/friends easily
- Export/import to keepass
- Access passwords anywhere (no need for sync)

# Pros

- Key based authentication
- Mobile interface
- Self-hosted
- Community edition is Free & Open source
- Web/Mobile/CLI interfaces

# Cons

- Mobile interface is limited, but improving

# Installing

# Download Free/Paid

https://signup.passbolt.com/pricing/pro

# Setup domain

- Point a sub-domain to your instance eg: passbolt.example.com

# Install Nginx

```
sudo apt install nginx
```

# Install passbolt pro

Ref:
https://help.passbolt.com/hosting/install/pro/debian

Installation on Debian 11.

```
sudo apt install passbolt-pro-server
```

# Install MariaDB

```
sudo apt install mariadb
```

# Passbolt Install: Setup Nginx

- Setup Nginx

| | |
|---|---|
| passbolt nginx SSL setup | none |
| passbolt domain name | passbolt.example.com |
| auto config nginx | Yes |

# Passbolt Install: Setup Passbolt DB

- MariaDB

| **username** | **root** |
| --- | --- |
| **pass** | |
| username | passboltadmin |
| pass | |
| db | passboltdb |

# Config Nginx for Passbolt

## /opt/nginx/conf/nginx.conf

```
server {

  listen 80;
  server_name passbolt.example.com;

  client_body_buffer_size     100K;
  client_header_buffer_size   1K;
  client_max_body_size        5M;

  client_body_timeout   10;
  client_header_timeout 10;
  keepalive_timeout     5 5;
  send_timeout          10;

  root /usr/share/php/passbolt/webroot;
  index index.php;
  error_log /var/log/nginx/passbolt-error.log info;
  access_log /var/log/nginx/passbolt-access.log;
```

# Config Nginx for Passbolt (cont)

```
  location / {
    try_files $uri $uri/ /index.php?$args;
  }

  location ~ \.php$ {
    try_files                 $uri =404;
    include                   fastcgi_params;
    fastcgi_pass              unix:/run/php/php7.4-fpm.sock;
    fastcgi_index             index.php;
    fastcgi_intercept_errors  on;
    fastcgi_split_path_info   ^(.+\.php)(.+)$;
    fastcgi_param             SCRIPT_FILENAME $document_root$fastcgi_sc
    fastcgi_param             SERVER_NAME $http_host;
    fastcgi_param PHP_VALUE   "upload_max_filesize=5M \n post_max_size=5
  }

}
```

# Install certbot

https://certbot.eff.org/instructions?ws=nginx&os=debiantesting

# Reconfigure passbolt for https

https://help.passbolt.com/configure/https/pro/debia

```
sudo dpkg-reconfigure passbolt-pro-server
```

Select auto for SSL setup

# Test

Access https://passbolt.example.com

# Config SSL manually

```
sudo certbot --nginx --nginx-ctl /opt/nginx/sbin/nginx --nginx-server-
```

# Setup Passbolt

Once installed you need to configure the rest through the web interface

# Begin

1. Go to https://passbolt.example.com
2. Click Start
3. Enter Subscription key

# Install Browser Addon

- Firefox
- Chrome

# Database setup

| | |
|---|---|
| URL | mysql://127.0.0.1:3306 |
| Username | passboltadmin |
| Password | |
| Database name | passboltdb |

# Server keys

| | |
|---|---|
| name | Example Passbolt Server |
| email | office@example.com |
| comment | |

# Emails

1. Fill in fields

| | |
|---|---|
| Sender name | Example Passbolt |
| Sender email | office@example.com |
| SMTP host | smtp.gmail.com |
| Use TLS | Yes |
| Port | 587 |
| Username | office@example.com |
| Password | |

2. Send test email

# Options

| | |
|---|---|
| Full base url | https://passbolt.example.com |
| Allow public registration | No |
| Force SSL | Yes |

# First User

1. Fill out details

| | |
|---|---|
| First name | Michael |
| Last name | Pope |
| Username | michael@example.com |

2. Fill in password and download and store your private key

# Done!

You can now login, start adding more computers and mobiles and import existing passwords.

# Mobile setup

# Setup Server

## Add to /etc/passbolt/passbolt.php

```
return [
// …
    // locate the passbolt section
    'passbolt' => [
        // insert the following plugin block after 'passbolt'
        'plugins' => [
            'mobile' => [
                'enabled' => true
            ],
            'jwtAuthentication' => [
                'enabled' => true
            ],
        ],          // leave the rest untouched, most likely gpg block, s
        // …
    ],
];
```

# Create JWT keys

```
sudo mkdir -m=770 /etc/passbolt/jwt
sudo chown www-data:www-data /etc/passbolt/jwt/
sudo su -s /bin/bash -c "/usr/share/php/passbolt/bin/cake passbolt crea
```

# Check with healthcheck

```
sudo su -s /bin/bash -c "/usr/share/php/passbolt/bin/cake passbolt hea
```

You should see this result at the end

```
JWT Authentication
[PASS] The JWT Authentication plugin is enabled
[PASS] The /etc/passbolt/jwt/ directory is not writable.
[PASS] A valid JWT key pair was found
```

# Restart nginx

```
systemctl restart passbolt
```

# Setup phone

1. Install app passbolt (Android 10+)
2. Open up passbolt in browser on pre-configured PC
3. Profile -> Mobile setup
4. Scan QR through the app
5. Done!

# Adding computers

1. Login to computer
2. Go to https://passbolt.example.com
3. Type in email, accept terms and hit next
4. Find email and click "start recovery"
5. Install the Add-on
6. Load in your private key which you got when creating the account
7. Enter passphrase
8. Pick color & characters for anti-phishing

# Next

1. Start using the CLI
2. Create a Emacs package to use CLI
3. Integrate with office custom software

# Demo

Go through videos on main page
https://www.passbolt.com/

# References

# Questions

| | |
|---|---|
| Email | map7@fastmail.com |
| Twitter | @map7 |
| Github | github: map7 |