
Obsidian.Docs Documentation

Release 0.1a1

ZA-PT

Jan 24, 2018

1	Getting Started	3
2	Installation Guide on Linux	5
2.1	Obtain Compiled Obsidian Files	5
2.2	Run Obsidian	5
2.3	Configure Obsidian	7
3	Installation Guide on Windows	9
4	Claims in Obsidian OAuth Server	11
4.1	Claims in OAuth Server	11
4.2	Claims list	11
5	Claims in Obsidian Management API	13
5.1	Introduce	13
5.2	Claims list	14
6	Contribution Guide	15
7	Toolkit Installation	17
8	Name Convention	19
8.1	For C# code	19
8.2	For Typescript code	20

Obsidian is an OAuth-based Authentication System. It also supports Resource Owner Password Credentials so you can use it as a normal CAS System.

CHAPTER 1

Getting Started

Installation Guide on Linux

This document is a guide for installation Obsidian on a Linux Distribution. For installation on Windows, see [Installation Guide on Windows](#)

2.1 Obtain Compiled Obsidian Files

2.1.1 Compile by configuration bash script

You need to compile Obsidian on your own because we have not provided compiled files. A bash script has been provided for configuring compiling environment.

You can configure compiling environment by the following command if in Ubuntu:

```
curl -o- https://raw.githubusercontent.com/ZA-PT/Obsidian/canary/configure_env/ubuntu/  
↪configure_env.sh | bash
```

After configuration, you can build and publish Obsidian:

```
./build.sh | dotnet publish
```

All runnable files can be found in `/home/Obsidian/src/Debug/netcoreapp2.0/publish`

Remember to obtain root permissions before you run.

2.2 Run Obsidian

Change your working directory of your shell to published Obsidian files, `/home/Obsidian/src/Debug/netcoreapp2.0/publish` in default. Run Obsidian by the following command:

```
dotnet Obsidian.dll
```

Obsidian will host itself and listen on port 5000 in default. You can visit <http://localhost:5000/> to access Obsidian.

2.2.1 Configure Reverse Proxy Server

You need to configure a reverse proxy server if you want to host more websites on your server and access through port 80. A reverse proxy server is able to transfer HTTP requests between Web Server and Obsidian.

In this sector, we use Nginx as web server and configure a reverse proxy server on it.

Install Nginx, add a new vhost with the following configuration:

```
server {
    listen 80;
    server_name yourdomain;
    location / {
        proxy_pass http://localhost:5000;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection keep-alive;
        proxy_set_header Host $http_host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

save and restart Nginx, now you can visit Obsidian through yourdomain.

2.2.2 Install Obsidian as Service

In order to make Obsidian start automatically, Obsidian must be installed as service or run by other monitoring software. In this sector, we use Supervisor as monitoring software, which can monitor Obsidian and restart it automatically.

Install Supervisor, add a new program with the following configuration:

```
[program:obsidian]
directory = /home/Obsidian/src/Debug/netcoreapp2.0/publish
command = dotnet Obsidian.dll
autostart = true
startsecs = 5
autorestart = true
startretries = 3
user = www
redirect_stderr = true
stdout_logfile_maxbytes = 20MB
stdout_logfile_backups = 20
stdout_logfile = /data/logs/Obsidian_stdout.log
```

Please make sure that every part of path exists, or it can not run correctly. Save this configuration files in Supervisor Configuration Directory.

Finally, refresh Supervisor:

```
supervisorctl update
```

Obsidian will automatically run now.

2.2.3 Install MongoDB

Obsidian stores its datas in MongoDB so you need to install Mongod. If your MongoDB listen at default port, Obsidian will work correctly.

If you want to set a custom MongoDB connection string, edit appsettings.json in Obsidian working directory, /home/Obsidian/src/Debug/netcoreapp2.0/publish/appsettings.json default:

```
{
  "ApplicationInsights": {
    "InstrumentationKey": ""
  },
  "Logging": {
    "IncludeScopes": false,
    "LogLevel": {
      "Default": "Debug",
      "System": "Information",
      "Microsoft": "Information"
    }
  },
  "OAuth20": {
    "TokenAudience": "ObsidianAud",
    "TokenIssuer": "Obsidian"
  },
  "ConnectionStrings": {
    "MongoDB": "mongodb://127.0.0.1:27017"
  }
}
```

replace “MongoDB” sector will correct connection string.

2.3 Configure Obsidian

Now visit <http://yourdomain/firstrun> to setup Obsidian. You need to provide administrator’s username, password and Obsidian URL.

Please make sure that Obsidian has permissions creating files in its working directory because The FirstRun program will create init.txt after setting up.

Finally, Obsidian will redirect you to Management Portal, where you can create scope,user or client.

Eventually, you have your Obsidian installed.

CHAPTER 3

Installation Guide on Windows

Claims in Obsidian OAuth Server

4.1 Claims in OAuth Server

Though Obsidian is mainly designed as an OAuth Authentication Server, the whole Obsidian program is commonly divided into three parts. On one hand, Obsidian's user model contains user's personal information which will be provided to third-part client on OAuth process, on the other hand, because it contains user's information, it can be regarded as a Resource Server.

Claims in user domain model mainly represent user's personal information instead of what a user can do.

4.2 Claims list

Model	ClaimType	Content
User	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier	User Id of user model
User	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name	Username of user model
UserProfile	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname	User's givenname
UserProfile	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname	User's surname
UserProfile	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/gender	User's gender
UserProfile	http://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress	User's email emailaddress

Claims in Obsidian Management API

5.1 Introduce

Obsidian uses Token-based Authentication and Claims-based Authorization.

If user wants to access a manament API action, a JsonWebToken must be added into the request.JsonWebToken generated by Obsidian OAuth Server contains claims that user owns. Some claims represent user's personal information, others represent permissions which user is endowed with.

Strictly speaking, A claim can not represent what a user can/can not do. However, if we want a user to be able to edit something, we can give it a claim that tells other it is an editor instead of a claim that contains what resource it can operate

In this section, we mainly list all available claims that can be given to user.

For claims represented user's personal information, see *Claims in Obsidian OAuth Server*

5.2 Claims list

Type	ClaimType	Enables
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/IsCreator	Represents user can add a new user
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/IsAcquirer	Represents user can query any user's information
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/UserName/IsEditor	Represents user can modify other user's user name
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/Password/IsEditor	Represents user can modify other user's password
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/Profile/IsEditor	Represents user can modify other user's profile
User	http://schema.za-pt.org/Obsidian/ManagementAPI/User/Claims/IsEditor	Represents user can modify other user's claim
Client	http://schema.za-pt.org/Obsidian/ManagementAPI/Client/IsCreator	Represents user can add a new client
Client	http://schema.za-pt.org/Obsidian/ManagementAPI/Client/IsAcquirer	Represents user can query information of a client
Client	http://schema.za-pt.org/Obsidian/ManagementAPI/Client/IsEditor	Represents user can modify a client
Client	http://schema.za-pt.org/Obsidian/ManagementAPI/Client/Secret/IsAcquirer	Represents user can query secret of a client
Client	http://schema.za-pt.org/Obsidian/ManagementAPI/Client/Secret/IsEditor	Represents user can modify secret of a client
Scope	http://schema.za-pt.org/Obsidian/ManagementAPI/Scope/IsCreator	Represents user can add a new permission scope
Scope	http://schema.za-pt.org/Obsidian/ManagementAPI/Scope/IsAcquirer	Represents user can query information of a permission scope
Scope	http://schema.za-pt.org/Obsidian/ManagementAPI/Scope/IsEditor	Represents user can modify a scope

CHAPTER 6

Contribution Guide

Obsidian is an open source software. You can re-develop Obsidian within the limits permitted by Apache 2.0 License. This chapter is to give someone who wants to participate in development of Obsidian.

CHAPTER 7

Toolkit Installation

1. Make sure these following tools are installed:

- .NET Core 1.1.1 with SDK 1.0.1
- Yarn Package Manager

2. Run the following commands:

```
cd ./src/Obsidian/  
dotnet restore  
dotnet --verbose build  
yarn
```


8.1 For C# code

References

- [Capitalization Conventions](#)
- [C# Coding Conventions \(C# Programming Guide\)](#)

```
using System;
using System.Threading.Tasks;

//For namespaces, use Pascal casing.
namespace NamingConversions
{
    // For classes, enums, use Pascal casing.
    public class SampleClass
    {
        //For constants, use Pascal casing.
        public const int Count = 0;

        //For static members, use Pascal casing.
        //For public fields, use Pascal casing.
        public static string Name = "Obsidian";

        //For private fields, use camel case with an underscore.
        private string _name;

        //For properties, use Pascal casing.
        public string StatusData { get; set; }

        //For methods, use Pascal casing.
        //For parameters, use camel casing.
        public void MyMethod(int myParameter)
        {
```

```
    }

    //For asynchronous methods, use Pascal casing with suffix "Async".
    public async Task ProcessAsync()
    {
    }

    //For events, use Pascal casing.
    public event EventHandler MyEvent;
}

//For interfaces, use Pascal casing with prefix "I".
public interface ISample
{
}
}
```

8.2 For Typescript code