

Selene Network Install and Setup

This post is going to outline what it takes to download, install and setup the Selene Network on your server.

It is expected that the domain that serves this code will be a standard WordPress type install secured by SSL (https not http).

Having a little understanding regarding how to install files on your web hosting service will be useful.

Expectations

Users of the network are going to expect that when they connect to your server, they will simply be able to append a '/dsn' to the URL and activate the Selene Network code. For example:

<https://YourDomain.Ext/dsn>

Likewise, any Selene Network compatible NFT project hosted on your server will need to be in an 'nfts' subdirectory and have a reasonable short name for indexing. The Agent NFT project that is part of the core demonstrated this structure. It will be referenced like this:

<https://YourDomain.Ext/nfts/agentv1>

The file that manages the custom settings for the individual installs will be stored in a 'src' directory off the domain like:

<https://YourDomain.Ext/src>

This 'src' directory will also hold the gallery JSON files and galleryv2 smart contract. Please note that all file system names should be lowercase (the code is case sensitive).

Download the Selene Network

The latest copy of the Selene Network code will be downloadable from this website:

<https://amorstyle.com/download>

Just click the zip file that you want to download.

Every official release of the code can be hash verified. To do that on windows, after the download, run the windows certification tool over the file like:

```
"certutil -hashfile the_downloaded_file.zip -sha1"
```

The resulting hash should match what is displayed on the website. This hash check shows that the ZIP file has not been change since the publish date. Note that if you receive a Selene Network ZIP file from someone, you can always verify it's hash via this download location.

Installing

Connect to your web hosting service and use their webspace explorer and find the location where you installed WordPress. At the same level as wp-admin, create four new directories 'dsn', 'src', 'private' and 'nfts'. Then, in the 'nfts' directory, create a 'agentv1' directory. In the 'src' folder create a 'galleryv2' directory.

Inside the 'dsn' directory, upload the dsn_*.zip file you downloaded above.

Once uploaded, extract the contents of dsn_*.zip into the 'dsn' directory.

Once that is completed, extract the contents of agentv1_*.zip (found in the docs directory) into the 'nfts/agentv1' directory.

Then, move the dsnreadme.php file from the 'dsn' folder to the 'src' folder. You will be editing this file to provide personalized settings for your install.

At this point, all the basic functionality is installed.

Setup and configuration

When the code loads, it reads from a configuration file called 'dsnreadme.php'. This is the file that you will want to edit in order to customize key elements that are displayed to the website visitor. This includes the agent and website ids, the name, the default gallery and a few other settings.

It is recommended that you edit the 'dsnreadme.php' file locally using something like notepad.exe.

If you are a registered Partner Agent in the network, you'll want to place your Id in the 'agent' field (replacing 1001).

If you are the proud owner of a Website NFT, you will place your token Id number in the 'website' field (replacing 1).

If you launch your own galleryv1 smart contract, you can update the 'GalleryAddr' field to reference your gallery smart contract.

If you registered multiple galleries to the same contract, you will also want to instruct the code which 'GalleryIndex' should be used.

Choose another name for the string that is displayed in the explorer tabs. That is the 'tabTitle' field which should not be 'AmorStyle Selene Network.'

Because the gallery code displays the newest entries on the top of the page and gallery contracts can hold unlimited numbers of contracts, the 'galleryLimit' field defines how many of the latest N number of gallery entries to show.

The 'limit' field is for displaying NFTs in the visitor's wallet.

Note that when you update your code to a newer version of the Selene Network core code, you should just be able to delete all the files in the dsn folder and then extract the latest code. The settings held in the dsnreadme.php should carry over to the next install.

This also holds for the NFT/DAT projects. Because they are in separate directories, the core code can be updated after validating that the existing projects work on the new codebase.

Upload your configuration settings

Once you've edited your configuration settings in the 'dsnreadme.php' file, upload that to the 'src' directory.

Done!

At this point, the code should be fully functional. Test it and see. Visit:

<https://YourDomain.Ext/dsn>

Reinstalling

If you have installed the code and you want to check to see if you're running the latest released version, look at the footer of the Selene Network pages. You'll see something like "Distributed Sales Network v1.02". If the version number that you're running matches the version on <https://AmorStyle.com/dsn>, then you're running the latest version.

Using your web hosting file explorer, remove all the files in the 'dsn' directory and then extract the new version of the code there. It's rare to see any changes in the agentv1 contract, thus that location which was installed originally will most likely not need to be updated.

At this point, you should be done.

Prelaunch testing

Before performing an upgrade of the core code, you should validate that your current collection of Selene Network compatible NFT smart contracts work with the new code.

There are multiple ways to perform this testing like installing the new code to a second server and validating it works before rolling out to your main site.

You could also test that your contracts are read correctly by visiting a server that is running the code you want to install and check that it reads your contracts correctly.

Also, before upgrading take note of the code version that you're running. If the upgrade exposes bugs, you can always revert back to the known working version.

One of the best ways to validate the new code supports your projects correctly is to have the new code unroll your gallery. That can be done like:

<https://amorstyle.com/dsn/?gallery=0xYourGalleryAddress&index=N>

Where you provide your gallery contract address and the index values that you use.

Bugs

All the code is provided so that if there is an issue, you can fix it yourself. The code is reasonably commented and contains debug information.

If there is something you want me to fix in the next version, you'll have to let me know and sell me on the idea that it's an important thing to fix. Bugs are part of life, but I want the product to work for you.

Feature requests

As with any product, there is probably room for improvement. If you have a request for new functionality or a recommendation that you'd like to see incorporated into the core code, please let me know.

Likewise, if you modify the code to do something cool and you'd like to share that with the community, please let me know or just share it with others.

Disclaimer

The code is provided 'as is' so use at your own risk. Because this code has dependencies, there are things that could break that are outside my control to fix, thus this code could break at any time for any reason. The good news is you have all the code, thus the ability to unblock yourself.

I have a personal interest in keeping the code running. I'll do what I reasonably can to keep it functional and share my code. But there are no guarantees.