How to access to your work inside a WebRunner.

(To develop with your favorite tools)

Every developer has their own favorite softwares configured in such a way to improve their development productivity.

Even if Ace Editor, including in the WebRunner web interface, has some code-facilities for developers, maybe you would prefer to work with your own tools.

Here are 3 solutions for you.

1/ Git.

This solution is perfect if you just want an access to your projects. The WebRunner allows you to create git repositories on-the-fly, and manage them thanks to the integrated git-command. You can create as much repositories as you wish.

Please, to prepare the environment, refer to this page of documentation :How to access your git repositories remotely.

Then, take a note of the "git_public_url" and "git_private_url" in the connection parameters of your instance (number 6 and 7) :

□

The "git_public_url" allows anybody to clone publicly any of your repositories, just by appending the repository's name to this url. If you want to push your local commits to the repository, please use the "git_private_url" (with your repository's name appended to it) as the origin address. The login and password are the ones you usually use to connect to your WebRunner.

Pros :	Cons :
• Really easy to setup : you have nothing to do except installing git on your personal computer. • Exchanging data is very fast. • You can use a normal development workflow using a Source Code Management System. • You can use for favorite development softwares.	• You will still be dependant of the WebRunner web UI. • You will need to use a console to prepare the server-side of the git environment.

2/ SSH.

Having a SSH access to your webrunner would allow you to use the integrated vim. It will also allow you to access to the whole slappartition.

In order to connect to your shell via the ssh command line, and using tools such as scp, you must provide your ssh public key to your WebRunner, using the parameter "user-authorized-key" in the parameters XML transmitted to your instance.

Pros :	Cons :
• You will have access to the whole slappart (and so the config files). • You can send documents using scp. • The WebRunner terminal can run wildely-used softwares, such as "vim", "nano", and "screen". • You can configure the terminal using a bashrc file. • Easier to use and faster than shellinabox.	• You need an IPv6 access. • You cannot use your favorite IDE (if it isn't vim).

3/ WebDav.

You can mount the working directory of your WebRunner as a DavFS file storage.

To do so, first take a note of the "webdav_url" parameter value in the connection parameters of your instance (it is number 5).

Then, open a console on your own computer. You must be logged in as root. First, download the "davfs2" package. Then, you can for example go to the "/tmp" directory and create a folder named "dav". Here are the commands to run on a debian environment :

```
$ su
$ aptitude install davfs2
$ cd /tmp
$ mkdir dav
```

Now, you need to mount the DavFS volume. The prompt will look like that :

□

Don't worry about the warning : the webdav server uses a self-signed certificate to allow https secure connections. The username and password are the ones you usually use to access to your webrunner.

You can now "cd dav" : you are now in the working directory of your webrunner, and you can access to your projects, compiled SRs, and running instances.

Pros :	Cons :
• You can access to your documents with any file manager (konqueror, nautilus, ...), or with command line. • You can use for favorite development softwares, and run commands such as "grep", "sed", ... • You also have access to your SR or instances' partitions.	• You need to be root, or setup a fstab rule. • Fetching or saving files can take some time.