**How to Set up Daul-Authentication to GitHub and AWS CodeCommit**

Both [GitHub](https://github.com/) and AWS [CodeCommit](https://aws.amazon.com/codecommit/) are used to version-control your source code. I use GitHub to share my code with the wider community and CodeCommit for production code for AWS. You may use other source control systems like [BitBucket](https://bitbucket.org/). Whatever the choice you make, it is convenient to set up your local machine to commit code to multiple Git-like code repositories.

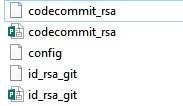
In this post, I will show you how to set up dual-authentication to GitHub and AWS CodeCommit repositories. You can pretty much use the same way to set up Git connections to other tools. At work, I do the dual-authentication to CodeCommit and BitBucket in the same way.

**Prerequisite**

We assume you have AWS and BitBucket account and of course Git installed.

**Steps**

1. Create SSH keys for both GitHub and CodeCommit. Use the online documentation [here](https://help.github.com/articles/connecting-to-github-with-ssh/) for GitHub and [here](https://docs.aws.amazon.com/codecommit/latest/userguide/setting-up-ssh-unixes.html) for CodeCommit.
2. Each SSH key pair will be created in the .ssh directory (usually located in your home directory). Name the keys so that you can differentiate them. Then, create config file as below.



1. In the config file, you can define hosts, user ids and the paths to the private keys as below. For CodeCommit, user can be retrieved from the management console as described [here](https://help.github.com/articles/connecting-to-github-with-ssh/). For GitHub, user is the user name.

[cc lang="text" tab\_size="4" lines="-1" line\_numbers="false"]

[/cc]

1. Create a repository named test\_github in GitHub and test\_codecommit in CodeCommit. Using UI is the easiest for GitHub. For CodeCommit, you can run AWS CLI command as below.

[cc lang="text" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

[/cc]

1. You can check the authentication in each system.

[cc lang="text" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

[/cc]

1. Let’s test by running the script below.

**BitBucket**

[cc lang="text" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

[/cc]

**CodeCommit**

[cc lang="text" tab\_size="4" lines="-1" theme="mac-classic" line\_numbers="false"]

[/cc]

You can see the file is committed to the remote branches for both. Yay!