

Deploying the Blockscout Infrastructure

Deployment with Terraform 12 is unstable due to these bugs[#144](#) [#147](#) [#148](#) [#149](#) . Please use TF 11.11 - 11.14 and following branch for deployment<https://github.com/poanetwork/blockscout-terraform/tree/before-t12> Go to[Deploying BlockScout](#) if you already have an infrastructure and simply want to install or update your BlockScout instance. 1) Check that all[infrastructure prerequisites](#) are installed with the correct version number.

2) Create the AWS access key and secret access key for user with sufficient permissions;

3) Create `hosts` file from `hosts.example` (`mv hosts.example hosts`) and adjust to your needs. Each host should represent each BlockScout instance you want to deploy.

Each host name should belong exactly to one group. Also, per Ansible requirements, hosts and groups names should be unique. The simplest `hosts` file with one BlockScout instance will look like:

...

Copy [group] host

...

Where [group] is a group name, which will be interpreted as a prefix for all created resources and `host` is a name of BlockScout instance.

4) For each host merge `infrastructure.yml.example` and `all.yml.example` config template files in `host_vars` folder into single config file with the same name as in `hosts` file:

...

Copy `cat host_vars/infrastructure.yml.example host_vars/all.yml.example > host_vars/host.yml`

...

5) For each group merge `infrastructure.yml.example` and `all.yml.example` config template files in `group_vars` folder into single config file with the same name as group name in `hosts` file:

...

Copy `cat group_vars/infrastructure.yml.example group_vars/all.yml.example > group_vars/group.yml`

...

6) [Adjust the variables](#) at `group_vars` and `host_vars` .

You can move variables between host and group vars depending on if variable should be applied to the host or to the entire group. Also, if you need to distribute variables across all the hosts/groups, you can add these variables to the `group_vars/all.yml` file.

More on variable precedence => [Official Ansible Docs](#) . 7) Run `ansible-playbook deploy_infra.yml`

- During the deployment the "diffs didn't match" error may occur, it will be ignored automatically. If Ansible play recap shows 0 failed plays, then the deployment was successful despite the error.
- Optionally, you may want to check the variables that were uploaded to the [Parameter Store](#)
- at AWS Console.
-

Last updated 2 months ago