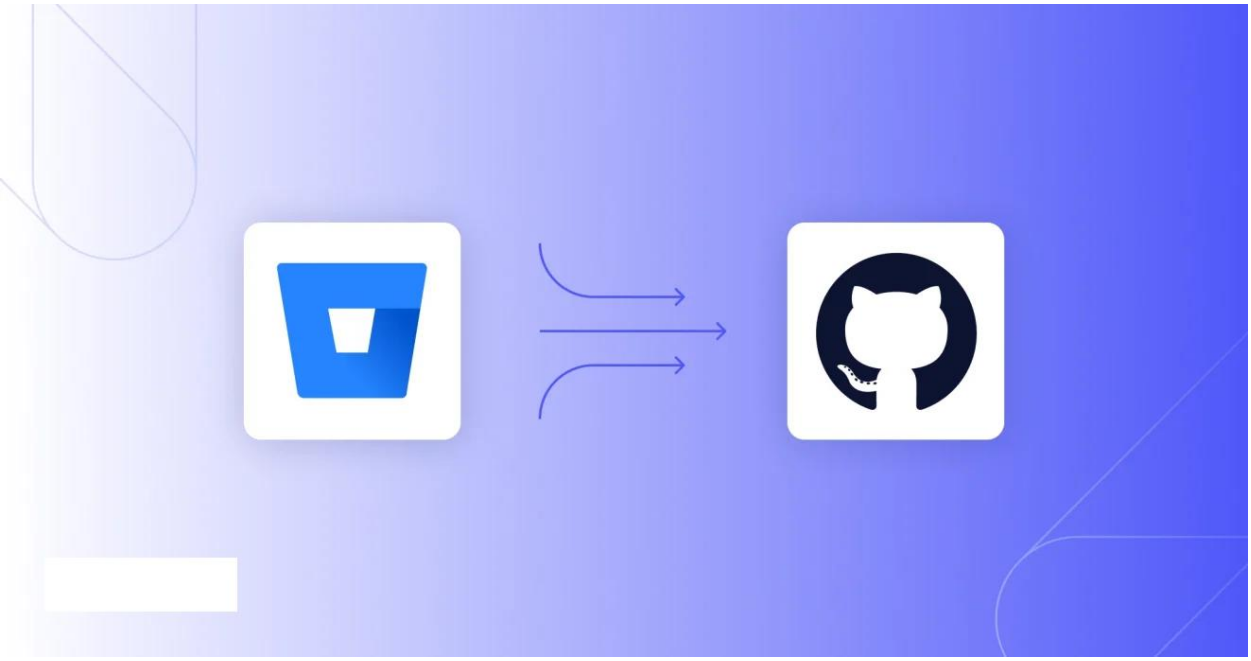
Migrating repositories from Bitbucket Server to GitHub Enterprise



Prerequisites: -

1) We strongly recommend that you perform a trial run of your migration and complete your production migration soon after. For more information [click here].
2) Ensure you understand the data that will be migrated and the known support limitations of the Importer. For more information [click here].
3) While not required, we recommend halting work during your production migration. The Importer doesn't support delta migrations, so any changes that happen during the migration will not migrate. If you choose not to halt work during your production migration, you'll need to manually migrate these changes.
4) **Bitbucket Server**:- You need the username and password for a Bitbucket Server account with admin or super admin permissions.
5) **Github Enterprise edition**:- For the destination organization on GitHub.com, you must be an organization owner or have the migrator role. For more information [click here]
6) AWS S3 storage bucket or Azure Blob Storage storage account.

STEPS:-

1) **Install the GitHub CLI - Click here to install. [gh –version (Minimum is 2.4.0)]**

```
sudo apt install gh
```

```
:~$ sudo apt  install gh
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
```

2) **Creating a personal access token for GitHub Enterprise Importer**
   a. If you are owner of Github organization or has admin access
      i. On Github home page click on profile icon and then settings
      ii. Click on Developer Settings
      iii. Click Personal access tokens > Tokens (classic)
      iv. Generate personal access token that can access the destination organization on GitHub [click here for more information].
      v. If SAML single sign-on is enforced for the organization(s) you need to access, authorize the personal access token for SSO.
      vi. Copy that Token and keep it safe place for further process.

OR

   b. Request owner of Github organization to provide Migrator role to you : Below steps should be executed by admin.
      i. Owner or administrator should login to Github via CLI
         1. gh auth login
         2. select Github.com
         3. select HTTPS
         4. Authenticate Git with Github credentials : yes
         5. Select Personal Access Token and then paste the PAT token.

```
:~$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? Yes
? How would you like to authenticate GitHub CLI? Paste an authentication token
Tip: you can generate a Personal Access Token here https://github.com/settings/tokens
The minimum required scopes are 'repo', 'read:org', 'workflow'.
? Paste your authentication token: ************************************
- gh config set -h github.com git_protocol https
√ Configured git protocol
! Authentication credentials saved in plain text
√ Logged in as
$
```

      ii. Follow 3rd step and install BBS2GH extension

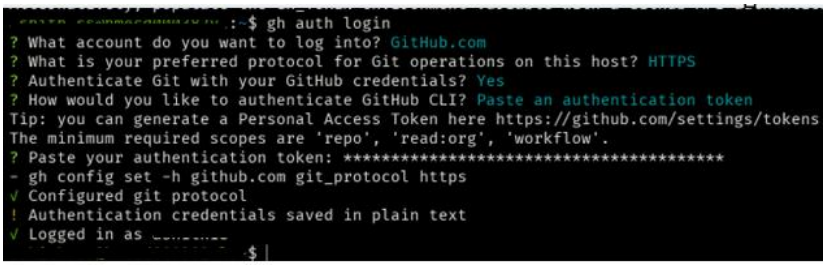
1. Grant migrator role to user :-

```
gh bbs2gh grant-migrator-role --github-org  <ORG_NAME>  --actor <username> --actor-type user
```

3) Install the BBS2GH extension.

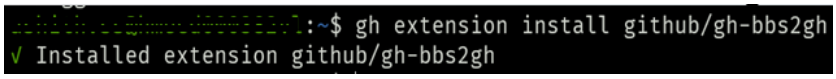
a. Before installing you need to login to Github via CLI

```
gh auth login
```



b.

```
gh extension install github/gh-bbs2gh
```



4) Your Bitbucket Server instance via SFTP, if your Bitbucket Server instance runs on Linux. In general, if you can access the server via SSH, then you can also use SFTP. In our case Bitbucket server is running in Linux so SSH is required

SSH AUTHENICATION

a. Generate ssh key

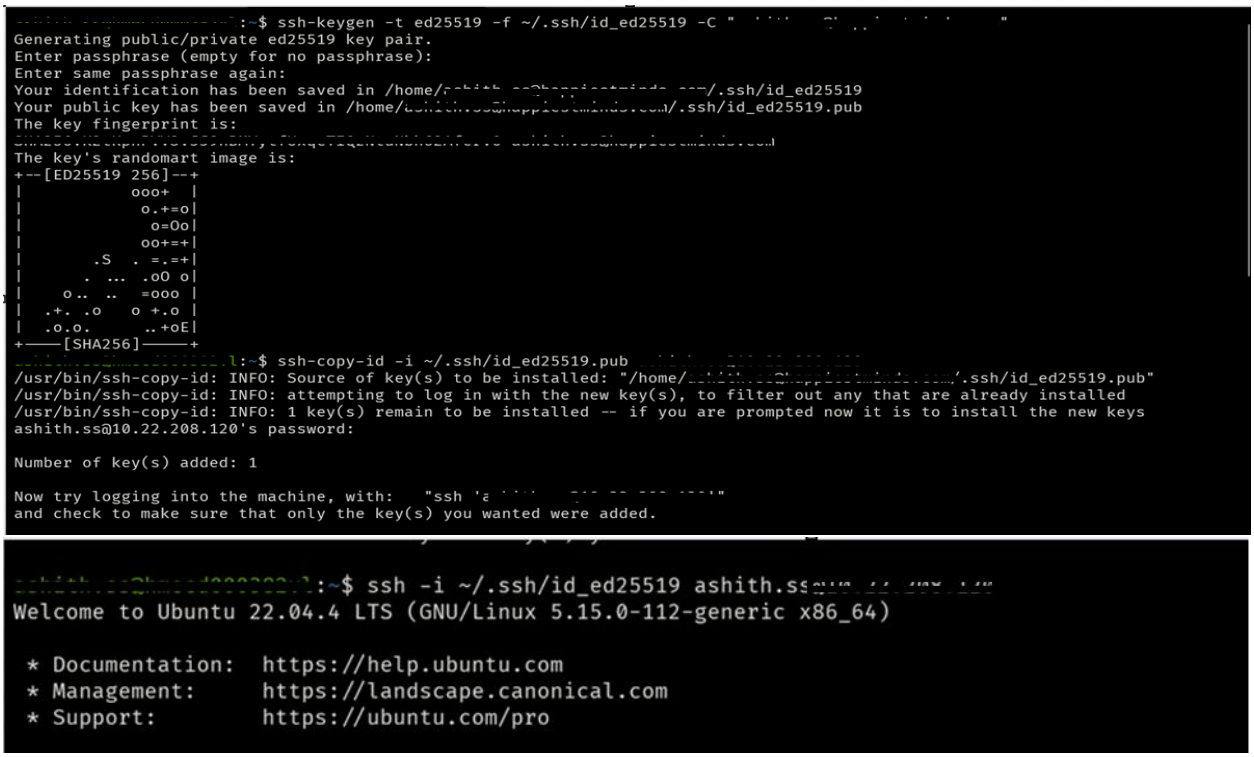
```
ssh-keygen -t ed25519 -f ~/.ssh/id_ed25519 -C "email ID"
```

b. Copy public key to bitbucket server and authenticate using using password

```
ssh-copy-id -i ~/.ssh/id_ed25519.pub <ssh user name>@<ip of bitbucker server>
```

c. Verify ssh is working for bitbucket server

```
ssh -i .ssh/id_ed25519 user name>@<ip of bitbucker server>
```



5) **Set environment variables**

---

export GH_PAT=”Token”
export BBS_USERNAME=”bitbucket username”
export BBS_PASSWORD=”bitbucket password”
export AWS_REGION=”S3 bucket Region”
export AWS_ACCESS_KEY_ID=”Access key which permission to copy file to particular bucket ”
export AWS_SECRET_ACCESS_KEY=”secret key”

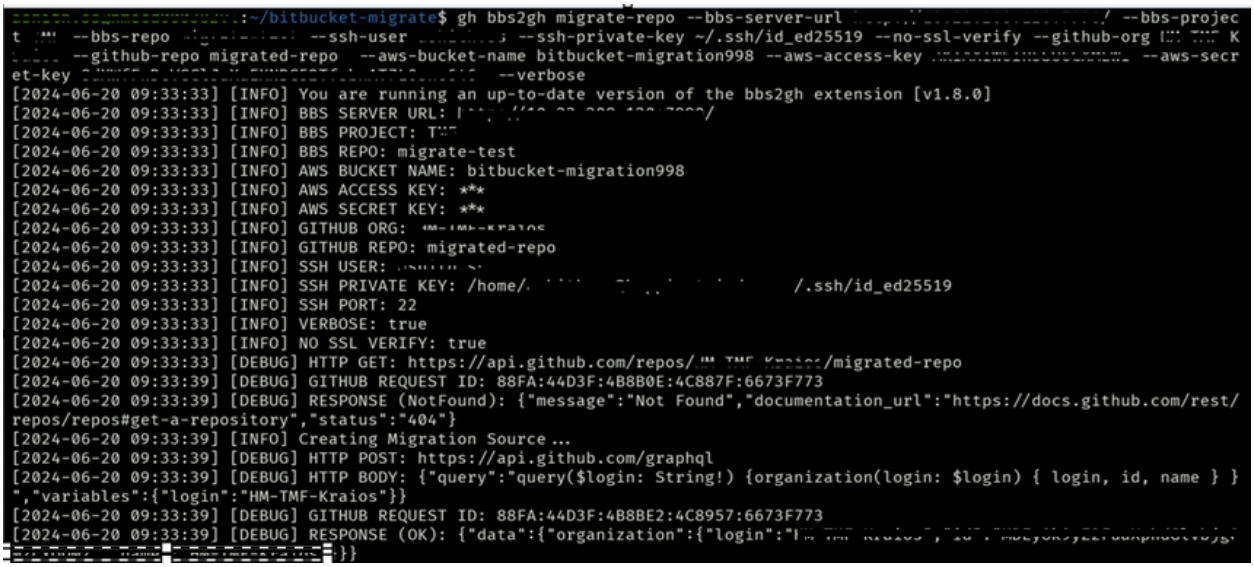
---

6) **In below command replace the placeholders in the command above with the following values**.

gh bbs2gh migrate-repo --bbs-server-url BBS-SERVER-URL \
--bbs-project PROJECT --bbs-repo CURRENT-NAME \
--github-org DESTINATION --github-repo NEW-NAME \
--ssh-user SSH-USER --ssh-private-key PATH-TO-KEY \
--no-ssl-verify \
--aws-bucket-name AWS-BUCKET-NAME \
--aws-access-key AWS_ACCESS_KEY_ID --aws-secret-key AWS_SECRET_ACCESS_KEY --verbose

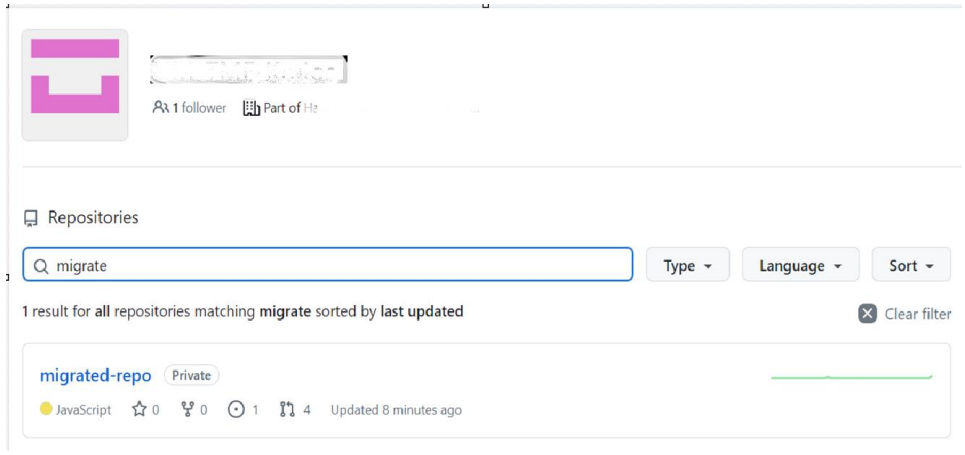
```
gh bbs2gh migrate-repo --bbs-server-url BBS-SERVER-URL  --bbs-project PROJECT  --bbs-repo CURRENT-NAME  --ssh-user SSH-USER  --ssh-private-key ~/.ssh/id_ed25519 --no-ssl-verify --github-org DESTINATION  --github-repo NEW-NAME  --aws-bucket-name AWS-BUCKET-NAME   --aws-access-key AWS_ACCESS_KEY_ID --aws-secret-key AWS_SECRET_ACCESS_KEY --verbose
```

[Note:- Use --no-ssl-verify when bitbucket server is running in HTTP and –verbose is to debug]



Placeholder	Value
BBS-SERVER-URL	The URL for your Bitbucket Server instance
PROJECT	The key for the Bitbucket Server project of the repository you want to migrate
CURRENT-NAME	The name of the repository you want to migrate
DESTINATION	Name of the destination organization
NEW-NAME	The name you want the migrated repository to have
SSH-USER	If your Bitbucket Server instance runs on Linux, the username to use when connecting t your Bitbucket Server via SFTP
PATH-TO-KEY	If your Bitbucket Server instance runs on Linux, the path to your SSH private key, such as ~/.ssh/id_rsa. For SSH key requirements, see "Managing access for a migration Bitbucket Server".
SMB-USER	If your Bitbucket Server instance runs on Windows, the username to use when connect your Bitbucket Server via SMB
AWS-BUCKET-NAME	The bucket name for your AWS S3 bucket

7) Verify Migrated Repository in Github



8) Tar file will be uploaded to S3 bucket