1 Introduction

Arseeding is a lightweight arweave data seed node. It is mainly used to synchronize, cache and broadcast arweave transactions and data.furthermore,arseeding support bundle transaction and upload bundle data to arweave.

Important:
arseeding is compatible with all http api interfaces of arweave node and also provides bundle transaction api.
Related articles:
arseeding server design
https://medium.com/everfinance/arseeding-server-design-4e684176555a

2 Getting Started

2.1 Get Arseeding

To start using arseeding, download arseeding from:
https://github.com/everFinance/arseeding/releases/tag/v1.0.3

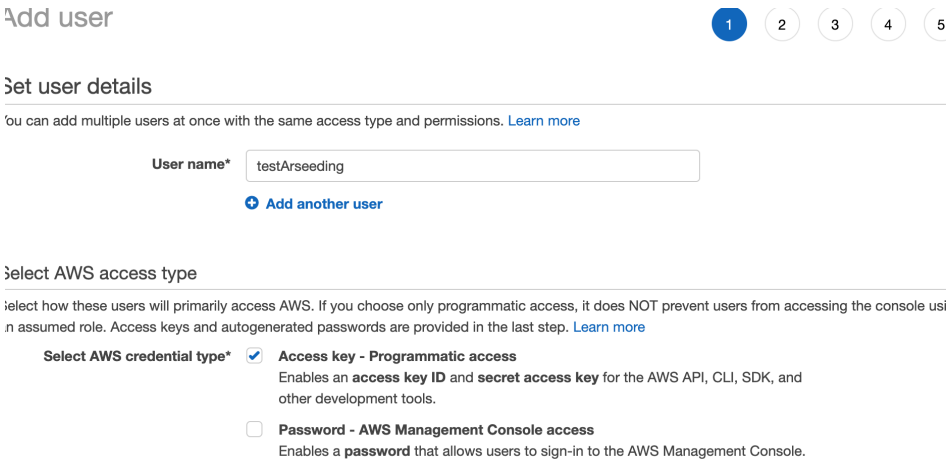
2.2 Running Preparation

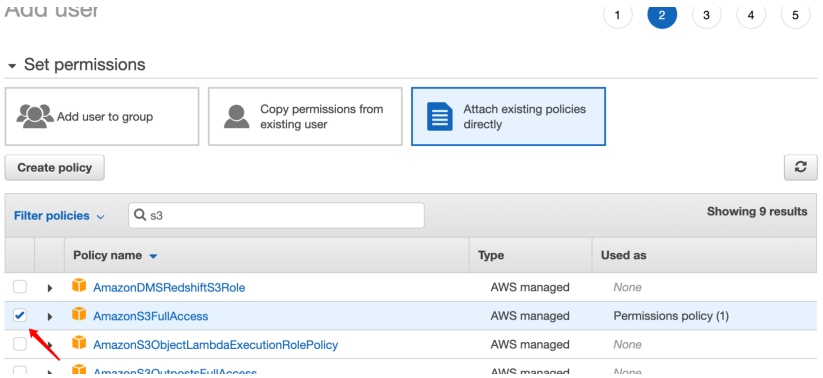
Arseeding Offers two storage types:BoltDB(local storage) and AWS S3(cloud storage).

we recommend using **s3** for storage so that you can get unlimited expansion(e.g many users upload data with your arseeding service,you should keep these big data for your users,at that moment local storage may not suit this situation)

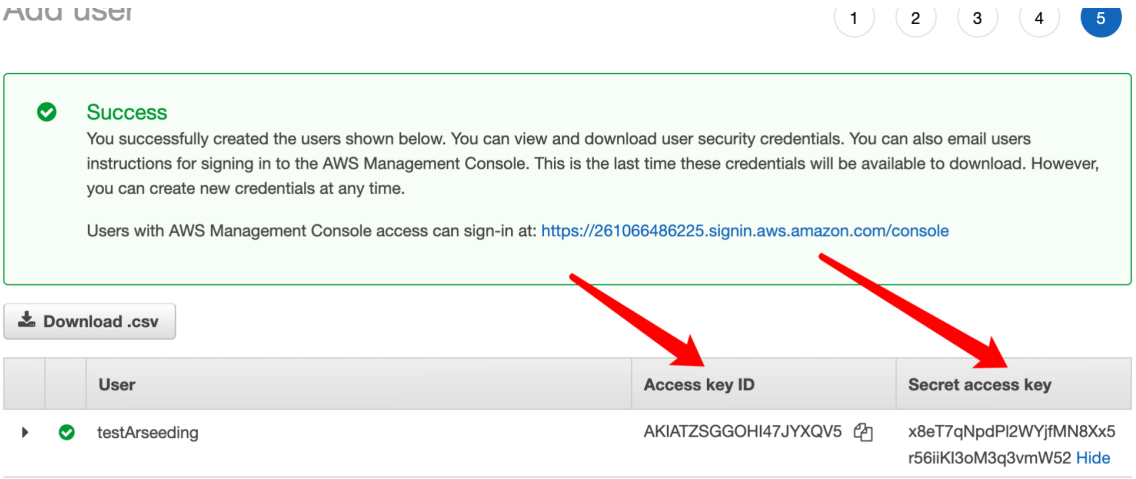
It's simple to start use s3

step1. register an aws account(if you haven't got one)
step2. create a IAM account follow the figures below





note: you can config these policies but this one must be chosen



These two keys will be used in arseeding when you access s3.
Also, you need to know where your bucket will be set,you can check this region info in aws, you should provide a proper bucket prefix when you start arseeding so that your bucket name will not conflict with other aws users who also use s3 service.

note:prefix shouldn't be too simple,(e.g S3_PREFIX=test, this bucket may be exist,you'd better choose a name represent the characteristics of your project,e.g S3_PREFIX=everpay)

2.3 Running Arseeding

you should config some environment variables first

If running arseeding with s3,you should export these variables first.
in your command line:

```
$ export USE_S3=true
$ export S3_ACC_KEY=Access key ID
$ export S3_SECRET_KEY=Secret access key
$ export S3_PREFIX=your prefix
```

```
$ export S3_REGION=region    // e.g ap-northeast-1
```

If you don't want to charge users for the arseeding service:

```
$ export NO_FEE=true
```

you should provide a mysql database DSN:

```
$ export MYSQL=your dsn // e.g
root@tcp(127.0.0.1:3306)/arseed?charset=utf8mb4&parseTime=True&loc=Local
```

you must got an AR wallet with sufficient balance so that you can upload data to arweave for your users,specific your wallet path:

```
$ export KEY_PATH=your ar keyfile     // e.g  ./data/keyfile.json
```

you can configure the port number(default is 8080):

```
$ export PORT=portNum  //e.g :8080
```

now you can run in command line:

```
$ sudo chmod 755 arseeding
$ ./arseeding
```

2.4 Running Arseeding With docker

first of all,get docker image:

```
$ docker pull everfinance/arseeding:v1.0.3
```

you need prepare environment variables like 2.3 and run docker with them

run with boltDB:

```
docker run -d \
--env DB_DIR=/arseeding/data/bolt \
--env KEY_PATH=/arseeding/data/keyfile.json \
--env PAY=https://api.everpay.io \
--env MYSQL="your mysql dsn" \
--env NO_FEE=true \ // “run with no fee mode”
-v /Your/KeyFile/Absolute/Path:/arseeding/data \
-p 8080:8080 \
everfinance/arseeding:v1.0.3 \
arseeding
```

run with s3:

```
docker run -d \
--env USE_S3=true \
--env S3_ACC_KEY=AccessKeyID \
--env S3_SECRET_KEY=SecretAccessKey \
```

```
--env S3_PREFIX=yourPrefix \
--env S3_REGION=region \
--env KEY_PATH=/arseeding/data/keyfile.json \
--env PAY=https://api.everpay.io \
--env MYSQL="your mysql dsn" \
--env NO_FEE=true \ // “run with no fee mode”
-v /Your/KeyFile/Absolute/Path:/arseeding/data \
-p 8080:8080 \
everfinance/arseeding:v1.0.3 \
arseeding
```

3 How to use Arseeding

Arseeding is a powerful infrastructure in arweave ecosystem. not only it provides the function of arweave gateway that you can qurey any arweave information and submit AR transanction，but also support synchronizing any arweave transaction to your arseeding storage and support ans-104 bundle transaction.

3.1 Compatible with arweave

figure 3.1: compatible APIs

You can call these apis just like calling the arweave gateway's api, you can get faster response than arweave gateway because arseeding cache some important information to memory.

3.2 Sync Broadcast Transaction

```
v1.POST("/task/:taskType/:arid", s.postTask)
v1.POST("/task/kill/:taskType/:arid", s.killTask)
v1.GET("/task/:taskType/:arid", s.getTask)
v1.GET("/task/cache", s.getCacheTasks)
```

figure 3.2: sync&broadcast APIs

arseeding provide three types of task:sync,broadcast,broadcast_meta.
If a transaction not exist in arseeding storage,use **sync**:
sync will download the transaction from arweave and store to arseeding.
broadcast task will broadcast an ar transaction to all arweave nodes that have been filtered.
kill task stops a specific task.

3.3 Bundle

Arseeding support bundle transaction.
Why use bundle transactions?

For example, there are so many users that each one wants to save a little bit of data to arweave.if you send each user's data with an **AR transaction** that will cost a lot of AR tokens.so there is another way that use bundle,each user's data will be packed in a bundle item with a itemId, all of the bundle items will be packed in **bundle data**,the bundle data will become ar transaction's data part, so that you just need to send only one ar transaction, each user can find their data by itemId.

how to create a bundle and send bundle item to arseeding,you can refer to：
https://github.com/everFinance/arseeding/blob/main/example/bundle-item/bundle_test.go

Furthermore:

you can learn more useful info from
arseeding sdk https://github.com/everFinance/arseeding/tree/main/sdk
goar https://github.com/everFinance/goar