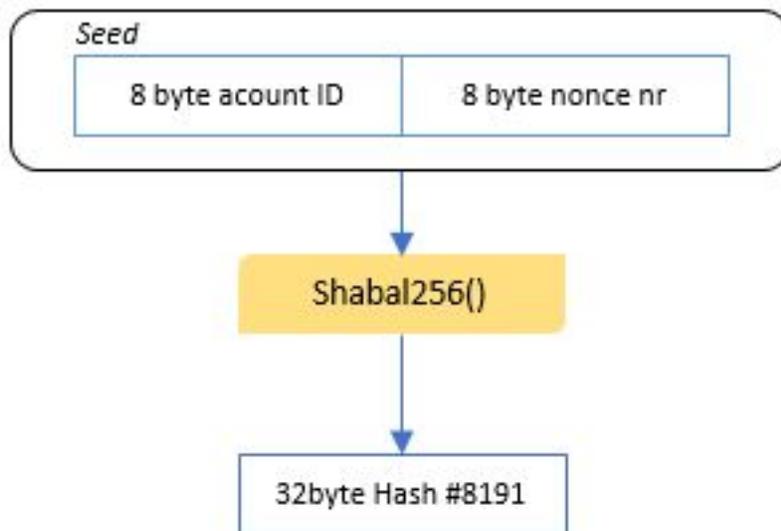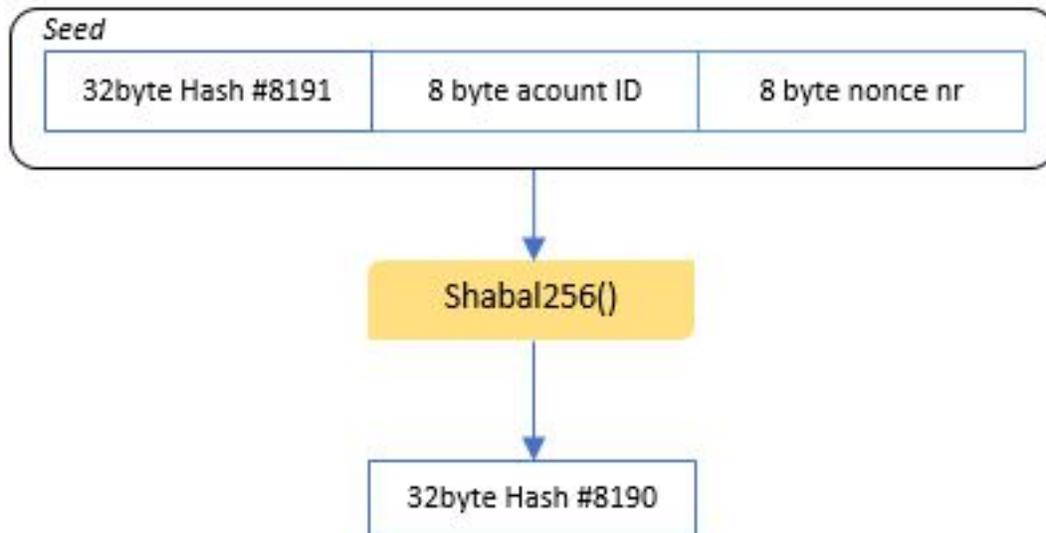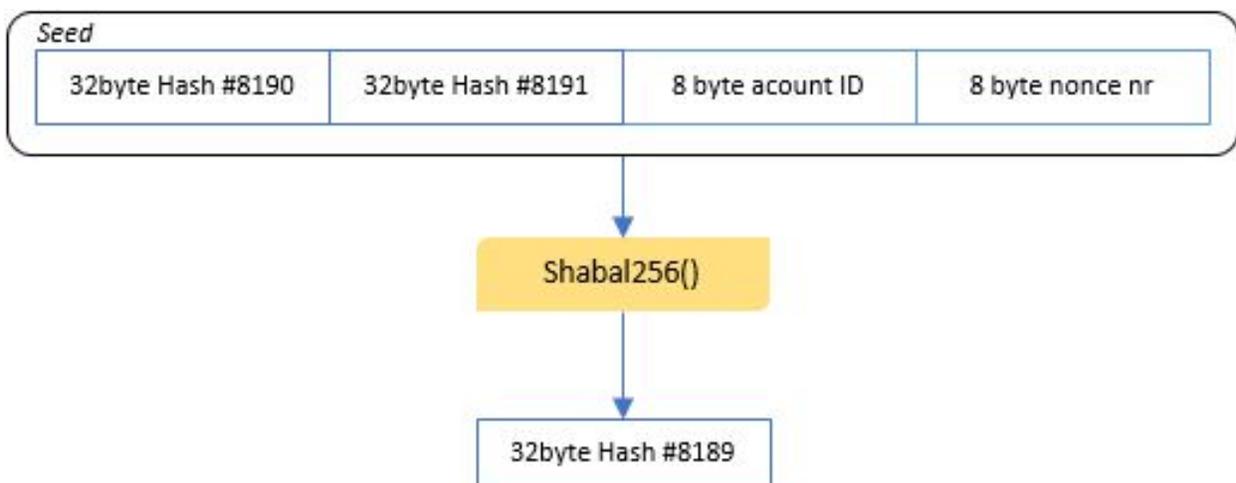# Generating a nonce

The first step in creating a nonce is to make the first seed. The seed is a 16byte long value containing the account id that we will be generating a nonce for and the nonce number. When this is done we start to feed the Shabal256 function to get our first hash.


We have produced the first hash. This is the last hash in the nonce. Hash #8191. Now we take this produced hash (#8191) and pre-append it to the starting seed. The result will now be our new seed for the next round of shabal256 computation.

```
Seed
┌─────────────────────┬──────────────────┬──────────────────┐
│  32byte Hash #8191  │  8 byte acount ID │  8 byte nonce nr │
└─────────────────────┴──────────────────┴──────────────────┘
                              │
                              ▼
                    ┌──────────────────┐
                    │    Shabal256()   │
                    └──────────────────┘
                              │
                              ▼
                    ┌──────────────────┐
                    │ 32byte Hash #8190│
                    └──────────────────┘
```

We now have produced two hashes. Hash #8191 and Hash #8190. This time we pre-append Hash 8190 to the last seed we used. The result will now be a new seed to feed Shabal256.

```
Seed
┌──────────────────┬──────────────────┬──────────────────┬──────────────────┐
│ 32byte Hash #8190│ 32byte Hash #8191│ 8 byte acount ID │  8 byte nonce nr │
└──────────────────┴──────────────────┴──────────────────┴──────────────────┘
                              │
                              ▼
                    ┌──────────────────┐
                    │    Shabal256()   │
                    └──────────────────┘
                              │
                              ▼
                    ┌──────────────────┐
                    │ 32byte Hash #8189│
                    └──────────────────┘
```

Once again, we have created a new hash. This procedure of pre-appending resulting hashes to a new seed will continue for all 8192 hashes we create for a nonce. After iteration 128 we have reached more than 4096 bytes in the seed. For all remaining iterations we will only read the last 4096 generated bytes.
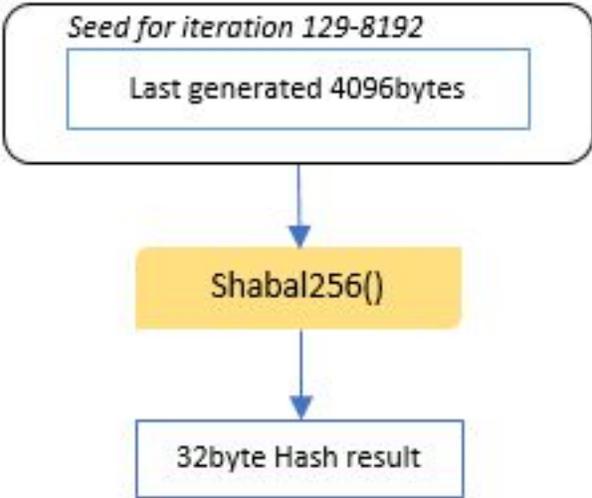
Once we have created 8192 hashes we are now going to make a Final hash. This is done by using all 8192 hashes and the first 16bytes as seed.
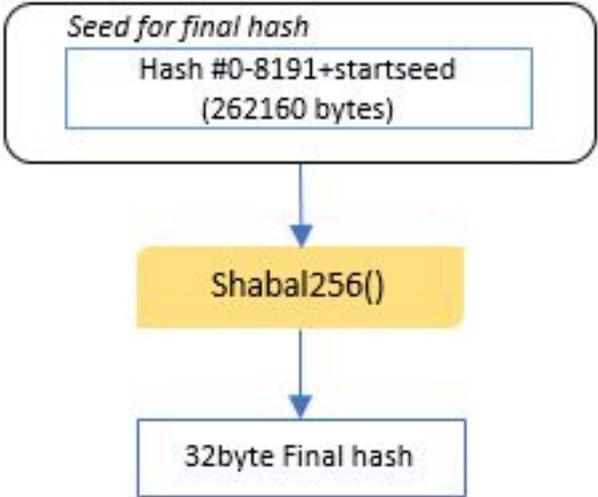
The final hash will now be used to xor all other hashes individually.

We have now created our nonce and can store it in a plot file before we continue to the next nonce.

Plot structure

When we are mining we read from one or more plot files. The miner software will open a plot file and seek the scoop locations to read the scoops data. If the plot file is unoptimized the scoop locations will be on more than one place. In the following example the miner will be seeking and reading scoop #403.
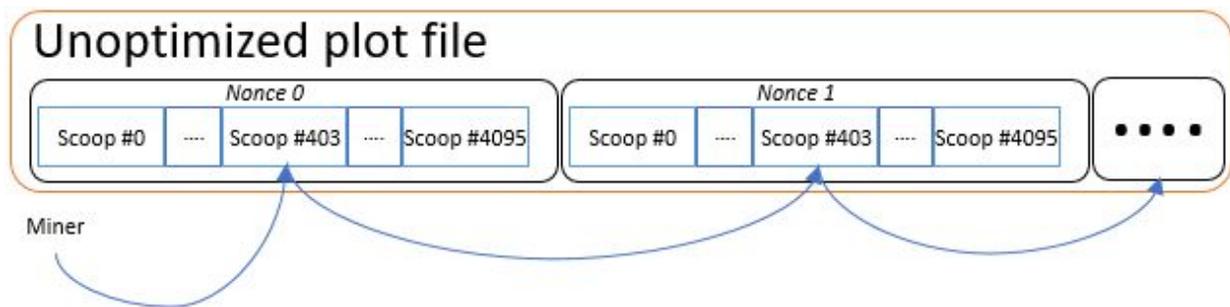
This is not the most effective way since the miner will spend a lot of time to seek new locations on the storage device to be able to read the scoops. To prevent this, we can optimize plots or use

plotter software that creates optimized plots from the beginning. Optimization is done by reordering the data in the plot file and grouping all data from the same scoop number together.