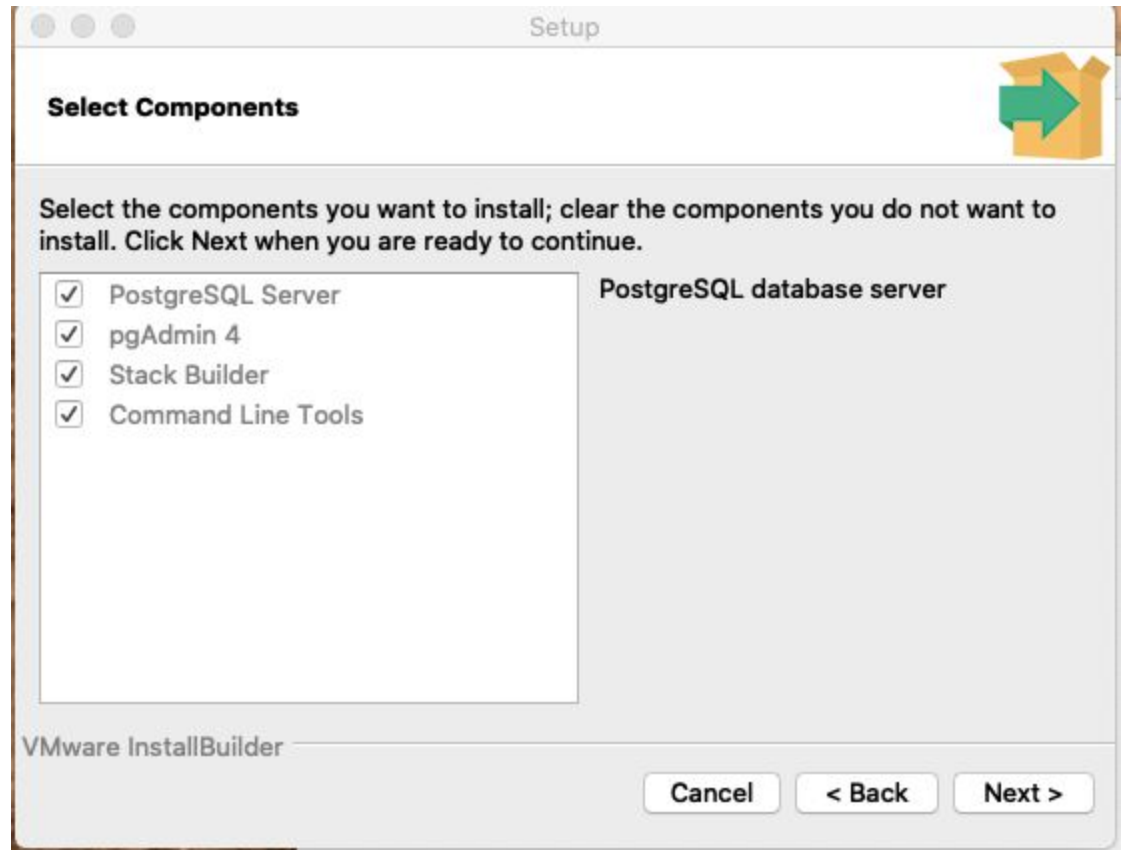


All the steps for getting chaind are technically on this page: <https://github.com/wealdtech/chaind>, but understanding them was a little bit complex. I hope that by reading this, you can save the time that I spent.



1. Download PostgreSQL & PGAdmin on your computer.
 - a. Go to <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads> and choose the latest version of PostgreSQL for your operating system.
 - b. Include all of the components when given the option and install the components.



2. Create a user called “chain” with a password (anything you want, just remember it) after going into your chaind application by:
 - a. Right-clicking on the “PostgreSQL 12” Server
 - b. Hover over “Create” and click on the “Group/Login Role” that pops up
 - c. You should see something like this:

The screenshot shows a window titled "Create - Login/Group Role" with a close button (X) in the top right corner. The window has several tabs: "General", "Definition", "Privileges", "Membership", "Parameters", "Security", and "SQL". The "General" tab is selected and active. It contains a "Name" label followed by an empty text input field, and a "Comments" label followed by a larger empty text area. At the bottom of the window, there is a red error bar with a warning icon and the text "Name cannot be empty." To the right of the error bar is a close button (X). Below the error bar are three buttons: an information icon (i), a question mark icon (?), and three action buttons: "Cancel", "Reset", and "Save".

- d. Type “chain” in the “Name” box, then go to “Definition” and type in your password. Finally, go to “Privileges” and click “yes” for “Can login” and “Superuser”, which will automatically select the first 6 choices.

You may be able to get away with just the “login” option, but I chose “Superuser” as well to be safe. Click save to finish creating the user.

Can login?	<input checked="" type="checkbox"/> Yes
Superuser?	<input checked="" type="checkbox"/> Yes
Create roles?	<input checked="" type="checkbox"/> Yes
Create databases?	<input checked="" type="checkbox"/> Yes
Update catalog?	<input checked="" type="checkbox"/> Yes
Inherit rights from the parent roles?	<input checked="" type="checkbox"/> Yes
Can initiate streaming replication and backups?	<input type="checkbox"/> No

i

?

✕ Cancel

↺ Reset

💾 Save

- e. Now, right-click on “Databases” under the server of your recently created user (the server was “PostgreSQL 12” for me) and hover over “Create” and click “Database”. Create a database called “chain” with owner “chain” and click save once you are done

Create - Database

General | Definition | Security | Parameters | Advanced | SQL

Database: chain

Owner: chain

Comment:

Buttons: [i] [?] [Cancel] [Reset] [Save]

3. If you are on a Mac OS like me, the binary files in the Github page won't work for you. At this point, you should
 - a. Download the Go compiler, if you don't already have it on your computer: <https://golang.org/doc/install>
 - b. Use the command `GO111MODULE=on go get github.com/wealdtech/chaind` to install chaind on your computer.
 - c. Now, we need to find out where exactly the Go compiler installed it: run `find go/*/chaind` to see where your chaind file is
 - d. For me it was in "go/bin/chaind" - remember this when you want to connect your node to your SQL database.

4. Set up your Teku node (instructions: <https://docs.teku.consensys.net/en/latest/HowTo/Get-Started/Run-Teku/>)
 - a. Install Homebrew: <https://brew.sh/> using the command on the first page. For me, the command only worked on Terminal and not iTerm, so you might have to experiment a bit.
 - b. Run "brew tap ConsenSys/teku" and then "brew install teku" to install Teku. Confirm it has been installed by doing "teku --version."
 - c. Start up Teku by typing "teku --rest-api-enabled --data-storage-mode=archive --data-path ~/Library/teku/mainnet --initial-state ~/Downloads/finalized-state.ssz"

- i. If you don't want to wait around 5 hours for your Mainnet beacon chain to dump all its data:
 1. Download the latest finalized state from:
["https://github.com/ajsutton/eth2-states/raw/master/mainnet/2021-02-16/finalized-state.ssz"](https://github.com/ajsutton/eth2-states/raw/master/mainnet/2021-02-16/finalized-state.ssz)
 2. After your `--initial-state` tag, type the absolute path to the location of the downloaded file (i.e. `--initial-state ~/[PATH TO DIRECTORY]/finalized-state.ssz`)
 3. WARNING: Do not include an `"=`" sign before the `'~'` sign, the `'~'` sign apparently needs space before it to do its thing. Therefore, be careful when typing in the tags for `--data-path` and `--initial-state`.
- ii. If you already have a directory called "Library/teku/beacon" where you have stored the test database, "prymont", specify another `--data-path` (you don't need to create it beforehand). E.g., `--data-path ~Library/teku/mainnet`