WORKSHOP: Making self-hosting p2p software scale with Nix(OS)

Eric Sirion

~~WORKSHOP: Making self-hosting p2p software scale with Nix(OS)~~

Eric Sirion

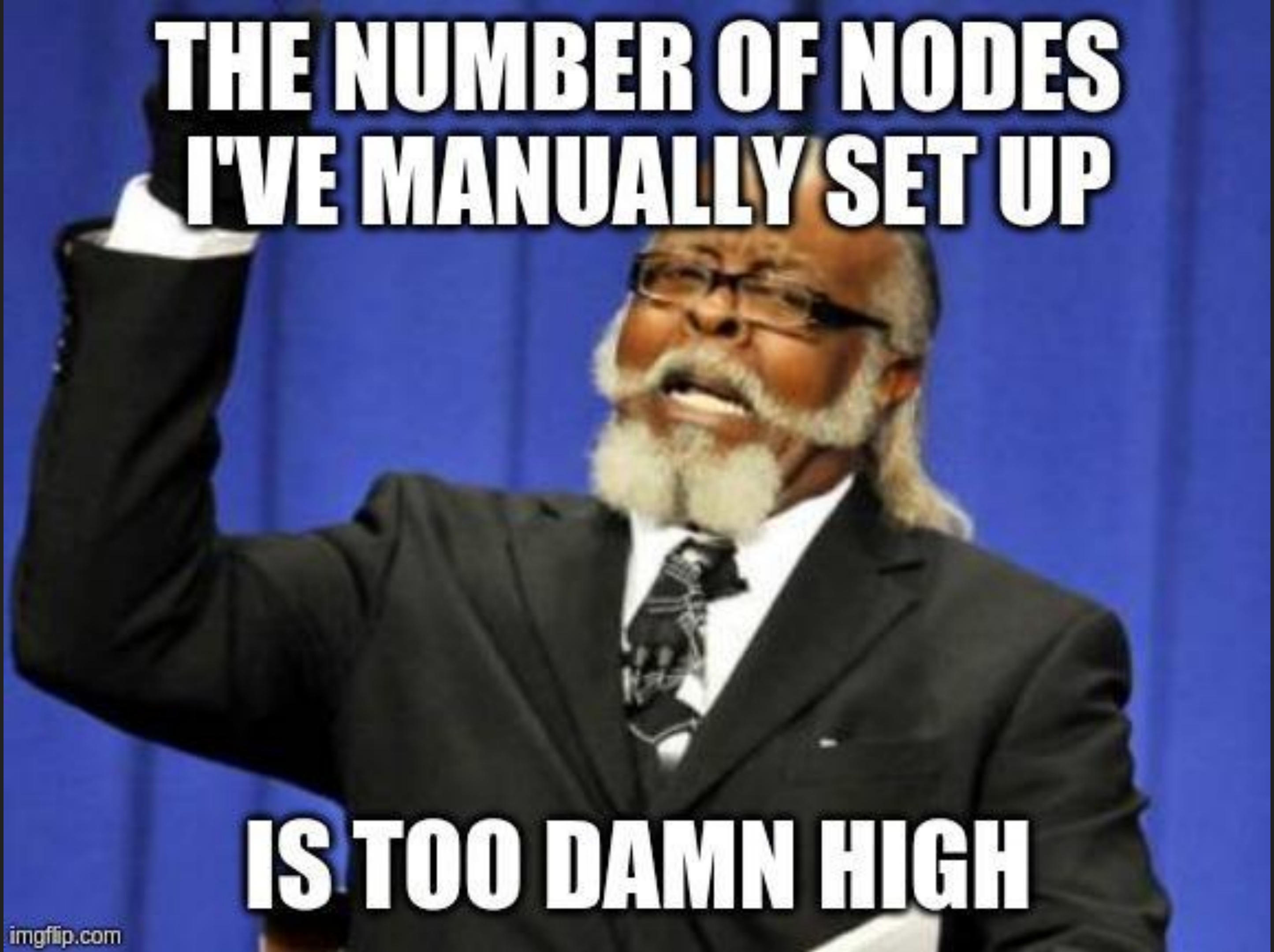
nix-bitcoin

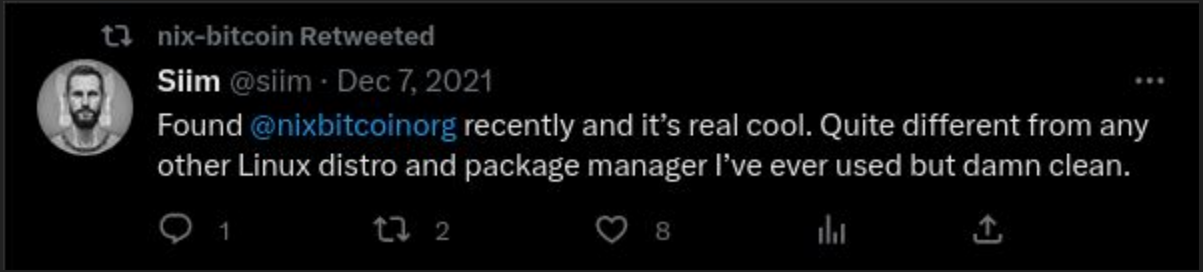
robust Bitcoin nodes for hackers (built on NixOS)

nixbitcoin.org

Jonas Nick (n1ckler)

Nix-bitcoin: A “Bitcoin node distribution” (like raspiblitz, Umbrel, etc).


THE NUMBER OF NODES
I'VE MANUALLY SET UP
IS TOO DAMN HIGH
imgflip.com

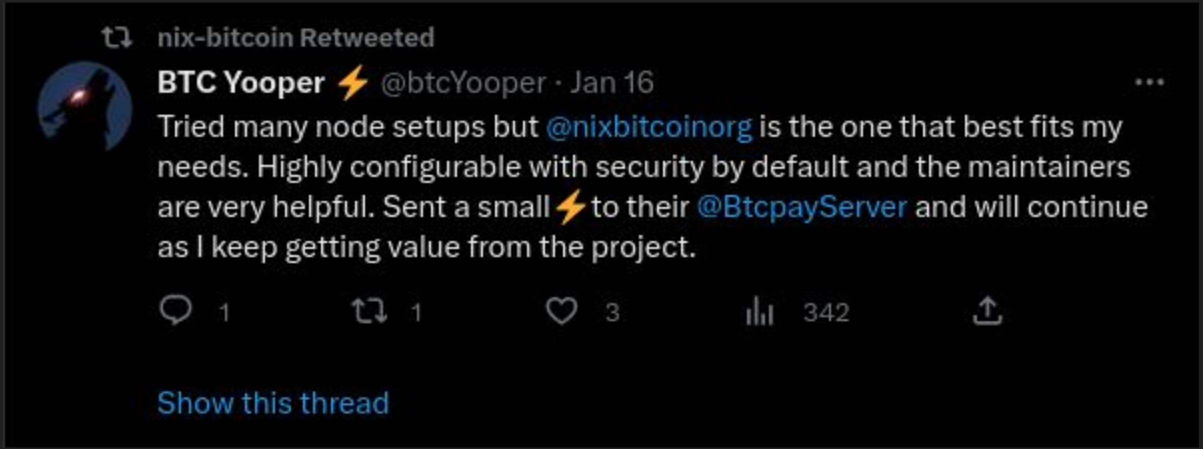


nix-bitcoin Retweeted

Siim @siim · Dec 7, 2021

Found @nixbitcoinorg recently and it’s real cool. Quite different from any other Linux distro and package manager I’ve ever used but damn clean.

💬 1 🔁 2 ♡ 8



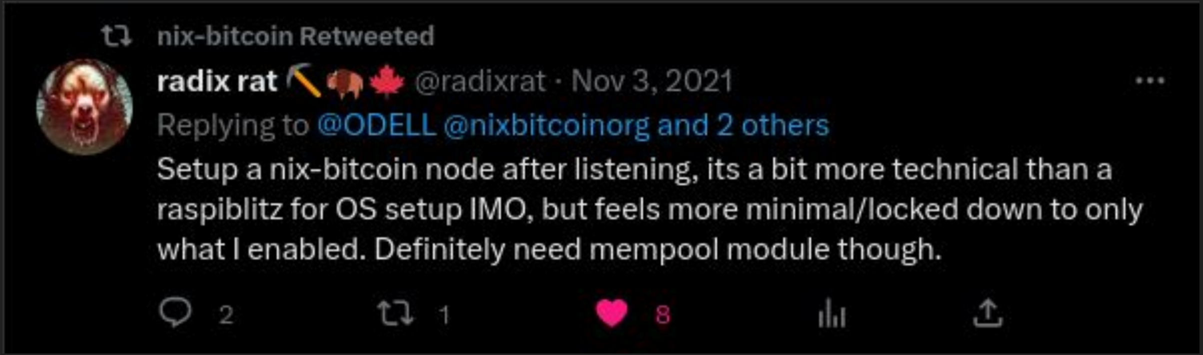
nix-bitcoin Retweeted

BTC Yooper ⚡ @btcYooper · Jan 16

Tried many node setups but @nixbitcoinorg is the one that best fits my needs. Highly configurable with security by default and the maintainers are very helpful. Sent a small ⚡ to their @BtcpayServer and will continue as I keep getting value from the project.

💬 1 🔁 1 ♡ 3 📊 342

Show this thread

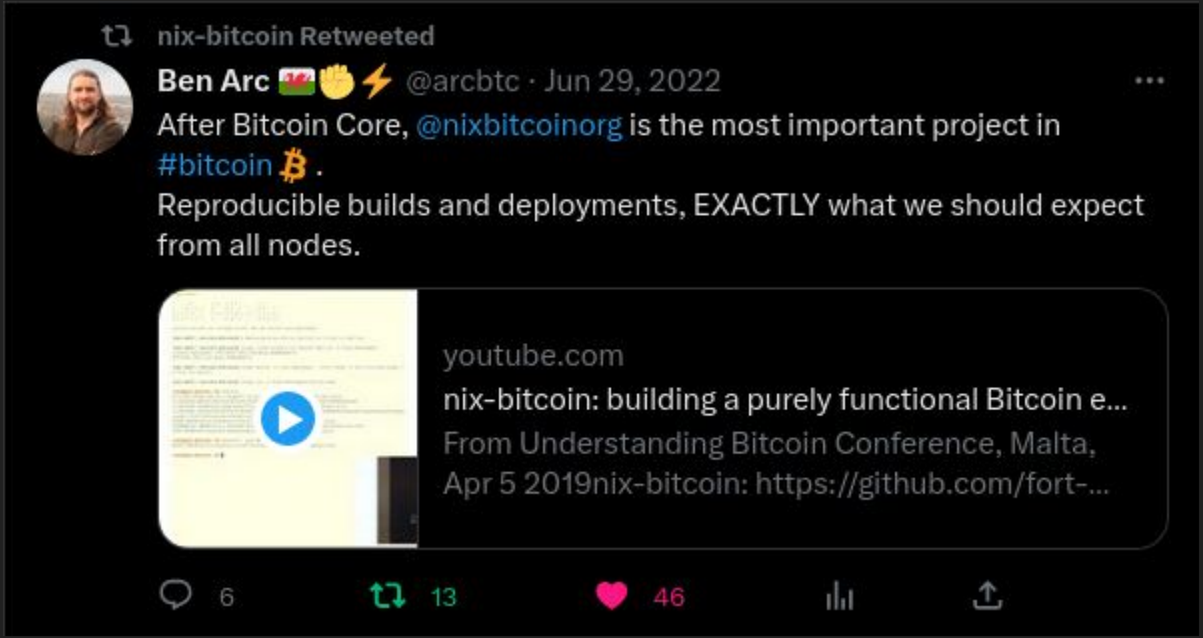


nix-bitcoin Retweeted

radix rat ⛏️🦬🍁 @radixrat · Nov 3, 2021

Replying to @ODELL @nixbitcoinorg and 2 others

Setup a nix-bitcoin node after listening, its a bit more technical than a raspiblitz for OS setup IMO, but feels more minimal/locked down to only what I enabled. Definitely need mempool module though.



nix-bitcoin Retweeted

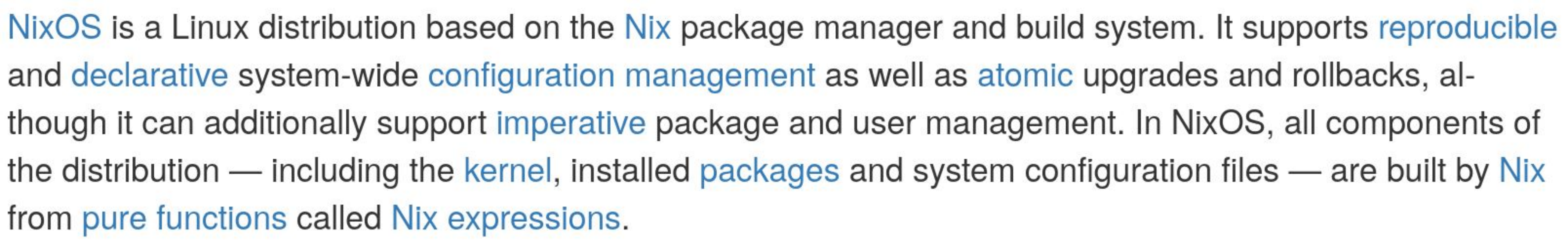
Ben Arc 🏴󠁧󠁢󠁷󠁬󠁳󠁿✊⚡ @arcbtc · Jun 29, 2022

After Bitcoin Core, @nixbitcoinorg is the most important project in #bitcoin ₿ .
Reproducible builds and deployments, EXACTLY what we should expect from all nodes.

youtube.com
nix-bitcoin: building a purely functional Bitcoin e...
From Understanding Bitcoin Conference, Malta, Apr 5 2019nix-bitcoin: https://github.com/fort-...

Why “damn clean, configurable, secure by default, minimal, reproducible”?

One crucial reason is that it’s build on NixOS!



NixOS is a Linux distribution based on the Nix package manager and build system. It supports reproducible and declarative system-wide configuration management as well as atomic upgrades and rollbacks, although it can additionally support imperative package and user management. In NixOS, all components of the distribution — including the kernel, installed packages and system configuration files — are built by Nix from pure functions called Nix expressions.

System-wide Config Management in Nix

configuration.nix (text files)

nix
tools

configuration.nix

```
{ config, pkgs, ... }: {
  imports = [
    ./hardware-configuration.nix
  ];
  services.bitcoind.enable = true;
  services.bitcoind.port = 8333;
  services.tor.hiddenServices.bitcoind = {
    map = [{port = config.services.bitcoind.port;}];
  };
}
```

```
$ nixos-rebuild switch
```

System-wide Config Management in Nix

1. The whole system config is just a few text files which can be kept under git version control.
2. Can easily deploy a configuration to a different machine
3. Nix expressions can be checked for well-formedness

```
services.bitcoind.port = 833w;
```

ERROR at build time

4. Allows nix-bitcoin to have an extensive test framework
5. Abstraction

nix-bitcoin Demo

System-wide Config Management in Nix

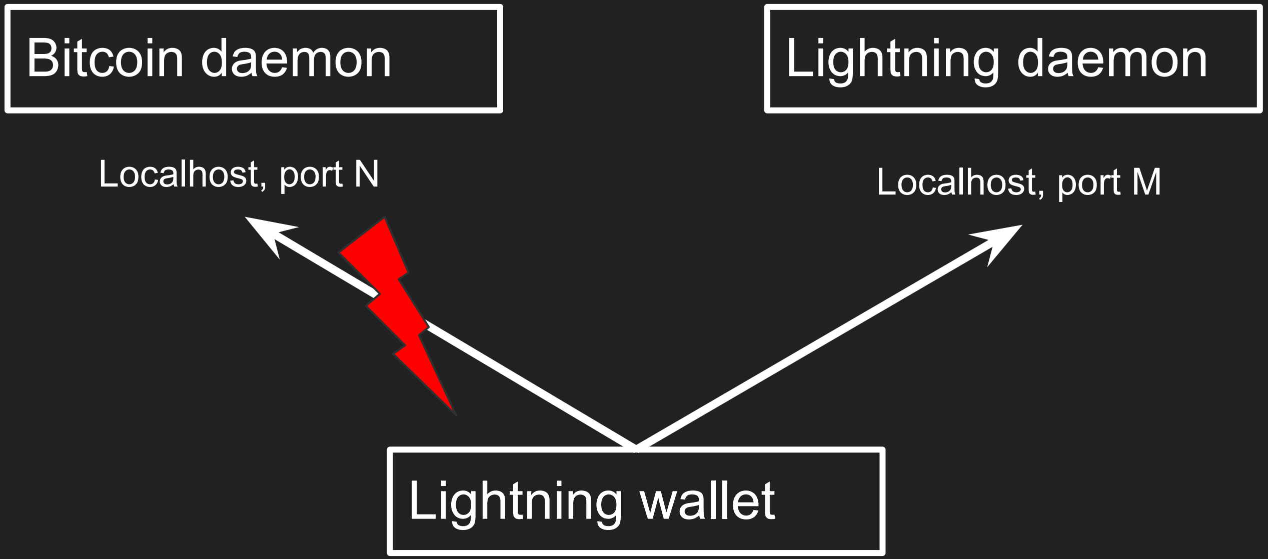
1. The whole system config is just a few text files which can be kept under git version control.
2. Can easily deploy a configuration to a different machine
3. Nix expressions can be checked for well-formedness

```
services.bitcoind.port = 833w;
```

ERROR at build time

4. Allows nix-bitcoin to have an extensive test framework
5. Abstraction

Abstraction: Network Namespace Example



Bitcoin daemon

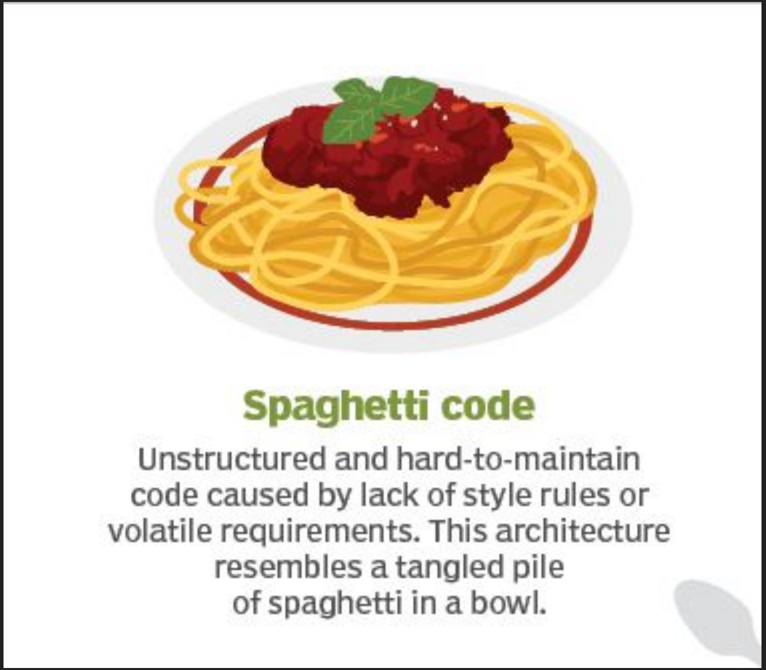
Lightning daemon

Localhost, port N

Localhost, port M

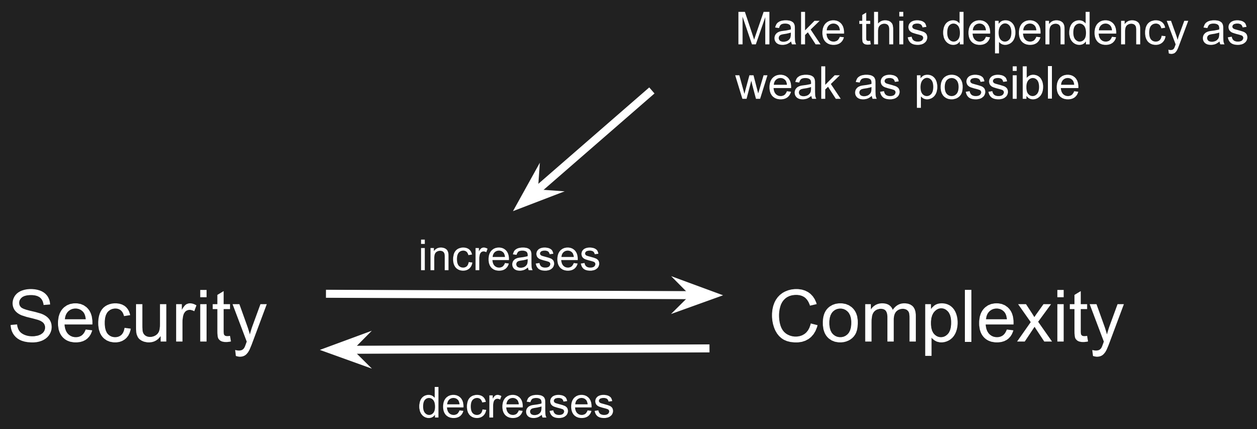
Lightning wallet

Solution: Put every service in own “network namespace” (Linux kernel feature). Then build internal router to connect namespaces.



Spaghetti code

Unstructured and hard-to-maintain code caused by lack of style rules or volatile requirements. This architecture resembles a tangled pile of spaghetti in a bowl.



Make this dependency as weak as possible

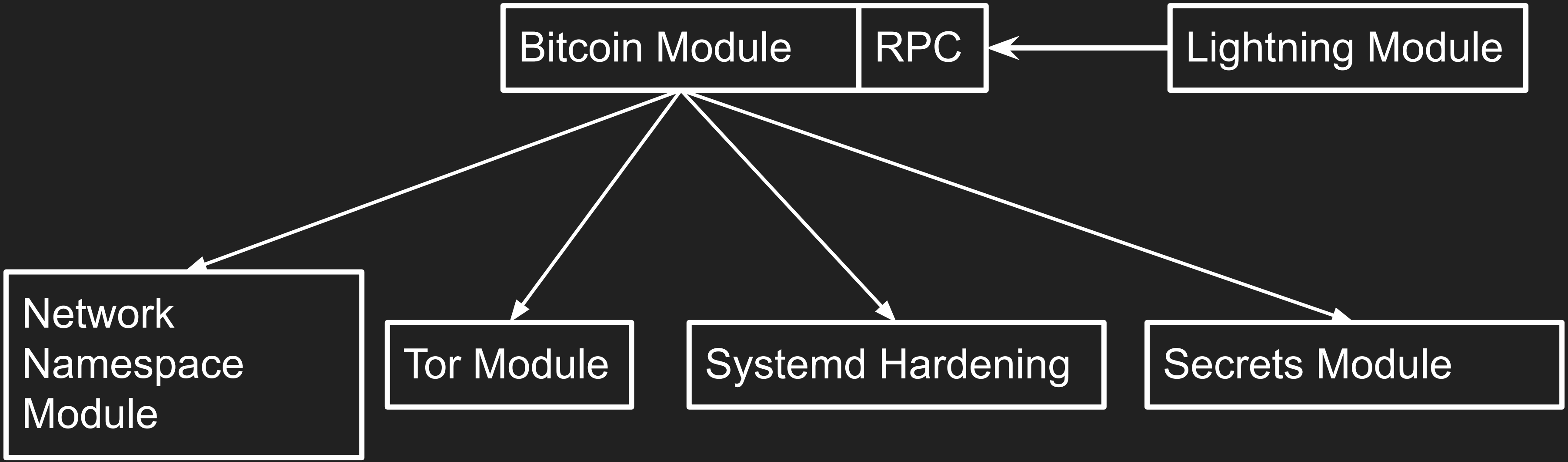
increases

Security

Complexity

decreases

Abstraction through (Nix) programming lang



Reproducibility

nix
tools

Dependencies

Uniquely identified via hashes

configuration.nix (text files)

Results in minimal dependencies

Reproducibility

Whole system config (including dependencies) is uniquely identified via hash

```
$ readlink /run/current-system

/nix/store/gzg4ld3i4vymbrrk49lrgr865z906qn5-nixos-system-ae
gon-23.05.20230509.9524f57
```

Format: /nix/store/<hash>-<output name>

Reproducibility

```
$ nix-store -q --references /run/current-system

/nix/store/il0y46mhxbpnvrfr4jr8vyqwazmg9x32-linux-6.2.14
/nix/store/mdcw7q80ndiyrwk85m7fwzfij1rl2201-etc
<...>
```

```
/nix/store/mdcw7q80ndiyrwk85m7fwzfij1rl2201-etc
-> /nix/store/ywcz7w8nwhcw6czdn2yhb8vwkii8rhwa-system-units
--> /nix/store/2xnybcwb4yw6nnh4w4r5c74c3yg4hy4k-unit-bitcoind.service
---> /nix/store/l3iggv1aagmgdf0ypywdgfc6n4q0vg1j-bitcoind-24.0.1
```

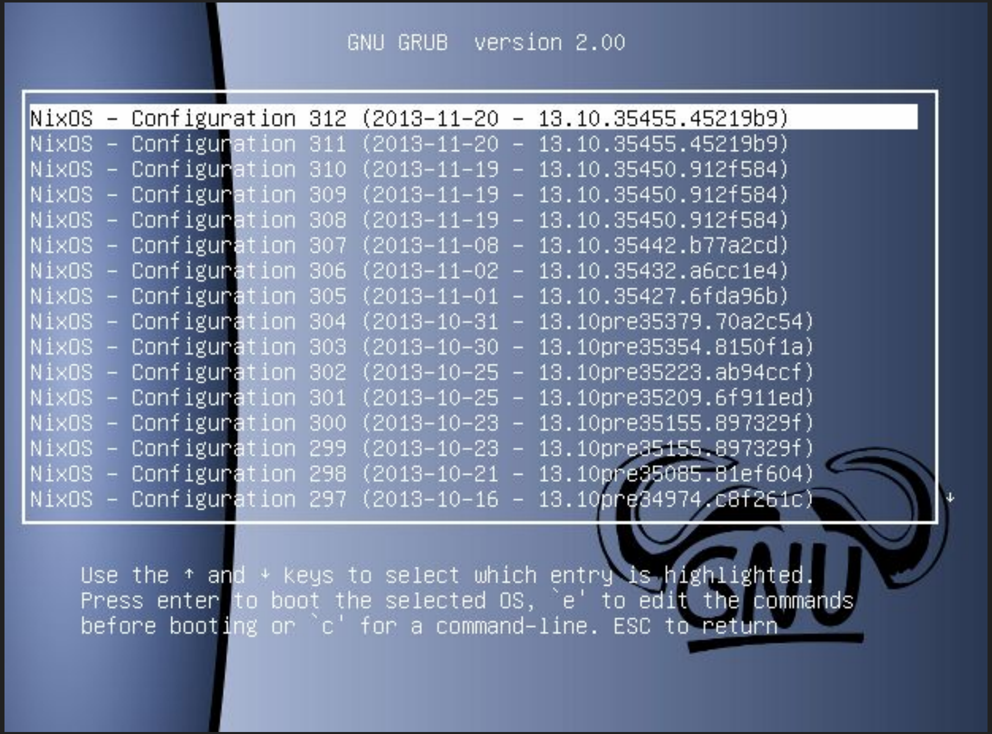
Reproducibility

```
$ nix-store -q --references
/nix/store/l3iggv1aagmgdf0ypywdgfc6n4q0vg1j-bitcoind-24.0.1
/nix/store/vnwdak3n1w2jjil119j65k8mw1z23p84-glibc-2.35-224
/nix/store/1rjl52i7gh0wasc7a1p82xk3lc79icww-sqlite-3.39.4
/nix/store/6plx60y4x4q2lfp6n7190kaihyxr7m1w-gcc-11.3.0-lib
/nix/store/7gd7g4b94sd80255cgvyan4ik0041wkh-zeromq-4.3.4
/nix/store/ah5bq30cclmcphcwhcab9y2hdpvklfb1-db-4.8.30
/nix/store/hdq6324rm7fjx3pbgdndmfdl7z62yf6r-libevent-2.1.12
/nix/store/jmbb7dcq6bczkvqd55ymxhknnihmzffl-miniupnpc-2.2.4
```

Reproducible

- Updates are easy
- Can build all dependencies from source to avoid trusting binary “caches”. 99.97% of the nixos minimal iso are binary reproducible.

Rollbacks



```
                    GNU GRUB  version 2.00

NixOS - Configuration 312 (2013-11-20 - 13.10.35455.45219b9)
NixOS - Configuration 311 (2013-11-20 - 13.10.35455.45219b9)
NixOS - Configuration 310 (2013-11-19 - 13.10.35450.912f584)
NixOS - Configuration 309 (2013-11-19 - 13.10.35450.912f584)
NixOS - Configuration 308 (2013-11-19 - 13.10.35450.912f584)
NixOS - Configuration 307 (2013-11-08 - 13.10.35442.b77a2cd)
NixOS - Configuration 306 (2013-11-02 - 13.10.35432.a6cc1e4)
NixOS - Configuration 305 (2013-11-01 - 13.10.35427.6fda96b)
NixOS - Configuration 304 (2013-10-31 - 13.10pre35379.70a2c54)
NixOS - Configuration 303 (2013-10-30 - 13.10pre35354.8150f1a)
NixOS - Configuration 302 (2013-10-25 - 13.10pre35223.ab94ccf)
NixOS - Configuration 301 (2013-10-25 - 13.10pre35209.6f911ed)
NixOS - Configuration 300 (2013-10-23 - 13.10pre35155.897329f)
NixOS - Configuration 299 (2013-10-23 - 13.10pre35155.897329f)
NixOS - Configuration 298 (2013-10-21 - 13.10pre35085.81ef604)
NixOS - Configuration 297 (2013-10-16 - 13.10pre34974.c8f261c)     ↓

    Use the ↑ and ↓ keys to select which entry is highlighted.
    Press enter to boot the selected OS, `e' to edit the commands
    before booting or `c' for a command-line. ESC to return
```

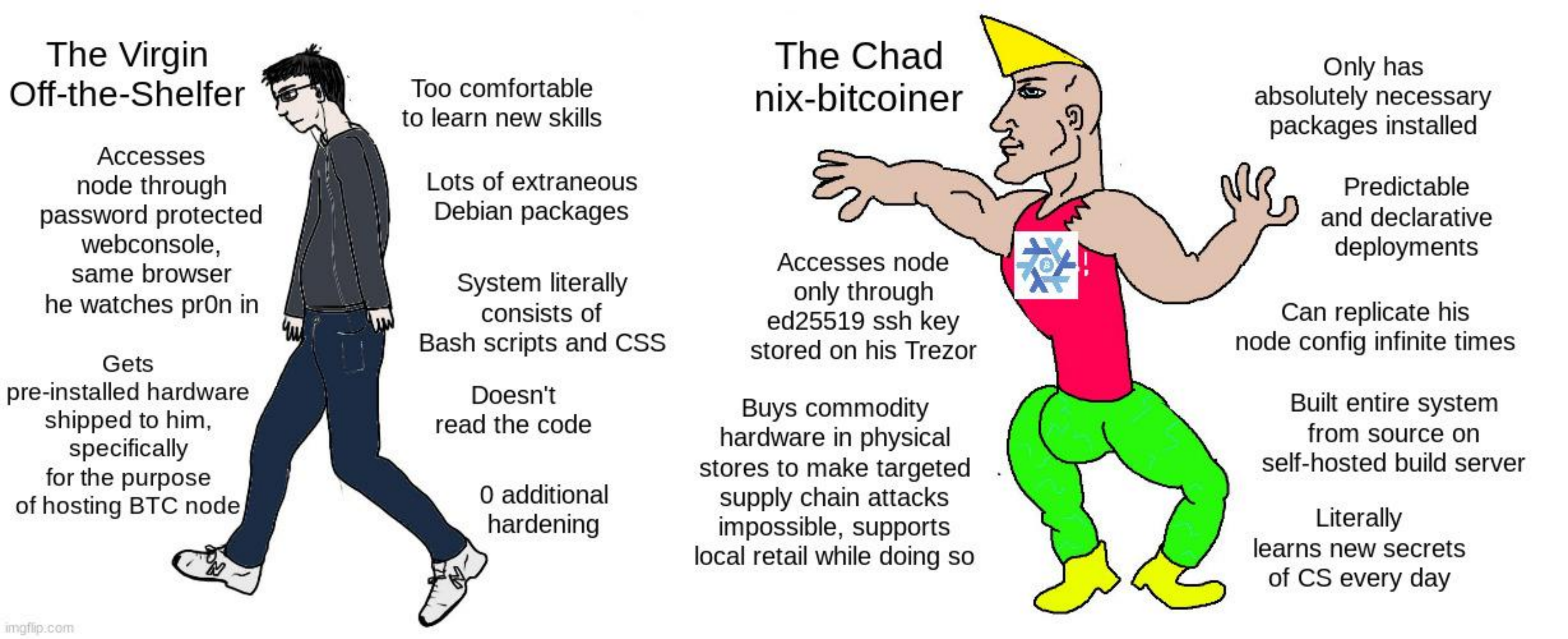
FUTURE work (™)

- Nix-bitcoin is highly customizable and extensible
  - Add new external modules?
  - Make user friendly node distribution based on nix-bitcoin

Before you start, some clarification

- Nix: package manager, available on most Linux distributions and MacOS
  - There’s a new feature called flakes, which basically revamped the UX of nix entirely
- NixOS: operating system, built on Nix
- Nixpkgs: “official” collection of Nix packages and NixOS modules

```
nix run github:fort-nix/nix-bitcoin
```


The Virgin Off-the-Shelfer
Accesses node through password protected webconsole, same browser he watches pr0n in
Gets pre-installed hardware shipped to him, specifically for the purpose of hosting BTC node
Too comfortable to learn new skills
Lots of extraneous Debian packages
System literally consists of Bash scripts and CSS
Doesn't read the code
0 additional hardening
The Chad nix-bitcoiner
Accesses node only through ed25519 ssh key stored on his Trezor
Buys commodity hardware in physical stores to make targeted supply chain attacks impossible, supports local retail while doing so
Only has absolutely necessary packages installed
Predictable and declarative deployments
Can replicate his node config infinite times
Built entire system from source on self-hosted build server
Literally learns new secrets of CS every day
imgflip.com

Slides at
[nickler.ninja/slides]

Resources

- nixbitcoin.org
- Examples
- Getting Started guide
- Nix-bitcoin community

- Zero-to-nix
- NixOS manual
- nix.dev
- Nix pills