

**Stredná priemyselná škola elektrotechnická**  
Hálova 16, 851 01 Bratislava

KOMPLEXNÁ ODBORNÁ PRÁCA

# LosOS

Bezstavový operačný systém pre vlastný cloud a mesh úložisko

Autor: Matúš Maštena  
Konzultant:  
Študijný odbor:  
Školský rok: 2026/2027

Bratislava 2027

## Čestné vyhlásenie

Vyhlasujem, že som komplexnú odbornú prácu vypracoval samostatne pod vedením konzultanta, s použitím uvedených zdrojov. Pri vývoji softvéru som používal nástroje asistované umelou inteligenciou; návrh systému, rozhodnutia o architektúre, recenzie a overenie každej zmeny sú mojou prácou a celá história vývoja je verejne dostupná v repozitári projektu.

V Bratislava, dňa .....

.....

Matúš Maštena

# Obsah

|                                                                   |           |
|-------------------------------------------------------------------|-----------|
| <b>LosOS: technická dokumentácia projektu .....</b>               | <b>1</b>  |
| <b>Abstrakt .....</b>                                             | <b>2</b>  |
| <b>Abstract .....</b>                                             | <b>3</b>  |
| <b>Zadanie a kde je zodpovedané .....</b>                         | <b>4</b>  |
| <b>Ako čítať .....</b>                                            | <b>5</b>  |
| <b>1 Úvod .....</b>                                               | <b>6</b>  |
| 1.1 Motivácia .....                                               | 6         |
| 1.2 Cieľ práce .....                                              | 6         |
| 1.3 Rozsah a čo práca nie je .....                                | 6         |
| 1.4 Metodika .....                                                | 6         |
| 1.5 Štruktúra dokumentu .....                                     | 7         |
| <b>2 Hlavné funkcionality a možnosti systému .....</b>            | <b>8</b>  |
| 2.1 Čo je LosOS .....                                             | 8         |
| 2.2 Čo na boxe beží .....                                         | 8         |
| 2.3 Funkcie pre majiteľa .....                                    | 9         |
| 2.4 Možnosti nasadenia .....                                      | 11        |
| 2.5 Mesh a trh .....                                              | 12        |
| 2.6 Porovnanie s bežnými riešeniami .....                         | 13        |
| 2.7 Čo LosOS nie je .....                                         | 14        |
| <b>3 Architektúra systému .....</b>                               | <b>15</b> |
| 3.1 Šesť rozhodnutí, ktoré nesú návrh .....                       | 15        |
| 3.2 Dva systémy z jednej sady modulov .....                       | 16        |
| 3.3 Úložisko .....                                                | 16        |
| 3.4 Záťaž a jednotný vzhľad .....                                 | 17        |
| 3.5 Edge .....                                                    | 17        |
| 3.6 Bezpečnostný model v skratke .....                            | 17        |
| 3.7 Prečo NixOS .....                                             | 18        |
| <b>4 Výber kernelu .....</b>                                      | <b>19</b> |
| 4.1 Čo jadro na boxe musí zvládnuť .....                          | 19        |
| 4.2 Kandidáti a rozhodnutie .....                                 | 19        |
| 4.3 Čo je na jadro navrstvené .....                               | 20        |
| 4.4 Tri odporúčania, ktoré sú zámerne vypnuté .....               | 21        |
| 4.5 Čo sledovanie najnovšieho jadra stojí a ako sa to platí ..... | 21        |
| 4.6 Zhrnutie .....                                                | 22        |
| <b>5 Inštalácia .....</b>                                         | <b>23</b> |
| 5.1 Inštalačné médiá .....                                        | 23        |
| 5.2 Požiadavky .....                                              | 23        |
| 5.3 Priebeh .....                                                 | 23        |
| 5.4 Čo inštalátor robí .....                                      | 24        |
| 5.5 Inštalácia vo virtuálnom stroji (ukážka) .....                | 25        |
| 5.6 Po inštalácii .....                                           | 26        |
| <b>6 Základná konfigurácia .....</b>                              | <b>27</b> |
| 6.1 Prvé spustenie: sprievodca .....                              | 27        |
| 6.2 Administračné stránky: panely a Použiť .....                  | 29        |
| 6.3 Dve veci, ktoré nie sú súčasťou popisu .....                  | 30        |

|           |                                                                  |           |
|-----------|------------------------------------------------------------------|-----------|
| 6.4       | Konfigurácia na úrovni systému .....                             | 31        |
| 6.5       | Príklad: zapnutie vzdialeného prístupu .....                     | 31        |
| <b>7</b>  | <b>Overenie správnosti konfigurácie a funkčnosti .....</b>       | <b>32</b> |
| 7.1       | Zásada: testy kódujú argumenty, nie len správanie .....          | 32        |
| 7.2       | Vrstva 1: automatizované testy .....                             | 32        |
| 7.3       | Hlavná skúška: mesh funguje, s edge proxy aj bez neho .....      | 33        |
| 7.4       | Overenie jedného boxu vo virtuálnom stroji .....                 | 35        |
| 7.5       | Vrstva 3: nezávislá kritická recenzia .....                      | 35        |
| 7.6       | Čo overené nie je .....                                          | 36        |
| <b>8</b>  | <b>Cenová ponuka hardvérového a softvérového vybavenia .....</b> | <b>37</b> |
| 8.1       | Z čoho sa realizácia skladá .....                                | 37        |
| 8.2       | Hardvér boxu .....                                               | 37        |
| 8.3       | Hardvér edge .....                                               | 38        |
| 8.4       | Softvér .....                                                    | 38        |
| 8.5       | Čo ponuka zámerne neobsahuje .....                               | 38        |
| 8.6       | Výsledok ponuky .....                                            | 39        |
| <b>9</b>  | <b>Technická dokumentácia projektu .....</b>                     | <b>40</b> |
| 9.1       | Vrstvy dokumentácie .....                                        | 40        |
| 9.2       | Príručka majiteľa .....                                          | 40        |
| 9.3       | Tento dokument .....                                             | 40        |
| 9.4       | Ako sa dokumentácia drží v súlade s kódom .....                  | 41        |
| 9.5       | Čo dokumentácii chýba .....                                      | 41        |
| <b>10</b> | <b>Záver .....</b>                                               | <b>42</b> |
| 10.1      | Čo bolo dosiahnuté .....                                         | 42        |
| 10.2      | Čo je plánované .....                                            | 42        |
| 10.3      | Poučenia .....                                                   | 42        |
| <b>11</b> | <b>Prílohy .....</b>                                             | <b>44</b> |
| 11.1      | Príloha A: Slovník pojmov .....                                  | 44        |
| 11.2      | Príloha B: Testy a kontroly .....                                | 44        |
| 11.3      | Príloha C: Štruktúra repozitára .....                            | 45        |
| 11.4      | Príloha D: Zdroje .....                                          | 45        |

# LosOS: technická dokumentácia projektu

**Komplexná odborná práca (KOP)** k praktickej časti odbornej zložky maturitnej skúšky. Stredná priemyselná škola elektrotechnická, Hálova 16, Bratislava, školský rok 2026/2027. Autor: Matúš Maštena. Konzultant: *(doplňuje autor)*.

## English summary

This chapter is the Slovak technical documentation handed in for the author's matriculation project (the KOP). It follows the four official requirements: (1) describe the system's main functions and options, (2) choose a kernel, install the system, configure it and verify it, (3) prepare a price offer for the hardware and software, (4) write the technical documentation. The English handbook around it is the source for most of its content; the chapter adds the kernel rationale, the verification procedure and the price offer. A PDF of the whole chapter is built from these pages by the handbook's CI job and published at [losos.dasmat.us/kop/LosOS-KOP.pdf](https://losos.dasmat.us/kop/LosOS-KOP.pdf).

# Abstrakt

LosOS je bezstavový operačný systém postavený na NixOS, ktorý z vyradeného mini-PC urobí domáci alebo firemný cloud, ktorý sa sám aktualizuje a spravuje sa výlučne cez webové rozhranie: bez SSH a bez prihlasovania do systému. Koreňový súborový systém je pri každom štarte postavený nanovo z jedného deklaratívneho popisu, trvalé dáta ležia na šifrovanom zväzku s kľúčom v čipe TPM. Na boxe beží Nextcloud (LosOS cloud) a Forgejo (LosOS Git) ako záťaž v lokálnom Kubernetes clustri a voliteľne sa box pripája k sieti ďalších boxov (mesh), ktorej požičiava voľný disk a procesor. Táto práca opisuje funkcie a možnosti systému, zdôvodňuje výber jadra, dokumentuje inštaláciu, základnú konfiguráciu a jej overenie, uvádza cenovú ponuku hardvéru a softvéru a opisuje, ako je projekt zdokumentovaný a testovaný.

# Abstract

LosOS is a stateless operating system built on NixOS that turns a repurposed mini-PC into a home or company cloud which updates itself and is administered only through a web page: no SSH, no shell logins. The root filesystem is rebuilt on every boot from one declarative description; durable data lives on an encrypted volume whose key is kept in the TPM chip. The box runs Nextcloud (LosOS cloud) and Forgejo (LosOS Git) as workloads in a local Kubernetes cluster and can optionally join a mesh of other boxes, lending them its spare disk and CPU. This document describes the system's functions and options, justifies the kernel choice, documents the installation, the basic configuration and its verification, gives a price offer for the hardware and software, and describes how the project is documented and tested.

## Zadanie a kde je zodpovedané

| Bod zadania                                                                                                                                    | Kapitola                                                                  |
|------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------|
| 1. Preskúmajte a opíšte hlavné funkcionality a možnosti vybraného operačného systému                                                           | 2 Funkcionality a možnosti, 3 Architektúra                                |
| 2a. Vyberte vhodný kernel                                                                                                                      | 4 Výber kernelu                                                           |
| 2b. Nainštalujte operačný systém                                                                                                               | 5 Inštalácia                                                              |
| 2c. Vykonajte základnú konfiguráciu                                                                                                            | 6 Základná konfigurácia                                                   |
| 2d. Overte správnosť konfigurácie a funkčnosť                                                                                                  | 7 Overenie: mesh dvoch boxov s edge aj bez neho, testy, kontrolné zoznamy |
| 3. Spracujte cenovú ponuku hardvérového a softvérového vybavenia (navrhnutý dovetok: <i>v prípade reálneho nasadenia</i> , čaká na potvrdenie) | 8 Cenová ponuka: dve konfigurácie reálneho nasadenia                      |
| 4. Vypracujte technickú dokumentáciu                                                                                                           | tento dokument a 9 Dokumentácia projektu                                  |

# Ako čítať

Dokument je písaný pre komisiu a konzultanta, teda pre technicky zdatného čitateľa, ktorý projekt nepozná. Každá kapitola začína tým, čo systém robí, a až potom vysvetľuje, prečo je to tak navrhnuté. Odkazy na súbory v repozitári (`modules/boot.nix` a podobne) smerujú do [github.com/dasmatus/losos](https://github.com/dasmatus/losos); každý významný súbor má v hlavičke komentár so zdôvodnením, prečo vyzerá tak, ako vyzerá. Obrázky sú snímky skutočného systému z ukážok vo virtuálnom stroji.

Zdrojom tohto dokumentu sú stránky Markdown v priečinku `handbook/docs/project/kop/` repozitára; z nich sa pri každej zmene zostaví aj PDF na odovzdanie (kapitola 9 hovorí ako).

# 1 Úvod

## 1.1 Motivácia

Fotky, dokumenty, kalendár a kód väčšiny ľudí ležia v cudzích cloudoch. Alternatíva, vlastný server, znamená údržbu: prihlasovať sa, aktualizovať, sledovať, či niečo nespadlo, a po čase nevedieť, v akom stave server vlastne je. Pritom doma alebo v kancelárii často stojí vyradené mini-PC, ktoré väčšinu dňa nerobí nič.

LosOS vznikol z dvoch otázok. Prvá: dá sa postaviť server, ktorý sa po inštalácii už nikdy nemusí spravovať inak než klikaním na webovej stránke, a ktorý sa sám opraví, keď sa niečo pokazí? Druhá: dá sa voľný disk a procesor takého servera bezpečne požičať iným ľuďom v sieti, a raz za to aj dostať zaplatené?

## 1.2 Cieľ práce

Cieľom je navrhnúť, implementovať, nainštalovať a zdokumentovať operačný systém pre takéto zariadenie (v práci ho voláme **box**), ktorý:

- sa spravuje len cez webové rozhranie, bez SSH a bez shellu;
- je **bezstavový**: pri každom štarte sa postaví nanovo z jedného textového popisu, takže nemôže „zvetrať“ do stavu, ktorý nikto nezapísal;
- chráni dáta šifrovaním disku s kľúčom v čipe TPM a bootuje bez obsluhy;
- poskytuje vlastný cloud (Nextcloud) a hosting kódu (Forgejo);
- sa voliteľne pripája do siete ďalších boxov (**mesh**) a požičiava im voľný disk a procesor podľa pravidiel, ktoré určí majiteľ;
- má pripravený trh, cez ktorý sa požičaná kapacita dá predávať.

## 1.3 Rozsah a čo práca nie je

Práca pokrýva operačný systém a jeho riadiacu rovinu: NixOS moduly, démon v Ruste, webové rozhranie, inštalátor, serverovú stranu siete (edge) a testy. Aplikácie samotné (Nextcloud, Forgejo, Kubernetes, Longhorn) sú prevzaté hotové projekty; práca ich konfiguruje, tematicky zjednocuje a izoluje, ale nemení ich kód.

Ukážka systému prebieha **vo virtuálnom stroji**, nie na fyzickom mini-PC. Tento spôsob bol s konzultantom dohodnutý, aby obhajoba nezávisela od sieťového pripojenia boxu v škole. Obraz pre virtuálny stroj sa zostavuje z toho istého zdroja ako inštalačné médium pre hardvér, takže komisia vidí presne ten systém, ktorý sa inštaluje na fyzický počítač.

## 1.4 Metodika

Projekt je verejný repozitár na GitHube s históriou každej zmeny. Pracovný postup bol pri každom kroku rovnaký:

1. zapísať návrh a jeho zdôvodnenie (do hlavičky súboru alebo do dokumentácie);
2. implementovať;
3. pokryť testom, ktorý zlyhá, keď niekto rozhodnutie neskôr zmení;
4. nechať zmenu prejsť kontinuálnou integráciou (CI) a ukážkou vo virtuálnom stroji.

Veľká časť kódu vznikla s pomocou AI asistenta ako nástroja. Rozhodnutia o architektúre, výber technológií, recenzia každej zmeny a overenie, že sa systém správa tak, ako tvrdí, boli prácou autora; v histórii repozitára je pri každej zmene vidieť, čo bolo navrhnuté, čo bolo vrátené a prečo. Kapitola

7 opisuje aj nezávislú kritickú recenziu, ktorú autor dal urobiť nad vlastným projektom, a ako sa jej nálezy opravili.

## **1.5 Štruktúra dokumentu**

Kapitoly 2 a 3 odpovedajú na prvý bod zadania (funkcionality, možnosti a architektúra). Kapitoly 4 až 7 na druhý bod (kernel, inštalácia, konfigurácia, overenie). Kapitola 8 je cenová ponuka, kapitola 9 opisuje dokumentáciu projektu a kapitola 10 je záver. Prílohy obsahujú slovník, zoznam testov, štruktúru repozitára a zdroje.

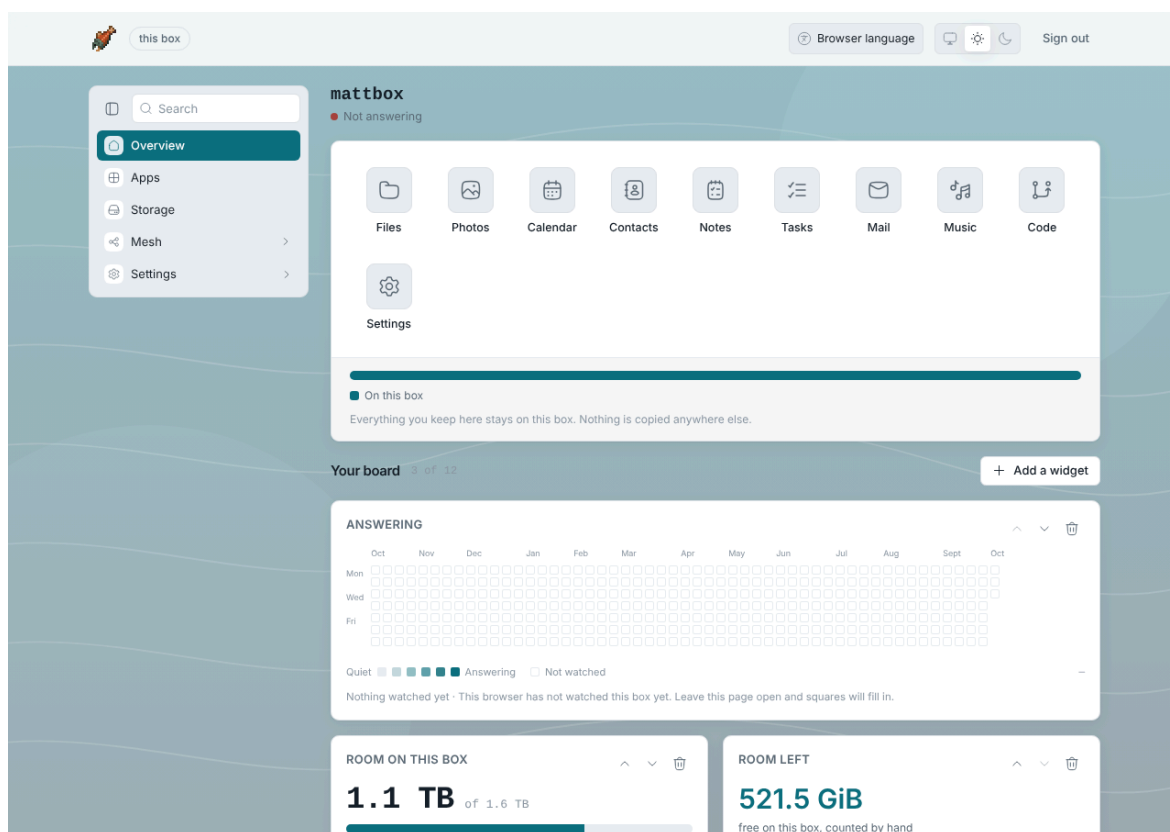
## 2 Hlavné funkcionality a možnosti systému

Táto kapitola odpovedá na prvý bod zadania: *preskúmajte a opíšte hlavné funkcionality a možnosti vybraného operačného systému*. Opisuje LosOS tak, ako ho vidí majiteľ boxu; kapitola 3 potom vysvetľuje, ako je to postavené.

### 2.1 Čo je LosOS

LosOS je operačný systém pre jeden účel: urobiť z malého počítača, spravidla z vyradeného kancelárskeho mini-PC, **zariadenie** (appliance). Majiteľ ho raz nainštaluje z USB kľúča, zapojí do siete a napájania, a odvtedy sa s ním stretáva iba cez webový prehliadač. Box nemá obrazovku na prihlásenie, nemá shell a nemá SSH server. Každú noc sa sám reštartuje a aktualizuje.

Systém je postavený na **NixOS**, linuxovej distribúcii, v ktorej je celý systém opísaný jedným deklaratívnym textom (tzv. *flake*). Z toho istého popisu sa zostavuje inštalčné médium, nainštalovaný systém, obraz pre virtuálny stroj aj automatizované testy.



Obrázok 1: Prehľad boxu v administračných stránkach: dlaždice aplikácií, karta úložiska a nástenka s widgetmi.

### 2.2 Čo na boxe beží

| Čo majiteľ vidí       | Čo to je                                                                                           |
|-----------------------|----------------------------------------------------------------------------------------------------|
| LosOS cloud           | Súbory, fotky, kalendár, kontakty, poznámky a úlohy. Postavené na Nextcloud, na adrese /nextcloud. |
| LosOS Git             | Repozitáre kódu. Postavené na Forgejo, na adrese /forgejo/, štandardne zapnuté.                    |
| Administračné stránky | Nastavenie a správa boxu. Na adrese /, dostupné len z lokálnej siete.                              |

| Čo majiteľ vidí   | Čo to je                                                                               |
|-------------------|----------------------------------------------------------------------------------------|
| Príručka          | Kompletný návod majiteľa vrátane vyhľadávania, na adrese /handbook/, aj bez internetu. |
| Mesh              | Voliteľné. Voľný disk a procesor požičaný iným boxom LosOS a požičaný od nich.         |
| Vzdialený prístup | Voliteľné. Verejná adresa cez server <i>edge</i> , bez otvárania portov doma.          |

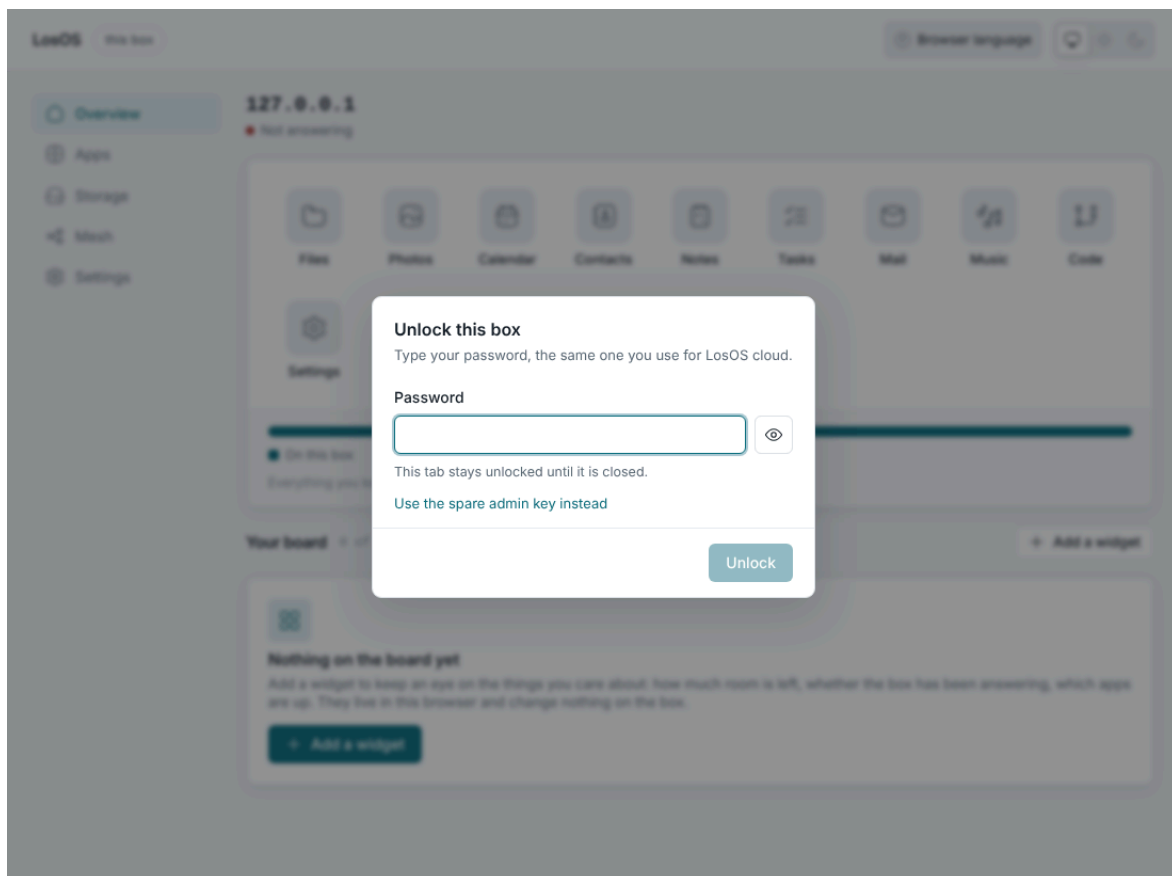
Obe aplikácie majú jednotný vzhľad a názvy LosOS cloud a LosOS Git; farby pochádzajú z jedného súboru tokenov (admin-ui/themes/tokens.css), ktorý zdieľajú administračné stránky, obe aplikácie aj príručka.

## 2.3 Funkcie pre majiteľa

### 2.3.1 Jedno heslo a náhradný kľúč

Heslo nastavené pri prvom spustení je heslo majiteľa pre LosOS cloud a zároveň odomyká administračné stránky. Box neudržiava druhú kópiu hesla: pri odomknutí sa opýta Nextcloudu cez vlastné lokálne spojenie, takže zmena hesla v Nextcloud platí okamžite aj pre administráciu. Nové heslo musí mať aspoň 12 znakov, malé aj veľké písmeno, číslicu a symbol.

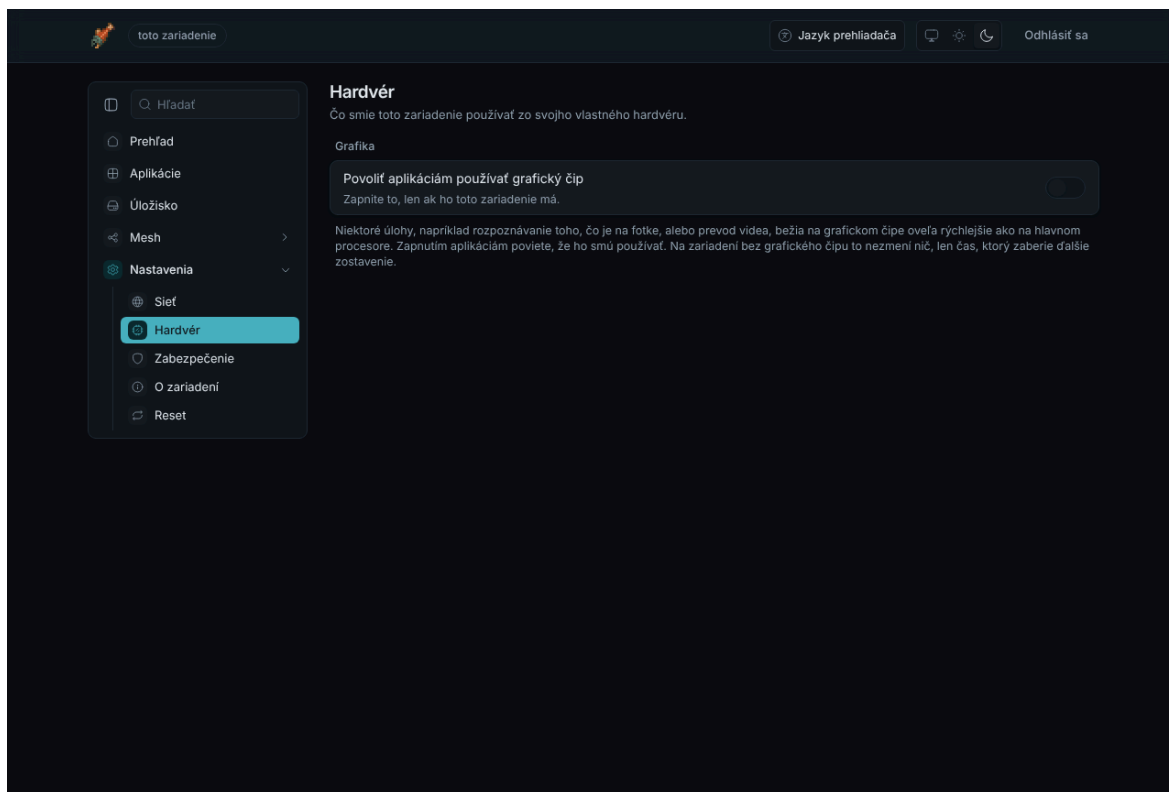
Pre jediný prípad, ktorý heslo nepokryje (LosOS cloud práve nebeží), existuje **náhradný administračný kľúč**: 64-znakový reťazec, ktorý si box vyrazil pri prvom štarte a ukázal ho majiteľovi raz, v sprievodcovi, s tlačidlami Kopírovať a Tlačiť. Na HTTPS spojení sa dá pridať aj **passkey**.



Obrázok 2: Odomknutie administračných stránok heslom; odkaz na náhradný kľúč je pre prípad, že LosOS cloud nebeží.

### 2.3.2 Administračné stránky

Jednostránková aplikácia (React) s piatimi sekciami: **Prehľad** (stav boxu, úložisko, dlaždice aplikácií, nástenka), **Aplikácie** (režim každej aplikácie a vyhľadávanie v katalógu aplikácií Nextcloudu), **Úložisko** (zaplnenie disku a rezerva, do ktorej box vie narášť), **Mesh** (pripojenie k iným boxom, hodiny, kedy box požičiava procesor, a zatiaľ sivý **Trh** so zdieľaním disku) a **Nastavenia** (Sieť, Hardvér, Zabezpečenie, Vzhľad, O zariadení, Reset). Rozhranie je v angličtine, slovenčine a nemčine, vo svetlej a tmavej téme.



Obrázok 3: Administračné stránky po slovensky, v tmavej téme: panel Hardvér.

Zmeny nastavení sa neukladajú „do systému“, ale do popisu boxu: tlačidlo **Použiť** zapíše voľby majiteľa do jedného súboru (`modules/overrides.nix`) a box sa z neho prestavia (*rebuild*). Zmena buď prejde celá, alebo vôbec; box medzitým ostáva dostupný. Kapitola 6 to opisuje podrobne.

### 2.3.3 Vzhľad a widgety

Majiteľ si môže nastaviť pozadie a závoj administračných stránok a na nástenku Prehľadu pridávať **widgety**: vstavané (napríklad história odpovedania boxu, voľné miesto), widgety zostavené z meraní boxu bez kódu a widgety **napísané ručne** v HTML a JavaScripte. Ručne písané widgety bežia v izolovanom rámci (`/widget - f frame/`) bez prístupu k administračnému kľúču a API; podrobnosti sú v kapitole o bezpečnosti.

### 2.3.4 Úložisko a rast disku

Inštalátor zámerne necháva asi 10 % disku nevyužitých ako rezervu. Panel Úložisko ju vie kedykoľvek pripojiť za behu, bez otvárania boxu: príkaz `losos-ctl grow` postupne vykoná `lvextend`, `cryptsetup resize` a `resize2fs`. Poradie je jediné správne a je overené testom (`tests/resize.nix`), lebo chyba v ňom je tichá.

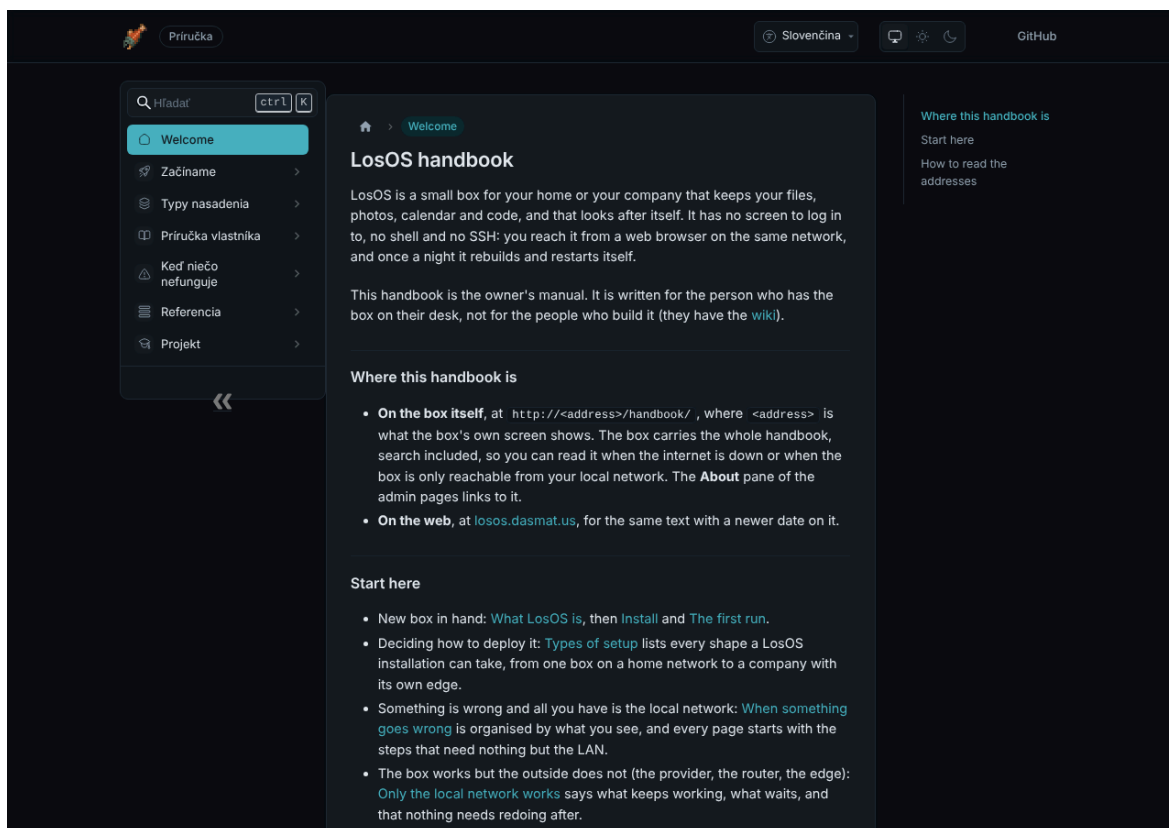
### 2.3.5 Aktualizácie a nočný reštart

Na každom boxe bežia tri časovače: o **00:07** bezpodmienečný reštart (koreň je `tmpfs`, takže reštart je oprava), o **03:00** prestavba systému zo zdroja aktualizácií a o **04:30** upratanie starých verzií (staršie

ako 14 dní) pri zachovaní posledných piatich generácií v boot menu. Zlyhaná aktualizácia nechá bežať predchádzajúci systém a ďalšiu noc sa skúsi znova.

### 2.3.6 Príručka na boxe

Každý box nesie celú príručku majiteľa vrátane offline vyhľadávania, lebo návod je najviac potrebný práve vtedy, keď funguje len lokálna sieť. Chybové hlásenia v administratívnych stránkach odkazujú na príslušnú stranu príručky („Čo robiť“).



Obrázok 4: Príručka majiteľa so slovenským rozhraním, tmavá téma.

## 2.4 Možnosti nasadenia

Box má niekoľko nezávislých „typov“, ktoré sa dajú kombinovať.

### 2.4.1 Spôsob nasadenia

| Typ                         | Pre koho                                    | Vzdialený prístup     | Mesh                       | Trh              |
|-----------------------------|---------------------------------------------|-----------------------|----------------------------|------------------|
| Samostatný box              | domácnosť, jeden stôl                       | nie                   | nie                        | nie              |
| Box s oficiálnym edge LosOS | majitelia, ktorí chcú verejnú adresu a mesh | áno                   | áno, s ostatnými           | áno, po otvorení |
| Firma s vlastným edge       | firma s viacerými boxmi v jednej sieti      | áno, cez vlastný edge | áno, medzi vlastnými boxmi | nie, zámerne     |

Každý box začína ako samostatný; ďalšie dva typy sú nastavenie, nie preinštalovanie. **Edge** je server (na internete alebo v lokálnej sieti), ku ktorému box drží tunel smerom von: cez neho dostane verejnú adresu bez otvárania portov, pripojí sa k meshu a (po otvorení) obchoduje na trhu. Oficiálny edge sa boxu preukazuje podpisom, ktorý box overuje verejným kľúčom projektu zabudovaným v systéme; firemný edge dostane objavovanie, vzdialený prístup a zdieľanie medzi vlastnými boxmi, ale nie obchodovanie.

## 2.4.2 Odomykanie disku a spôsob štartu

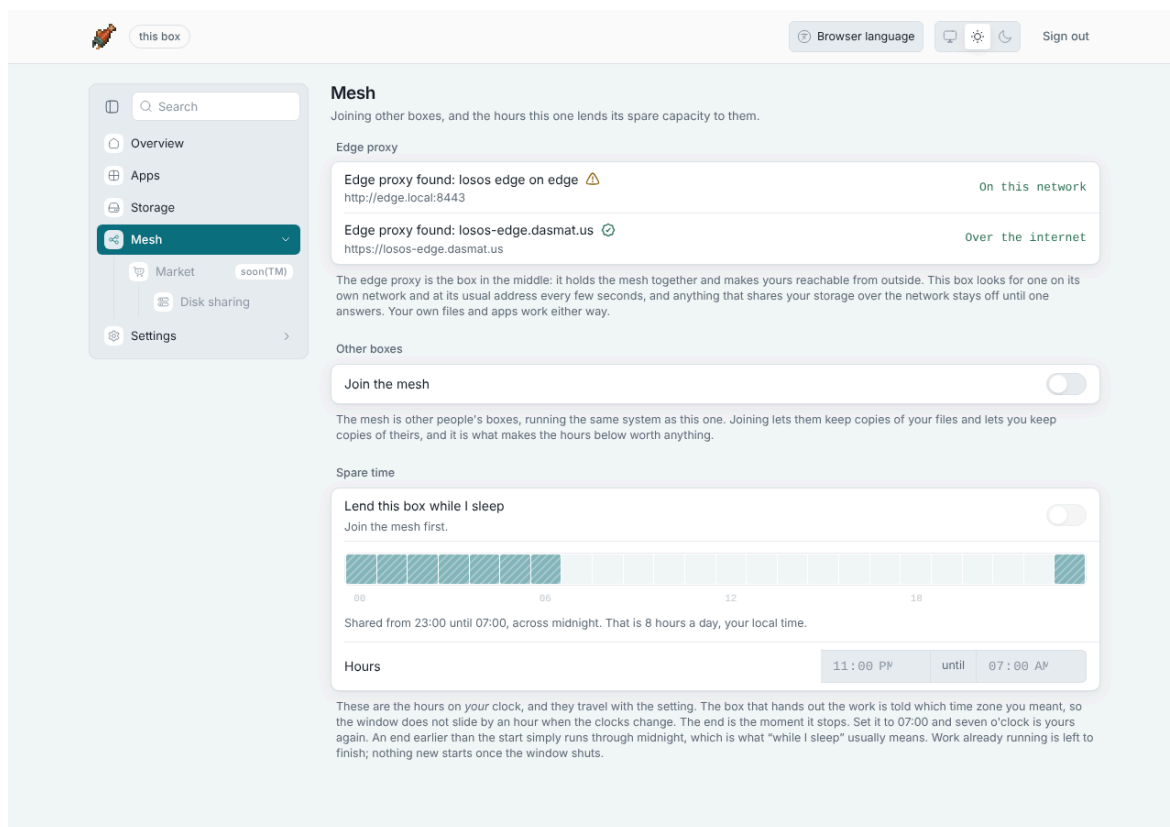
Disk je vždy šifrovaný (LUKS) a formátovaný bez obsluhy z náhodného kľúča. Na stroji s čipom **TPM 2.0** sa kľúč zapečatí do čipu a na boot oddiele nie je žiadne tajomstvo; bez čipu (alebo s `--no-tpm`) sa kľúč vloží do initrd na nešifrovanom boot oddiele, čo je slabšia, ale funkčná konfigurácia pre virtuálne stroje a testy. Inštalátor podporuje **UEFI** (systemd-boot) aj **BIOS** (GRUB) a vie režim rozpoznať sám. Jediná podporovaná architektúra je **x86-64**: flake zostavuje len x86\_64-linux, takže počítače s ARM, vrátane Macov s čipom Apple M, LosOS nespustia ani ako hostiteľ virtuálneho stroja bez pomalej emulácie.

## 2.4.3 Režim aplikácií

LosOS cloud a LosOS Git bežia ako **záťaž (workload)** v lokálnom Kubernetes clustri boxu (k3s), z obrazu zostaveného spolu so systémom, alebo **natívne** ako systémové služby. Konfigurácia Nextcloudu je jedna pre oba režimy, aby sa nerozišli.

## 2.4.4 Režim úložiska a požičiavanie procesora

Box si disk buď **necháva pre seba**, alebo **zdieľa** voľné miesto s meshom ako súčasť replikovaného poolu (Longhorn). Zdieľané dáta ležia v oddelenom účte shared v adresári šifrovanom vlastným kľúčom (fscrypt), ktorý sa načíta len počas zdieľania. Procesor box požičiava len v **okne** hodín, ktoré majiteľ určí, a len keď je box nečinný; okno sa vyhodnocuje v časovom pásme boxu, takže sa pri zmene času neposúva.



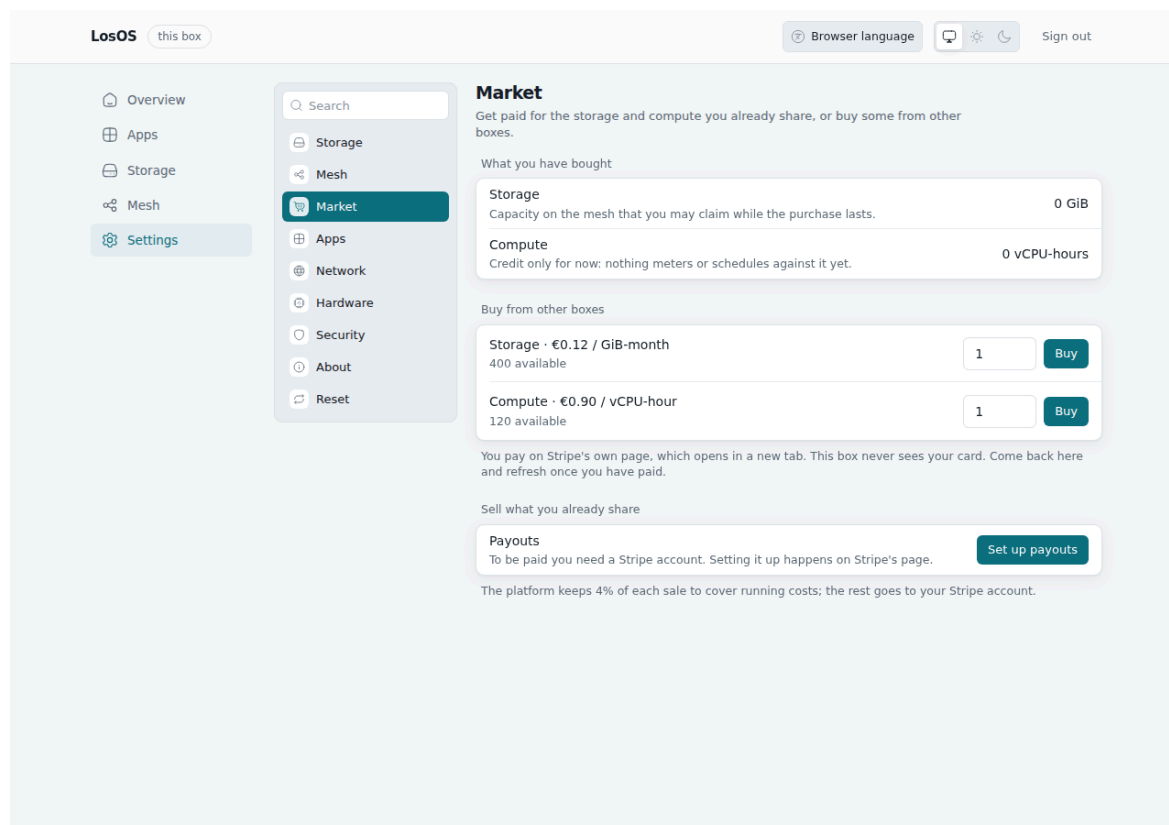
Obrázok 5: Panel Mesh s dvoma nájdеныmi edge servermi: firemný v lokálnej sieti (s výstrahou, že nie je oficiálny) a oficiálny cez internet.

## 2.5 Mesh a trh

**Mesh** je sieť boxov jedného edge. Úložisko: Longhorn replikuje zväzky naprieč boxmi, ktoré zdieľajú miesto; box zdieľa len to, čo sám nepoužíva, a vlastné súbory majiteľa do poolu nikdy nevstupujú. Výpočty: edge označí uzol mimo okna alebo pri zaťažení ako neplánovateľný, takže

cudzie úlohy bežia len v hodinách, ktoré majiteľ povolil. Box bez dosiahnuteľného edge zdieľanie odmietne, aby nesľúbil miesto poolu, ku ktorému sa nedostane.

**Trh** je voliteľná nadstavba: majiteľ predáva GiB-mesiaca úložiska alebo vCPU-hodinu výpočtu, ale iba to, čo už zdieľa. Platby idú cez Stripe Connect; platforma si necháva 4 % (400 základných bodov, strop 20 %) a peniaze sa nikdy nedotknú boxu. Kód trhu je hotový od konca po koniec a otestovaný proti náhradám Stripe; otvorí sa po založení podnikania, ktoré musí stať za platformou Stripe. Dovtedy je karta Trh v rozhraní sivá.



Obrázok 6: Panel Trh v stave, keď bude otvorený: nákup úložiska a výpočtu, nastavenie výplat.

## 2.6 Porovnanie s bežnými riešeniami

| Vlastnosť                            | Bežný NAS (Synology, QNAP) | Server s Nextcloudom (Ubuntu, Docker) | LosOS                                                      |
|--------------------------------------|----------------------------|---------------------------------------|------------------------------------------------------------|
| Správa                               | webové rozhranie výrobcu   | SSH, terminál, ručná údržba           | len webové rozhranie, žiadny shell                         |
| Stav systému                         | mení sa časom, drift       | mení sa časom, drift                  | pri každom štarte nanovo z jedného popisu                  |
| Aktualizácie                         | závislé od výrobcu         | ručné                                 | automatické každú noc, s návratom na predchádzajúci systém |
| Šifrovanie disku bez zadávania hesla | zriedka                    | ručne (LUKS + TPM)                    | štandardne, kľúč v TPM                                     |
| Hardvér                              | vlastný hardvér výrobcu    | ľubovoľný                             | ľubovoľný x86-64, aj vyradený                              |
| Zdieľanie kapacity s inými           | nie                        | nie                                   | mesh, voliteľne trh                                        |

| Vlastnosť          | Bežný NAS (Synology, QNAP) | Server s Nextcloudom (Ubuntu, Docker) | LosOS                                                         |
|--------------------|----------------------------|---------------------------------------|---------------------------------------------------------------|
| Reprodukovateľnosť | nie                        | čiastočne (Docker Compose)            | celý systém vrátane aplikácií je jeden zdroj, testovaný vo VM |

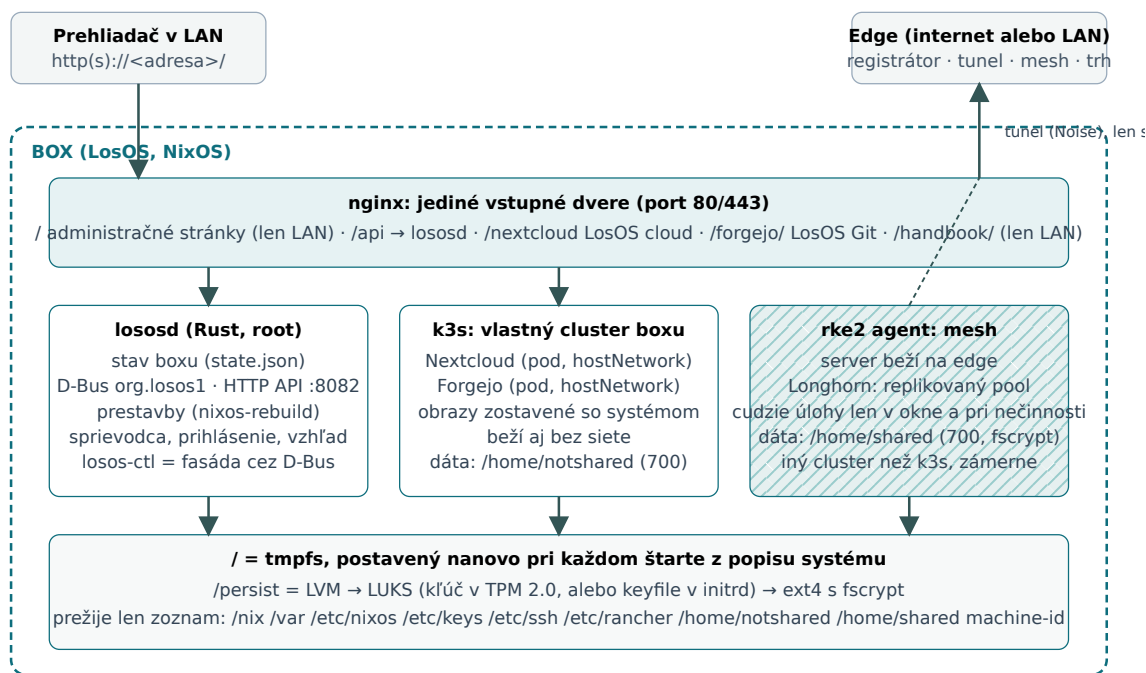
Cena za tieto vlastnosti je obmedzenie: LosOS nie je všeobecný server. Nedá sa naň doinštalovať ľubovoľný softvér inak než zmenou popisu systému, a to je zámer: box sa nedá dostať do stavu, ktorý nikto nezapísal.

## 2.7 Čo LosOS nie je

- **Nie je záloha.** Box je jedna kópia; ďalšiu si majiteľ drží inde (klienti Nextcloudu to uľahčujú).
- **Nie je hotový produkt.** Kapitola 10 hovorí, čo je hotové a čo plánované.

## 3 Architektúra systému

Táto kapitola dopĺňa prvý bod zadania o pohľad dovnútra: ako je systém navrhnutý a prečo. Je to zároveň návrh architektúry operačného systému pre mesh úložisko, ktorý bol súčasťou pôvodného zadania projektu.



Obrázok 7: Architektúra boxu: jedny vstupné dvere, tri bloky (riadiaci démon, vlastný cluster, mesh cluster) a jediné trvalé úložisko.

### 3.1 Šesť rozhodnutí, ktoré nesú návrh

1. **Bezstavový koreň.** Systémový disk je tmpfs postavený pri každom štarte z deklaratívneho popisu (NixOS). Prežije len vymenovaný zoznam adresárov na šifrovanom zväzku /persist, ktoré modul *impermanence* pripojí späť. Box preto nemá drift konfigurácie a nepotrebuje shell: reštart je oprava.
2. **Dva účty, dve dátové domény.** notshared (uid 1000) vlastní súbory majiteľa, shared (uid 1001) vlastní kópie pre mesh, v adresári šifrovanom vlastným kľúčom (fscrypt), ktorý sa načíta len počas zdieľania. Obe domovské zložky majú práva 700 a vlastnú skupinu, takže ani jedna nevie čítať druhú. Test tests/impermanence.nix to overuje, lebo prvý návrh (spoločná skupina users a práva 750) izoláciu potichu rušil.
3. **Dva clustre.** Vlastné aplikácie boxu bežia v jeho vlastnom Kubernetes clustri (k3s, bez sieťového pluginu, pody v sieti hostiteľa). Mesh je iný cluster (rke2; agent na boxe, server na edge). Agent sa bez svojho servera nespustí a box sa každú noc o 00:07 reštartuje, takže keby vlastné služby bežali v mesh clustri, výpadok edge cez polnoc by ich zhodil. Stavové adresáre oboch sa nekrížia (/var/lib/rancher/{k3s, rke2}).
4. **Jedny vstupné dvere.** Nginx drží jediné verejné porty a smeruje podľa cesty. Administračný povrch (/ , /api/, /setup/, /handbook/) je chránený podľa zdrojovej adresy (len LAN) a prísnou Content-Security-Policy; aplikácie sú dostupné odšadiaľ, kde je box dosiahnuteľný. Guard zámerne odmieta aj loopback, lebo tunel z edge prichádza z 127.0.0.1.

5. **Koreňový démon s typovaným API.** `lososd` (Rust) vlastní stav boxu (`/var/lib/losos/state.json`, jediný zapisovateľ, atomický zápis), vystavuje jednu metódu na operáciu na systémovej zbernici D-Bus (`org.losos1`) a JSON API na loopbacku (`127.0.0.1:8082`) s tokenom. Prestavby spúšťa ako prechodné `systemd` jednotky (`losos-rebuild-<job>`) a po vlastnom reštarte (ktorý `nixos-rebuild switch` spôsobí) sa k nim znovu pripojí. `losos-ctl` je tenká fasáda, ktorá volania posielá démonu.
6. **Všetko je prestavba.** Nastavenia, aktualizácie aj reset idú jednou cestou: zapísať popis, zostaviť systém, prepnúť. Režim zlyhania je jednotný (predchádzajúci systém beží ďalej) a nočná aktualizácia je ten istý kód, ktorý majiteľ spúšťa ručne tlačidlom Použiť.



#### Rovnaká cesta pre tri udalosti:

- majiteľ stlačí Použiť v administrácii (`lososd` spustí `losos-rebuild-<job>`)
- nočná aktualizácia o 03:00 (`system.autoUpgrade` zo zdroja `losos.upgradeFlakeUri`)
- obnovenie továrenského nastavenia (prázdny `overrides.nix`)

Obrázok 8: Životný cyklus systému: jeden popis, jedno zostavenie, jedno prepnutie, pre ručnú zmenu, nočnú aktualizáciu aj reset.

## 3.2 Dva systémy z jednej sady modulov

Flake exportuje dve konfigurácie NixOS, ktoré zdieľajú moduly v `modules/`:

| Výstup                                   | Čo to je                                                                         | Moduly                                                                                                            |
|------------------------------------------|----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| <code>nixosConfigurations.iso</code>     | živé inštalačné ISO, nesie rozloženie diskov (disko) a inštalátor                | <code>options</code> , <code>disko</code> , <code>installer</code> , <code>cache</code>                           |
| <code>nixosConfigurations.install</code> | nainštalovaný systém na disku                                                    | všetky: <code>boot</code> , <code>impermanence</code> , <code>hardening</code> , služby, démon, aktualizácie, ... |
| <code>nixosModules.edge</code>           | serverová strana (VPS alebo stroj v LAN): Traefik, tunel, registrátor, mesh, trh | samostatný modul                                                                                                  |

Všetky projektové voľby žijú pod `options.losos.*` (`modules/options.nix`) s predvolenými hodnotami v `defaults.nix`; moduly nikdy nečítajú holé `config.X`, vždy `config.losos.X`. Dva súbory sú špecifické pre každý box a zapisuje ich inštalátor alebo démon: `install-target.nix` (disky, firmvér, režim odomykania) a `overrides.nix` (voľby majiteľa).

## 3.3 Úložisko

Každý disk z `losos.targetDrives` dostane GPT a jeden fyzický zväzok LVM; všetky sa spoja do skupiny `persist-vg` s jedným logickým zväzkom `persist`, ktorý je LUKS2 a vnútri ext4 s funkciou `encrypt` (`fscrypt` ju potrebuje a `btrfs` ju neponúka). Prvý disk nesie ešte ESP (500 MiB) a v režime BIOS 1 MiB oddiel pre GRUB. Je to zrefazenie, nie RAID: kapacita sa sčíta, výpadok jedného disku zväzok zničí. Pre zariadenie, ktorého dáta mesh replikuje Longhornom, je to prijateľné a je to zdokumentované.

Kľúč zväzku je 4096 hexadecimálnych znakov (2048 náhodných bajtov) v `/etc/keys/persist-keyfile`, úmyselne bez NUL bajtov a bez konca riadka: disko ho formátovaniu odovzdáva cez

substitúciu príkazu, ktorá NUL bajty zahadzuje, kým `initrd` a `systemd-cryptenroll` čítajú súbor surový. Pôvodná verzia zapisovala surové bajty a prakticky každá inštalácia bez TPM sa formatovala iným kľúčom, než akým sa potom odomykala. Chyba vyšla najavo až v teste s emulovaným TPM (`tests/tpm.nix`).

### 3.4 Závaž a jednotný vzhľad

Nextcloud a Forgejo bežia ako statické pody v k3s s `hostNetwork`, takže odpovedajú na loopbacku a `nginx` ich vidí ako lokálne. Obrazy sa zostavujú spolu so systémom (`flake/images.nix`), takže aktualizácia prinesie otestovanú dvojicu. Konfigurácia Nextcloudu je jedna dátová štruktúra (`modules/nextcloud-stack.nix`) pre natívny aj kontajnerový režim.

Obe aplikácie dostanú tému postavenú na spoločnom `tokens.css`: Nextcloud priečinok témy `losos`, Forgejo tri CSS súbory témy a `DEFAULT_THEME`. Logá sa kreslia z jedného zdroja (`admin-ui/themes/brand/`) pri zostavení.

### 3.5 Edge

Edge je modul NixOS: Traefik s automatickým TLS vpredu, server tunela rathole za ním, registrátor `losos-registrar` (Rust), ktorý oba konfiguruje podľa boxov, ktoré sa registrujú a posielajú heartbeat, a voliteľne riadiaca rovina meshu (`rke2` server s Longhornom) a trh (Stripe Connect). Tunel používa šifrovanie Noise; box si verejný kľúč edge zapamätá pri prvom kontakte a zmenený odmietne. Kľúč Stripe drží oddelený proces `losos-stripe-gate`, ktorý s registrátorom hovorí cez unixový socket a nikdy mu kľúč neukáže. Pre prezentácie beží riadiaca časť registrátora aj ako jedna funkcia na Vercel s registrom v malej databáze (Neon Postgres); tunel, mesh a platby tam bežať nemôžu.

### 3.6 Bezpečnostný model v skratke

Hranice dôvery (plný text je v `docs/security-model.md`):

1. **Internet** → **edge**. Len edge je na internete; box počúva zvonku na ničom a tunel otvára smerom von.
2. **Edge** → **box**. Tunel je šifrovaný od konca po koniec; box pripne kľúč edge.
3. **LAN** → **box**. HTTPS s certifikátom, ktorý si box vydal sám a majiteľ ho raz nainštaluje (sprievodca, krok 1). Do tej chvíle by niekto na tej istej Wi-Fi mohol zachytiť prvé použitie.
4. **Súbory majiteľa** ↔ **kópia meshu**. Dva účty, oddelené skupiny, práva 700, fscrypt kľúč len počas zdieľania.
5. **Oficiálny edge** ↔ **ľubovoľný edge**. Verejný kľúč projektu v systéme; obchodovanie len s oficiálnym. Súkromnú polovicu kľúča vytvára a používa len nástroj na počítači autora projektu (`losos-registrar provision`), ktorý najprv prihlási osobu cez GitHub a odmietne každého, kto nie je na zozname povolených účtov zapísanom v repozitári; žiadny box ani edge kľúč nikdy nedrží.

Administratívny kľúč je mocný ako root; vzniká na boxe, box ho vydá raz v sprievodcovi a inak ho vydá len prehliadaču, ktorý preukáže heslo majiteľa. Desať neúspešných pokusov z jednej adresy spúšťa narastajúce oneskorenie; zmeny nastavení sa zapisujú do audit logu. Hardening je opísaný v kapitole 4.

Známe obmedzenia sú uvedené otvorene: jeden pôvod (origin) pre administráciu a obe aplikácie, krádež celého boxu s čipom TPM, audit log bez rotácie, škrtenie podľa adresy, a certifikát boxu, ktorý je zároveň koreňová autorita bez obmedzení (správny tvar je oddelená CA s obmedzením mena a krátky serverový certifikát; je to v pláne).

### 3.7 Prečo NixOS

Celý box vrátane aplikácií je jeden popis, ktorý sa dá zostaviť, otestovať vo virtuálnych strojoch a na boxe bez obsluhy prestavať. Repozitár nesie sedemnášť VM testov a sadu invariantov vyhodnocovaných pri zostavení, ktoré zlyhajú, keď niekto zmení nosnú predvolenú hodnotu. To je to, čo robí „bez shellu“ možným: box nikdy nie je v stave, ktorý nikto nezapísal.

## 4 Výber kernelu

Bod 2a zadania: *vyberte vhodný kernel pre daný operačný systém*. LosOS beží na nainštalovanom boxe s **najnovším stabilným jadrom Linuxu**, ktoré NixOS balí ako `linuxPackages_latest` (`modules/boot.nix`). Inštalčné médium a edge server ostávajú na predvolenom jadre NixOS (LTS), lebo požiadavky na ne sú iné. Táto kapitola zdôvodňuje tú voľbu proti trom alternatívam a opisuje, čo je na jadro navrstvené a čo voľba stojí.

### 4.1 Čo jadro na boxe musí zvládnuť

Kernel sa nevyberá abstraktne; vyberá sa pre konkrétnu triedu hardvéru a konkrétny spôsob prevádzky.

- **Hardvér:** vyradené kancelárske mini-PC (Intel NUC a jeho klony, Lenovo ThinkCentre Tiny, HP EliteDesk Mini, Dell OptiPlex Micro, novšie stroje s Intel N100/N305 alebo AMD Ryzen). Majú NVMe disky, 2,5 GbE radiče Realtek, moduly Wi-Fi 6/6E Intel, moderné riadenie napájania (s2idle, ASPM) a firmvérový TPM 2.0 (Intel PTT, AMD fTPM). Ovládače a opravy pre tento hardvér vznikajú v hlavnej vývojovej vetve (mainline) a do LTS vetiev sa dostávajú neskôr alebo vôbec.
- **Prevádzka:** box sa aktualizuje každú noc sám, bez obsluhy, a nemá shell, z ktorého by sa problém riešil ručne. Stabilita sa teda musí zabezpečiť inak než „nemeniť nič“: testami pred vydaním a návratom k predchádzajúcej generácii pri zlyhaní.
- **Záťaž:** dva Kubernetes runtime (k3s a rke2 agent), containerd, Longhorn (iSCSI, dm-crypt), fscrypt na ext4, TPM2 v initrd (systemd stage 1). Jadro teda musí mať zapnuté overlayfs, user namespaces, cgroups v2, nf\_conntrack, vxlan, iscsi\_tcp a dm\_crypt.
- **Bezpečnosť:** box je stále zapnutý, pripojený k sieti a požičiava procesor cudzím úlohám. Mitigácie hardvérových zraniteľností a obmedzenie útočnej plochy jadra sú dôležitejšie než na pracovnej stanici.

### 4.2 Kandidáti a rozhodnutie

| Kandidát                                                              | Čo ponúka                                                                   | Hodnotenie                                                                                                                                                                                                                                                                                                                                |
|-----------------------------------------------------------------------|-----------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Najnovšie stabilné jadro</b> ( <code>linuxPackages_latest</code> ) | nové vydanie približne každých 9 až 10 týždňov, najnovšie ovládače a opravy | <b>Zvolené.</b> Hardvérové problémy mini-PC sa opravujú najprv tu. Box sa aktualizuje sám, takže sledovanie najnovšieho jadra nestojí žiadnu ručnú prácu. Riziko regresie sa platí testami a návratom.                                                                                                                                    |
| <b>LTS jadro</b> (predvolené v NixOS)                                 | menej zmien, dva a viac rokov opráv                                         | Zaostáva v hardvéri. Stabilita, ktorú LTS kupuje („menej prekvapení na ručne aktualizovanom serveri“), je na boxe zabezpečená testami a nočným návratom; za LTS by sa platilo horšou podporou nového hardvéru. Zostáva na ISO a edge, kde hardvér nie je problém.                                                                         |
| <b>Hardened jadro</b> ( <code>linux_hardened</code> )                 | mitigácie zakompilované navyše                                              | V <code>nixpkgs</code> , ktoré flake sleduje, už <b>neexistuje</b> : <code>linux_hardened</code> je throw "removed due to lack of maintenance" a profil <code>hardened.nix</code> bol odstránený v NixOS 26.05. Jeho užitočná časť sa dá nastaviť na bežnom jadre (nižšie), čo je presne to, čo robí <code>modules/hardening.nix</code> . |
| <b>Real-time jadro</b> ( <code>PREEMPT_RT</code> )                    | deterministická latencia                                                    | Nič na boxe to nepotrebuje; stálo by priepustnosť.                                                                                                                                                                                                                                                                                        |

| Kandidát                                     | Čo ponúka     | Hodnotenie                                                                                                                                                           |
|----------------------------------------------|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Jadro iného systému (napr. Windows, FreeBSD) | iný ekosystém | Bezstavový systém postavený z textu pri každom štarte, fscrypt, TPM2 v initrd a Kubernetes runtime viažu projekt na Linux; na Windows sa takýto návrh postaviť nedá. |

Rozhodnutie je v kóde jedným riadkom a jeho zdôvodnenie v komentári nad ním (`modules/boot.nix`):

```
# Latest upstream kernel. The losos appliance runs on repurposed mini-PCs
# whose NVMe/Wi-Fi/sleep hardware quirks are fixed fastest in mainline, so
# track linuxPackages_latest rather than the nixpkgs default LTS. ...
boot.kernelPackages = pkgs.linuxPackages_latest;
```

Voľba je **ohraničená na nainštalovaný box**: `boot.nix` importuje len systém `install`. Inštalačné ISO potrebuje predovšetkým načítať `squashfs` a nestarať sa o nič iné, a `edge` beží na VPS s virtuálnym hardvérom, kde najnovšie ovládače nič neprinášajú.

### 4.3 Čo je na jadro navrstvené

Projekt nezabalil upstreamový profil (nijaký už nie je), ale skladá `hardening` z primitív odporúčaných projektom KSPP (Kernel Self Protection Project). Vrstva `losos.hardening.enable` (predvolene zapnutá) je tá, ktorá „nič nestojí“: nemôže box odpojiť od siete, zastaviť službu ani zabrániť naplánovaniu podu.

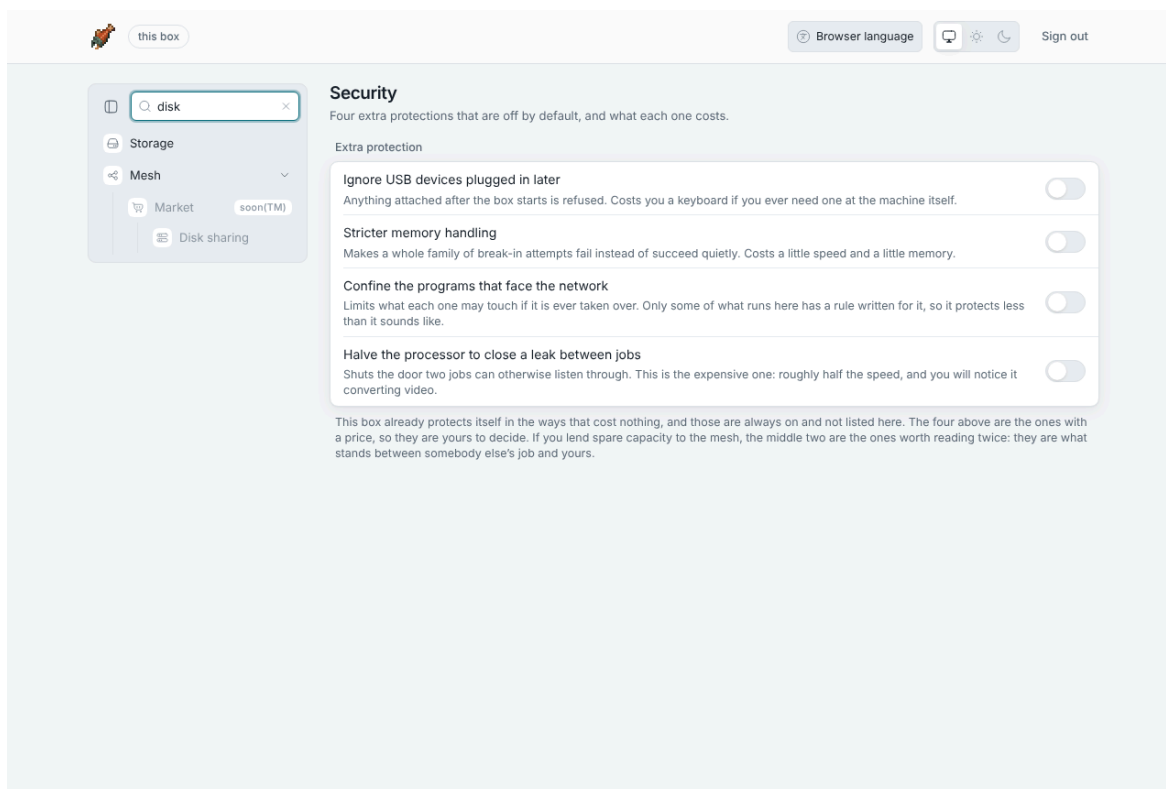
**Parametre jadra pri štarte** (`boot.kernelParams`): `slab_nomerge`, `init_on_alloc=1`, `init_on_free=1`, `page_alloc.shuffle=1`, `randomize_kstack_offset=on`, `vsyscall=none`, `debugfs=off`, vynútené mitigácie `spectre_v2=on`, `spec_store_bypass_disable=on`, `llt=flush`, `mds=full`, `tsx=off`, `tsx_async_abort=full`, `kvm.nx_huge_pages=force`, a IOMMU (`intel_iommu=on`, `amd_iommu=force_isolation`, `iommu.strict=1`, `efi=disable_early_pci_dma`) proti DMA útokom cez periférie.

**Sysctl**: skrytie ukazovateľov a logov jadra (`kernel.kptr_restrict=2`, `dmesg_restrict=1`), `perf_event_paranoid=3`, `yama_ptrace_scope=2`, `kernel.sysrq=0`, `unprivileged_bpf_disabled=1`, `randomize_va_space=2`, vypnuté `core dumpy`, `fs.protected_*`, `vm.unprivileged_userfaultfd=0` a sprísnený sieťový stack (SYN cookies, žiadne ICMP redirecty a zdrojové smerovanie, voľné `rp_filter=2`).

**Blacklist modulov**, ktorý blokuje aj ručný `modprobe` (samotné `boot.blacklistedKernelModules` bráni len automatickému načítaniu): staré a zriedkavé sieťové protokoly (`dccp`, `sctp`, `rds`, `tipc`, ...), ovládače súborových systémov, ktoré parsujú cudzie obrazy, a cesty s DMA (`FireWire`, `Thunderbolt`). Nič z toho, čo potrebuje container runtime, CNI alebo Longhorn, na zozname nie je.

**Ďalšie**: `tmpfs /tmp`, `noexec` na `/dev/shm`, `security.protectKernelImage` (zakázaný `kexec` a hibernácia), `dbus-broker`, a `systemd sandboxing` služieb, ktoré čelia sieti (`nginx`, `avahi-daemon`, `lososd: ProtectHome`, `ProtectKernelModules`, `ProtectKernelTunables`, ...).

**Štyri voliteľné prepínače**, ktoré môžu niečo pokaziť, má majiteľ na paneli Zabezpečenie: `AppArmor`, `hardened_malloc` (`GrapheneOS`), vypnutie SMT (polovica jadier) a `USBGuard` (zariadenia pripojené po štarte sa odmietnu).



Obrázok 9: Panel Zabezpečenie: štyri voliteľné ochrany s cenou, ktorú každá má.

## 4.4 Tri odporúčania, ktoré sú zámerne vypnuté

Publikované bezpečnostné zoznamy sú písané pre stroje bez container runtime. Tri ich položky by box rozbili, a preto sú vypnuté a test `tests/hardening.nix` zlyhá, ak ich niekto zapne:

| Nastavenie                               | Prečo nie                                                                                              |
|------------------------------------------|--------------------------------------------------------------------------------------------------------|
| prísny <code>rp_filter=1</code>          | zahadzuje multicast odpovede mDNS, jediný spôsob, ako nájsť box bez shellu; a nefunguje pod ním Calico |
| <code>user.max_user_namespaces=0</code>  | zastaví oba kubelety aj containerd                                                                     |
| <code>noexec</code> na <code>/tmp</code> | tam bežia zostavenia Nix a nočná prestavba je jediná cesta samoopravy                                  |

Pre rovnaký dôvod musia ostať načítateľné `overlay`, `br_netfilter`, `veth`, `vxlan`, rodina `nf_conntrack`, `iscsi_tcp` a `dm_crypt`, a `net.ipv4.ip_forward` ostáva zapisovateľné za behu, lebo `k3s` a `rke2` si ho nastavujú samy.

## 4.5 Čo sledovanie najnovšieho jadra stojí a ako sa to platí

- **Rast boot oddielu.** Takmer každá nočná aktualizácia prinesie nové jadro a `initrd` do 500 MiB ESP. Boot menu je obmedzené na päť generácií (`configurationLimit = 5` pre `systemd-boot` aj `GRUB`) a `nix.gc` o 04:30 maže verzie staršie ako 14 dní; `tests/invariants.nix` oboje pripína, lebo bez nich sa box za pár mesiacov zaplní a prestavba o 03:00 zlyhá na kroku bootloadera. Toto bola skutočná medzera nájdená recenziou (kapitola 7).
- **Jadro, ktoré niečo pokazí.** Predchádzajúci systém ostáva v boot menu a ďalšiu noc sa prestavba skúsi znova. VM testy bootujú nainštalovaný systém, inštalátor, odomýkanie z TPM, rast disku a `hardening` pri každej zmene týchto modulov.

- **Kľúč v TPM nie je viazaný na merania (PCR).** Viazanie na merania by po každej aktualizácii jadra alebo firmvéru box zamklo a nikto by nemal kde zadať obnovovací kľúč. Dôsledok (zlodej s celým boxom ho vie nabootovať) je priznaný v bezpečnostnom modeli.
- **Rýchlosť aktualizácie.** Jadro sa sťahuje z verejnej cache NixOS a z vlastnej cache projektu (`proxy.losos.dasmat.us`), takže sa na boxe nekompiluje.

## 4.6 Zhrnutie

Najnovšie stabilné jadro je pre túto triedu hardvéru a tento spôsob prevádzky správna voľba: podpora hardvéru je najlepšia, cena (viac zmien) je zaplatená mechanizmami, ktoré box aj tak potrebuje (testy, generácie, GC), a bezpečnosť, ktorú by prinieslo hardened jadro, je navrstvená ako konfigurácia s testom, ktorý chráni každé rozhodnutie vrátane tých „vypnutých“.

## 5 Inštalácia

Bod 2b zadania: *nainštalujte operačný systém*. Inštalácia je navrhnutá pre človeka, ktorý nikdy nevidel terminál: jedna otázka a potom čakanie. Inštalátor **zmaže všetky pevné disky** v stroji.

### 5.1 Inštalačné médiá

| Médium                   | Veľkosť | Použitie                                                                                  |
|--------------------------|---------|-------------------------------------------------------------------------------------------|
| Inštalačné ISO           | malé    | bežná cesta; priložené ku každému vydaniu na GitHubu. Cieľ si systém stiahne.             |
| ISO s celým systémom     | veľké   | nesie zostavený systém, inštalácia kopíruje namiesto sťahovania. Zostavuje sa ručne.      |
| Demo obraz disku (QCOW2) | veľké   | predinštalovaný virtuálny disk pre QEMU alebo virt-manager, bez šifrovania a inštalátora. |

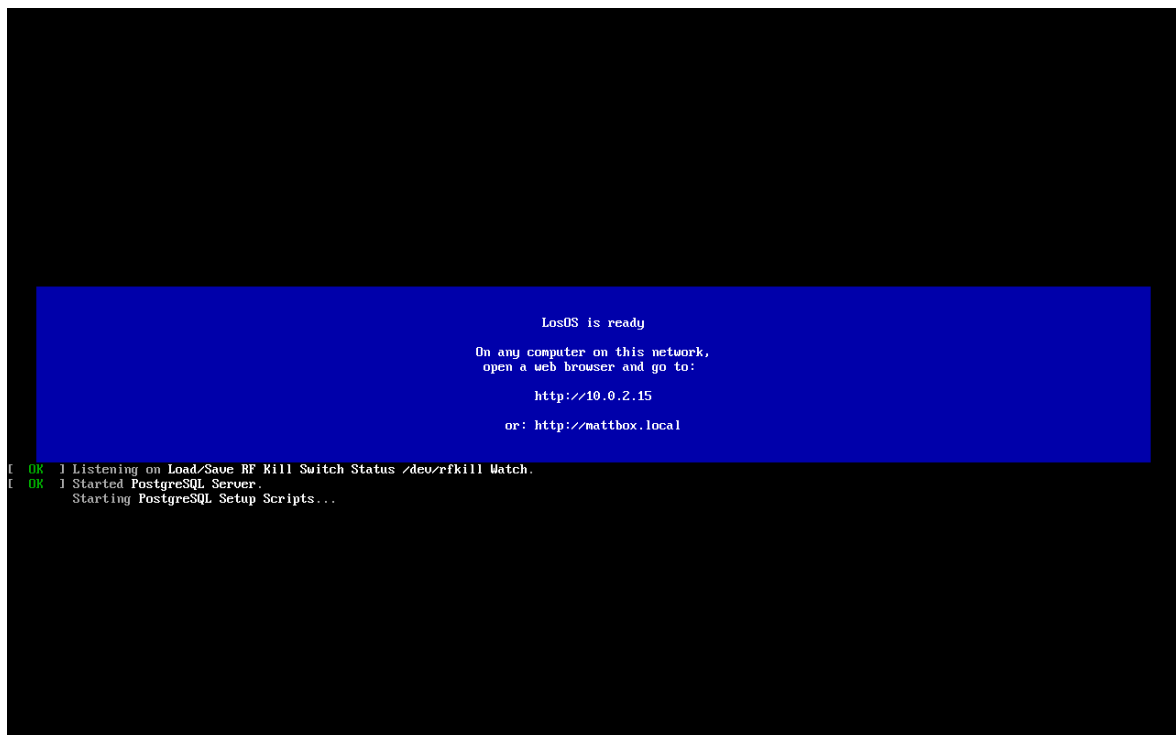
Inštalačné ISO sa zostavuje v CI pri každej zmene a CI ho aj **bootuje** (pod OVMF aj SeaBIOS); test prejde, keď inštalátor pošle DNS dotaz na `github.com`, teda keď došiel až po klonovanie zdroja. Každá zmena je tak overená skutočným štartom média.

### 5.2 Požiadavky

- Procesor **x86-64** (Intel alebo AMD), aspoň 4 GiB pamäte a disk 40 GiB alebo väčší. Väčšina kancelárskych mini-PC má firmvérový TPM 2.0. Flake zostavuje výlučne pre `x86_64-linux`: na počítači s procesorom ARM, teda ani na Macu s čipom Apple M, sa LosOS nainštalovať nedá; aj ukážka vo virtuálnom stroji na takom hostiteľovi beží len s emuláciou x86-64, ktorá je rádovo pomalšia. Inak platí, že čokoľvek s x86-64 a dostatkom pamäte a disku systém pohodlne utiahne, aj vyradený kancelársky stroj.
- USB kľúč 2 GiB alebo väčší.
- Káblové pripojenie k sieti s prístupom na internet: inštalátor sťahuje systém, ktorý inštaluje (z cache, nie kompilovaním).
- Obrazovka a klávesnica len na inštaláciu. Potom obrazovka ukazuje len banner s adresou boxu.
- **Secure Boot vypnutý**. Nič v LosOS nie je podpísané; so zapnutým Secure Boot firmvér kľúč odmietne, niekedy potichu.

### 5.3 Priebeh

1. Stiahnuť ISO a jeho `.sha256` z posledného vydania, zapísať na USB kľúč (balenaEtcher, Rufus v režime DD, alebo dd).
2. Nabootovať z kľúča. Menu ponúkne **UEFI**, **BIOS** alebo **autodetekciu** a po 30 sekundách bez odpovede zvolí autodetekciu, takže aj bezobslužný štart nainštaluje. Autodetekcia zvolí režim, v ktorom bol kľúč sám nabootovaný (nepítomnosť `/sys/firmware/efi` znamená BIOS).
3. Čakať. Inštalátor nájde všetky pevné disky, spojí ich do jedného šifrovaného zväzku, stiahne systém a nainštaluje ho. Ku koncu vypíše `unlock: TPM alebo unlock: keyfile in the initrd`.
4. Vybrať kľúč a nechať stroj reštartovať. Obrazovka ukáže modrý banner s adresou boxu; tú treba otvoriť v prehliadači na ľubovoľnom počítači v tej istej sieti a pokračovať prvým spustením (kapitola 6).



Obrázok 10: Banner na tty1 po prvom štarte: IP adresa a názov boxu, nič iné na obrazovke nie je.

## 5.4 Čo inštalátor robí

Inštalátor je podprikaz `losos-ctl install` (Rust, backend/src/installer.rs), zabalený ako `losos-install`, ktorý sa na ISO spúšťa automaticky ako shell na tty1. Jeho plán je **dáta**: funkcia `plan_install` vráti zoznam krokov a testy overujú poradie deštruktívnych krokov bez toho, aby čokoľvek formátovali.

1. **Detekcia diskov.** Všetky pevné disky okrem inštalačného média sa stanú `losos.targetDrives`.
2. **Zápis `install-target.nix`.** Disky, režim firmvéru (`losos.bios`) a režim odomykania (`losos.tpm.enable`) sa zapíšu do súboru, ktorý bude súčasťou popisu nainštalovaného boxu. Vďaka tomu neskoršia aktualizácia z verejného zdroja (github:) nezabudne, aké disky a firmvér box má; `flake.nix` uprednostní živú kópiu v `/etc/nixos`, a preto obe cesty prestavby bežia s `--impure` (invariant to pripína).
3. **Kľúč.** `ensure_keyfile` zapíše 4096 hexadecimálnych znakov do `/etc/keys/persist-keyfile`.
4. **disko.** Rozloženie z `modules/disko.nix`: GPT, ESP 500 MiB (a 1 MiB oddiel pre GRUB v režime BIOS), LVM fyzické zväzky → `persist-vg` → LV `persist` → LUKS2 → ext4 s encrypt. Zväzok sa zámerne nenafúkne na celú skupinu: `losos.storage.fillPercent` necháva rezervu na rast za behu.
5. **TPM.** Ak je `/dev/tpmrm0` (alebo `--tpm`), hneď po formátovaní sa spustí `systemd-cryptenroll --tpm2-device=auto --tpm2-pcrs= --unlock-key-file=...`; kľúč sa zapečatí do čipu bez viazania na PCR a `initrd` nenesie žiadne tajomstvo. Bez čipu (alebo `--no-tpm`) sa kľúč vloží do `initrd` ako `/crypto_keyfile.bin`. Obe podoby `crypttab` sú výlučné: `systemd-cryptsetup` s kľúčovým súborom *aj* `tpm2-device=` by čítal súbor ako zapečatený blob.
6. **nixos-install** systému `install` zo zdroja `flake`, so zapísaným `install-target.nix`.
7. **Reštart.**

```
=== losos-install returned: proof of the software TPM (take 5) ===
Software TPM seen by the guest?
cpu----- 1 root root 10, 224 Oct 5 13:54 /dev/tpm0
cpu----- 1 root root 252, 65536 Oct 5 13:54 /dev/tpmrm0
TPM major version: 2
=== LUKS2 header of /dev/persist-ug/persist: ===
Keyslots:
 0: luks2
 1: luks2
Tokens:
 0: systemd-tpm2
    Keyslot: 1
=== keyfile stays on the encrypted volume only: ===
total 4
-rw----- 1 root root 4096 Oct 5 14:07 persist-keyfile
background.png
converted-font.pf2
grub
=== owners of the persisted sources (created by id, no chown-by-name): ===
700 1000:1000 /mnt/persist/home/notshared
700 1001:1001 /mnt/persist/home/shared
755 0:0 /mnt/persist/etc/rancher
755 0:0 /mnt/persist/var
[root@nixos:~]#
```

Obrázok 11: Dôkaz z inštalácie vo virtuálnom stroji s emulovaným TPM: LUKS hlavička má token `systemd-tpm2` v slot 1 a kľúčový súbor ostal len na šifrovanom zväzku. Vlastníci trvalých adresárov sú nastavení podľa uid.

## 5.5 Inštalácia vo virtuálnom stroji (ukážka)

Ukážka na obhajobe beží vo virtuálnom stroji (dohodnuté s konzultantom). Dve veci sú iné než na hardvéri:

- **TPM** musí byť emulovaný (`swtpm`) pri inštalácii aj pri každom ďalšom štarte, s rovnakým stavovým adresárom; inak sa box nainštaluje v režime kľúčového súboru, čo tiež funguje.
- Hostiteľ virtuálneho stroja spravidla **nerozlišuje .local meno** boxu (`libvirt`, `VirtualBox NAT`), takže sa box otvára cez IP adresu z banneru.

```
mkdir -p tpm
swtpm socket --tpmstate dir=tpm --ctrl type=unixio,path=tpm/sock --tpm2 --
daemon
qemu-img create -f raw disk.img 40G
qemu-system-x86_64 -m 4096 -smp 2 -enable-kvm -machine q35 \
  -drive file=losos.iso,media=cdrom,readonly=on \
  -drive file=disk.img,format=raw,if=virtio \
  -chardev socket,id=chrtpm,path=tpm/sock -tpmdev
emulator,id=tpm0,chardev=chrtpm \
  -device tpm-tis,tpmdev=tpm0 \
  -nic user,hostfwd=tcp:127.0.0.1:80-:80
```

`-enable-kvm` predpokladá hostiteľa s procesorom x86-64 a zapnutou virtualizáciou. Na hostiteľovi ARM (Mac s čipom Apple M) treba prepínač vynechať a QEMU emuluje x86-64 softvérovo (TCG); inštalácia potom trvá desiatky minút namiesto jednotiek a hodí sa len na overenie, nie na predvádzanie.

Bez `-bios` bootuje QEMU SeaBIOS, takže inštalátor zvolí GRUB; druhý štart ide bez riadku s `media=cdrom`. Záznamy ukážok (inštalácia od ISO po prvé prihlásenie, pri emulácii bez KVM asi 45 až 50 minút; so KVM podstatne rýchlejšie) sú súčasťou materiálov k obhajobe.

Čo sa pri ukážkach vo VM našlo a opravilo, je v kapitole 7: chýbajúce virtio ovládače v `initrd` (box po inštalácii nenabootoval v QEMU), kľúčový súbor s NUL bajtmi (box stál na výzve na heslo) a `/run/nextcloud` vytváraný bez práv v obraze (Nextcloud nikdy nenaštartoval v režime záťaže). Každá z nich je dnes pokrytá testom alebo zmenou obrazu.

## 5.6 Po inštalácii

Box po prvom štarte: odomkne `/persist` z TPM, postaví `tmpfs` koreň a pripojí späť trvalé adresáre, spustí `lososd` (ktorý si pri prvom štarte vyrazí administračný kľúč), `k3s` so statickými podmi Nextcloudu a Forgejo, `nginx` a banner na `tty1`. Administračné stránky odpovedajú zhruba do dvoch minút; Nextcloud dokončí svoj prvý štart o pár minút neskôr (sprievodca na to čaká a ukazuje dôvod).

## 6 Základná konfigurácia

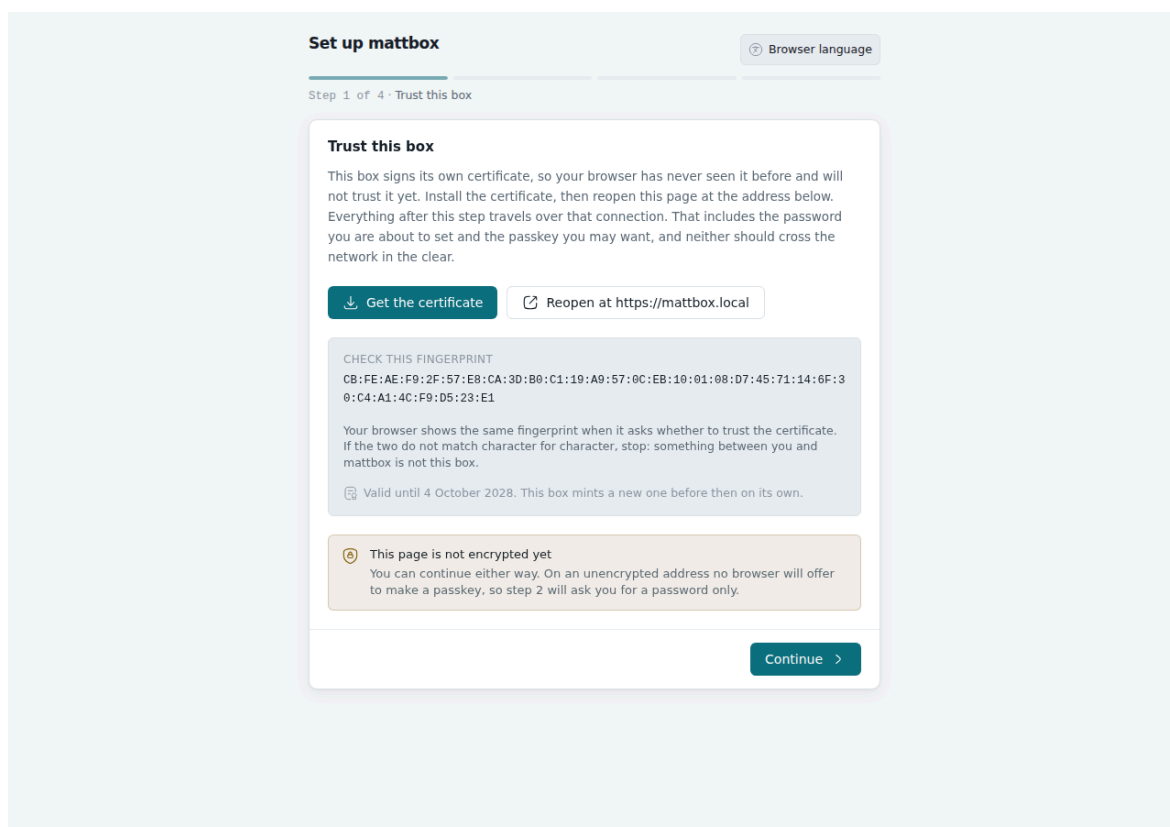
Bod 2c zadania: *vykonajte základnú konfiguráciu operačného systému*. Na boxe LosOS sa konfiguruje dvoma spôsobmi: pri **prvom spustení** cez sprievodcu a potom priebežne cez **administračné stránky**. Oba zapisujú do toho istého popisu systému; terminál sa nepoužíva nikdy.

### 6.1 Prvé spustenie: sprievodca

Pri prvom otvorení adresy boxu sa zobrazí trojkrokový sprievodca. Treba ho prejsť z počítača vo vlastnej sieti krátko po inštalácii: kým nie je dokončený, box patrí tomu, kto ho dokončí ako prvý (tzv. *claim*). Zvonku siete box nikto zabrať nemôže a bez preinštalovania ho zabrať znova nejde.

#### 6.1.1 Krok 1: Dôverovať tomuto boxu

Box obsluhuje HTTPS s certifikátom, ktorý si vydal sám. Sprievodca ukáže jeden riadok pre terminál počítača, na ktorom sa stránka otvorila (`curl -fsSL http://<adresa>/setup/trust.sh | sh` na macOS a Linuxe, `irm http://<adresa>/setup/trust.ps1 | iex` na Windows), ktorý pridá **jeden certifikát** do úložísk, ktoré prehliadače čítajú, len pre aktuálneho používateľa a bez práv správcu. Skript pochádza z boxu v lokálnej sieti, dá sa prečítať vopred a vypíše odtlačok certifikátu na porovnanie so stránkou. Telefóny dostanú súbor `losos-ca.crt` na ručné pridanie. Krok sa dá preskočiť; bez neho nebude k dispozícii passkey.



Obrázok 12: Krok 1 sprievodcu: odtlačok certifikátu na porovnanie a upozornenie, že stránka ešte nie je šifrovaná.

#### 6.1.2 Krok 2: Prihlasovanie

Majiteľ zvolí **heslo**: aspoň 12 znakov, veľké aj malé písmeno, číslica a symbol; sprievodca odškrtnáva pravidlá počas písania. Krok počká, kým LosOS cloud dokončí svoj prvý štart (na čerstvom boxe pár minút), a potom heslo nastaví. Prihlasovacie meno je `notshared`. Po nastavení sa raz zobrazí

**náhradný administračný kľúč s tlačidlami Kopírovať a Tlačiť. Na HTTPS spojení sa dá pridať passkey.**

Set up losos Browser language

Step 2 of 3 - Choose how you sign in

**Choose how you sign in**

This box made itself a random password when it was installed and showed it to nobody, which is why nothing can sign in yet. Choose one now. It is the one password for everything here: LosOS cloud, the apps on your phone and computer, and these admin pages.

**New password**

Four rules, and these are all of them. A few unrelated words joined with a dash, with a capital letter and a number somewhere, is the easiest way to meet them.

- ☐ At least 12 characters
- ☐ A capital letter and a small letter
- ☐ A digit
- ☐ A symbol, such as - or !

**Type it again**

Nothing on this box can tell you what you typed, so a slip here means starting the box over.

[Set a different password](#)

**Password set**

YOUR SIGN-IN NAME  
**not shared**

This is the name to type on the sign-in page at the end of the wizard, and in the desktop and phone apps.

**Your spare admin key, shown only now**

From now on your password unlocks these admin pages. This key is the spare: it opens them even while LosOS cloud is not running to check the password. The box shows it only now and keeps no copy a browser can ask for again, so copy it into a password manager, or print it and keep the sheet.

SPARE ADMIN KEY  
**7f3a9c1e5b2d4f6a8c9e1b3d5f7a9c2e4b6d8f0a1c3e5b7d9f1a3c5e7b9d2f4a**

[Copy the key](#) [Print](#)

**Add a passkey as well**

A passkey signs you in with your face, your fingerprint or your screen lock, from this browser on this device. It is an addition, not a replacement: the desktop and phone apps still sign in with the password you just set.

[Create passkey](#)

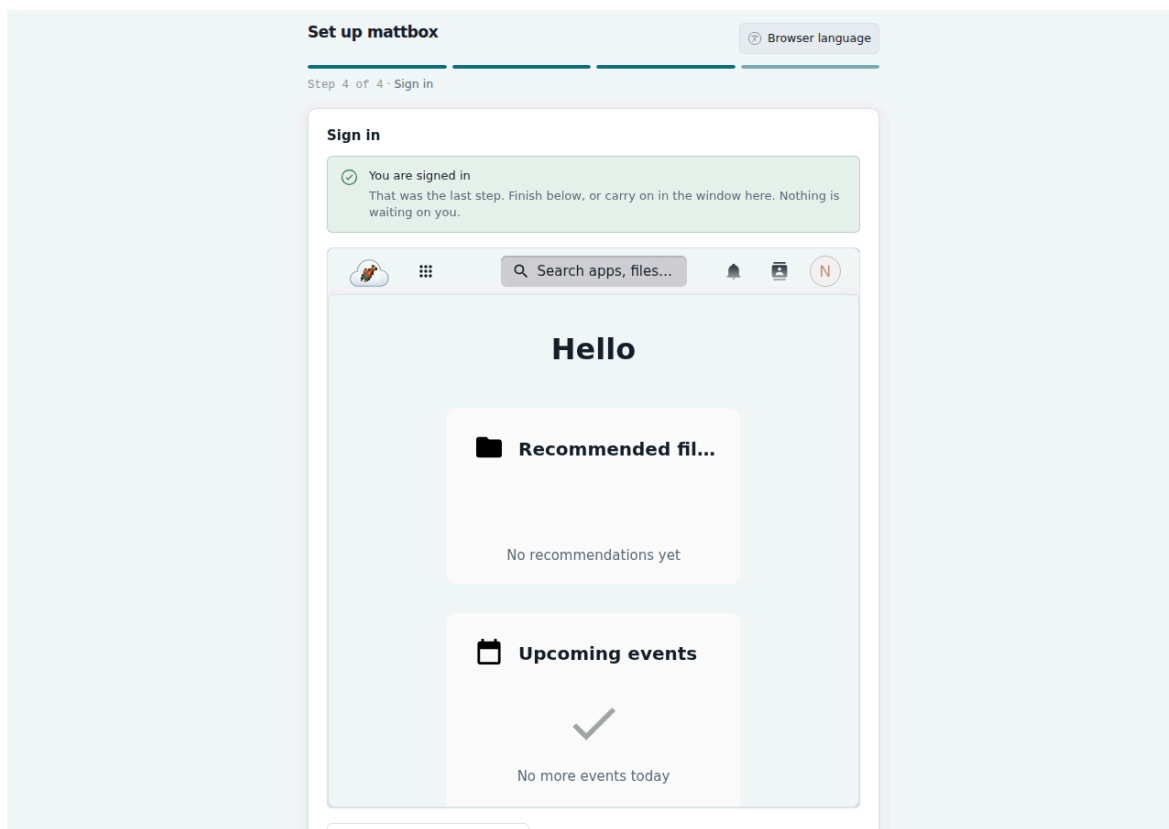
This device has no built-in fingerprint or face unlock, so your browser will ask for a security key.

[< Back](#) [Continue >](#)

Obrázok 13: Krok 2 sprievodcu po nastavení hesla: prihlasovacie meno a jediné zobrazenie náhradného kľúča.

### 6.1.3 Krok 3: Prihlásenie

Sprievodca otvorí LosOS cloud vo vlastnej stránke, majiteľ sa prihlási práve nastaveným heslom a sprievodca ho prevedie na prehľad administračných stránok.



Obrázok 14: Posledný krok sprievodcu: prihlásenie do LosOS cloudu priamo na stránke sprievodcu.

## 6.2 Administračné stránky: panely a Použiť

Box nemá nastavenia v bežnom zmysle; má **popis seba samého** v Nixe a panely nastavení upravujú jeho malú časť. Pri úprave poľa sa dole zobrazí lišta *N zmien ešte nepoužitých* s tlačidlami Zahodiť a Použiť. Pole, ktoré neprejde kontrolou (názov s medzerou, port mimo rozsahu), je označené a Použiť čaká. Nič sa boxu nedotkne, kým sa Použiť nestlačí.

Čo Použiť robí:

1. Démon zapíše voľby do `modules/overrides.nix` na `/etc/nixos` (jediný zdroj pravdy pre voľby majiteľa; `backend/src/overrides.rs`).
2. Spustí prestavbu ako úlohu na pozadí (`losos-rebuild-<job>`); widget **Zmeny** na Prehľade ukazuje, či každá prešla.
3. Nový systém sa prepne. Niektoré zmeny reštartujú samotný démon uprostred; ten úlohu po štarte znovu prevezme (`startSupervisor`).
4. Oznámenie povie *Zmeny použité* alebo *Zmeny sa nepodarilo použiť* s dôvodom.

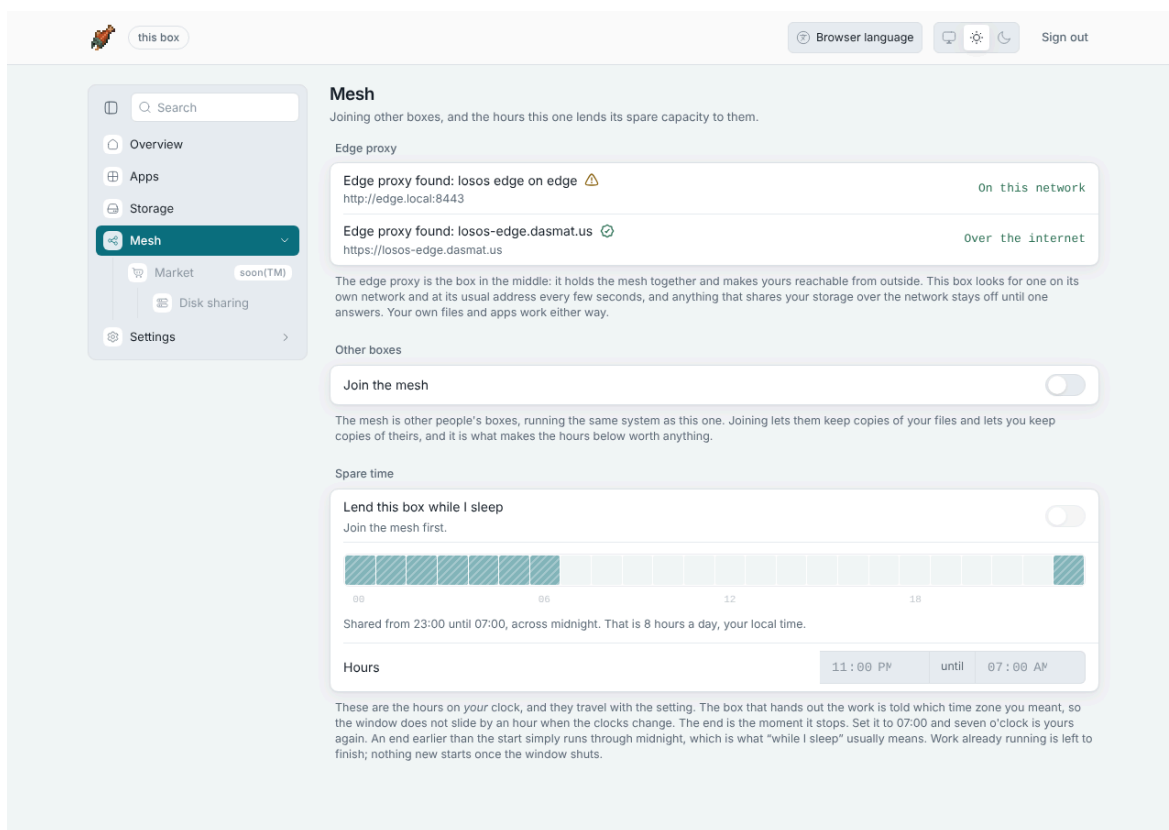
Zmena buď prejde celá, alebo vôbec: zlyhaná prestavba nechá bežať predchádzajúce nastavenia. Prestavba trvá niekoľko minút; box je medzitým dostupný.

### 6.2.1 Čo sa dá nastaviť

| Panel        | Nastavenia                                                                                                                 |
|--------------|----------------------------------------------------------------------------------------------------------------------------|
| Sieť         | názov boxu ( <code>&lt;názov&gt;.local</code> ), šifrované spojenie (HTTPS), dostupnosť zvonku (tunel k edge), adresa edge |
| Hardvér      | povoliť aplikáciám grafický čip                                                                                            |
| Zabezpečenie | štyri voliteľné ochrany: USBGuard, hardened_malloc, AppArmor, vypnutie SMT                                                 |

| Panel       | Nastavenia                                                                         |
|-------------|------------------------------------------------------------------------------------|
| Vzhľad      | pozadie, závoj, ručne písané widgety (bez prestavby)                               |
| O zariadení | verzia, časovače, odkaz na príručku, odznak režimu úložiska                        |
| Reset       | obnovenie továrenského nastavenia (prázdny <code>overrides.nix</code> a prestavba) |
| Aplikácie   | LosOS Git zapnuté či vypnuté, režim aplikácií, vyhľadávanie v katalógu Nextcloudu  |
| Úložisko    | zaplnenie disku, použitie rezervy ( <code>losos-ctl grow</code> )                  |
| Mesh        | pripojiť sa k meshu, požičovať procesor, okno hodín (v časovom pásme boxu)         |
| Trh         | zdieľanie disku a obchodovanie; zatiaľ sivé                                        |

Tabuľka zobrazení na voľby `losos.*` je v príručke (*Settings to options*). Všetky voľby žijú pod `options.losos.*`, takže to, čo panely neponúkajú, sa dá nastaviť úpravou popisu systému (napríklad zdroj aktualizácií `losos.upgradeFlakeUri`).



Obrázok 15: Panel Mesh: edge nájdený v sieti aj cez internet, prepínače meshu a okno hodín.

Prepínače meshu a zdieľania disku sú **len na zapnutie s edge v dosahu** (*enable-only* brána): kým box edge nenašiel (`GET /api/edge` hlási *none*), sú sivé s dôvodom a priame volanie API odpovie 409. Zapnuté zdieľanie sa pri strate edge nevypína samo, box ho však znovu ponúkne až po jeho návrate. Kapitola 7 to overuje ako scenár A a B.

## 6.3 Dve veci, ktoré nie sú súčasťou popisu

**Vzhľad** (pozadie, závoj, ručne písané widgety) drží démon ako dokument v `/var` a platí okamžite pre každý prehliadač. **Nástěnka** widgetov je uložená v prehliadači, v ktorom ju majiteľ poskladal. Obe prežijú reštart, ale ani jedna nepotrebuje prestavbu.

## 6.4 Konfigurácia na úrovni systému

Pre úplnosť, konfigurácia, ktorú inštalátor a predvolené hodnoty vykonajú bez zásahu majiteľa, a ktorá sa dá zmeniť len úpravou popisu:

- **Používatelia:** notshared (uid 1000) a shared (uid 1001), bez hesla, s vlastnými skupinami a domovmi 700; žiadne iné interaktívne účty, `services.openssh.enable = false`.
- **Sieť:** DHCP, mDNS/DNS-SD cez Avahi (box sa hlási ako `<názov>.local`), nginx na portoch 80 a 443, časové pásmo a locale v `configuration.nix`.
- **Trvalé adresáre:** zoznam v `modules/impermanence.nix`; čokoľvek nové, čo má prežiť reštart, musí pribudnúť tam.
- **Aktualizácie:** `system.autoUpgrade` o 03:00 zo zdroja `losos.upgradeFlakeUri` (predvolene vlastná kópia popisu `git+file:///etc/nixos#install`, ktorá softvér nemení; verejný zdroj `github:dasmatus/losos#install` ho mení). Fragment `#install` je nosný: bez neho `nixos-rebuild` hľadá konfiguráciu podľa názvu hostiteľa, ktorú flake neexportuje, a nočná aktualizácia potichu zlyháva.
- **Reštart** o 00:07 s `Persistent=true`, **GC** o 04:30.
- **Cache:** box sťahuje zostavené cesty z `cache.nixos.org` a z vlastnej cache projektu (`losos.cache.*`).

## 6.5 Príklad: zapnutie vzdialeného prístupu

1. Prevádzkovateľ edge zapíše box do zoznamu povolených (`losos.edge.tenants.<id>` s tokenom).
2. Majiteľ na paneli Sieť zapne *Dostupný zvonku* a (ak edge nie je v tej istej sieti) zadá jeho adresu; Použiť.
3. Box sa zaregistruje na registrátore, stiahne verejný kľúč tunela a pripne ho, otvorí tunel smerom von. Edge vytvorí trasu a certifikát pre `https://<názov>.<doména edge>`.
4. Administratívne stránky ostávajú aj cez tunel len pre LAN: tunel prichádza z loopbacku a guard loopback odmieta.

## 7 Overenie správnosti konfigurácie a funkčnosti

Bod 2d zadania: *overte správnosť konfigurácie a funkčnosť operačného systému*. Overenie má v projekte jednu hlavnú skúšku a tri vrstvy. Hlavná skúška je mesh v dvoch scenároch: **bez edge proxy** (box zdieľanie odmietne a slúži lokálne) a **s edge proxy** (dva boxy v jednej sieti so spoločným úložiskom), lebo správnosť konfigurácie tohto systému znamená, že mesh funguje práve vtedy, keď má. Vrstvy sú automatizované testy, ktoré bežia pri každej zmene; ručný kontrolný zoznam vo virtuálnom stroji, ktorý je zároveň scenárom ukážky; a nezávislá kritická recenzia, ktorej nálezy sa opravili a pokryli testami.

### 7.1 Zásada: testy kódujú argumenty, nie len správanie

Box nemá shell, takže chyba, ktorá sa prejaví až na nainštalovanom stroji, nemá kde byť opravená ručne. Testy preto nechránia len to, že niečo funguje, ale aj *prečo* je niečo nastavené práve tak: `tests/hardening.nix` overuje, čo je zámerne vypnuté; `tests/resize.nix` overuje poradie `lvextend` → `cryptsetup` `resize` → `resize2fs`; `tests/front-vhost.nix` overuje, že loopback dostane 403, a vysvetľuje, prečo by 200 bola chyba; `tests/install.nix` overuje funkciu `encrypt` na `ext4`, nie len typ súborového systému. Kto neskôr rozhodnutie zmení, dostane červený test namiesto rozbitého boxu v skrini.

### 7.2 Vrstva 1: automatizované testy

#### 7.2.1 Jednotkové testy (Rust)

Vrstva príkazov démona je napísaná proti rozhraniu `Losos` s reálnou (`io_backend`) a pamäťovou (`fake`) implementáciou, takže stavový automat sa testuje bez súborového systému. Plánovače (inštalácia, rast disku, obnova, trh) vracajú plán ako dáta a testy overujú poradie deštruktívnych krokov bez formátovania. Spolu vyše 200 testov v `truch crate` (`backend`, `backend-registrar`, `edge-vercel`); bežia cez `cargo test` v CI aj lokálne (`devenv test`). Oba `crate` musia byť čisté pre `clippy` -D warnings a `rustfmt`.

#### 7.2.2 Testy vo virtuálnych strojoch (NixOS)

Sedemnášť testov v `tests/` bootuje skutočné systémy z tých istých modulov, z ktorých sa inštaluje. Prehľad:

| Test                          | Čo overuje                                                                                                                                                                            |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>install.nix</code>      | inštalátor na troch prázdnych diskoch: detekcia, disko, LVM, LUKS, <code>ext4 encrypt</code> , otvorenie zväzku kľúčom                                                                |
| <code>tpm.nix</code>          | od konca po koniec: formát, zapečatenie do <code>swtpm</code> , reštart, odomknutie bez obsluhy                                                                                       |
| <code>impermanence.nix</code> | <code>tmpfs</code> koreň, zoznam trvalých adresárov, práva 700 a oddelené skupiny domén                                                                                               |
| <code>hardening.nix</code>    | obe polovice: čo je zapnuté a čo je zámerne vypnuté; povrch jadra pre <code>k3s</code> , <code>rke2</code> , <code>containerd</code> a <code>Longhorn</code>                          |
| <code>resize.nix</code>       | <code>rast /persist</code> za behu v správnom poradí                                                                                                                                  |
| <code>admin-vm.nix</code>     | <code>lososd</code> : D-Bus, HTTP API, token, prestavba                                                                                                                               |
| <code>front-vhost.nix</code>  | <code>nginx</code> : smerovanie podľa cesty, <code>guard len</code> pre LAN, 403 na loopbacku, CSP vrátane <code>arm</code> pre <code>/widget-frame/</code> a <code>/handbook/</code> |
| <code>setup.nix</code>        | trasy prvého spustenia, certifikát, <code>trust.sh</code> , pôvody pre „nájdi môj box“                                                                                                |
| <code>tls.nix</code>          | vlastný certifikát boxu a cache                                                                                                                                                       |

| Test                | Čo overuje                                                                |
|---------------------|---------------------------------------------------------------------------|
| keyring.nix         | klúčenka tajomstiev boxu (fscrypt kľúč cez TPM)                           |
| console.nix         | banner na tty1 s adresou                                                  |
| cluster-vm.nix      | koexistencia dvoch Kubernetes runtime na jednom hostiteľovi               |
| edge-vm.nix         | edge + box: registrácia, tunel                                            |
| market-vm.nix       | produkčné zapojenie Stripe gate                                           |
| nextcloud-httpd.nix | Apache v pode Nextcloudu proti fixtúre webroot (front-controller rewrite) |
| admin-ui.nix        | produkčný zväzok SPA v prehliadači so skutočnou hlavičkou CSP             |
| design-system.nix   | render React obalu dizajnového systému (len vývoj)                        |

tests/invariants.nix nie je VM: pri vyhodnotení konfigurácie install pripína hodnoty, ktoré si box nemôže dovoliť stratiť driftom predvolenej hodnoty (GC, strop boot menu, režim odomykania, - -impure, fragment #install, cache). Beží v nix flake check - -no-build, teda v CI pri nulových nákladoch na zostavenie.

VM testy potrebujú KVM a bežia lokálne pred zlúčením zmeny (nix build .#checks.x86\_64-linux.<názov>); bez KVM bežia pod TCG pomalšie, ale bežia.

### 7.2.3 Testy v prehliadači

admin-ui/app/tests/\*.browser.mjs (Playwright, Chromium) spúšťajú skutočný zostavený SPA so skutočnou hlavičkou CSP: sprievodca (36 scenárov), aplikácia (10), oznámenia, vzhľad. Zlyhajú pri každom novom porušení CSP, takže výmena knižnice, ktorá vkladá štýly za behu, zafarbí kontrolu na červeno.

### 7.2.4 Kontinuálna integrácia

.github/workflows/ci.yml pri každom push a pull requeste: zhoda pinov (flake.lock = devenv.yaml = devenv.lock), clippy a rustfmt, testy oboch crate, nix flake check s invariantmi, zostavenie balíkov (lososd, registrátor, administratívne UI, príručka, OCI obrazy), **zostavenie inštalovateľného ISO a jeho boot** pod OVMF aj SeaBIOS (tests/iso-boot.py čaká na DNS dotaz inštalátora), publikovanie zostavených ciest do cache. Príručka má vlastný workflow (typová kontrola, kontrola odkazov a tokenov, zostavenie, nasadenie na GitHub Pages, zostavenie tohto PDF).

## 7.3 Hlavná skúška: mesh funguje, s edge proxy aj bez neho

Správnosť konfigurácie tohto systému sa nedá overiť na jednom stroji, lebo jeho zmysel je v sieti: box musí nájsť edge, pripojiť sa k meshu a zdieľané úložisko musí byť vidieť z oboch strán. Rovnako dôležité je však to, čo box urobí, keď edge v sieti **nie je**: musí to zistiť sám, zdieľanie odmietnuť a ďalej slúžiť svojmu majiteľovi lokálne. Hlavné overenie sú preto **dva scenáre na jednej sieti bez internetu**, každý s očakávaným výsledkom, ktoré sa predvedú za sebou na tej istej zostave dvoch boxov.

### 7.3.1 Scenár A: bez edge proxy

Zostava: dva boxy v jednej sieti, žiadny edge (alebo edge vypnutý).

| #  | Krok                                                         | Očakávaný výsledok                                                                                         |
|----|--------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|
| A1 | spustiť oba boxy a prejsť sprievodcom                        | Nextcloud a adminské rozhranie fungujú na každom boxe samostatne; panel Mesh hlási <i>Edge proxy: none</i> |
| A2 | pokúsiť sa zapnúť <i>Join the mesh</i> alebo zdieľanie disku | prepínače sú sivé s dôvodom; priame volanie API odpovie 409 s vetou, že edge nebol nájdený                 |
| A3 | nechať boxy bežať a sledovať panel Mesh                      | box skenuje sieť ďalej (mDNS <i>_losos-edge._tcp</i> a nastavená adresa), stav sa nemení, nič sa nepokazí  |
| A4 | reštartovať box                                              | po štarte je stav rovnaký: lokálne služby bežia, mesh je vypnutý, nič sa nepokúša pripojiť naslepo         |

Výsledok scenára A: box bez edge je plnohodnotné lokálne úložisko a **nikdy nezačne zdieľať disk do siete, v ktorej nie je dôveryhodný edge**. Toto je brána *enable-only* z kapitoly 6: zdieľanie sa dá zapnúť len s edge v dosahu.

### 7.3.2 Scenár B: s edge proxy

Zostava: tie isté dva boxy plus jeden edge v tej istej sieti.

| #  | Krok                                                          | Očakávaný výsledok                                                                                                                            |
|----|---------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| B1 | spustiť edge (alebo ho zapnúť k zostave zo scenára A)         | do minúty oba boxy ukážu na paneli Mesh <i>Edge proxy found</i> so zdrojom <i>On this network</i>                                             |
| B2 | na oboch boxoch zapnúť <i>Join the mesh</i>                   | prestavba prejde; <code>kubectl get nodes</code> na edge ukáže oba uzly v stave Ready                                                         |
| B3 | na oboch boxoch zapnúť zdieľanie disku                        | Longhorn na edge ukáže dva uzly s pridelenou kapacitou; adresár <code>/home/shared</code> je odomknutý (fscrypt)                              |
| B4 | vytvoriť zväzok v poole a zapísať doň dáta                    | zväzok má repliky na oboch boxoch; dáta sú čitateľné po odpojení jedného boxu                                                                 |
| B5 | vypnúť edge                                                   | do minúty oba boxy hlásia <i>Edge proxy: none</i> ; zapnutie zdieľania odpovie 409; vlastné aplikácie boxov bežia ďalej (návrat do scenára A) |
| B6 | zapnúť edge späť                                              | brána sa otvorí bez zásahu; uzly sa vrátia do Ready bez opätovného pridania ( <code>/etc/rancher</code> prežil)                               |
| B7 | nastaviť okno 23:00 až 07:00 a pozrieť uzol na edge mimo okna | uzol má taint <code>NoSchedule</code> ; v okne a pri nečinnosti taint zmizne                                                                  |
| B8 | pripojiť neoficiálny edge (bez podpisu koreňovým kľúčom)      | zdieľanie je dovolené, trh hlási <i>noOfficialEdge</i> a objednávka odpovie 409                                                               |

Výsledok scenára B: **úložisko dvoch boxov je spojené do jedného poolu a vidieť ho z oboch strán**, členstvo prežije výpadok edge aj reštart, a obchodovanie sa povolí len oficiálnemu edge.

### 7.3.3 Čím sú scenáre doložené

- **Automatizovaný test `tests/edge-lan.nix`** (pripravovaný v PR #75): dva virtuálne stroje na jednej sieti, edge s modulom edge a ohlasovaním služby `_losos-edge._tcp` cez mDNS, box s riadiacou rovinou zariadenia. Test prejde oba scenáre v jednom behu: box bez edge zdieľanie odmietne (A2); s edge v dosahu je prepnutie do režimu mesh prijaté (B1, B2); keď edge zmizne, box to do dvoch skenov zbadá a zapnutie zdieľania odmietne s vlastnou vetou (B5); po návrate edge sa brána znovu otvorí (B6). Druhý, nepodpísaný edge v tom istom teste overuje, že obchodovanie sa povolí len oficiálnemu edge (B8).
- **Ukážka `demo/edge-lan/run.sh`** (ten istý PR): skript postaví edge VM ako smerovač virtuálnej siete (VDE prepínač, 10.77.0.1/24, DHCP, NAT), nainštaluje box z inštalačného ISO, zaberie ho, a prevedie oba scenáre cez API: box bez edge zdieľanie odmietne; edge sa zapne, box ho nájde

sám, panel Mesh to ukáže a dovoľí zdieľať úložisko; edge sa vypne, box to do minúty zbadá a zdieľanie odmietne; edge sa vráti a brána sa otvorí. Nahrávka oboch vetiev je súčasťou materiálov k obhajobe. Skript je zároveň základom reprodukovateľného firemného nasadenia (edge a boxy v jednej sieti), takže overenie konfigurácie a návod pre operátora sú jeden a ten istý postup.

Body A1 až A4, B1, B2, B5, B6 a B8 pokrýva test `edge-lan.nix` a ukážka; B3, B4 a B7 sú overené v `cluster-vm.nix` (koexistencia runtime) a na edge ručne, a sú to body, ktoré ukážka na obhajobe predvedie naživo.

## 7.4 Overenie jedného boxu vo virtuálnom stroji

Kontrolný zoznam po inštalácii, s očakávaným výsledkom. Je to zároveň scenár ukážky na obhajobe; pri každom bode je uvedený, ktorý bod zadania dokladá.

| #  | Krok                                                                                  | Očakávaný výsledok                                                                                                      | Dokladá |
|----|---------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|---------|
| 1  | Nabootovať ISO, nechať menu vybrať autodetekciu                                       | inštalátor beží bez otázok, vypíše <code>unlock: TPM (so swtpm)</code> alebo <code>unlock: keyfile in the initrd</code> | 2b      |
| 2  | Reštart bez média                                                                     | žiadna výzva na heslo; modrý banner s IP adresou a <code>&lt;názov&gt;.local</code>                                     | 2b, 2a  |
| 3  | Otvoriť IP adresu v prehliadači                                                       | sprievodca, krok 1 s odtlačkom certifikátu                                                                              | 2c      |
| 4  | Nastaviť heslo (12+ znakov, pravidlá sa odškrtnú)                                     | <i>Password set</i> , náhradný kľúč zobrazený raz                                                                       | 2c      |
| 5  | Prihlásiť sa do LosOS cloudu v kroku 3                                                | Nextcloud prihlásený, presmerovanie na prehľad                                                                          | 2c, 2d  |
| 6  | Otvoriť <code>/forgejo/</code>                                                        | LosOS Git odpovedá, téma <code>losos-auto</code>                                                                        | 2d      |
| 7  | Zavrieť kartu a otvoriť administráciu znova, odomknúť heslom                          | dialóg <i>Unlock this box</i> prijme heslo; nesprávne heslo dá 401                                                      | 2d      |
| 8  | Zmeniť názov boxu na paneli Sieť, Použiť                                              | oznámenie <i>Changes applied</i> , box odpovedá na novom <code>.local</code> mene, IP nezmenená                         | 2c, 2d  |
| 9  | Vytvoriť súbor mimo <code>/persist</code> (napr. cez widget) a nahráť súbor do cloudu | po reštarte prvý zmizol, druhý ostal                                                                                    | 2d      |
| 10 | Pozrieť <code>uname -r a /proc/cmdline</code> (cez konzolu VM, v ukážke)              | najnovšie stabilné jadro; KSPP parametre ( <code>slab_nomerge</code> , <code>init_on_alloc=1</code> , ...)              | 2a      |
| 11 | <code>cryptsetup luksDump /dev/persist-vg/persist</code> (konzola VM)                 | dva kľúčové sloty, token <code>systemd-tpm2</code> v slot 1                                                             | 2b      |
| 12 | Panel Úložisko → Použiť rezervu                                                       | zvázok narastie za behu, <code>df</code> ukáže viac miesta                                                              | 2d      |
| 13 | Panel Mesh bez edge                                                                   | <i>Edge proxy: none</i> , prepínače zdieľania sivé s dôvodom                                                            | 2d      |
| 14 | <code>curl http://127.0.0.1/</code> na boxe                                           | 403 (guard len pre LAN), kým <code>curl http://127.0.0.1:8082/api/health</code> odpovie                                 | 2d      |

Body 10, 11 a 14 vyžadujú konzolu, ktorú nainštalovaný box nemá; v ukážke sa robia z inštaláčného média alebo z VM testu, nie z boxu. Body 1 až 9 sú to, čo majiteľ naozaj robí.

## 7.5 Vrstva 3: nezávislá kritická recenzia

5. októbra 2026 autor nechal nad repozitárom urobiť recenziu z pohľadu skúšajúceho („čo by tu komisia našla ako prvé“), s nálezmi overenými v kóde. Nástroje (clippy, testy, typová kontrola) nenašli nič; nálezy boli o správaní a dokumentácii. Najdôležitejšie a ich stav:

| Nález                                                                                                        | Oprava                                                                                                                                                         |
|--------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ISO spúšťalo inštalátor bez TPM, hoci dokumentácia opisovala TPM ako predvolené                              | inštalátor autodetekuje <code>/dev/tpmrm0</code> , zapečatí kľúč hneď po formáte, <code>--no-tpm</code> je opt-out (PR #47, #48); <code>tests/tpm.nix</code>   |
| aktualizácia z <code>github</code> : zdroja by zabudla disky a režim odomykania boxu                         | živé <code>install-target.nix</code> a <code>overrides.nix</code> z <code>/etc/nixos</code> , obe cesty prestavby s <code>--impure</code> ; invariant (PR #49) |
| sprievodca nevedel box zabráť (volal trasu s tokenom, ktorý ešte nemal)                                      | krok 2 najprv volá <code>POST /api/setup/claim</code> (PR #45)                                                                                                 |
| administračný kľúč sa vydal raz a nikde nezobrazil                                                           | zobrazenie v sprievodcovi s Kopírovať a Tlačiť; neskôr heslo ako hlavná cesta (PR #65)                                                                         |
| žiadne GC a žiadny strop generácií na samoaktualizujúcom sa boxe                                             | <code>nix.gc</code> 14 dní, <code>configurationLimit = 5</code> ; invariant                                                                                    |
| piny <code>nixpkgs</code> v <code>devenv.yaml</code> a <code>flake.lock</code> sa rozišli a CI to nestrážilo | úloha <code>pins</code> v CI                                                                                                                                   |
| kľúčový súbor LUKS so surovými bajtmi (NUL) sa formátoval iným kľúčom, než akým sa odomykal                  | hexadecimálny zápis, test otvorenia kľúčom hneď po disko (PR #47)                                                                                              |

Ponechané a priznané v bezpečnostnom modeli: certifikát boxu je zároveň CA bez obmedzení; synchronné obsluhy HTTP v démone; trh nikdy nebežal proti skutočnému testovaciemu režimu Stripe (bežal proti náhradám s rovnakým rozhraním). Recenzia a pripravené odpovede sú súčasťou materiálov k obhajobe.

## 7.6 Čo overené nie je

Pre úplnosť: Nextcloud a Forgejo ako celok nebootujú v žiadnom VM teste (obraz Nextcloudu má 2,6 GiB a jeho zostavenie v testovacom VM by trvalo hodiny); ich funkčnosť sa overuje ručne vo VM ukážke a Apache podu samostatným testom. Mesh s Longhornom a taintom beží v `cluster-vm.nix` len po koexistenciu runtime; objavenie edge a brána zdieľania majú test `edge-lan.nix` (PR #75), replikácia a taint sa overujú na edge ručne podľa zoznamu B3, B4 a B7. Trh je overený proti náhradám Stripe a apiservera.

## 8 Cenová ponuka hardvérového a softvérového vybavenia

Bod 3 zadania: *spracujte cenovú ponuku hardvérového a softvérového vybavenia potrebného na realizáciu projektu v prípade reálneho nasadenia.*

### Znenie bodu 3

Dovetok „v prípade reálneho nasadenia“ je doplnenie, ktoré autor navrhol konzultantovi 7. októbra 2026 spolu s pripomienkami k zadaniu; čaká na potvrdenie. Dôvod: obhajoba prebieha vo virtuálnom stroji, ktorý nič nestojí, takže ponuka má zmysel len pre skutočné nasadenie na hardvér. Ak konzultant doplnenie neprijme, platí pôvodné znenie bez dovetku a kapitola sa nemení, lebo aj tak oceňuje reálne nasadenie.

Ponuka preto oceňuje **reálne nasadenie**: jeden fyzický box pre jednotlivca alebo kanceláriu, a firemnú zostavu s vlastným edge. Ukážka na obhajobe beží vo virtuálnom stroji a nie je položkou ponuky. Táto kapitola hovorí, z čoho sa ponuka skladá a prečo; konkrétne položky s cenami a zdrojmi sú v samostatnom dokumente **Cenová ponuka LosOS** a tabuľke *cenova-ponuka.xlsx*, ktoré sú súčasťou materiálov k obhajobe (autor sa rozhodol udržiavať ich mimo repozitára kódu, lebo ceny sa menia a repozitár nesie len kód, Markdown a obrázky). Pri odovzdaní sa dokument s cenami prikladá za touto kapitolou; jeho výsledok je zhrnutý na konci kapitoly.

### 8.1 Z čoho sa realizácia skladá

LosOS má dve polohy a ponuka ich oddeľuje:

1. **Jeden box doma alebo v kancelárii.** Potrebujeme hardvér boxu, USB kľúč na inštaláciu a sieť, ktorá už existuje. Softvér je celý slobodný, teda bez licenčných poplatkov.
2. **Firemné nasadenie s vlastným edge** (niekoľko boxov v jednej sieti, verejná adresa a mesh medzi nimi). K tomu pribúda edge server, doména a prípadne hosting; a prácu inštalácie a zaškolenia, ktorú autor plánuje ponúkať po maturite.

Tretia poloha, **trh**, nie je položka ponuky, ale obchodný model: majiteľ boxu predáva kapacitu, ktorú už zdieľa, a platforma si necháva 4 %.

### 8.2 Hardvér boxu

Požiadavky vyplývajú z kódu, nie z marketingu: 64-bitový procesor x86-64, aspoň 4 GiB pamäte (8 GiB pre pohodlný beh Nextcloudu a Forgejo spolu s Kubernetes), disk 40 GiB alebo väčší (reálne podľa toho, koľko dát má box držať a zdieľať), káblová sieť, a pre predvolený spôsob odomykania disku čip TPM 2.0 (firmvérový Intel PTT alebo AMD fTPM stačí; bez neho box beží v režime kľúčového súboru). Grafický čip nie je potrebný.

Ponuka preto pracuje s dvoma úrovňami:

| Úroveň  | Čo to je                                                                                                                       | Pre koho                    |
|---------|--------------------------------------------------------------------------------------------------------------------------------|-----------------------------|
| Úsporná | repasované kancelárske mini-PC (ThinkCentre Tiny, EliteDesk Mini, OptiPlex Micro) z druhej ruky, 8 GiB RAM, SSD 256 GiB a viac | domácnosť, ukážka, prvý box |

| Úroveň          | Čo to je                                                                          | Pre koho                         |
|-----------------|-----------------------------------------------------------------------------------|----------------------------------|
| Priestrannejšia | nové mini-PC s Intel N100/N305 alebo Ryzen, 16 GiB RAM, NVMe 1 TB a viac, 2,5 GbE | firma, box so zdieľaním do meshu |

K obom patrí USB kľúč 2 GiB a viac (jednorazovo) a sieťový kábel. Druhý disk nie je potrebný: inštalátor spája všetky pevné disky do jedného zväzku, a redundanciu rieši mesh, nie RAID.

### 8.3 Hardvér edge

Edge je modul NixOS a jeho nároky sú malé: registrátor v Ruste, Traefik, rathole a voliteľne rke2 server s Longhornom. Pre firemné nasadenie v jednej sieti by prichádzalo do úvahy aj zariadenie triedy Raspberry Pi (ARM64), flake však dnes zostavuje len pre x86\_64-Linux, takže je to položka na overenie, nie hotová možnosť; istá voľba je malý x86-64 stroj v sieti alebo, pre verejnú adresu z internetu, malý VPS. Ponuka uvádza obe možnosti vedľa seba.

### 8.4 Softvér

Celý softvérový stack je slobodný softvér; licenčné náklady sú nulové. Ponuka ich uvádza, aby bolo vidieť, čo projekt používa a za akých podmienok:

| Zložka                               | Licencia          | Úloha v LosOS                       |
|--------------------------------------|-------------------|-------------------------------------|
| LosOS (tento projekt)                | AGPL-3.0-or-later | moduly, démon, UI, inštalátor, edge |
| NixOS / nixpkgs                      | MIT               | základ systému, zostavenie, cache   |
| Linux                                | GPL-2.0           | jadro                               |
| Nextcloud                            | AGPL-3.0          | LosOS cloud                         |
| Forgejo                              | GPL-3.0-or-later  | LosOS Git                           |
| k3s, rke2, containerd                | Apache-2.0        | beh aplikácií a mesh                |
| Longhorn                             | Apache-2.0        | replikované úložisko meshu          |
| Traefik, rathole                     | MIT, Apache-2.0   | TLS a tunel na edge                 |
| disko, impermanence                  | MIT               | rozloženie diskov, bezstavový koreň |
| systemd (cryptenroll), LUKS, fscrypt | LGPL/GPL          | šifrovanie a odomýkanie             |

Platené služby vstupujú až s firemným nasadením alebo trhom: doména (autorova `dasmat.us` stojí približne 8 EUR ročne), hosting edge (VPS) alebo demo hostiteľa riadiacej roviny (Vercel a Neon vo voľných úrovniach), poplatky Stripe pri trhu (pre karty EÚ približne 1,5 % + 0,25 EUR za transakciu, čo pri 4 % platforme znamená bod zvratu okolo 10 EUR na objednávku; pod ním platforma dopláca, čo recenzia v kapitole 7 odhalila a ponuka s tým počíta minimom objednávky).

### 8.5 Čo ponuka zámerne neobsahuje

- **Cenu práce autora na vývoji.** Projekt je maturitná práca; ponuka oceňuje, čo si zákazník kupuje, nie čo vznik systému stál.
- **Prémiový hardvér.** Systém beží pohodlne na ľubovoľnom stroji x86-64 s dostatkom pamäte a disku; drahší box neprinesie nič, čo by softvér využil. Jediná tvrdá hranica je architektúra: ARM (vrátane Macov s čipom Apple M) podporovaná nie je.
- **Zálohovanie.** Box je jedna kópia; druhú si drží majiteľ, kde chce.

## 8.6 Výsledok ponuky

Dokument s cenami (stav k 7. októbru 2026, slovenské maloobchodné ceny s DPH 23 %, zdroj a dátum pri každej položke) počíta dve konfigurácie reálneho nasadenia:

| Konfigurácia                 | Z čoho sa skladá                                                           | Jednorazovo | Ročne    |
|------------------------------|----------------------------------------------------------------------------|-------------|----------|
| <b>A: jeden fyzický box</b>  | jeden box (Acer Veriton N150), doména (Namecheap, 7,79 € ročne), elektrina | 452,59 €    | 25,31 €  |
| A + voliteľný edge v cloude  | k tomu Hetzner CX23 ako edge                                               | 452,59 €    | 113,72 € |
| <b>B: nasadenie vo firme</b> | tri boxy, lokálny x86 edge, prepínač, doména, elektrina                    | 1 795,59 €  | 74,37 €  |

Softvér nestojí nič: každý komponent je slobodný (tabuľka licencií vyššie), takže softvérová časť ponuky sú len prevádzkové služby (doména, prípadne VPS). Predpoklady: elektrina 0,20 € za kWh, práca autora neocenená (pozri predchádzajúcu časť), Raspberry Pi ako edge je uvedené len orientačne a označené *na overenie*, lebo flake zatiaľ zostavuje len x86\_64. Konfigurácia A je najmenšie reálne nasadenie a zároveň systém, ktorý sa predvádza na obhajobe, tam však vo virtuálnom stroji (so schváleným nahradením fyzického boxu), takže ukážka sama nič nestojí; konfigurácia B je ponuka, s ktorou chce autor po maturite osloviť firmy.

## 9 Technická dokumentácia projektu

Bod 4 zadania: *vypracujte technickú dokumentáciu k projektu*. Tento dokument je jej odovzdávaná podoba; táto kapitola opisuje, z čoho dokumentácia projektu pozostáva, pre koho je ktorá časť, ako sa zostavuje a publikuje, a ako sa udržiava v súlade s kódom.

### 9.1 Vrstvy dokumentácie

| Vrstva                               | Pre koho                    | Kde                                                                            |
|--------------------------------------|-----------------------------|--------------------------------------------------------------------------------|
| <b>Príručka majiteľa</b> (tento web) | človek s boxom na stole     | handbook/docs/, publikovaná na losos.dasmat.us a na každom boxe pod /handbook/ |
| <b>Technická dokumentácia (KOP)</b>  | komisia, konzultant         | handbook/docs/project/kop/, táto kapitola príručky a PDF                       |
| <b>Vývojárska wiki</b>               | kto systém stavia a mení    | wiki/ v repozitári, publikovaná do GitHub wiki                                 |
| <b>CLAUDE.md a hlavičky súborov</b>  | vývojár (a AI asistent)     | koreň repozitára; prvé desiatky riadkov takmer každého .nix a .rs súboru       |
| <b>Bezpečnostný model a baseline</b> | recenzent                   | docs/security-model.md, docs/baseline.md                                       |
| <b>Drôtový kontrakt API</b>          | klienti démona              | backend/schema.json                                                            |
| <b>Testy</b>                         | každý, kto mení rozhodnutie | tests/, s komentárom, prečo test existuje                                      |

Rozdelenie podľa čitateľa je zámerné. Príručka nikdy nepredpokladá terminál a každá stránka o problémoch začína krokmi, ktoré potrebujú len lokálnu sieť. Wiki predpokladá vývojára s Nixom a Rustom. Hlavičky súborov zaznamenávajú *prečo* je kód taký, aký je, spravidla aj chybu, ktorá bola raz zaplatená; recenzent, ktorý sa pýta „prečo“, nájde odpoveď najprv tam.

### 9.2 Príručka majiteľa

Príručka je web postavený v Docusaurus so šiestimi časťami: Začíname, Typy nasadenia, Príručka vlastníka, Keď niečo nefunguje (podľa príznaku), Referencia a Projekt. Má offline vyhľadávanie (index je súčasťou zväzku), svetlú a tmavú tému zhodnú s administráčnymi stránkami (paleta je bajt po bajte tá istá a kontrola `check:tokens` zlyhá, keď sa rozíde) a rozhranie v angličtine, slovenčine a nemčine; stránky sú zatiaľ anglické a prekladajú sa po jednej.

Zostavuje sa dvakrát z jedného zdroja: verejná stránka na GitHub Pages pri každom push do main (`.github/workflows/handbook.yml`) a kópia na box (`losos-handbook` vo `flake/packages.nix`, koreň `/handbook/`), ktorú nginx podáva len z LAN pod vlastnou politikou CSP. Chybové hlásenia administráčnych stránok odkazujú na konkrétnu stránku príručky; kontrola `check:links` zlyhá, keď hlásenie odkazuje na stránku, ktorá neexistuje. Rozbitý odkaz je v návode chyba, nie varovanie, preto `onBrokenLinks: 'throw'`.

### 9.3 Tento dokument

Zdrojom sú stránky Markdown v `handbook/docs/project/kop/` s obrázkami vedľa nich. Z nich sa zostavuje:

- **web**: ako kapitola príručky (toto, čo čítate), s navigáciou, vyhľadávaním a oboma témami;

- **PDF na odovzdanie:** skript `handbook/scripts/build-kop.sh` zreťazí kapitoly v poradí, pandoc ich prevedie do Typstu s vlastnou šablónou (titulná strana, obsah, číslované kapitoly, popisy obrázkov) a Typst vysádza PDF. Beží v CI pri každej zmene príručky a výsledok sa publikuje na `losos.dasmat.us/kop/LosOS-KOP.pdf`; ten istý skript beží aj lokálne (`nix shell --inputs-from . nixpkgs#pandoc nixpkgs#typst`).

Jeden zdroj pre web aj PDF znamená, že dokument sa nemôže rozísť sám so sebou, a Markdown v repozitári znamená, že každá zmena dokumentu prechádza tým istým pull requestom a recenziou ako kód.

## 9.4 Ako sa dokumentácia drží v súlade s kódom

- **Tvrdenia s testom.** Čo dokumentácia tvrdí o správaní (poradie krokov pri raste disku, čo je zámerne vypnuté, 403 na loopbacku), má test, ktorý zlyhá, keď kód tvrdenie prestane spĺňať.
- **Pull request šablóna.** Každá zmena viditeľná človeku musí mať v popise snímku pred a po, pri rovnakej veľkosti okna; čo sa netestovalo, musí byť povedané (VM testy bez KVM sa uvádzajú ako „nеспustené“, nie zamlčané).
- **Kontroly v CI.** Tokeny príručky proti UI, odkazy z chybových hlásení, typová kontrola, zostavenie bez rozbitých odkazov.
- **Pravidlá písania.** Príručka vedie tým, čo čitateľ vidí; príčina druhá, náprava tretia; IP adresa vždy pred `.local` menom; pomenovať pull request, keď sa správanie nedávno zmenilo, aby čitateľ so starším boxom vedel, prečo sa jeho obrazovka líši.

## 9.5 Čo dokumentácii chýba

Otvorene: stránky príručky zatiaľ nie sú preložené do slovenčiny a nemčiny (preklad je pripravený technicky, nie obsahovo); niekoľko návrhových špecifikácií v `docs/superpowers/` nesie zastarané statusy z čias pred prepisom do Rustu; a časť wiki je stručnejšia než príručka, lebo ťažisko sa presunulo k nej. Zoznam je v kapitole 10 medzi plánovanou prácou.

# 10 Záver

## 10.1 Čo bolo dosiahnuté

Zadanie žiadalo preskúmať a opísať funkcionality a možnosti operačného systému, vybrať jadro, systém nainštalovať, nakonfigurovať a overiť, spracovať cenovú ponuku a vypracovať technickú dokumentáciu. Všetky štyri body sú splnené a každý má v tomto dokumente kapitolu.

Hotové a otestované je: bezstavové zariadenie na NixOS so šifrovaným trvalým zväzkom, odomykanie z TPM aj z kľúčového súboru, inštalácia UEFI aj BIOS bez obsluhy, administratívne stránky s jedným heslom a náhradným kľúčom, LosOS cloud a LosOS Git ako záťaž v lokálnom clustri s jednotným vzhľadom, nočná samooprava a samoaktualizácia so stropom generácií a GC, rast disku za behu, hardening postavený z primitív KSPP s testom chránených výnimiek, edge s tunelom, meshom a oknom výpočtov, trh od konca po koniec proti náhradám Stripe, riadiaca rovina edge pre prezentácie na Vercel, príručka majiteľa na každom boxe a táto dokumentácia.

Projekt má približne 45 000 riadkov vlastného kódu v Nixe, Ruste a TypeScripte, vyše 200 jednotkových testov, sedemnášť testov vo virtuálnych strojoch, invarianty vyhodnocované pri zostavení, testy v prehliadači a CI, ktoré inštalčné médium nielen zostaví, ale aj nabootuje.

## 10.2 Čo je plánované

- **Otvorenie trhu** po založení podnikania, ktoré bude platformou Stripe; dovtedy ostáva karta Trh sivá. Prvý beh proti testovaciemu režimu Stripe.
- **Objavovanie edge a oficiálna identita edge** (pripravované v PR #75): box hľadá edge v LAN aj na nastavenej adrese, odmieta zdieľanie bez dosiahnuteľného edge a obchoduje len s edge podpísaným koreňovým kľúčom projektu.
- **Firemné nasadenie** ako produkt: reprodukovateľný skript pre edge a boxy v jednej sieti, návod pre operátora firmy.
- **Preklady príručky** do slovenčiny a nemčiny.
- **Oddelená CA s obmedzením mena** namiesto certifikátu boxu, ktorý je zároveň koreňovou autoritou.
- **Rozšírenie VM testov** o boot nainštalovaného systému až po prihlásenie do Nextcloudu (dnes to overuje ukážka, nie test).

## 10.3 Poučenia

- **Testy majú chrániť rozhodnutia, nie len správanie.** Najcennejšie testy projektu sú tie, ktoré zlyhajú, keď niekto „opraví“ niečo, čo je vypnuté zámerne.
- **Lintre a kontroly do CI od prvého dňa.** Kód v Ruste driftoval bez rustfmt a clippy, piny nixpkgs sa rozišli bez kontroly; obe veci stáli viac času dodatočne než by stáli na začiatku.
- **Chyby, ktoré vidno len na skutočnom štarte.** Dve najvážnejšie chyby projektu (kľúčový súbor s NUL bajtmí, chýbajúce virtio ovládače v initrd) prešli všetkými testami, lebo žiadny test nebootoval nainštalovaný systém až do konca. Ukážka vo virtuálnom stroji nie je len prezentácia; je to test.
- **Nezávislá recenzia vlastnej práce** našla za jedno popoludnie sedem vecí, ktoré by komisia našla tiež. Oplatí sa ju urobiť skôr, než ju urobí niekto iný.
- **Infraštruktúra CI má limity, ktoré formujú architektúru.** Desafinútový strop bývalého hostiteľa CI viedol k rozdeleniu úloh a k importu artefaktov medzi nimi; štruktúra ostala, aj keď strop padol, lebo dôvody ju prežili.

Jednou vetou: LosOS robí z vyradeného mini-PC vlastný cloud, ktorý sa sám spravuje, a z jeho voľných zdrojov susedskú sieť, z ktorej sa raz dá aj zarábať.

# 11 Prílohy

## 11.1 Príloha A: Slovník pojmov

| Pojem                  | Význam                                                                                 |
|------------------------|----------------------------------------------------------------------------------------|
| box                    | počítač s nainštalovaným LosOS (appliance)                                             |
| edge                   | server, ku ktorému box drží tunel: verejná adresa, mesh, trh                           |
| mesh                   | sieť boxov jedného edge, ktoré si zdieľajú voľný disk a procesor                       |
| trh                    | voliteľný predaj a nákup zdieľanej kapacity cez Stripe Connect, 4 % platforme          |
| LosOS cloud, LosOS Git | mená, pod ktorými majiteľ vidí Nextcloud a Forgejo                                     |
| administračné stránky  | webové rozhranie boxu na /, len z LAN                                                  |
| sprievodca             | trojkrokové prvé spustenie: certifikát, heslo, prihlásenie                             |
| náhradný kľúč          | 64-znakový administračný kľúč, zobrazený raz; mocný ako root                           |
| claim                  | zabratie čerstvého boxu prvým, kto dokončí sprievodcu                                  |
| impermanence           | model, v ktorom koreň je tmpfs a prežije len zoznam adresárov na /persist              |
| /persist               | šifrovaný zväzok (LVM → LUKS2 → ext4) so všetkým trvalým stavom                        |
| TPM 2.0                | čip, do ktorého inštalátor zapečatí kľúč disku                                         |
| keyfile režim          | odomykanie kľúčom v initrd na nešifrovanom boot oddiele (bez TPM)                      |
| flake                  | deklaratívny popis celého systému v Nixe                                               |
| generácia              | jedna zostavená verzia systému; posledných päť ostáva v boot menu                      |
| prestavba (rebuild)    | zostavenie a prepnutie systému z popisu; cesta pre Použiť, nočnú aktualizáciu aj reset |
| overrides.nix          | súbor s voľbami majiteľa, ktorý zapisuje démon                                         |
| install-target.nix     | súbor s diskami, firmvérom a režimom odomykania boxu, ktorý zapisuje inštalátor        |
| lososd, lososctl       | riadiaci démon a jeho fasáda v príkazovom riadku                                       |
| k3s, rke2              | dva Kubernetes runtime: vlastný cluster boxu a agent meshu                             |
| Longhorn               | replikované blokové úložisko meshu                                                     |
| fscrypt                | šifrovanie adresára shared vlastným kľúčom, načítaným len počas zdieľania              |
| okno                   | hodiny, v ktorých box požičiava procesor, v časovom pásme boxu                         |
| KSP                    | Kernel Self Protection Project, zdroj odporúčaní pre hardening jadra                   |
| CSP                    | Content-Security-Policy, hlavička obmedzujúca, čo smie stránka načítať a spustiť       |
| widget                 | dlaždica na nástenke Prehľadu; ručne písané bežia v izolovanom rámci                   |

## 11.2 Príloha B: Testy a kontroly

**Testy vo virtuálnych strojoch** (tests/\*.nix, nix build .#checks.x86\_64-linux.<názov>): install, tpm, impermanence, hardening, resize, admin-vm (losos-admin-daemon), front-vhost, setup, tls, keyring, console, cluster-vm, edge-vm, market-vm, nextcloud-httpd, admin-ui, design-system.

**Pri vyhodnotení:** tests/invariants.nix (losos-invariants).

**Jednotkové:** `cargo test` v `backend/`, `backend-registrar/`, `edge-vercel/`; `cargo clippy --all-targets -- -D warnings`; `cargo fmt --check`.

**V prehliadači:** `admin-ui/app/tests/{wizard,app,toasts,look}.browser.mjs`.

**CI** (`.github/workflows/ci.yml`): `pins`, `clippy`, `test`, `eval`, `losos-ctl`, `registrar-and-ui` (vrátane príručky), `images`, `publish-cache`, `iso` (zostavenie a boot pod OVMF a SeaBIOS), `release-media` pri značke `v*`. `handbook.yml`: `typecheck`, `check` (odkazy, tokeny), `build`, `deploy` na Pages, PDF tohto dokumentu.

**Dev prostredie:** `devenv shell`, `devenv test` (všetko okrem VM testov a ISO).

## 11.3 Príloha C: Štruktúra repozitára

|                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                   |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>flake.nix</code> , <code>flake.lock</code>                                                                                                                                                                                                                                                                                                                                                    | sysémy <code>iso</code> a <code>install</code> , modul <code>edge</code> , balíky, <code>checks</code>                                                                                                            |
| <code>modules/</code>                                                                                                                                                                                                                                                                                                                                                                               | NixOS moduly; voľby v <code>options.nix</code> , predvolené v                                                                                                                                                     |
| <code>defaults.nix</code>                                                                                                                                                                                                                                                                                                                                                                           |                                                                                                                                                                                                                   |
| <code>boot</code> , <code>disko</code> , <code>impermanence</code> , <code>hardening</code> , <code>configuration</code> , <code>services</code> , <code>workloads</code> ,<br><code>cluster</code> , <code>containers (nginx)</code> , <code>daemon</code> , <code>setup</code> , <code>tls</code> , <code>updates</code> , <code>console</code> , <code>edge</code> ,<br><code>proxy</code> , ... |                                                                                                                                                                                                                   |
| <code>backend/</code>                                                                                                                                                                                                                                                                                                                                                                               | <code>lososd</code> + <code>losos-ctl</code> (Rust): <code>installer</code> , <code>grow</code> , <code>setup</code> ,<br><code>signin</code> , <code>look</code> , <code>market</code> , <code>supervisor</code> |
| <code>backend-registrar/</code>                                                                                                                                                                                                                                                                                                                                                                     | registrátor <code>edge</code> (Rust): <code>server</code> , <code>join</code> , <code>window</code> ,<br><code>market</code> , <code>stripe_gate</code> , <code>identity</code>                                   |
| <code>edge-vercel/</code>                                                                                                                                                                                                                                                                                                                                                                           | riadiaca rovina <code>edge</code> ako funkcia Vercel (demo)                                                                                                                                                       |
| <code>admin-ui/app/</code><br><code>shadcn/ui</code>                                                                                                                                                                                                                                                                                                                                                | administračné stránky (React, Vite, Tailwind,                                                                                                                                                                     |
| <code>admin-ui/themes/</code>                                                                                                                                                                                                                                                                                                                                                                       | <code>tokens.css</code> , témy Nextcloud a Forgejo, logá                                                                                                                                                          |
| <code>handbook/</code><br>( <code>docs/project/kop</code> )                                                                                                                                                                                                                                                                                                                                         | príručka majiteľa (Docusaurus) a tento dokument                                                                                                                                                                   |
| <code>tests/</code>                                                                                                                                                                                                                                                                                                                                                                                 | VM testy, invarianty, <code>iso-boot.py</code>                                                                                                                                                                    |
| <code>wiki/</code>                                                                                                                                                                                                                                                                                                                                                                                  | zdroj vývojárskej wiki                                                                                                                                                                                            |
| <code>docs/</code>                                                                                                                                                                                                                                                                                                                                                                                  | bezpečnostný model, baseline, návrhové špecifikácie                                                                                                                                                               |
| <code>flake/</code><br>( <code>mold</code> , <code>ccache</code> , <code>sccache</code> )                                                                                                                                                                                                                                                                                                           | balíky, obrazy OCI, obrazy diskov, <code>fast-build</code>                                                                                                                                                        |
| <code>.github/</code>                                                                                                                                                                                                                                                                                                                                                                               | CI, akcie, šablóna pull requestu, snímky k PR                                                                                                                                                                     |

## 11.4 Príloha D: Zdroje

- Repozitár projektu: <https://github.com/dasmatus/losos> (kód, wiki, vydania)
- Príručka majiteľa: <https://losos.dasmat.us>
- NixOS manual a nixpkgs: <https://nixos.org/manual/nixos/unstable/>
- impermanence: <https://github.com/nix-community/impermanence>
- disko: <https://github.com/nix-community/disko>
- Kernel Self Protection Project, odporúčané nastavenia: [https://kspp.github.io/Recommended\\_Settings](https://kspp.github.io/Recommended_Settings)
- `systemd-cryptenroll(1)`, `crypttab(5)`, `systemd-cryptsetup`
- Linux kernel documentation: `fscrypt`, `LUKS2` (`cryptsetup`)
- k3s: <https://docs.k3s.io>; rke2: <https://docs.rke2.io>
- Longhorn: <https://longhorn.io/docs>
- Nextcloud Server Administration Manual: <https://docs.nextcloud.com>
- Forgejo documentation: <https://forgejo.org/docs>
- Traefik: <https://doc.traefik.io/traefik>; rathole: <https://github.com/rapiz1/rathole>

- Stripe Connect, destination charges: <https://docs.stripe.com/connect>
- Docusaurus: <https://docusaurus.io>; pandoc: <https://pandoc.org>; Typst: <https://typst.app/docs>
- Stredná priemyselná škola elektrotechnická, Hálova 16: Maturitná skúška, informačný materiál školy (predbežný harmonogram 2026/2027)