dvb-WarpPool

Votre propre pool de minage solo Bitcoin

Pool solo public 0 % — Manuel de l'opérateur

Exploiter dvb-WarpPool comme un pool ouvert, non-custodial et sans frais

Version 1.52.0 · Édition novembre 2026

Sommaire

1 Ce qu'est un pool public 0 %

2 Activer le mode per-worker

3 Ce que les mineurs saisissent

4 Pages de portefeuille publiques & espace propriétaire

5 Confidentialité de l'opérateur

6 Accepter les dons (facultatif)

7 Profil de durcissement public

8 TLS

9 Derrière un reverse proxy

10 Liste de contrôle de déploiement

11 Support & liens

1 · Ce qu'est un pool public 0 %

Un pool public 0 % transforme dvb-WarpPool en un **pool solo ouvert, non-custodial et sans frais** : chaque mineur qui se connecte mine vers *sa propre* adresse Bitcoin — tirée de l'identifiant Stratum — et la totalité de la récompense de bloc revient directement à celui qui trouve le bloc, on-chain. Le pool ne détient jamais de fonds. Il n'y a rien à retirer, et rien qui exige de faire confiance.

Techniquement, il s'agit du **mode de paiement per-worker**. Dans le mode `single` par défaut, la récompense entière paie une seule adresse d'opérateur — la configuration ordinaire « solo + amis », où chaque mineur travaille pour *votre* portefeuille. En mode **per-worker**, chaque mineur fournit sa propre adresse comme nom d'utilisateur Stratum, et la coinbase est construite par connexion pour payer exactement cette adresse : **une sortie, la subvention complète, aucun frais.**

Les quatre propriétés qui le définissent

- **Non-custodial.** Le pool ne touche jamais aux fonds de personne. Le réseau Bitcoin paie directement celui qui trouve le bloc.
- **Per-worker.** Différents mineurs peuvent utiliser différentes adresses ; chacun conserve la récompense complète des blocs qu'il trouve.
- **Récompense complète pour celui qui trouve.** Une unique sortie de coinbase paie l'intégralité de la subvention plus les frais à l'adresse figurant dans l'identifiant du mineur gagnant — 0 % de prélèvement.
- **Fail-closed.** Le démon refuse de démarrer si des frais ou un don d'opérateur sont configurés en même temps que le mode per-worker, de sorte que vous ne pouvez pas accidentellement exploiter un pool « public » qui prélèverait discrètement une part de la récompense.

> **Garde-fou d'honnêteté — à lire attentivement.** dvb-WarpPool **n'a pas encore trouvé de bloc sur le mainnet.** Le chemin de consensus (coinbase per-worker → un vrai nœud Bitcoin Core accepte le bloc → la récompense paie celui qui trouve) est vérifié de bout en bout sur regtest et signet, mais le premier bloc mainnet reste encore « pas encore ». Faites la promotion du pool pour exactement ce qu'il est — un pool solo neuf, ouvert et non-custodial — et **jamais** comme « éprouvé au combat ». Si vous publiez une page d'accueil, gardez ce garde-fou visible.

> **Comparaison avec le mode `single`.** Si vous voulez seulement miner vos propres machines, ou un groupe fixe d'amis, tous payant un seul portefeuille, restez en mode `single` — voir le *Manuel utilisateur* principal. Ce manuel de l'opérateur porte spécifiquement sur l'ouverture du pool afin que des inconnus puissent y diriger leurs mineurs et que chacun conserve ce qu'il trouve.

2 · Activer le mode per-worker

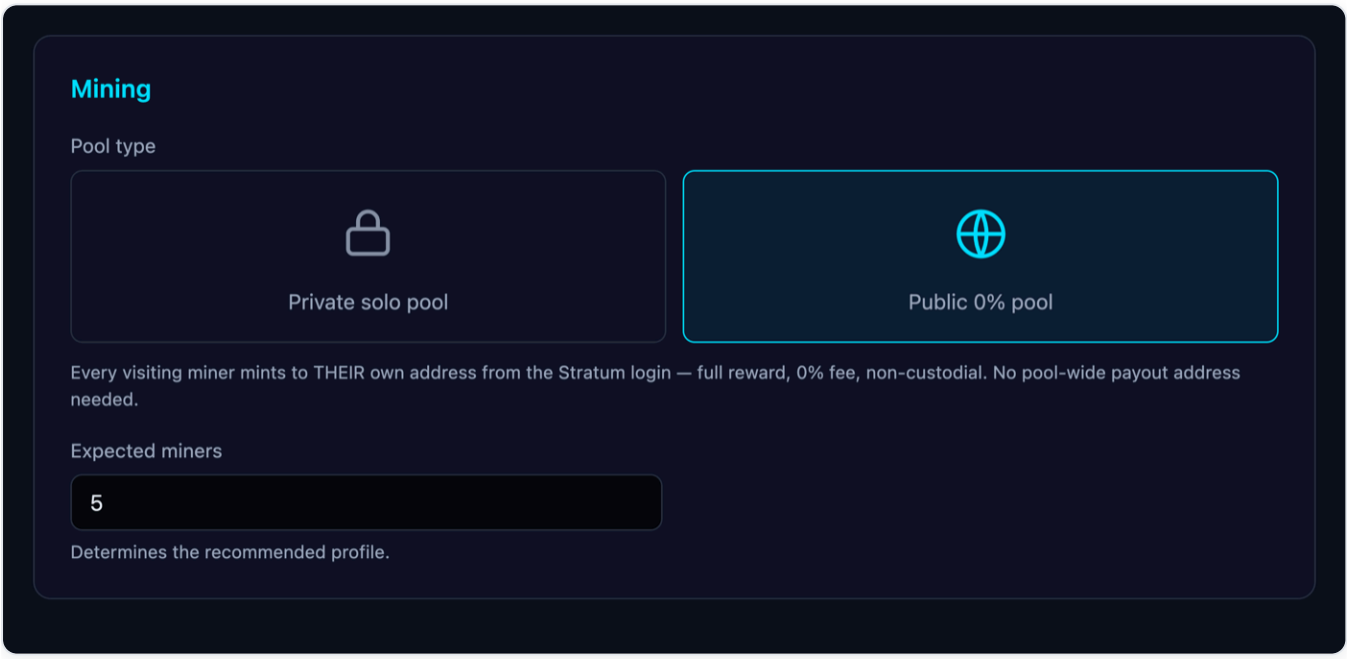
Il existe deux voies vers le même résultat : l'assistant de configuration (recommandé pour une nouvelle installation) ou une modification d'une seule ligne dans `config.toml`. Les deux écrivent le même drapeau `payout_mode = "per_worker"`.

Voie A — l'assistant de configuration

Lorsque vous installez via l'assistant de configuration, sa carte **Minage** propose un choix de **type de pool** : **Pool solo privé** ou **Pool public 0%**. Choisir **Pool public 0%** fait trois choses pour vous :

- écrit `payout_mode = "per_worker"` ;
- masque le champ **adresse de paiement** à l'échelle du pool (il est inutilisé dans ce mode — chaque mineur apporte la sienne) ;
- force les **frais du pool** à 0 et masque les champs de frais.

Aucune édition manuelle n'est requise.



La carte Minage de l'assistant de configuration : choisir « Pool public 0% » bascule le pool vers les paiements per-worker et masque les champs d'adresse de paiement et de frais.

Voie B — `config.toml`

Si vous gérez la configuration à la main, un seul drapeau constitue tout l'interrupteur de la fonctionnalité :

```
[mining]
payout_mode = "per_worker"
# payout_address is IGNORED in per_worker mode (each miner brings its own).
# pool_fee_percent / operator_donation_percent MUST be 0 — the daemon
# refuses to start otherwise (full reward goes to the finder).
```

Fail-closed sur les frais ou les dons

Le démon **échoue en mode fermé (fail closed)** au démarrage si un `pool_fee_percent` ou un `operator_donation_percent` non nul est configuré en même temps que `per_worker`. Il ne démarrera pas tant que les frais et le don ne seront pas revenus à zéro. C'est un invariant de sécurité délibéré : un pool per-worker paie à celui qui trouve la récompense *complète*, il n'existe donc aucune configuration dans laquelle il pourrait discrètement prélever une part.

> **Les fautes de frappe ne paient jamais la mauvaise personne.** Une adresse d'identifiant invalide ou du mauvais réseau est **rejetée** à `mining.authorize` (Stratum V1) ou à l'ouverture du canal (Stratum V2). Il n'y a **aucun repli** vers une adresse d'opérateur — une adresse mal saisie est refusée, jamais discrètement redirigée.

3 · Ce que les mineurs saisissent

Un mineur qui rejoint votre pool public configure son identifiant Stratum exactement comme sur n'importe quel pool solo, avec une seule règle : **le nom d'utilisateur est la propre adresse Bitcoin du mineur**, éventuellement suivie d'une étiquette de machine après un point.

```
Username:  bc1qyour-own-address.rig1
Password:  x   (anything — it is not used)
```

- Tout ce qui précède **le premier point** est l'adresse de paiement ; ce qui suit est une étiquette de machine libre utilisée uniquement pour les statistiques.
- Le mot de passe peut être n'importe quoi (couramment `x`).

Types d'adresses pris en charge

Toute adresse mainnet valide fonctionne. Le pool accepte :

Type	Préfixe d'exemple
P2PKH (legacy)	`1…`
P2SH (y compris SegWit encapsulé)	`3…`
P2WPKH (SegWit natif)	`bc1q…`
P2WSH (script SegWit natif)	`bc1q…`
P2TR (Taproot)	`bc1p…`

> **Mauvaise adresse = refusée, pas redirigée.** Si l'identifiant n'est pas une adresse valide pour ce réseau, le mineur est rejeté au moment de l'autorisation. Il n'y a pas d'adresse de repli, donc une faute de frappe échoue simplement à se connecter au lieu de payer la récompense de bloc au mauvais destinataire.

4 · Pages de portefeuille publiques & espace propriétaire

Chaque adresse qui mine sur votre pool obtient sa propre page de statut sans connexion — le flux « recherche mon adresse » qu'un pool solo public est censé proposer. Les mineurs mettent la leur en favori ; personne n'a besoin de compte.

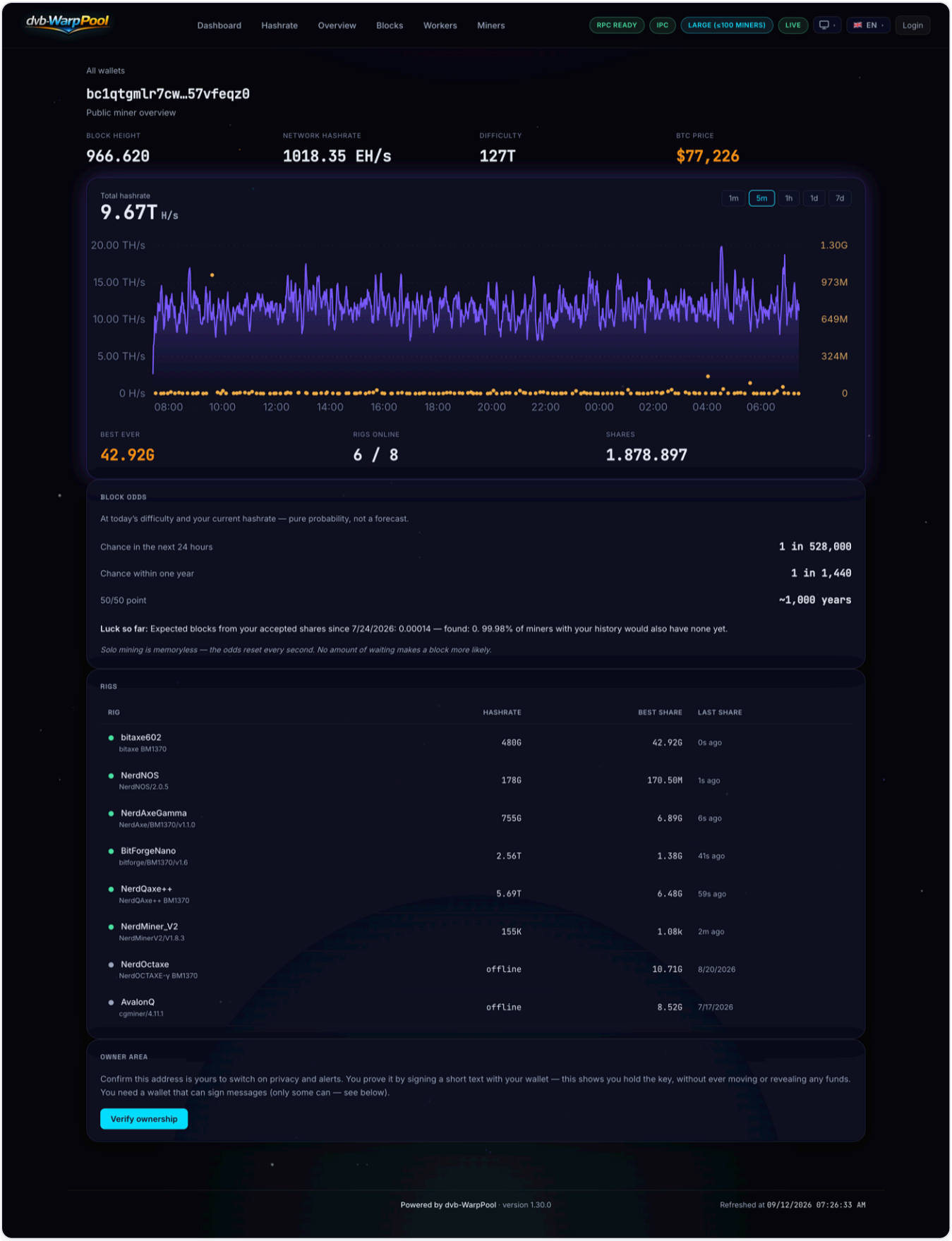
La page publique à `/users/<address>`

Ouvrez `/users/<address>` et vous voyez, pour cette adresse :

- le **hashrate total** en gros caractères, additionné sur chaque machine connectée sous l'adresse, avec une bascule de fenêtre 1 m / 5 m / 1 h / 1 j / 7 j ;
- un **graphique de hashrate sur 24 heures** où les meilleures parts apparaissent sous forme de points ;
- la **meilleure difficulté de part jamais atteinte**, les **machines en ligne** et les **parts acceptées cumulées** ;
- un **tableau par machine** — une ligne par identifiant `address.rig`, avec un point d'état en ligne, le hashrate, la meilleure part, la dernière part et le firmware, plus une colonne WarpTune si le propriétaire partage ce badge ;
- un **panneau des chances de bloc** honnête (voir ci-dessous).

Aucune connexion, aucune intervention de l'opérateur — quiconque possède l'adresse peut l'ouvrir.

Le **panneau des chances de bloc** remplace l'unique « temps estimé jusqu'au bloc » d'autrefois — une moyenne qui se lit comme « jamais » pour une machine domestique et masque tout l'intérêt du minage solo : le processus **n'a pas de mémoire**, chaque seconde est un tirage neuf et rien ne se rapproche. À la place, il montre la vraie distribution : la **chance dans les 24 prochaines heures** et **sur un an** (probabilités faibles sous la forme « 1 sur N », pourcentages arrondis vers le bas pour ne jamais surestimer), le **point 50/50** (la médiane, pas la moyenne), et la **chance jusqu'à présent** — blocs attendus vs trouvés, avec un honnête « N % des mineurs avec votre historique n'en auraient pas encore trouvé non plus. ». Une note rappelant cette absence de mémoire rend le point central explicite : aucune durée d'attente ne rend un bloc plus probable.



Une page publique par portefeuille à /users/<address> : hashrate agrégé, ventilation par machine et meilleure part — sans connexion requise.

JSON compatible ckpool. Les mêmes chiffres sont exposés via `GET /api/users/<address>` au format de champs ckpool (`hashrate1m` , `hashrate1hr` , `bestshare` , `worker[]` , …), de sorte que le firmware des mineurs et les outils de statistiques tels que `ckstats` peuvent le lire directement. Les deux routes sont en HTTP simple sous le port de statut, derrière le même reverse proxy que le tableau de bord.

L'espace propriétaire en opt-in

Par défaut, une page de portefeuille est publique. Un mineur qui souhaite plus de contrôle peut **prouver la propriété** de l'adresse et débloquer un espace propriétaire — entièrement par mineur, rien que l'opérateur ne configure :

1. La page remet un court **message de défi** à usage unique.
2. Le mineur le signe dans la boîte *Signer un message* de son propre portefeuille — la fonctionnalité standard BIP-137 « vérifier une adresse » d'Electrum, Sparrow, Bitcoin Core et des portefeuilles matériels. **Aucune clé privée ne quitte le portefeuille, aucun fonds ne bouge, aucun compte n'est créé** — le pool ne fait que vérifier la signature.
3. Une signature valide place un cookie de propriétaire de courte durée pour cette adresse et fait apparaître trois commandes :
   - **Confidentialité** — masque l'adresse de la liste publique des portefeuilles et vide ses données `/users/<address>` pour tout le monde sauf le propriétaire vérifié (le lien direct continue de fonctionner, pour lui).
   - **Notifications** — dirige un sujet ntfy personnel ou un webhook vers les étapes marquantes de cette adresse : *bloc trouvé*, un nouveau *meilleur share personnel* (sur l'ensemble de ses machines), la première entrée dans le *top 100 de tous les temps* du pool, et *machine hors ligne*. Chaque type possède son propre délai anti-spam, et un bloc trouvé envoie un seul message plutôt que trois alertes.
   - **Badge WarpTune** (depuis v1.31.0) — créez un **jeton d'envoi** pour cette adresse, placez-le dans la variable d'environnement dont le nom figure dans `warptune.toml` (`pool_push_token_env`) sur la machine où tourne votre programme compagnon dvb-WarpTune, et réglez `pool_push = true` . Le programme compagnon envoie alors son snapshot à ce pool et le tableau des rigs affiche un badge par rig : mode, fréquence, simulation / en direct / pilotage coupé / obsolète ; survolez-le pour voir la dernière décision. Les visiteurs ne voient le badge réduit qu'après activation de **Afficher le badge aux visiteurs**. Le jeton n'est affiché qu'une seule fois, peut être renouvelé ou révoqué ici, et le pool ne renvoie jamais rien à un mineur.

Texte dans la coinbase (v1.45, pools per-worker) : une fois la propriété de votre adresse prouvée, vous pouvez définir jusqu'à 16 caractères (lettres, chiffres, ponctuation, espaces) qui entrent dans la coinbase de chaque bloc que ce pool construit pour votre adresse, juste après le tag du pool — votre trace sur la chaîne, visible dans tout explorateur. Prend effet au prochain job ; l'opérateur peut désactiver la fonction.

Taproot fonctionne aussi maintenant. La signature de propriété couvre les adresses à clé unique **P2PKH / P2WPKH / P2SH-P2WPKH** (BIP-137) et **Taproot (P2TR)** via **BIP-322** — signez avec un portefeuille qui le prend en charge, comme Sparrow. L'espace propriétaire ne nécessite **aucune configuration de l'opérateur** — il est piloté entièrement par le mineur.

5 · Confidentialité de l'opérateur

Un pool mainnet public ne devrait pas distribuer une liste énumérable de l'adresse et de la configuration de machine de chacun. dvb-WarpPool vous offre **deux couches indépendantes** de contrôle. Toutes deux ont par défaut le comportement ouvert, donc le durcissement ici est en opt-in.

Couche A — config : exposition du répertoire & du classement

Deux clés dans `[privacy]` définissent l'exposition par défaut de toute l'interface. Dans les deux cas, l'**opérateur connecté voit toujours les données complètes** — celles-ci n'affectent que les visiteurs anonymes, alors gardez l'authentification admin activée.

```
[privacy]
# Hide the browsable Workers/wallet directory from anonymous visitors. Each
# /users/<address> page still works by direct link; the logged-in operator
# still sees the full directory (so keep admin auth enabled).
wallet_directory_public = false
# Mask the Bitcoin address in the public best-shares leaderboard, server-side.
# The logged-in operator still sees full addresses.
leaderboard_addresses_public = false
```

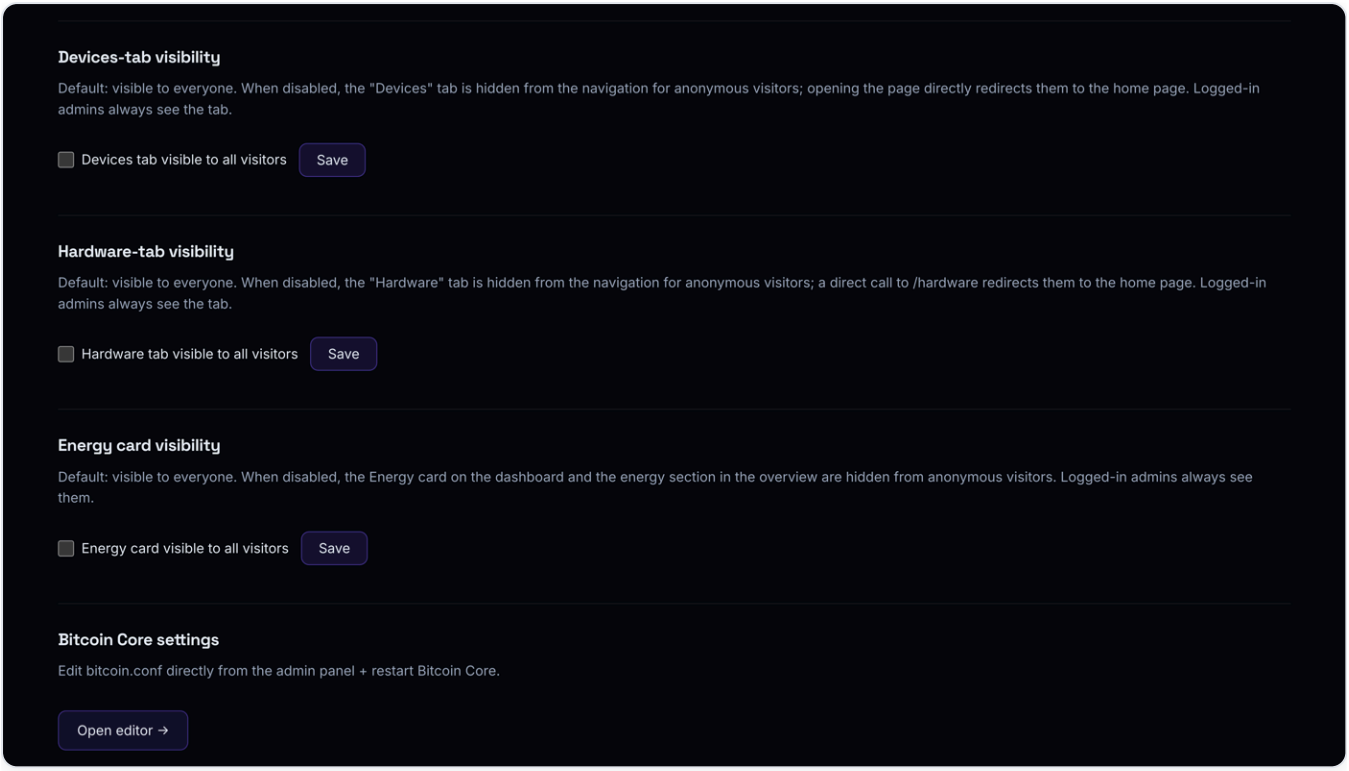
Avec cette posture, les mineurs atteignent leur propre page par lien, mais le pool ne publie pas de liste navigable de tout le monde, et les adresses du classement sont masquées côté serveur. Cela ne touche ni aux paiements ni à la construction des blocs.

Couche B — bascules de visibilité de l'interface Admin

Sous **Admin** → **Profil**, un ensemble de bascules de visibilité masquent des vues entières aux visiteurs anonymes :

- l'onglet **Appareils** (la route `/miners` — le gestionnaire de périphériques interactif avec la télémétrie par périphérique) ;
- l'onglet **Matériel** (ressources de l'hôte et image du profil de taille) ;
- la carte **Énergie** sur le tableau de bord.

> **Quel onglet la bascule masque-t-elle ?** La bascule de visibilité contrôle l'onglet **Appareils** — la route `/miners` avec la télémétrie de périphériques en direct — **et non** les listes **Mineurs** ou **Workers** (Workers = portefeuilles, Mineurs = logins Stratum, Appareils = le matériel) ; la bascule concerne la vue du gestionnaire de périphériques.



Admin → Profil : les bascules de visibilité masquent les vues Appareils/Matériel/Énergie aux visiteurs anonymes ; les adresses du classement public sont masquées côté serveur.

> **Posture recommandée pour un mainnet public.** Réglez `wallet_directory_public = false` et `leaderboard_addresses_public = false`, et masquez les vues Appareils, Matériel et Énergie aux visiteurs anonymes. Les mineurs atteignent toujours leur propre page `/users/<address>` par lien ; l'opérateur conserve la vue d'ensemble complète lorsqu'il est connecté.

Supprimer un worker

Si un mineur vous demande de disparaître du classement, ou si d'anciens noms de test l'encombrent, ouvrez **Admin → Workers**. **Supprimer** fait oublier un worker qui n'est pas connecté : son entrée, ses meilleures shares (il quitte donc le classement) et sa mémoire VarDiff. L'historique des shares et les blocs trouvés sont conservés. Un worker connecté ne peut pas être supprimé : le mineur doit d'abord se déconnecter, et s'il se reconnecte plus tard, il repart simplement de zéro. Un mineur qui vient de s'arrêter apparaît comme **actif récemment** et peut déjà être supprimé. Les noms dont la rétention a déjà supprimé l'entrée de worker, mais qui figurent encore au classement, sont listés dans un second tableau, **Résidus sans entrée de worker**, et se suppriment de la même façon. Chaque tentative est consignée dans le journal d'audit.

6 · Accepter les dons (facultatif)

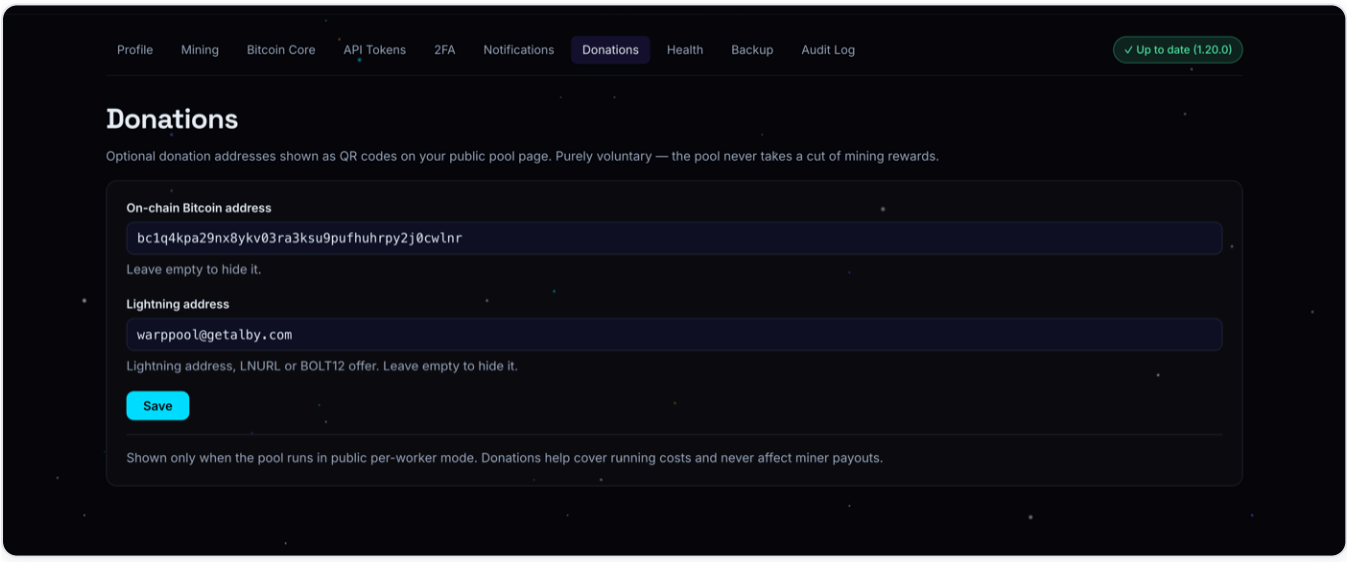
Comme un pool per-worker prélève **0 %** de chaque récompense, la seule façon de financer l'hébergement est un don volontaire. Le tableau de bord prend cela en charge de façon intégrée, et toute la fonctionnalité n'apparaît **qu'en mode per-worker** — un pool solo privé ne l'affiche jamais.

Définir les adresses

Définissez-les sous **Admin** → **Dons**. L'entrée de navigation est **masquée sauf si le pool fonctionne en mode per-worker**.

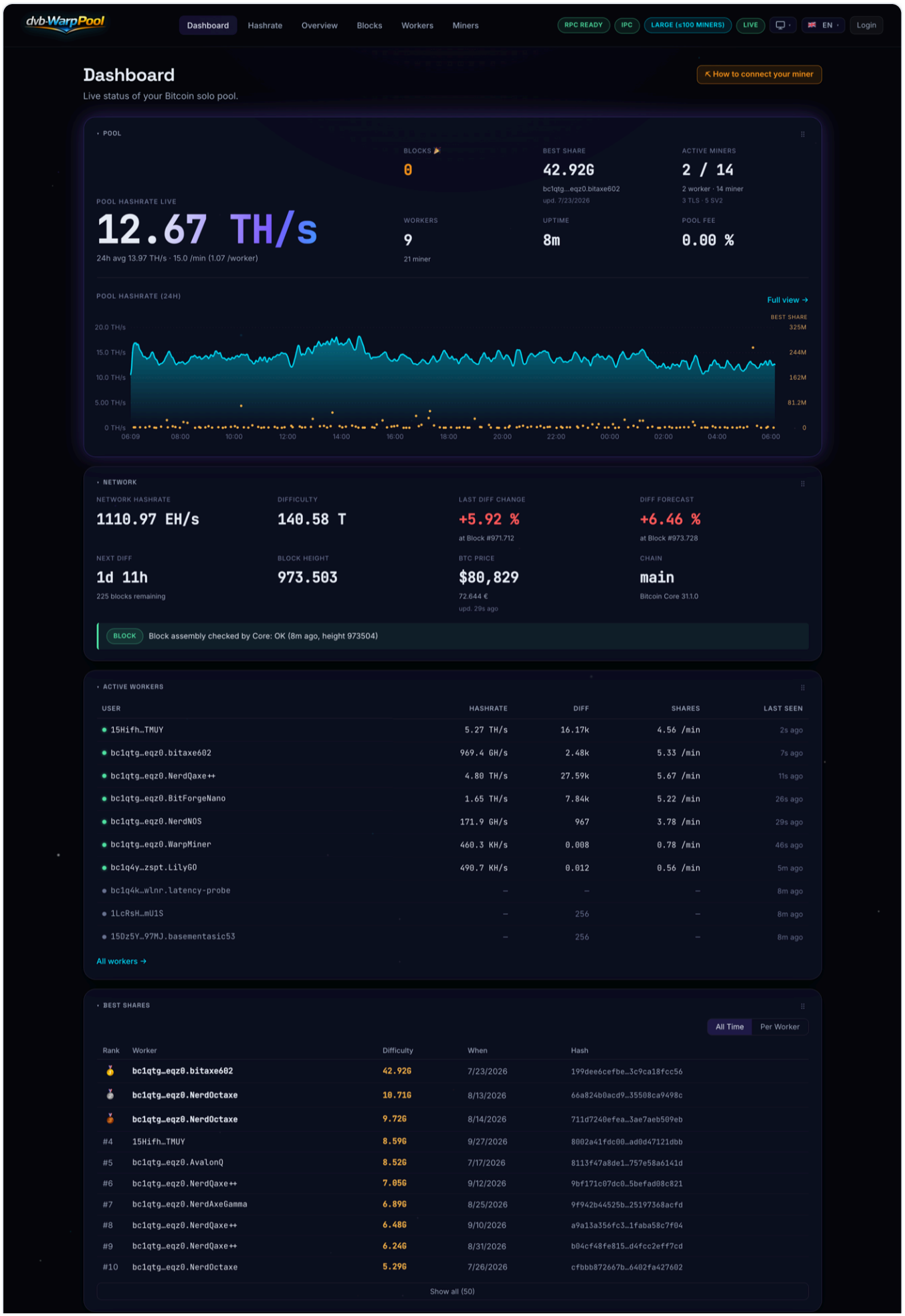
Champ	Ce qu'il faut saisir
Adresse Bitcoin on-chain	Toute adresse Bitcoin valide. Elle est validée à l'enregistrement et affichée sous forme de QR code `bitcoin:` que le portefeuille du visiteur peut ouvrir directement.
Adresse Lightning	Une adresse Lightning, un LNURL ou une chaîne BOLT12, stockée telle que vous la saisissez. Affichée dans son propre champ de copie et son QR code.

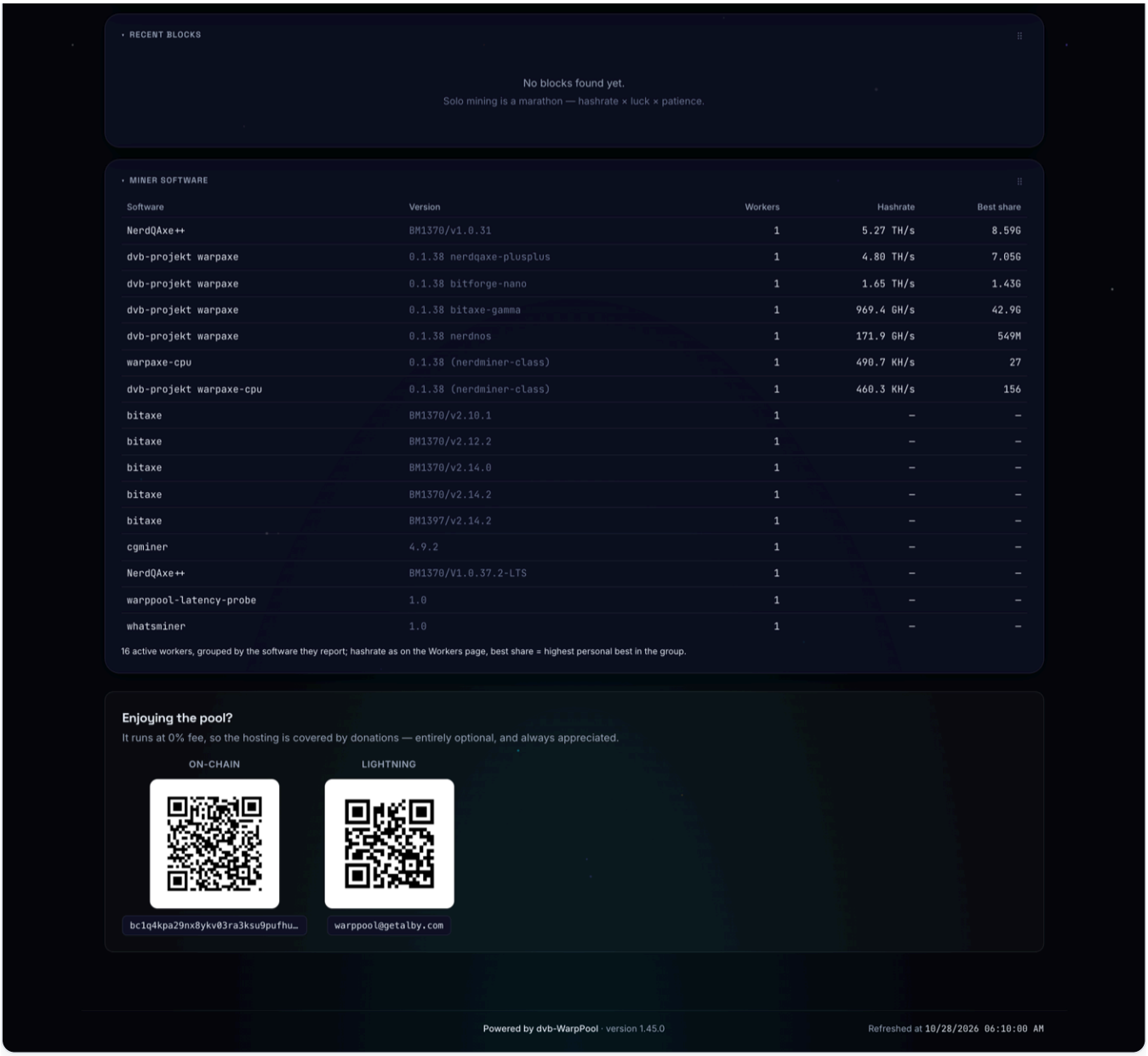
Laissez l'un ou l'autre champ vide pour masquer cette option. Lorsqu'au moins un est renseigné, une carte **« Le pool vous plaît ? »** avec des boutons de copie et des QR codes apparaît en **bas du tableau de bord public**.



Admin → Dons (pools per-worker uniquement) : définissez une adresse de pourboire on-chain et/ou Lightning ; une carte de soutien apparaît alors sur le tableau de bord public.

> **Les dons ne touchent jamais à la récompense de bloc.** Ces adresses sont stockées comme paramètres du pool (et non dans `config.toml`), donc l'enregistrement prend effet immédiatement **sans redémarrage**. Ce sont de simples métadonnées d'opérateur — elles ne sont **jamais mêlées à la construction de la coinbase**, donc en définir une ne peut pas affecter la règle fail-closed selon laquelle un pool per-worker paie à celui qui trouve la récompense complète. Un « don » ici est un visiteur qui envoie des sats à une adresse affichée, non une part de la récompense de bloc de quiconque.


dvb-WarpPool
Dashboard
Hashrate
Overview
Blocks
Workers
Miners
RPC READY
IPC
LARGE (≤100 MINERS)
LIVE
EN
Login
Dashboard
Live status of your Bitcoin solo pool.
How to connect your miner
POOL
POOL HASHRATE LIVE
12.67 TH/s
24h avg 13.97 TH/s · 15.0 /min (1.07 /worker)
BLOCKS
0
WORKERS
9
21 miner
BEST SHARE
42.92G
bc1qtg…eqz0.bitaxe602
upd. 7/23/2026
UPTIME
8m
ACTIVE MINERS
2 / 14
2 worker · 14 miner
3 TLS · 5 SV2
POOL FEE
0.00 %
POOL HASHRATE (24H)
Full view →
BEST SHARE
20.0 TH/s
15.0 TH/s
10.0 TH/s
5.00 TH/s
0 TH/s
325M
244M
162M
81.2M
0
06:09
08:00
10:00
12:00
14:00
16:00
18:00
20:00
22:00
00:00
02:00
04:00
06:00
NETWORK
NETWORK HASHRATE
1110.97 EH/s
DIFFICULTY
140.58 T
LAST DIFF CHANGE
+5.92 %
at Block #971.712
DIFF FORECAST
+6.46 %
at Block #973.728
NEXT DIFF
1d 11h
225 blocks remaining
BLOCK HEIGHT
973.503
BTC PRICE
$80,829
72.644 €
upd. 29s ago
CHAIN
main
Bitcoin Core 31.1.0
BLOCK
Block assembly checked by Core: OK (8m ago, height 973504)
ACTIVE WORKERS
USER
HASHRATE
DIFF
SHARES
LAST SEEN
15Hifh…TMUY
5.27 TH/s
16.17k
4.56 /min
2s ago
bc1qtg…eqz0.bitaxe602
969.4 GH/s
2.48k
5.33 /min
7s ago
bc1qtg…eqz0.NerdQaxe++
4.80 TH/s
27.59k
5.67 /min
11s ago
bc1qtg…eqz0.BitForgeNano
1.65 TH/s
7.84k
5.22 /min
26s ago
bc1qtg…eqz0.NerdNOS
171.9 GH/s
967
3.78 /min
29s ago
bc1qtg…eqz0.WarpMiner
460.3 KH/s
0.008
0.78 /min
46s ago
bc1q4y…zspt.Lily60
490.7 KH/s
0.012
0.56 /min
5m ago
bc1q4k…Wlnr.latency-probe
–
–
–
8m ago
1LcRsH…mU1S
–
256
–
8m ago
15Dz5Y…97MJ.basementasic53
–
256
–
8m ago
All workers →
BEST SHARES
All Time
Per Worker
Rank
Worker
Difficulty
When
Hash
bc1qtg…eqz0.bitaxe602
42.92G
7/23/2026
199dee6cefbe…3c9ca18fcc56
bc1qtg…eqz0.NerdOctaxe
10.71G
8/13/2026
66a824b0acd9…35508ca9498c
bc1qtg…eqz0.NerdOctaxe
9.72G
8/14/2026
711d7240efea…3ae7aeb509eb
#4
15Hifh…TMUY
8.59G
9/27/2026
8002a41fdc00…ad0d47121dbb
#5
bc1qtg…eqz0.AvalonQ
8.52G
7/17/2026
8113f47a8de1…757e58a6141d
#6
bc1qtg…eqz0.NerdQaxe++
7.05G
9/12/2026
9bf171c07dc0…5befad08c821
#7
bc1qtg…eqz0.NerdAxeGamma
6.89G
8/25/2026
9f942b44525b…25197368acfd
#8
bc1qtg…eqz0.NerdQaxe++
6.48G
9/10/2026
a9a13a356fc3…1faba58c7f04
#9
bc1qtg…eqz0.NerdQaxe++
6.24G
8/31/2026
b04cf48fe815…d4fcc2eff7cd
#10
bc1qtg…eqz0.NerdOctaxe
5.29G
7/26/2026
cfbbb872667b…6402fa427602
Show all (50)



Le tableau de bord public tel qu'un visiteur anonyme le voit — statistiques du pool et du réseau en direct, un classement des meilleures parts avec adresses masquées (chaque mineur conserve son étiquette de machine), et la carte de soutien facultative avec les QR codes de dons en bas.

Annoncer une maintenance à vos mineurs

Un pool public a des visiteurs que vous ne pouvez pas joindre sur Discord. Sous **Admin** → **Bannière**, vous pouvez afficher une courte annonce — une fenêtre de maintenance, une resynchronisation du nœud prévue — sous la forme d'un bandeau défilant en haut de la page du pool, là où tous ceux qui minent chez vous la verront. Vous fixez la durée pendant laquelle elle doit rester affichée (minutes, heures ou jours) et elle disparaît d'elle-même une fois ce délai écoulé, de sorte qu'une annonce oubliée ne peut pas s'éterniser ; un bouton Arrêter y met fin plus tôt. Le texte est affiché en texte brut et limité à 200 caractères. Annoncer ainsi les interruptions prévues est tout simplement une question de savoir-vivre sur un pool sur lequel d'autres comptent.

7 · Profil de durcissement public

Un pool exposé à Internet a besoin des contrôles anti-abus qu'un pool réservé aux amis peut ignorer. Chaque clé ci-dessous existe déjà ; aucune n'est spécifique au per-worker — elles s'appliquent à tout déploiement public.

```
[stratum]
# Bound how many simultaneous V1 connections one source IP may hold. On a
# public pool this stops a single host from hogging slots. Leave 0 only if you
# expect many miners behind one NAT.
max_connections_per_ip = 50

[ratelimit]
enabled = true
connects_per_sec = 5
connect_burst = 20
auths_per_sec = 1.0
auth_burst = 10

# Auto-ban IPs that keep tripping the connect limit (escalating). Loopback and
# private LAN are never banned.
autoban_enabled = true
autoban_strikes = 3
autoban_base_secs = 60
autoban_max_secs = 86400
autoban_window_secs = 600

# Ban a worker that floods invalid shares (share-flood defence). Off by
# default; turn on for a public pool.
reject_ban_enabled = true
reject_ban_min_samples = 100
reject_ban_ratio = 0.9

# Optional subnet access control. `ban_cidrs` denies ranges outright;
# `allow_only = true` flips to a strict whitelist (only allow_cidrs connect).
# ban_cidrs = ["203.0.113.0/24"]
# allow_only = false
# allow_cidrs = ["192.168.0.0/16"]
```

Les clients IPv6 sont comptés par leur /64 (depuis la v1.50). Une connexion internet avec IPv6 reçoit généralement tout un bloc d'adresses — un « /64 », la première moitié de l'adresse — et l'appareil qui se trouve derrière peut choisir n'importe quelle adresse de ce bloc. Pour qu'un même client ne puisse pas contourner les limites en changeant d'adresse, chaque limite par client ci-dessus (débits de connexion et d'authentification, l'auto-ban, le ban anti-flood de shares, `max_connections_per_ip`) compte un client IPv6 public par son /64. Un ban que vous ajoutez à la main dans **Admin → Profil → Liste de blocage d'IP** sur une adresse IPv6 bannit tout son /64 ; la liste affiche et débannit toujours l'adresse telle que vous l'avez saisie. Les adresses **unique-local** (`fc00::/7`) et **link-local** (`fe80::/10`) sont traitées comme votre propre LAN, comme les adresses IPv4 privées : comptées par adresse et jamais auto-bannies. Pour l'IPv4, rien ne change.

Pannes du nœud et vos mineurs. Si Bitcoin Core ne fournit aucun template de bloc pendant 10 minutes, le pool déconnecte tous les mineurs et refuse les nouvelles connexions jusqu'à ce que les templates

reviennent, afin que les mineurs hachent sur leur pool de secours plutôt que sur un job périmé (actif par défaut depuis la v1.45 ; `[server] node_unhealthy_disconnect` et `node_unhealthy_grace_secs`). Dites aux personnes qui minent sur votre pool de régler un pool de secours dans leur firmware.

Variante sur invitation

Combinez les paiements per-worker avec une **liste blanche d'adresses** afin que seules des adresses connues puissent miner. Un mineur dont l'adresse de paiement n'est pas répertoriée est rejeté sur les deux transports (autorisation V1 et ouverture de canal Sv2) :

```
[mining]
payout_mode = "per_worker"
address_allowlist = ["bc1qalice...", "bc1qbob..."]
```

Une `address_allowlist` vide (la valeur par défaut) laisse le pool ouvert à tous. Notez que `allow_only` / `allow_cidrs` régissent *quels réseaux* peuvent se connecter, tandis que `address_allowlist` régit *quelles adresses de paiement* peuvent miner — les deux sont indépendants.

8 · TLS

Le firmware qui ne fait confiance qu'aux autorités de certification publiques a besoin d'un vrai certificat. Les clés concernées vivent dans `[stratum]` :

```
[stratum]
stratum_tls_listen = ":3334"
tls_cert_path = "/etc/dvb-warppool/tls/cert.pem"
tls_key_path  = "/etc/dvb-warppool/tls/key.pem"
tls_reload_secs = 300
```

Laissez les deux chemins **non définis** pour utiliser le certificat auto-signé généré automatiquement par le démon — convient au firmware de classe NerdMiner/Bitaxe qui vous laisse l'épingler.

Avec le certificat auto-signé, le panneau « Comment connecter votre mineur » du pool affiche aussi une ligne **Empreinte de la clé TLS (SHA-256)**, une chaîne comme `AB:CD:…` qui identifie la clé du pool. Un firmware qui épingle un certificat auto-signé par son empreinte affiche la même valeur avant de faire confiance au pool — **WarpAxe 0.1.41 et versions ultérieures** le fait et demande au propriétaire du mineur de la confirmer —, les personnes qui minent sur votre pool peuvent donc comparer les deux valeurs. Avec un certificat signé par une CA, la ligne n'est pas affichée, car cette vérification n'est pas nécessaire.

Les renouvellements se gèrent tout seuls. Un vrai certificat Let's Encrypt est remplacé tous les deux ou trois mois, et ce qui l'a émis — Caddy, certbot, votre propre PKI — se contente de réécrire ces deux fichiers. Le pool s'en aperçoit et se met à servir le nouveau certificat de lui-même. Vous n'avez pas besoin de le redémarrer, et les mineurs connectés à ce moment-là le restent : ils conservent le certificat déjà négocié, seules les nouvelles connexions reçoivent le nouveau. `tls_reload_secs` définit la fréquence de vérification (300 secondes par défaut ; réglez-le sur `0` pour désactiver le contrôle). Si un renouvellement est surpris à moitié écrit, le pool continue de servir l'ancien certificat et réessaie à la vérification suivante, plutôt que d'abandonner le listener.

> **Stratum V2.** Si vous exécutez aussi Stratum V2, il écoute sur sa propre adresse `sv2_listen` (par défaut `:34254`). Le transport Sv2 est toujours chiffré par NOISE, il n'a donc besoin d'aucun certificat TLS séparé. Les mineurs ont en outre besoin de la clé publique d'autorité NOISE du pool pour pouvoir se connecter. Les mineurs choisissent entre un canal *standard* et un canal *étendu* dans leur propre firmware ; orientez-les vers *étendu*, qui leur permet de vérifier la coinbase qu'ils hachent. Les deux sont acceptés ici et aucun ne demande de réglage de votre côté.

> **Un seul port pour tout.** Avec `single_port = true` dans `[stratum]`, le port en clair (`[server] pool_listen`, 3333) accepte aussi Stratum V1 sur TLS et Stratum V2. Chaque connexion est reconnue à ses premiers octets et confiée au serveur correspondant, avec les mêmes limites, bannissements et compteurs que sur le port propre à ce protocole. TLS et Sv2 ont toujours besoin de leurs propres listeners configurés comme ci-dessus, et ces ports continuent de fonctionner. Derrière un équilibreur TCP, l'en-tête PROXY vient en premier et est vérifié avant que le protocole ne soit reconnu (chapitre 9). Désactivé par défaut.

9 · Derrière un reverse proxy

Lorsque l'interface de statut est servie en HTTPS via nginx ou Caddy, définissez trois clés dans `[server]` :

```
[server]
status_public_url = "https://pool.example.com"
trust_proxy_headers = true    # so per-IP limits see the real client IP
cookie_secure = true          # UI is HTTPS
# status_tls_listen = ":18443"   # or terminate TLS at the proxy instead
```

`status_public_url` permet à l'interface de construire des URL absolues pour les liens de web-push et d'e-mail. Réglez `cookie_secure = true` chaque fois que l'interface est atteinte en HTTPS.

> **Ne faites confiance qu'aux en-têtes de proxy que vous contrôlez.** Avec `trust_proxy_headers = true`, le démon fait confiance à l'en-tête `X-Forwarded-For` pour identifier la véritable IP du client. Ne l'activez **que** lorsqu'un proxy que vous contrôlez définit réellement cet en-tête — sinon un client pourrait **usurper son IP** et passer directement au travers de vos limites et bannissements par IP.

Stratum derrière un équilibreur TCP : le protocole PROXY

Les ports Stratum sont du TCP brut et passent par un équilibreur L4 (HAProxy en mode TCP, nginx `stream`), jamais par le proxy HTTP. Derrière un tel équilibreur, le pool ne verrait que l'adresse de l'équilibreur — un seul mineur fautif pourrait le faire bannir. Depuis la v1.45, l'équilibreur peut envoyer d'abord l'adresse réelle du client (protocole PROXY v1/v2, HAProxy `send-proxy-v2`, nginx `proxy_protocol on`) ; le pool ne la lit **que** depuis les adresses que vous listez, qui **doivent** alors l'envoyer :

```
[server]
pool_listen = "127.0.0.1:13333"       # daemon sur des ports internes, l'équilibreur possède
[stratum]
stratum_tls_listen = "127.0.0.1:13334"
sv2_listen = "127.0.0.1:13336"
proxy_protocol = true
proxy_protocol_trusted = ["127.0.0.1"]   # adresse(s) de l'équilibreur
```

Les limites, les bannissements, le plafond par IP et la page Workers montrent alors le mineur, pas l'équilibreur. Attention : toute connexion depuis une adresse de confiance doit porter l'en-tête — un mineur ou sidecar local qui se connecte directement aux ports du daemon est coupé ; dirigez-le vers l'équilibreur.

10 · Liste de contrôle de déploiement

Parcourez-la avant d'ouvrir le pool au public :

- ☐ `payout_mode = "per_worker"`, avec **aucun** frais de pool ni don d'opérateur (le démon démarre — il échoue en mode fermé sinon).
- ☐ Bitcoin Core accessible ; envisagez `[node] template_source = "ipc"` pour une latence plus faible (nécessite Core ≥ 31 compilé avec IPC — GBT reste la valeur par défaut).
- ☐ `[ratelimit] enabled = true` avec `autoban_enabled` et `reject_ban_enabled` activés.
- ☐ `max_connections_per_ip` défini (sauf si vos mineurs sont derrière un seul NAT).
- ☐ TLS devant Stratum (votre propre certificat pour le firmware n'acceptant que les CA publiques).
- ☐ Reverse proxy : `trust_proxy_headers` et `cookie_secure` définis correctement.
- ☐ Testez un mineur avec `your-address.rig1` ; confirmez qu'il apparaît à `/users/<address>` et que ses parts sont acceptées.
- ☐ Si vous publiez une page d'accueil, gardez les garde-fous d'honnêteté : pas de « éprouvé au combat » ; le bloc mainnet, c'est « pas encore ».
- ☐ Pied de page `[legal]` facultatif (liens conditions/vie privée) pour un déploiement public ou commercial.

11 · Support & liens

Guide en ligne	docs.warppool.org/public-pool-setup.html
Sources & tickets	git.warppool.org/dvb-projekt/dvb-WarpPool
Comparaison des fonctionnalités	docs.warppool.org/comparison.html
Contact	dvbprojekt@gmx.de

dvb-WarpPool est un logiciel libre et open-source (AGPL-3.0-or-later, avec une licence commerciale disponible). Ce manuel décrit la Version 1.52.0. Les écrans peuvent différer légèrement d'une version à l'autre. Le minage solo est par nature probabiliste ; rien ici n'est une promesse de récompense. Un pool public 0 % est non-custodial — le pool ne détient jamais les fonds de quiconque, et celui qui trouve est payé la récompense complète directement par le réseau Bitcoin.

Ouvert, non-custodial, sans frais — et que celui qui trouve garde tout.