SETUP GUIDE

Linux Security Setup Guide for Web3 Teams

A self-service baseline that works without device management

Prepared for	Web3 team members	Classification	Internal
Prepared by	Oak Security	Date	2026-06-17

Oak Security GmbH

1. What This Guide Is For

This guide is for anyone in a Web3 team who uses a Linux desktop or laptop for work, whether you are a developer, an operator, or in a non-technical role. Most of the setup is designed to work **without centralized device management (MDM)**: you should be able to apply the baseline yourself, without turning your machine into a centrally managed device. Section 2 describes lightweight management options for teams that decide they want them.

The aim is to keep a normal laptop problem, such as a bad browser extension, a suspicious download, a stolen session, or an unlocked screen, from turning into a company-wide incident. It is not to make every machine identical.

> **NOTE**
> **Baseline rule:** any machine used for work should be encrypted, updated, locked when unattended, separated from risky personal browsing, and kept free of unnecessary extensions and local secrets.

For the most sensitive work, such as production administration, cloud/KMS access, DNS or registrar changes, treasury signing, source-control organization administration, and release approvals, a dedicated hardened work device is still the better option.

Note on distributions: Linux varies. Commands below assume a mainstream desktop (Ubuntu/Debian or Fedora/RHEL with GNOME). On other distributions or desktop environments, the concept is the same even if the exact command, package name, or settings path differs. Use your distribution's documentation when in doubt.

2. Lightweight Management Options for Company Machines

This guide is deliberately self-service friendly. It does not require MDM, and it should not be used as a reason to manage personal devices without a clear company decision.

If your team starts issuing company-owned machines, the same baseline can move from "please configure this yourself" to "the company enforces this by default." Linux endpoint management is less mature than on macOS or Windows, so most small teams lean on configuration management and golden images rather than a classic MDM.

Practical Options

Option	Best Fit	Notes
Configuration management (Ansible, Salt)	Team comfortable with infrastructure-as-code that wants reproducible, reviewable baselines.	The most common approach for Linux fleets. Apply hardening as code and track drift.
osquery plus a fleet manager (Fleet, Kolide)	Team that wants visibility and compliance checks rather than heavy enforcement.	Strong for inventory and posture reporting; lighter on enforcement.
Microsoft Intune for Linux	Team already on Entra ID that needs basic compliance and conditional access on supported distros.	Linux coverage is limited compared to Windows or macOS. Verify your distribution is supported.

Standardization Without Full MDM

For a distributed team, you can still distribute consistent settings:

- **Ansible playbooks** or **Salt states** that apply the baseline and can be re-run to correct drift. The closest analog to a managed configuration profile.
- **dconf system profiles** to set and lock GNOME settings (screen lock, sharing) machine-wide.

- **Golden images / preseed / kickstart / cloud-init** so new machines start from an encrypted, hardened baseline.

Treat these as standardization, not full management. They do not replace MDM for inventory, remote wipe, recovery-key escrow, forced updates, app deployment, or reliable offboarding.

A Sensible First Management Rollout

Start with company-owned devices for high-risk users only:

- head of engineering;
- production administrators;
- release owners;
- treasury signers;
- DNS, registrar, source-control organization, hosting, and cloud/KMS administrators.

For those devices, enforce only the controls that reduce the most risk:

- LUKS full-disk encryption, with the passphrase or recovery material escrowed;
- automatic security updates;
- screen lock and password requirement;
- host firewall on (default deny incoming);
- unused services and sharing off;
- basic package and extension inventory;
- a remote wipe or rebuild path for lost or compromised devices;
- clear ownership and offboarding.

Keep the self-check below for personal devices. Use management for company machines where the company has a legitimate need to enforce and recover the device.

3. Quick Setup Checklist

These are the settings every work machine should have before it is used for work.

System Settings

Setting	What To Do	Where To Check
Updates	Install pending updates. Enable automatic security updates (`unattended-upgrades` on Debian/Ubuntu, `dnf-automatic` on Fedora/RHEL).	Your package manager or GNOME Software
Encryption	Use LUKS full-disk encryption. This is set at install time; if the disk is not encrypted, plan a reinstall with LUKS. Store the passphrase in your password manager.	Set during OS installation; verify with `lsblk` and `cryptsetup status`
Screen lock	Require a password after the screen blanks, and lock on suspend. Use a short blank-screen delay.	Settings > Privacy > Screen Lock (GNOME)
Firewall	Enable the host firewall with a default-deny inbound policy (`ufw` on Ubuntu, `firewalld` on Fedora/RHEL).	`sudo ufw status` or `sudo firewall-cmd --state`
Services and sharing	Disable services you do not use: SSH server, Samba, VNC/RDP, Avahi, printer sharing. Check what is listening.	`systemctl list-unit-files --state=enabled`, `ss -tulpn`

Work Separation

Area	What To Do
Browser profile	Create a dedicated browser profile for work. Use it for email/workspace, source control, chat, docs,

	hosting, cloud, registrar access, and the password manager.
Personal browsing	Keep personal browsing, airdrops, trading, wallet testing, and random web apps out of the work profile.
Wallet activity	Do not mix personal wallet activity with production, admin, or company SaaS sessions. Use a separate profile or browser.
Extensions	Remove extensions you do not need. Do not install wallet, coupon, scraping, AI helper, or unknown extensions in the work profile.

Credentials, Keyrings, and Local Data

Area	What To Do
Passwords	Store work credentials in the team-approved password manager. Do not store them in plain text files, screenshots, chat, shell history, or local notes.
Keyring / browser sync	The login keyring (GNOME Keyring, KWallet) usually unlocks with your login password. Browser account sync can push passwords and passkeys to personal devices. Use them for work credentials only when the team accepts that model.
MFA	Prefer passkeys where supported. Use FIDO2 hardware security keys for critical accounts, and consider `pam-u2f` for login or `sudo`.
Local secrets	Check `~/Downloads`, `~/Desktop`, screenshots, shell history, dotfiles, `.env` files, `~/.ssh`, `~/.aws`, `~/.config`, `~/.kube`, `~/.docker/config.json`, `~/.netrc`, and GnuPG keys. Remove anything that should not be local.
SSH keys	Use a separate SSH key for work. Do not sync private SSH keys through Nextcloud, Dropbox, Google Drive, or chat.

Backups	Use encrypted backups (for example `restic` or `borg` with encryption). Do not back up seed phrases, plaintext secrets, private keys, or sensitive screenshots into uncontrolled personal cloud storage.

4. Password Managers, Passkeys, and Hardware Keys

Use different storage models for different risk levels. A synced password manager is a good default for normal work credentials. It is not the same thing as a hardware-bound key.

Storage Model	Use For	Caveat
Team-approved password manager	Normal work passwords, recovery codes, low-risk SaaS credentials, and shared team credentials.	Good organizational control, but synced vault secrets are still usable if the vault and endpoint are compromised.
Login keyring (GNOME Keyring, KWallet) / browser sync	Personal credentials, and low-risk work credentials if the team accepts the model.	The login keyring typically unlocks with your session password; browser sync may push credentials and passkeys to personal devices.
FIDO2 hardware security key	Critical accounts such as email/workspace, source-control admin, cloud, hosting, registrar, and password-manager admin.	Less convenient, but portable and bound to a physical authenticator. Can also protect login/`sudo` via `pam_u2f`.
Hardware-backed SSH key (FIDO2 `sk-ed25519`)	Production or sensitive repository and infrastructure access.	Requires setup discipline and backup keys, but avoids storing reusable private keys in files or synced vaults.

Practical Policy

- Use the team-approved password manager for normal work credentials.
- Use FIDO2 hardware keys for critical accounts.
- Do not treat synced password-manager passkeys as equivalent to hardware-bound credentials.
- Do not store SSH private keys, production secrets, API tokens, wallet seed phrases, or critical recovery codes in plaintext files, the login keyring, screenshots, or local notes.

- If browser sync or a synced keyring holds a work credential, understand that every synced device becomes part of the access surface.

5. Keyrings, Browser Sync, and Personal Cloud

The login keyring, browser account sync, and personal cloud folders are convenient and generally well-designed for personal use. In a work context they can sync or expose work credentials, passkeys, and files in ways the company does not know about and cannot clean up during offboarding or an incident.

Recommended Approach

Topic	Guidance
Browser password sync	Use it for personal credentials. For work credentials, use the team-approved password manager by default.
Browser account sync	Treat sync as a path to every device signed in to the same personal browser account. If a work credential or passkey is stored there, every synced device becomes part of the access surface. Prefer a separate work profile that is not signed in to a personal account.
Passkeys	Passkeys are good, but synced passkeys are still credentials. For critical accounts, prefer FIDO2 hardware keys.
Login keyring	It is normal for apps to store some tokens in the keyring. Remember it often unlocks automatically with your login. Do not manually store production secrets, seed phrases, API keys, KMS material, recovery keys, or SSH private keys there for convenience.
Personal cloud sync	Do not keep work secrets or confidential files in personal Nextcloud, Dropbox, or Google Drive folders, including synced Desktop and Documents directories. Use the approved company storage location.

If you use browser sync or a synced keyring for any work credential, make sure:

- The associated account has a strong unique password and MFA.
- No one else uses that account or your local login.
- Every synced device is encrypted, updated, and locked.
- Old or unused devices have been removed from the account.
- You understand that incident cleanup may require reviewing every synced device, not only the machine used for work.

6. SSH Keys and Developer Credentials

SSH keys, access tokens, cloud credentials, and API keys deserve separate handling. They are often more useful to an attacker than a normal website password.

Baseline Rules

- Use a separate SSH key for work. Keys live in `~/.ssh`; keep permissions tight (`700` on the directory, `600` on private keys).
- Prefer modern key types such as ed25519 unless a service requires something else.
- Protect file-based private keys with a passphrase, and use `ssh-agent` (or the keyring agent) rather than unprotected keys.
- For production or sensitive infrastructure access, prefer hardware-backed SSH keys (FIDO2 `sk-ed25519`).
- Do not store private keys in Nextcloud, Dropbox, Google Drive, chat, docs, screenshots, or email.
- Do not share private keys. Share public keys only.
- Do not reuse personal SSH keys for work infrastructure or repositories.
- Avoid SSH agent forwarding unless there is a clear need; prefer `ProxyJump`.
- Remove old keys from source-control accounts and servers when they are no longer needed.
- Rotate SSH keys and tokens after suspicious device activity.

Suggested Developer Self-Check

Check	Pass Condition
Separate key	Work uses a dedicated SSH key.
Permissions	`~/.ssh` and private keys have correct restrictive permissions.
Private key storage	Private key is not stored in any synced folder or chat history.

Passphrase	File-based private key has a passphrase.
Hardware-backed key	Critical infrastructure access uses a hardware-backed SSH key where practical.
Source-control keys	Your source-control account has no stale or unknown SSH keys.
Tokens	Access tokens and API keys have clear purpose, limited scope, and expiry where possible.

7. The Recommended Setup Flow

Step 1: Update the System

Install pending updates through your package manager or GNOME Software, and enable automatic security updates (`unattended-upgrades` on Debian/Ubuntu, `dnf-automatic` on Fedora/RHEL).

Do not sit on security updates for weeks. If an update has to be delayed because it may break a tool, say so in the team channel and set a date to revisit it. Watch for kernel updates that require a reboot to take effect.

Step 2: Confirm Full-Disk Encryption

Verify the system disk is encrypted with LUKS (`lsblk` should show a `crypt` device; `sudo cryptsetup status` confirms it).

LUKS is normally enabled at install time. If the disk is not encrypted, plan a reinstall with encryption rather than relying on a partial fix. Store the passphrase in your password manager. Do not save it as a screenshot, local file, or chat message on the same machine. A recovery passphrase stored on the device it unlocks provides no recovery.

Step 3: Lock the Device Properly

Open **Settings > Privacy > Screen Lock** (GNOME) or the equivalent for your desktop.

- Require a password after the screen blanks.
- Lock automatically on suspend.
- Use a short blank-screen delay.
- Lock the screen manually with **Super+L** before stepping away.

Step 4: Enable the Firewall and Turn Off Services

Enable the host firewall with a default-deny inbound policy:

- Ubuntu/Debian: `sudo ufw default deny incoming` then `sudo ufw enable`.

- Fedora/RHEL: ensure `firewalld` is running and use a restrictive zone.

Then disable services you do not actively use. Check what is enabled and listening with `systemctl list-unit-files --state=enabled` and `ss -tulpn`, and turn off:

- SSH server (`sshd`) if you do not accept inbound SSH
- Samba / NFS file sharing
- VNC / RDP remote desktop
- Avahi / mDNS
- printer sharing (CUPS remote access)

If you need one of these for a real workflow, document it and turn it off when you no longer need it.

Step 5: Separate Work from Personal Browsing

Create a dedicated browser profile for work. This is one of the highest-value controls in this setup.

Use the work profile for company tools: email/workspace, source control, chat, docs, hosting, cloud, registrar access, password manager, and similar services.

Do not use that profile for personal browsing, crypto experiments, unknown dapps, airdrops, trading, wallet testing, or random extensions. Those activities should live in a separate profile or browser.

Step 6: Clean Up Extensions

Review extensions in every browser used for work.

Remove anything you do not actively need. Be especially careful with extensions that can read or change page content, inspect traffic, manage downloads, automate browser activity, or interact with wallets.

Do not install extensions from search ads, social links, chat links, or "quick fix" instructions. If you need a new extension for work, check the publisher, permissions, reviews, and installation source before installing it. The same caution applies to GNOME Shell extensions and unfamiliar APT/DNF repositories or PPAs.

Step 7: Secure Accounts

For work-related accounts:

- Use the password manager.
- Use unique passwords.
- Prefer passkeys where supported.
- Use FIDO2 hardware keys for critical accounts.
- Review recovery email and phone settings.
- Remove stale sessions and unknown devices.

Critical accounts include email/workspace, source control, chat, password manager, hosting, cloud, registrar, docs, treasury tooling, and wallet-related services.

Step 8: Remove Local Secrets

Machines collect sensitive material over time. Check the obvious places first:

- `~/Downloads`
- `~/Desktop`
- Screenshots
- Shell history (`~/.bash_history`, `~/.zsh_history`)
- Dotfiles
- `.env` files
- SSH private keys (`~/.ssh`)
- Cloud and tooling configs (`~/.aws`, `~/.kube`, `~/.docker/config.json`, `~/.netrc`)
- Access tokens
- API keys
- Wallet seed phrases or private keys
- Keyring entries that are no longer needed

Production secrets should not live on a work laptop unless there is a documented reason. If a secret has been shared through chat or stored locally for convenience, treat that as something to clean up, not as normal practice.

8. Extra Rules for Critical-Access Users

Some people have access where a laptop compromise could become a company-level incident. This includes production administrators, release owners, DNS or registrar admins, treasury signers, source-control organization admins, and anyone with cloud/KMS or signing authority.

If this applies to you, use the stricter profile below.

Control	Requirement
Dedicated work context	Use a dedicated work device where practical. If that is not possible, use a dedicated local user account or browser profile for privileged work. Highly targeted users may consider a compartmentalized OS such as Qubes.
Hardware-backed MFA	Use FIDO2 hardware keys for critical accounts where supported, and consider `pam-u2f` for login and `sudo`.
Sync hygiene	Do not store critical credentials or synced passkeys in browser sync or a synced keyring unless explicitly accepted by the team.
Password manager	Use the team-approved password manager for normal work credentials, but do not treat synced vault passkeys as hardware-bound credentials.
Browser extensions	No unapproved extensions in the production or admin browser profile.
SSH and tokens	Keep SSH keys and API tokens scoped, separate from personal use, and out of synced folders. Prefer hardware-backed SSH keys for sensitive access.
Wallet separation	Do not combine personal wallet activity with production or admin browser sessions.
Remote access	Keep `sshd`, VNC, and RDP disabled unless

	explicitly needed.
Travel	Do not travel with all critical devices, signing devices, or recovery material together.
Suspicious activity	Rotate relevant credentials and tokens before resuming privileged work.
Attack-surface reduction	Keep Secure Boot on, keep SELinux or AppArmor in enforcing mode, install minimal software, and avoid disabling kernel hardening for convenience.

9. Things To Avoid

These habits are common, but they create unnecessary risk.

- Installing browser extensions casually.
- Installing wallet extensions from search results, ads, social posts, or chat links.
- Keeping company credentials in plain text files, screenshots, local notes, chat, shell history, or personal cloud folders.
- Using browser sync or a synced keyring for critical work credentials without team agreement.
- Treating synced password-manager passkeys as equivalent to hardware-bound credentials.
- Syncing private SSH keys through Nextcloud, Dropbox, Google Drive, or similar services.
- Using the same browser profile for personal wallet activity and production or admin work.
- Leaving `sshd`, VNC, RDP, or Samba enabled without a clear need.
- Running shell commands pasted from strangers, support chats, Telegram, Discord, social media, or search results. Be especially wary of `curl ... | sudo bash` and "just run this one-liner" instructions.
- Adding unknown APT/DNF repositories or PPAs, or running `sudo` without understanding what a command does.
- Disabling SELinux, AppArmor, or the firewall for convenience.
- Treating a suspicious personal wallet event as unrelated to work when the same device is used for company access.

10. If Something Suspicious Happens

For a serious suspected compromise, do not try to "clean up" first. The safer sequence is: isolate the device, preserve evidence, and use a clean device to revoke access.

Immediate Actions

1. Stop using the device for work.
2. Disconnect the device from the network: turn off Wi-Fi and unplug Ethernet.
3. Keep the device powered on and locked if you can do so safely. Do not keep working on it.
4. Do not delete files, clear logs, uninstall tools, or "clean up" before someone has decided whether evidence matters.
5. Use a different trusted device to notify the security owner or head of engineering.

Powering off is acceptable if you cannot isolate the device, if you believe it is actively causing harm, or if you are dealing with theft, seizure, or personal safety. Otherwise, leave it powered on and disconnected so there is a better chance of understanding what happened.

Revoke Access from a Clean Device

From a different trusted device, force sign-out and revoke sessions for accounts used on the suspect machine.

Account Type	Action
Email / workspace	Sign out all sessions, review trusted devices, rotate password if needed, review OAuth apps.
Source control	Revoke sessions, remove unknown SSH keys, revoke personal access tokens, rotate affected tokens.
Password manager	Sign out other sessions, review device list, rotate exposed vault items.
Chat / docs / hosting / cloud / registrar	Revoke sessions, review connected apps, rotate credentials where available.

Cloud / KMS-related access	Revoke sessions, rotate keys, review IAM users, roles, access keys, and recent activity.
SSH	Remove affected public keys from servers and source-control accounts, then generate new keys from a clean device.
Wallets / treasury tooling	Treat as high risk. Stop signing until the device and credentials have been reviewed.

Rotate Credentials

Rotate anything that may have been exposed:

- Passwords used on the device.
- Passkeys or security keys registered from the device, if compromise is plausible.
- Personal access tokens.
- SSH keys.
- API keys.
- Cloud access keys.
- Wallet-related credentials.
- Recovery codes stored locally or in browser/keyring sync.

If the password manager was unlocked during the incident, assume more exposure and rotate the most sensitive credentials first.

Decide Whether to Rebuild

If compromise cannot be ruled out, rebuild the device before using it for work again. A rebuild should include:

- Wipe and reinstall the OS with LUKS encryption.
- Reapply this baseline.
- Reinstall only required software.
- Recreate SSH keys and tokens from a clean state.
- Reconnect accounts only after old sessions have been revoked.

If you have critical access, do not resume production, admin, treasury, DNS, or release work from the device until the security owner agrees it is safe.

11. Monthly Self-Check

Once a month, do this short review. It should take a few minutes.

Check	Done
The system is up to date and automatic security updates are on.	
LUKS encryption is active and the passphrase is stored safely.	
The firewall is on with default-deny inbound.	
Unused services and sharing (`sshd`, Samba, VNC/RDP) are off.	
Screen lock is required after a short delay and on suspend.	
The work browser profile is separate from personal browsing.	
Browser and GNOME Shell extensions have been reviewed.	
Critical accounts use FIDO2 hardware keys where supported.	
Normal work credentials are in the team-approved password manager.	
Browser sync and keyring use has been reviewed for work credentials.	
SSH keys, access tokens, and API keys have been reviewed.	
No company secrets are sitting in obvious local or synced-cloud locations.	
Stale sessions, devices, and tokens have been reviewed.	

12. References

- CIS Benchmarks (Ubuntu Linux, Red Hat Enterprise Linux, Debian): https://www.cisecurity.org/cis-benchmarks
- ANSSI Configuration Recommendations of a GNU/Linux System: https://cyber.gouv.fr/en/publications/configuration-recommendations-gnulinux-system
- Ubuntu Security: https://ubuntu.com/security
- LUKS / cryptsetup: https://gitlab.com/cryptsetup/cryptsetup/-/wikis/home
- ufw (Uncomplicated Firewall): https://help.ubuntu.com/community/UFW
- firewalld: https://firewalld.org/documentation/
- OpenSSH manual: https://www.openssh.com/manual.html
- GNOME Keyring: https://wiki.gnome.org/Projects/GnomeKeyring
- pam-u2f (hardware-key login): https://developers.yubico.com/pam-u2f/
- Microsoft Intune for Linux: https://learn.microsoft.com/en-us/mem/intune/fundamentals/deployment-guide-platform-linux
- Fleet (osquery management): https://fleetdm.com/