Security reset checklist.

Start with the account that resets your other accounts. Then work down the list. Tick a box when you have checked the setting or completed the action.

01 / START HERE · YOUR MAIN EMAIL

- [ ] **Open your email provider's security settings.**
  Use its app or type its address yourself. Add a passkey, or turn on multi-factor authentication (MFA). Check your recovery email and phone; remove anything you do not recognise. Save backup codes somewhere safe, separate from the device you might lose.

02 / EVERYONE · WORK THROUGH THESE NEXT

- [ ] **Install updates. Finish the restart.**
  Check your phone, computer, browser and apps. Enable automatic security updates where available; replace software that no longer receives security fixes.

- [ ] **Let a password manager create unique passwords.**
  Protect the manager with a strong, unique master password and MFA. Replace reused passwords, starting with email, banking and other important accounts.

- [ ] **Add passkeys or MFA to important accounts.**
  Prefer passkeys or security keys where supported; otherwise use an authenticator app. SMS is still better than password-only sign-in. Never approve an unexpected login prompt.

- [ ] **Check how you would get back in.**
  Review recovery methods and backup codes for key accounts. Add a second supported sign-in method where possible. Check the provider's recovery steps before changing phones.

- [ ] **Replace exposed credentials; review account access.**
  Change a leaked password and every account where you reused it. Review active sessions and connected apps; sign out or revoke anything unfamiliar. Use the provider's account-recovery process if you are locked out.

These steps reduce common risks. They cannot guarantee complete protection. For a work account or device, follow your organisation's process and report suspected compromise to its IT or security team.

Build things? Protect the keys.

Use this addendum if you run apps, deploy code or connect AI tools to API accounts. If you do not, skip to the commitment below.

03 / API ACCOUNTS, CODE & DEPLOYMENTS

- [ ] **Find the accounts and services you actually run.**
  List cloud, hosting, source control and AI providers. Apply security fixes to deployed apps, runtimes and dependencies; test changes and confirm the fixed version is live.

- [ ] **Move secrets into managed secret storage.**
  Keep API keys and tokens out of source code, browser bundles, logs, chat and screenshots. Use your host's secret store or a secrets manager, injected at runtime. Keep local secret files out of version control.

- [ ] **Limit what each key can do.**
  Use separate credentials for each app and environment. Grant only needed permissions, remove unused tokens, and set usage alerts and provider-supported limits. Do not assume a budget alert stops spending.

- [ ] **If a secret leaks, revoke or rotate it promptly.**
  Follow the provider's procedure: replace the secret in every dependent service, verify the change and revoke the old value. If abuse is ongoing, revoke immediately even if service is interrupted. Review access and usage logs.

- [ ] **Clean up the leak and prevent a repeat.**
  Deleting a file or commit does not invalidate a secret. After revocation, remove exposed values from code and logs; handle repository history as needed. Enable secret scanning or push protection where available.

Make the next step specific.

When I _____ (a routine or time)

I will open _____ (account or device)

and complete _____.

Example: After lunch today, I will open my main email's security settings and add a passkey. Write the action here; never write passwords, keys or recovery codes.

Trusted guidance · keep this sheet, use the live instructions

CISA — Secure Our World · updates, passwords and MFA
cisa.gov/secure-our-world

UK NCSC — Top tips for staying secure online · practical account security
ncsc.gov.uk/collection/top-tips-for-staying-secure-online

GitHub Docs — Removing sensitive data from a repository · revoke first, then clean up
docs.github.com → Authentication → Removing sensitive data from a repository