Security Tips for Startups

Some low-hanging fruit to improve your software and IT security for free

Two/Multi-Factor Authentication (“2FA”, “MFA”)

Require 2FA with {Google Authenticator, YubiKey, etc} for:

- (Required) Infrastructure root accounts: AWS, GCP, Github admin, Google Apps admin
- (Required) Dev accounts with access to customer or other critical data
- (Strongly recommended): All employee accounts for Github, Google Apps, or equivalents

Why

- Prevents immediate account takeover if password is compromised, allows time to reset password in this case

Encrypt all laptops

During laptop setup for all employees, enable hard disk encryption and explain to the new employee(s) that they need a unique, reasonably secure login password for their laptop

Why

- Laptops are frequently stolen. Without disk encryption, the data on disk can be easily read by a reasonably determined hacker by mounting the drive on a different computer. Encryption gives you time to revoke the employees credentials, and some peace-of-mind that the data probably won’t be read.

No important secrets in source control (git(hub), etc)

Write a quick policy for engineers to read as part of onboarding that includes “no secrets checked into source control” and instructions on how to manage secrets (options: environment variables, encrypt the secrets before checking into source control, Hashicorp Vault, etc).

Why

- Prevents a compromise of your Github (or other source management) account from giving attacker your secrets

No secrets in Slack/Hipchat, Email, etc

Instead, share credentials via:

- PGP encrypt, then email (keep team members' public PGP keys on wiki)
- Airdrop (protocol works over SSL [source])
- Other encrypted messaging solution
- Drop on secured server with shared SSH access

Why

- Prevents your secrets from being compromised if the logging systems of these services are compromised; prevents your secrets from being read out of an email as the mail passes through various SMTP servers; prevents a historical log of these secrets if an account is compromised

Unique passwords: use password managers

Require engineers to use a password manager and generate unique passwords for each critical account. Recommend your favorite (and pay for it if it’s not free).

Why

- Prevents compromise of single password from turning into compromise of many accounts

Other tips

- Hash your user's passwords and secrets using the latest and greatest schemes: Password Storage Cheat Sheet
- Use SSL everywhere
- Be careful of monitoring and logging systems, that may accidentally send secrets in your runtime application environment to third-party systems
- Create a blameless culture of continuous learning! Everyone makes mistakes, credentials can be rotated. Many developers from larger environments are accustomed to dedicated security teams handling a lot of these concerns: in a startup, everyone is a bit of an "infrastructure engineer" so it's worthwhile to get people talking about them and asking questions.