

Mac App Store submission â Step-by-step

This file contains exact commands and actions to build, sign, package and upload the macOS app to the Mac App Store.

Prerequisites

- A Mac with Xcode and Xcode command-line tools installed.
- Rust toolchain with cargo and rustup.
- An Apple Developer Program account and App Store Connect access.
- The repository contains helper scripts: scripts/build_macos_app.sh, scripts/make_icns.sh, scripts/p12_to_base64.sh

Summary of steps

- 1) Create app icon (.icns) from icon.iconset (macOS)
- 2) Export Apple certificates (.p12) from Keychain Access (Apple Distribution and Installer certs)
- 3) Convert .p12 files to base64 for GitHub secrets
- 4) (Local) Build, sign, and package the app locally (recommended to test)
- 5) Verify codesign and Gatekeeper behavior locally
- 6) Upload App Store Connect API key and certificate secrets to GitHub
- 7) Trigger CI workflow to build & upload automatically
- 8) Finalize metadata and submit for review in App Store Connect

Exact commands (copy/paste)

```
# 1) Make .icns from icon.iconset (on macOS)
./scripts/make_icns.sh packaging/macOS/icon.iconset packaging/macOS/resources/app.icns
ls -l packaging/macOS/resources/app.icns

# 2) Export certificates from Keychain Access (GUI recommended) or use security CLI
# GUI: Keychain Access -> select certificate -> Right-click -> Export -> choose .p12
# CLI example (replace identity and output path):
# security export -k ~/Library/Keychains/login.keychain-db -t identities -f pkcs12 -o ~/Desktop/apple_dist.p12
# security export -k ~/Library/Keychains/installer.keychain-db -t identities -f pkcs12 -o ~/Desktop/installer.p12

# 3) Convert .p12 to base64 for GitHub secrets
./scripts/p12_to_base64.sh ~/Desktop/apple_dist.p12 > apple_dist_b64.txt
./scripts/p12_to_base64.sh ~/Desktop/installer.p12 > installer_b64.txt

# 4) Local build & package (macOS)
rustup target add x86_64-apple-darwin aarch64-apple-darwin
export APP_NAME="Time Warp"
export EXECUTABLE_NAME="time-warp"
export BUNDLE_ID="org.honeybadger.timewarp"
export VERSION="3.0.0"
export BUILD_VERSION="300"
export SIGN_IDENTITY="Apple Distribution: Your Name (TEAMID)"
export INSTALLER_IDENTITY="3rd Party Mac Developer Installer: Your Name (TEAMID)"
bash scripts/build_macos_app.sh

# 5) Verify signing and Gatekeeper
codesign --verify --deep --strict --verbose=2 "dist/${APP_NAME}.app" || true
spctl -a -t install -v "dist/${APP_NAME}.app" || true
spctl -a -t install -v "dist/${APP_NAME}-${VERSION}.pkg" || true

# 6) Upload GitHub Secrets (using gh CLI)
# gh auth login --username="Your GitHub Username" --password="Your GitHub Password"
```