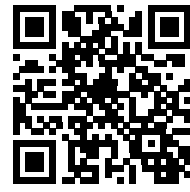




Download on the
App Store



 Website



NEW: Label AI images
EU AI Act Art. 50

Stego-Lab

User Manual & Technical Documentation

iOS · iPadOS · macOS

Steganography · Watermarking · EXIF · Batch · AI Labelling

Property	Value
Product	Stego-Lab
Platforms	iOS 17+ · iPadOS 17+ · macOS 14+
Version	1.1 · built on 2026-09-22, 14:19
Manual language	English
Privacy	100 % local — no cloud
GitLab	gitlab.com/RaithChr/stego-lab

Table of Contents

1 Introduction	5
1.1 What is Stego-Lab?	5
1.2 Use Cases	5
1.3 Key Differentiators	5
2 Installation & Getting Started	7
2.1 System Requirements	7
2.2 Installation	7
2.3 First Launch	8
3 User Interface	9
3.1 Main Navigation	9
3.2 Encode Area	9
3.3 Decode Area	10
4 Steganography	11
4.1 Fundamentals	11
4.2 LSB Method	11
4.3 Embedding Content	11
4.4 Password Protection	12
4.5 Decoding & Capacity	12
5 Watermarking	14
5.1 Visible Watermarks	14
5.2 Invisible & Combined Watermarks	14
5.3 Positioning & Batch	14
6 EXIF & Metadata Editing	16
6.1 EXIF Fundamentals	16
6.2 Readable Fields	16
6.3 Editable Fields	16
6.4 GPS Data	17
6.5 Delete All Metadata	18
7 Batch Processing & Supported Formats	19
7 Batch Processing	19
8 Supported Formats	19
9 Labelling AI Images	22
9.1 Legal Framework	22
9.2 Labelling in Encode and Batch	22
9.3 Fields Written	22
9.4 Detection on Read	23

9.5 Free vs. Pro	23
9.6 Limitations	23
9.7 Legal Notice	23
10 Security & Privacy	24
10.1 Local Processing	24
10.2 Encryption Details	24
10.3 Steganographic Security	24
11 Technical Specifications	26
11.1 Performance Data	26
11.2 Memory Requirements	26
11.3 Algorithms & Standards	26
12 Use Case Examples	28
12.1 Private Document Protection (Pro)	28
12.2 Copyright Proof (Pro)	28
12.3 Photographer Workflow, 500 images (Pro)	28
12.4 Educational Context (Pro)	29
13 FAQ	30
14 Technical Appendix	32
14.1 LSB Encode Process	32
14.2 Internal Frame Format	32
14.3 Error Handling	33
15 Glossary	35
16 Market Positioning	37
16.1 Feature Comparison	37
16.2 Target Audiences	37
17 Workflow Reference	38
17.1 Encode Workflow (Pro)	38
17.2 EXIF Cleanup	38
17.3 Batch for Social Media (Pro)	38
17.4 Label an AI Image	39
18 Security Assessment	40
18.1 Threat Model	40
18.2 Password Recommendations	40
18.3 Internal Pre-Release Security Review	40
19 Support & Contact	42
19.1 Support Channels	42
19.2 Known Limitations	42

20 List of Tables	43
21 Version History	45
Planned Features	46
22 In-App Purchase Stego-Lab Pro	47
22.1 Free and Pro at a Glance	47
22.2 Purchase and Restore	47
23 About the Developer / Imprint	48
23.1 Developer Details	48
23.2 Other Projects by the Developer	48
23.3 GitLab Repository	49
23.4 Website	49
24 Legal & Privacy	50
24.1 Privacy Policy (Summary)	50
24.2 License and Terms of Use	50
24.3 Frameworks and Third-Party Components	51
24.4 Copyright & Trademarks	52

Introduction

CHAPTER 1

1.1 What is Stego-Lab?

Stego-Lab is a native app for iPhone, iPad, and Mac that combines five tool areas: steganography, watermarking, EXIF/metadata editing, labelling of AI-generated images, and batch image processing. All image data is processed exclusively on the device — no cloud, no external servers. The app itself opens no network connections; the only exceptions are the in-app purchase through Apple’s StoreKit and links that you choose to open in the browser.

1.2 Use Cases

Domain	Description
Private archiving	Embed sensitive documents invisibly in photos
Copyright protection	Embed an invisible author notice (text, image or file) — not a cryptographic signature; does not survive JPEG recompression (Pro)
Photography	Clean EXIF, remove GPS, set copyright
Photo series	Batch watermarks on entire photo collections
Journalism	Embed evidence material encrypted
Education and teaching	Demonstrate steganography concepts in practice
Content creators	Copyright watermarks for social media
IT security	Investigate covert image channels

1.3 Key Differentiators

Feature	Description
Fully local	No cloud, no telemetry; the app itself opens no network connection (exception: in-app purchase through StoreKit)
Cross-platform	iPhone, iPad, Mac — same data format (PV2) and same functional core; operation, file selection and some extra functions differ by platform
Combined tools	Stego + watermark + EXIF + AI labelling + batch in one app
AES-256 encryption	Optional password protection (AES-256-GCM, Pro) for hidden text, images and files; the format header and the invisible AI label deliberately remain unencrypted

Feature	Description
Lossless output	PNG + TIFF for maximum data integrity

Installation & Getting Started

CHAPTER 2

2.1 System Requirements

Platform	Minimum	Recommended	Architecture
iPhone	iOS 17.0	Latest iOS version maintained by Apple	ARM64 (A12 or newer)
iPad	iPadOS 17.0	Latest iPadOS version maintained by Apple	ARM64 (A10 or newer)
Mac	macOS 14.0	Latest macOS version maintained by Apple	Apple Silicon + Intel
Storage	Free space for the output files (PNG/TIFF are larger than JPEG)	—	—
RAM	No technical minimum	4 GB+ (iPhone/iPad) or 8 GB+ (Mac) for large images	—
Network	Not required	Only for the in-app purchase	Offline

▲ **No longer supported: iOS 16 / iPadOS 16 and macOS 13 (Ventura).** As of version 1.1, Stego-Lab requires at least iOS 17 / iPadOS 17 or macOS 14. The reason is the switch of the development environment to Xcode 27: since then the app uses system interfaces that only exist from these versions on. In addition, Apple no longer ships security updates for macOS 13. Devices that cannot go beyond iOS 16 or macOS 13 can keep using version 1.0 but receive no further app updates. To protect confidential data, use the newest system version your device supports, with all available security updates.

2.2 Installation

Step	iOS / iPadOS	macOS
1	Open the App Store	Open the Mac App Store
2	Search for “Stego-Lab”	Search for “Stego-Lab”
3	Tap “Get”	Click “Get”
4	Confirm with Face ID / Touch ID	Confirm with your Apple Account
5	Launch the app from the Home Screen	Launch the app from the Applications folder (up to macOS 15 also via Launchpad, from macOS 26 via “Apps”)

✓ Stego-Lab requires no registration and no account. All features work offline; only purchasing and restoring Stego-Lab Pro require a connection to the App Store.

2.3 First Launch

Stego-Lab asks for no permission on first launch. On iPhone and iPad, images are chosen with the system photo picker, which needs no access to the whole photo library; only when you first save to Photos does iOS ask for permission to add photos. Alternatively, images can be loaded from the Files app. On Mac, images are opened via Finder (Open dialog or drag and drop). Camera access is requested only at the first direct capture (Pro). The app's user interface is in German; the tab and button names quoted in this manual are the original labels.

User Interface

CHAPTER 3

3.1 Main Navigation

Area	Symbol (iPhone/iPad · Mac)	Function
“Encode” (Mac: “Verstecken”)	arrow.up.square · eye.slash.fill	Embed content into images
“Decode” (Mac: “Entdecken”)	arrow.down.square · eye.fill	Extract embedded content
“EXIF”	info.square · tag	View and edit metadata
“Batch”	square.stack.3d.up.fill · wand.and.stars	Process multiple images (Pro)
“Info”	info.circle · info.circle	Help, Pro window and app details. On Mac the tab order is Verstecken, Entdecken, Batch, EXIF, Info

■ The user interface of Stego-Lab 1.1 is in German; on iPhone and iPad the tabs carry English names (Encode, Decode, EXIF, Batch, Info), on the Mac German ones (Verstecken, Entdecken, Batch, EXIF, Info). This manual quotes the original label and gives the translation in parentheses.

3.2 Encode Area

Element	Description
Carrier image	Any image (PNG, TIFF, JPEG, HEIC, BMP, RAW); the result is always saved losslessly as PNG or TIFF
Content type	Text, image or file — image and file with Pro only
Direct capture	Capture a photo with the camera and embed it in one step; never lands in the photo library (Pro)
Quality slider	Live slider balances the capture resolution against carrier capacity
Password	Optional — activates AES-256-GCM (Pro)
Output format	PNG or TIFF (iPhone/iPad: PNG by default; Mac: a TIFF carrier stays TIFF, otherwise PNG)
Capacity indicator	Real-time bar with percentage, plus the content size relative to the capacity
Encode button	The “Kodieren” button (Mac: “Kodieren und Speichern”) starts the invisible embedding (Pro); without Pro only the carrier with AI label can be saved

Capture a Photo Directly

In the Encode area the image to embed can also be captured directly with the camera: take a photo and hide it inside the carrier in a single step, with no detour through the photo library. The capture is never saved to the photo library. On Mac it stays exclusively in memory. On iPhone and iPad the app stores a temporary copy in its protected app container for further processing; it is deleted when you reset or replace the content, and at the latest at the next app launch. A live quality slider balances resolution against the carrier's remaining capacity, and the EXIF orientation is burned straight into the pixels so the image embeds upright. On Mac the app supports Continuity Camera (using an iPhone as an external camera). Direct capture is part of Stego-Lab Pro.

✓ A photo taken with the built-in camera is never written to the photo library. On Mac it stays exclusively in memory; on iPhone and iPad it exists only as a temporary file in the app container until it has been processed — in keeping with the on-device, no-cloud principle of Stego-Lab.

Photo from the Share Menu

A carrier image can also be brought in from outside the app — straight from the system's Share menu. On iPhone and iPad, pick an image in the Photos or Files app, tap Share and then "Copy to Stego-Lab"; the photo is loaded right away as the carrier into the "Encode" tab, which opens automatically. On Mac, Stego-Lab appears in Finder under "Open With" (right-click an image) — or drag the image onto the app icon in the Dock; there it lands in the "Verstecken" tab. This brings a photo straight into the embedding workflow without going through the image picker.

✓ Stego-Lab registers with the system purely as a recipient for images — no sign-in, no extra service. On iPhone and iPad the handed-over copy is removed right after loading; the photo stays in memory only, until you save the finished output yourself.

3.3 Decode Area

Element	Description
Load image	iPhone/iPad: from Photos or Files (PNG, JPG, TIFF, HEIC, RAW); Mac: "Bild öffnen..." (PNG, JPEG, TIFF, BMP) or drag and drop
Password	Enter it if the content is encrypted (Pro)
Decode	Extracts embedded content — Stego-Lab format, Stegosuite PNG (LSB), legacy Stegosuite 0.8 JPG (F5). Text without password is free, image and file content requires Pro
Preview	Decoded content displayed directly; an accompanying message (e.g. for multi-block carriers) appears in the banner above
Export	Copy or share text (free); image and file export with Pro
Legacy notice	For PV1 and Stegosuite content an advisory appears — weaker crypto, read-only (PV1 = development format from before the first release, never shipped)

3.4 – 3.6 EXIF · Batch · Info

EXIF area: display all metadata and edit writable fields directly. Batch area: process multiple images simultaneously with progress bar. Info area: quick reference and app version details.

Steganography

CHAPTER 4

4.1 Fundamentals

Steganography is the technique of hiding information within an inconspicuous carrier medium. Unlike cryptography, steganography conceals the very existence of the message.

Stego-Lab implements image steganography via the Least-Significant-Bit method (LSB): the least significant bits of color channels R, G, B are replaced by payload bits. Visual effects are imperceptible to the human eye.

4.2 LSB Method

Each color channel is an 8-bit value (0–255). An LSB change corresponds to a color difference of ± 1 — far below the human perception threshold.

Channel	Original bit (LSB)	Payload bit	Resulting bit	Colour difference
Red	1101 0110	1	1101 0111	+1
Green	1010 0011	0	1010 0010	-1
Blue	0011 1001	1	0011 1001	0

Parameter	Value	Note
Bits per channel	1 (fixed, not configurable)	3 channels: R, G, B
Payload capacity	3 bits per pixel	= 3/8 byte per pixel, minus framing data (see 4.5)
PSNR (quality)	≥ 51 dB	≈ 51.1 dB at full load; not visually perceptible, statistically detectable
Channels used	R, G, B	Alpha channel is not used; carrier transparency is not preserved
Required output format	PNG or TIFF	JPEG destroys LSB data!

▲ IMPORTANT: JPEG compression irreversibly destroys LSB data. Stego-Lab always outputs PNG or TIFF for steganography.

4.3 Embedding Content

Embedding Text

Text is embedded as UTF-8 byte sequence. All Unicode characters, line breaks, and special characters are fully supported.

Resolution	Capacity (bytes)	Capacity (characters)
1080 × 1920 (Full HD)	approx. 777,600 B	approx. 777,600 ASCII
2160 × 3840 (4 K)	approx. 3,110,400 B	approx. 3 million ASCII
4000 × 6000 (24 MP)	approx. 9,000,000 B	approx. 9 million ASCII

Images & Files

Images and files of any type (PDF, DOCX, ZIP ...) are embedded exactly as the app receives them: byte by byte, without re-encoding and without additional compression – a JPEG stays a JPEG. The only exception is a picture taken with the camera function: it exists only in memory, so it is encoded losslessly as PNG and, if necessary, scaled to the capacity of the carrier image. The original filename is placed in front of the payload (2-byte length + name in UTF-8) and is encrypted together with the content when a password is set. The 10-byte header holds only the identifier, version, payload length and flags; no MIME type is stored.

4.4 Password Protection

Optional AES-256-GCM encryption (AEAD), part of Stego-Lab Pro. Without a password the payload is stored in plain text in the LSB layer. The key is derived from the password via PBKDF2-HMAC-SHA256 (600,000 iterations, random 16-byte salt per encryption). Without the correct password the embedded bits can be read out but not decrypted; with a sufficiently strong password, guessing it is computationally infeasible. Weak passwords can be found by dictionary attacks – Stego-Lab itself offers such a check in the Decode area.

Parameter	Value
Algorithm	AES-256-GCM (AEAD)
Key derivation	PBKDF2-HMAC-SHA256
Iterations	600,000 (OWASP 2023 recommendation)
Salt	16 bytes, cryptographically random, new for every encryption
Nonce	12 bytes, cryptographically random (GCM standard)
Integrity	16-byte GCM auth tag (built in, no separate HMAC)

▲ There is no backdoor, no master key and no recovery function: if the password is lost, the content cannot be recovered – not even by the developer. A strong password protects the content permanently; a weak one can be found by a dictionary attack. Choose sufficiently long passwords and keep them safe.

4.5 Decoding & Capacity

Decode: read the first 80 LSB bits (10-byte header), check the identifier (PV2; the development format PV1 is accepted read-only), evaluate length and flags, then read salt and nonce if present and the payload. For encrypted content the key is derived via PBKDF2-HMAC-SHA256 (600,000 iterations) from the password and the stored salt, and the payload is decrypted with AES-256-GCM, verifying the 16-byte authentication tag. If the tag is invalid (wrong password or modified data), nothing is output. If no known identifier is found, Stego-Lab

tries, read-only, the legacy formats Stegosuite (PNG) and, for JPEG carriers, F5.

Formula	Value
Capacity (bytes) = $\lfloor W \times H \times 3 \div 8 \rfloor$ – framing data	W, H = width and height in pixels; 3 channels × 1 bit; framing data see below (at least 10 bytes)
Example: $3024 \times 4032 \times 3 \div 8$	= 4,572,288 bytes ≈ 4.57 MB (4.36 MiB); usable without password 4,572,278 bytes, with password 4,572,234 bytes
Framing data (Stego-Lab internal)	10-byte header (identifier 4 + version 1 + length 4 + flags 1); with a password an additional 44 bytes (salt 16 + nonce 12 + GCM tag 16), 54 bytes in total; for images and files an additional 2 bytes + filename in UTF-8 (inside the ciphertext when a password is set); with AI labelling (Pro, batch) an additional 7-byte frame + labelling data (at most 2,048 bytes), unencrypted

Watermarking

CHAPTER 5

5.1 Visible Watermarks

Parameter	Text watermark	Image watermark
Content type	Any UTF-8 text	PNG, JPEG, HEIC, TIFF
Opacity	15 – 100 % (default 65 %)	15 – 100 % (default 65 %)
Position	9 fixed positions in a 3×3 grid, default: bottom right	9 fixed positions in a 3×3 grid, default: bottom right
Size	2 – 10 % of the shorter image edge (minimum 24 px)	8 – 45 % of the image width (minimum 48 px)
Rendering	White, bold system font, black drop shadow — colour, typeface and rotation cannot be chosen	Original colours of the logo, no rotation
Availability	Batch area: one image free, several images with Pro	Batch area: one image free, several images with Pro

5.2 Invisible & Combined Watermarks

Invisible: An identifier is hidden in the image via LSB. Visually identical to the original — extractable via Stego-Lab at any time. Combined: apply visible and invisible watermark simultaneously to one image.

5.3 Positioning & Batch

Position	Label in the app	Description
Top left	"Oben L"	Top-left corner
Top centre	"Oben M"	Top centre of the image
Top right	"Oben R"	Top-right corner
Middle left	"Links M"	Vertically centred, left
Centre	"Mitte"	Fully centred
Middle right	"Rechts M"	Vertically centred, right
Bottom left	"Unten L"	Bottom-left corner
Bottom centre	"Unten M"	Bottom centre of the image

Position	Label in the app	Description
Bottom right	"Unten R"	Bottom-right corner — default

Batch watermarking: same settings applied to any number of images in one run (Pro; a single image with a visible watermark is free) — visible, invisible, or combined.

EXIF & Metadata Editing

CHAPTER 6

6.1 EXIF Fundamentals

EXIF (Exchangeable Image File Format) is the widely used standard for storing metadata in image files. It contains technical camera data, GPS coordinates, timestamps and rights information. Stego-Lab reads and writes these fields through Apple's ImageIO without fixing a particular EXIF version.

6.2 Readable Fields

EXIF tag	Field (label in the app)	Example value
0x010F	"Kamera-Hersteller" (camera make)	Apple
0x0110	"Kamera-Modell" (camera model)	iPhone 15 Pro
0xA434	"Objektiv" (lens model)	iPhone 15 Pro back camera
0x9003	"Aufnahmedatum" (DateTimeOriginal)	2024:03:15 14:32:00
0x9004	"Digitalisiert" (DateTimeDigitized)	2024:03:15 14:32:00
0x829A	"Belichtungszeit" (exposure time)	0.001 (= 1/1000 s)
0x829D	"Blende" (f-number)	1.8
0x8827	"ISO" (ISO speed)	64
0x920A	"Brennweite" (focal length)	6.765
0x013B	"Künstler" (Artist)	John Doe
0x8298	"Copyright"	© 2024 J. Doe
0x010E	"Beschreibung" (ImageDescription)	Alpine panorama
0x9286	"Kommentar" (UserComment)	Free text
0x0131	"Software"	Stego-Lab 1.1

6.3 Editable Fields

Field	Editable	Note
"Beschreibung" (description)	YES (Free)	Free text
"Künstler" (Artist)	YES (Free)	Photographer's name
"Copyright"	YES (Free)	Copyright notice
"Software"	YES (Free)	Editing software
"Kamera-Hersteller", "Kamera-Modell" (camera make, model)	YES (Pro)	Free text (Mac: "Kamera (Marke)", "Kamera (Modell)")
"Objektiv" (lens)	YES (Pro)	Free text
"Aufnahmedatum" (capture date)	YES (Pro)	EXIF format YYYY:MM:DD HH:MM:SS (Mac: "Datum (Original)")
"Digitalisiert" (digitised)	YES (Pro)	EXIF format YYYY:MM:DD HH:MM:SS (Mac: "Datum (Digitalisiert)")
"Belichtungszeit" (exposure time)	YES (Pro)	Seconds, decimal (e.g. 0.001); invalid input is silently discarded
"Blende" (aperture)	YES (Pro)	F-number (e.g. 2.8); invalid input is silently discarded
"ISO"	YES (Pro)	Integer (e.g. 800)
"Brennweite" (focal length)	YES (Pro)	Millimetres, decimal (e.g. 50); invalid input is silently discarded
"Kommentar" (comment)	YES (Pro)	Free text (EXIF UserComment)
GPS: "Breitengrad", "Längengrad", "Höhe" (latitude, longitude, altitude)	YES (Pro)	Decimal number; negative = south, west or below sea level. Delete and/or set anew — an empty field removes only that value
All fields	—	An empty field removes the value. Takes effect with "Änderungen anwenden" (apply changes; Mac: "Metadaten anwenden")

6.4 GPS Data

GPS tag	Description	Format
GPSLatitude	Latitude	File: degrees, min., sec. + N/S — app: decimal degrees, negative = south (Pro)
GPSLongitude	Longitude	File: degrees, min., sec. + E/W — app: decimal degrees, negative = west (Pro)
GPSAltitude	Altitude above sea level	File: metres + reference — app: metres, negative = below sea level (Pro)

GPS tag	Description	Format
GPSTimeStamp	GPS time	UTC HH:MM:SS — not shown in the app
GPSTimeStamp	GPS date	YYYY:MM:DD — not shown in the app
GPSImgDirection	Camera direction	0 – 360° — not shown in the app
Removal	Partial or complete	Clearing the coordinate fields removes only latitude, longitude and altitude; all other GPS tags stay in the file. Only "Alle Metadaten löschen" (delete all metadata, Free) removes them completely — together with all other metadata

▲ GPS data contains the exact capture location. Always check and remove if needed before sharing images!

6.5 Delete All Metadata

This function removes EXIF, GPS, IPTC and XMP metadata as well as camera and lens data and embedded thumbnails from the image — ideal before sharing or publishing.

The EXIF area provides the "Alle Metadaten löschen" (delete all metadata) button for this (free, with a confirmation dialog). The deletion only takes effect with "Änderungen anwenden" (apply changes; Mac: "Metadaten anwenden"); until then, sharing and saving are locked. In the Batch area, the "Alle Metadaten löschen" toggle (Pro) cleans an entire image series in a single run; if AI labelling is active there at the same time, the label is written again after the deletion. If the image carries an AI label (see chapter 9), only its metadata part is removed. A visible label and an invisible label inside the image data are kept, because they are part of the pixels — the confirmation dialog points this out.

■ What is kept: image content, colour profile (ICC) and image orientation. JPEG and HEIC files are re-encoded in the process — an additional lossy compression step. PNG and TIFF remain lossless.

Batch Processing & Supported Formats

CHAPTER 7

7 Batch Processing

Parameter	Value
Max. images	Unlimited (depends on RAM; max. 250 MB or 300 megapixels per file)
Processing	Sequential, with progress bar ("Bearbeite x / y")
Error handling	Files that cannot be loaded are reported at import and left out; an error during processing aborts the run. No log.
Cancelling	Cannot be cancelled while running
Output	Temporary folder; file name = original + "_watermark" in every mode (with a counter on name clashes). Then iPhone/iPad: "Exportordner teilen..." or "Alle in Fotos speichern"; Mac: "Im Finder zeigen" or "In Ordner kopieren..."
Export format	PNG, TIFF or JPG; JPG only in "Sichtbar" (visible) mode and not while the robust AI layer is active
Modes	"Sichtbar" (visible) · "Unsichtbar" (invisible) · "Beides" (both); plus AI labelling and "Alle Metadaten löschen" (delete all metadata)
Availability	One image with a visible watermark free; several images, "Unsichtbar" (invisible), "Beides" (both), "Alle Metadaten löschen" (delete all metadata) and the robust AI layer with Stego-Lab Pro

8 Supported Formats

Input Formats

Format	Full name	Lossless	Loadable as carrier
PNG	Portable Network Graphics	YES	YES — primary format; result saved as PNG or TIFF
TIFF	Tagged Image File Format	YES	YES — result saved as PNG or TIFF
JPEG / JPG	Joint Photographic Experts Group	NO	YES — result is always saved as PNG or TIFF; in addition, reading of old Stegosuite 0.8 JPGs (F5, since 1.1)
HEIC	HEIF container (ISO/IEC 23008-12) with HEVC coding	NO	YES — result saved as PNG or TIFF (Mac: via drag & drop or "Open with")

Format	Full name	Lossless	Loadable as carrier
DNG	Adobe Digital Negative (RAW)	mostly	YES, provided the system supports the RAW model — result saved as PNG or TIFF (Mac: via drag & drop or "Open with")
CR2 / CR3	Canon RAW	mostly	YES, provided the system supports the RAW model — result saved as PNG or TIFF (Mac: via drag & drop or "Open with")
NEF	Nikon RAW	mostly	YES, provided the system supports the RAW model — result saved as PNG or TIFF (Mac: via drag & drop or "Open with")
ARW	Sony RAW	mostly	YES, provided the system supports the RAW model — result saved as PNG or TIFF (Mac: via drag & drop or "Open with")
BMP	Bitmap	YES	YES — result saved as PNG or TIFF

Output Formats

Format	Steganography	Watermark	EXIF	Quality
PNG	YES (default)	YES	YES (if the source is PNG)	Lossless
TIFF	YES	YES	YES (if the source is TIFF; fallback for unknown source types)	Lossless
JPEG	NO — only reading of old Stegosuite 0.8 JPGs (F5)	YES (batch mode "Sichtbar" (visible) only, without the robust AI layer)	YES (if the source is JPEG)	System default, not adjustable
HEIC	NO	NO	YES (if the source is HEIC)	System default, not adjustable

Recommendations

Use case	Input format	Output format
Steganography (text)	Any image (PNG, TIFF, JPEG, HEIC, BMP, RAW)	PNG or TIFF
Steganography (file)	Any image (PNG, TIFF, JPEG, HEIC, BMP, RAW)	PNG or TIFF

Use case	Input format	Output format
Visible watermark (batch; one image free)	Any	JPG (social media) / PNG or TIFF (archive)
Invisible watermark (batch, Pro)	Any	PNG or TIFF
EXIF editing	Any	Same format as the source (TIFF for unknown source types)
Removing GPS	JPEG, HEIC, PNG, TIFF	Same format as the source
Long-term archiving	TIFF	TIFF (uncompressed)

Labelling AI Images

CHAPTER 9

9.1 Legal Framework

Since 2 August 2026, Article 50 of the EU AI Act applies with two separate obligations. Providers of AI systems that generate images must label their output in a machine-readable way (para. 2); systems placed on the market before 2 August 2026 benefit from a transition period until 2 December 2026. Deployers who use an AI system to generate or manipulate images that constitute a deep fake within the meaning of Article 3(60) must disclose that the content has been artificially generated or manipulated (para. 4). This covers images that resemble real persons, objects, places or events and would falsely appear authentic. For evidently artistic, creative or satirical works, a notice that does not hamper the display of the work is sufficient; purely personal, non-professional use is not covered. There is no general labelling obligation for every AI-edited image.

Stego-Lab does not generate AI content itself and is therefore not a provider under para. 2. The app is a tool for deployers who want or need to implement a disclosure under para. 4: it combines a visible label with machine-readable metadata; in the Batch area (Pro) the "Robuste unsichtbare Schicht" (robust invisible layer) is added — robust against the removal of metadata, not against lossy recompression (see 9.6).

artificialintelligenceact.eu/en/transparency-rules-article-50

9.2 Labelling in Encode and Batch

One switch is enough: in the "AI Labelling" section of Encode and Batch, "Label as AI image" activates the preset. Two variants are available, "fully AI-generated" and "AI-edited"; an optional "Generator" text field accepts e.g. "Midjourney v7" and is written only to the metadata, never into the visible label.

The visible label adapts automatically: a minimum-contrast rule based on luminance measurement (light text on a dark background or vice versa) and a minimum size of 3% of the shorter image edge keep it legible. Its position is freely selectable from a 3×3 grid (default: bottom left). A three-stage status light next to the switch shows at a glance whether the current setting is Art. 50 compliant.

Encode already labels completely on the Free tier: visible label plus metadata, even without any hidden content of your own. Batch adds the robust invisible layer for Stego-Lab Pro and labels any number of images in one run.

9.3 Fields Written

Field	Value / content	Format
IPTC Digital Source Type — fully AI-generated	http://cv.iptc.org/newscodes/digitalsourcetype/trainedAlgorithmicMedia	IPTC / XMP
IPTC Digital Source Type — AI-edited	http://cv.iptc.org/newscodes/digitalsourcetype/compositeWithTrainedAlgorithmicMedia	IPTC / XMP
XMP Iptc4xmpExt:DigitalSourceType	Same value as IPTC, additionally as an XMP property (PNG via iTXt)	XMP

Field	Value / content	Format
EXIF UserComment + TIFF ImageDescription	AI-generated content · KI-generierter Inhalt (EU AI Act Art. 50)	Plain text
Invisible layer "AI-DISCLOSURE v1"	JSON holding variant, generator, issuer and timestamp; prefix "AIDISC1:"	PV2 trailer, flag bit 3 (Pro)

9.4 Detection on Read

Decode and EXIF show a neutral "AI labelling detected" badge as soon as one of the three sources is found — with no judgment and no claim about the image's origin. When several sources are found, priority runs invisible layer, then metadata, then plain-text hint; if the sources disagree, a neutral note appears instead of a warning.

9.5 Free vs. Pro

Building block	Tier
Visible AI label	Free
Metadata labelling (IPTC/XMP/EXIF)	Free
Robust invisible layer	Pro
Batch labelling of many images	Pro

9.6 Limitations

▲ Limitations: IPTC/XMP Digital Source Type is reliably written and re-read by Stego-Lab for PNG, TIFF, and JPEG. The plain-text hint in TIFF ImageDescription is correctly written to the file for all three formats, but Stego-Lab's own reader only recovers it for TIFF carriers, not PNG or JPEG. The middle dot „." in the EXIF UserComment plain text is corrupted to a question mark for all three formats — genuine, format-independent data loss. The invisible layer does not survive JPEG recompression or social-media uploads; it is an archival/proof feature for PNG/TIFF, not a C2PA replacement. The EU-wide unified label will only follow once the Commission's final code of practice is in place — until then, the text label remains the standard.

9.7 Legal Notice

■ Stego-Lab does not generate AI content and makes no claim about an image's origin. Labelling is based solely on the user's own input. The app is a tool to help meet transparency obligations, not legal advice. Responsibility for the accuracy and completeness of the labelling remains with the user.

Security & Privacy

CHAPTER 10

10.1 Local Processing

Data category	Processing	Network
Images	Local only	NONE
Embedded content	Local only	NONE
Passwords	In RAM only, never persisted	NONE
EXIF data	Local only	NONE
Crash reports	No crash reporting in the app (system reports go to Apple only with user consent)	NONE
Analytics / telemetry	Not present	NONE
Advertising	Not present	NONE

10.2 Encryption Details

Step	Method	Parameters
1. Salt	SecRandomCopyBytes (Apple Security)	16 bytes, cryptographically random, new for every encryption
2. Nonce	Apple CryptoKit (AES.GCM.seal)	12 bytes, random, new for every encryption (GCM standard)
3. Key	PBKDF2-HMAC-SHA256	600,000 iterations, 32-byte key
4. Enc.	AES-256-GCM (AEAD)	16-byte authentication tag appended to the ciphertext
5. Frame	Header salt nonce ciphertext+tag AI trailer if present	Binary, in the LSB layer; header and AI trailer remain unencrypted
6. Verify	GCM authentication tag	Checked during decode; no separate HMAC

10.3 Steganographic Security

Attack	Without password	With AES-256
Signature check (format known)	Immediately detectable: the identifier "PV2" is stored in plain text in the LSB layer	Immediately detectable; content and filename protected

Attack	Without password	With AES-256
Visual inspection	Safe	Safe
RS analysis	Detectable	Channel visible, content protected
Chi-squared test	Detectable	Channel visible, content protected
Brute force (content)	Content directly readable	Computationally infeasible (strong password)
Password dictionary attack	N/A	Possible with a weak password
JPEG conversion	Content destroyed	Content destroyed

✓ Recommendation: always combine LSB steganography with AES-256. Content remains unreadable even if the channel is statistically detected.

Technical Specifications

CHAPTER 11

11.1 Performance Data

Operation	Image size	iPhone 15 Pro	iPad Pro M4	MacBook Pro M3
Encode text 1 KB	12 MP (4032×3024)	< 0.5 s	< 0.5 s	< 0.2 s
Encode file 1 MB	12 MP	~1.5 s	~1.2 s	~0.5 s
Encode file 10 MB	48 MP	~8 s	~6 s	~2.5 s
Decode 1 KB	12 MP	< 0.5 s	< 0.5 s	< 0.2 s
Decode 1 MB	12 MP	~1.5 s	~1.2 s	~0.5 s
Visible watermark	12 MP	~0.8 s	~0.6 s	~0.3 s
Reading EXIF	Any	< 0.1 s	< 0.1 s	< 0.1 s
Batch WM, 100 images	avg. 8 MP	~80 s	~60 s	~25 s

11.2 Memory Requirements

Image size	RAM (RGBA)	RAM (RGB)	Recommended device RAM
8 MP (3264×2448)	~32 MB	~24 MB	3 GB
12 MP (4032×3024)	~49 MB	~37 MB	4 GB
24 MP (6000×4000)	~96 MB	~72 MB	6 GB
48 MP (8064×6048)	~196 MB	~147 MB	8 GB
108 MP (12000×9000)	~432 MB	~324 MB	16 GB

11.3 Algorithms & Standards

Algorithm	Use	Specification
LSB steganography	Primary embedding method (1 bit per channel, R/G/B, sequential)	Own frame format PV2, source code published under the MIT licence
AES-256-GCM	Payload encryption (AEAD); the 12-byte nonce is generated by CryptoKit when sealing	FIPS 197, NIST SP 800-38D

Algorithm	Use	Specification
PBKDF2-HMAC-SHA256	Key derivation, 600,000 iterations	RFC 8018 (formerly RFC 2898), NIST SP 800-132
GCM authentication tag	Integrity check (16 bytes, no separate HMAC)	NIST SP 800-38D
SecRandomCopyBytes	Generation of the 16-byte salt	Apple Security framework
HKDF-SHA256	Key derivation for legacy format PV1 (development format before 1.0, never released) – read only	RFC 5869
AES-128-CTR	Stegosuite compatibility – read only	NIST SP 800-38A
PBKDF2-HMAC-SHA1	Key derivation for Stegosuite – read only	RFC 8018
MD5	Seed of the Stegosuite position generator – no security function, read only	RFC 1321
SHA-1	Pseudo-random generator for F5 coefficient selection – no security function, read only	FIPS 180-4
EXIF 2.32	Metadata standard	CIPA DC-X008-Translation-2019
PNG	Lossless output format	ISO/IEC 15948:2004
TIFF 6.0	Lossless output format	Adobe TIFF specification
UTF-8	Text encoding	RFC 3629, Unicode 15.0

Use Case Examples

CHAPTER 12

12.1 Private Document Protection (Pro)

Step	Action
1	Open the Encode area
2	Load a holiday photo as carrier (PNG/TIFF recommended)
3	Content type: File → select Pass-Scan.pdf
4	Enter a strong password (recommended: at least 12 characters, better a passphrase)
5	Start Encode → save as PNG
6	Delete the original PDF after securing the PNG
Decode	Load PNG → Decode → password → extract file

12.2 Copyright Proof (Pro)

Step	Action
1	Batch → apply visible WM: "@Name © 2025" → output PNG
2	EXIF → set Copyright + "Künstler" (Artist) (Free), "Änderungen anwenden" (apply changes)
3	Encode: embed the invisible watermark text into the image marked this way
4	Output as PNG, password optional
5	Publish
Proof	Decode → invisible WM proves authorship (only if passed on losslessly as PNG/TIFF)
Caution	Keep this order: a visible watermark applied afterwards over already hidden data destroys it. Alternative in a single pass: batch mode "Beides" (both).

12.3 Photographer Workflow, 500 images (Pro)

Step	Action
1	Batch → select 500 images
2	Visible WM: "© 2025 Name" — position "Unten M" (bottom centre), opacity 25–40 %
3	Enable the "Alle Metadaten löschen" (delete all metadata) toggle — removes EXIF, GPS, IPTC and XMP from all output images
4	The author credit is in the visible watermark; there is no field-by-field bulk EXIF editing (Artist/Copyright) — if needed, per image in the EXIF area
5	Export format: JPG for delivery (quality not adjustable, system default); for a PNG archive, start a second run
6	Start batch
Result	500 images in ~10 minutes (guide value)

12.4 Educational Context (Pro)

Learning objective	Demonstration with Stego-Lab
Understand LSB steganography	Embed text; compare original and result with an external image tool (histogram) — Stego-Lab itself shows no histogram
Experience AES encryption	Enable password protection, decode without password → error
Understand PBKDF2 and salt	Embed the same password twice → a different result each time because of the random salt
Analyse the EXIF standard	Fully analyse a camera image
GPS privacy	View GPS → remove it ("Alle Metadaten löschen" is Free, the GPS fields are Pro) → check the result
Lossless vs. lossy	Encode into PNG, save as JPEG with an external tool → decode fails
Capacity calculation	Image dimensions → formula → compare with the app

FAQ

CHAPTER 13

Q1: Use JPEG for steganography?

A: For embedding, no: JPEG compression irreversibly destroys LSB data, so Stego-Lab only outputs PNG or TIFF. For reading, the app additionally supports old Stegosuite 0.8 JPGs using the F5 method (decode only).

Q2: What is the maximum embeddable content size?

A: Capacity (bytes) = $(W \times H \times 3) \div 8$ – framing data (10 bytes, 54 bytes with a password, plus the filename). A 12 MP image (3024 × 4032) offers ≈ 4.57 MB (4.36 MiB). Stego-Lab shows the remaining capacity in real time.

Q3: What if I forget the password?

A: The password is never stored; there is no backdoor, no master key and no recovery. Only the optional word-list attack in the Decode area (Pro) can find a weak password that is contained in the list – with a strong password the content remains permanently inaccessible.

Q4: Can stego data be detected?

A: Visually: No. Statistically detectable (RS, Chi-squared). Content still unreadable with AES-256.

Q5: Works fully offline?

A: Yes. No network connections whatsoever. All features available offline.

Q6: Images stored in the cloud?

A: No. Stego-Lab transmits no data. All processing is exclusively local.

Q7: Decode Mac-encoded images on iPhone?

A: Yes. Internal format is cross-platform compatible.

Q8: Best format for maximum quality?

A: TIFF (uncompressed) for archive. PNG for lossless with smaller file size.

Q9: RAW formats supported?

A: As input: DNG, CR2, CR3, NEF, ARW, ORF ... For steganography the output is always PNG or TIFF. For purely visible watermarks in batch mode, JPG is also possible (not when the robust invisible layer is active); the EXIF area exports in the format of the source file.

Q10: Different passwords per image?

A: Yes. Each image can have an individual password.

Q11: Shortcuts app supported?

A: Planned for a future version.

Q12: Visible vs. invisible watermark?

A: Visible: drawn directly into image. Invisible: hidden via LSB, extractable only via app.

Q13: Do I have to label AI images?

A: It depends. Since 2 August 2026, Art. 50(4) EU AI Act obliges deployers to disclose, for published deep fakes, that the content was AI-generated or AI-edited; for evidently artistic or satirical works a reduced notice is sufficient, and purely private use is not covered. Providers of AI systems are subject to the machine-readable marking obligation under para. 2. Stego-Lab provides the tool, not legal advice (see 9.7).

Q14: Does the labelling survive social media or JPEG compression?

A: The visible label yes (part of the pixels). Metadata depends on the platform — many strip it on upload. The invisible layer (LSB) does not.

Q15: How do I remove all metadata at once?

A: EXIF area: "Alle Metadaten löschen" (delete all metadata) button (Free), then "Änderungen anwenden" (apply changes; Mac: "Metadaten anwenden"). For an entire image series: Batch area, "Alle Metadaten löschen" toggle (Pro).

Technical Appendix

CHAPTER 14

14.1 LSB Encode Process

Step	Description
1. Prepare payload	Text as UTF-8 bytes; images and files as unmodified original bytes, without re-encoding and without compression (only camera photos are saved as PNG first). For images and files the filename is prepended: 2-byte length (big-endian) + name in UTF-8.
2. Encrypt (optional, Pro)	With a password: 16-byte random salt (SecRandomCopyBytes), 256-bit key via PBKDF2-HMAC-SHA256 (600,000 iterations), AES-256-GCM with a random 12-byte nonce (CryptoKit). The 16-byte authentication tag is appended to the ciphertext; the filename is encrypted as well.
3. Build frame	10-byte header (identifier "PV2", version, length, flags), followed by salt and nonce when encrypted, then the payload or the ciphertext with its tag, and, if present, the unencrypted AI-labelling trailer. If the frame does not fit into the image, Stego-Lab aborts before modifying any pixel.
4. Replace LSBs	The frame is written bit by bit, most significant bit of each byte first: per pixel successively into the least significant bit of R, G and B, row by row from the first pixel (row-major order). Exactly 1 bit per channel; the alpha channel is not used.
5. Output	Saved as PNG or TIFF (lossless, 8 bits per channel); carrier transparency is not preserved. The LSB data remain intact.

14.2 Internal Frame Format

The internal Stego-Lab frame header is written first into the carrier image LSB layer, followed by payload data. This format is cross-platform compatible (iOS, iPadOS, macOS).

Section	Size	Content
MAGIC	4 bytes	0x50 0x56 0x32 0x00 ("PV2" + null byte) – marks Stego-Lab data in the LSB layer; "PV1" (development format) is also accepted read-only
VERSION	1 byte	Format version, currently 0x02 (not evaluated when reading)
PAYLOAD_LENGTH	4 bytes	uint32, big-endian: payload length in bytes – when encrypted, the length of the ciphertext including the 16-byte tag (excluding salt and nonce); otherwise the length of the plain-text payload including the filename prefix

Section	Size	Content
FLAGS	1 byte	Bit 0 = content type (1 = text, 0 = image/file) · bit 1 = encrypted · bit 2 = filename prefix at the start of the payload · bit 3 = AI-labelling trailer follows (since 1.1, batch with robust layer only, Pro) · bits 4–7 = 0. No type/MIME field, no compression.
SALT	16 bytes	Only if bit 1 is set: random salt for PBKDF2 (SecRandomCopyBytes)
NONCE	12 bytes	Only if bit 1 is set: random AES-GCM nonce (CryptoKit)
PAYLOAD	= PAYLOAD_LENGTH	Without password: [filename prefix] + content in plain text. With password: AES-256-GCM ciphertext of [filename prefix] + content, ending with the 16-byte authentication tag
– Filename prefix	2 + n bytes	Only if bit 2 is set, part of the payload (included in PAYLOAD_LENGTH): length n (uint16, big-endian) + filename in UTF-8; encrypted together with the content when a password is set
AI TRAILER	7 + n bytes	Only if bit 3 is set, after the payload, not included in PAYLOAD_LENGTH: "AIDT" + version 0x01 + length n (uint16, big-endian) + "AIDISC1:" followed by JSON ($n \leq 2,048$); always unencrypted, only with PNG/TIFF output

14.3 Error Handling

Error message	Cause	Remedy
Note	The user interface of Stego-Lab 1.1 is in German; messages are quoted verbatim with a translation in brackets. Numbers in the messages depend on the image.	—
„Zu groß: 5.2 MB (max 4.4 MB)“ (too large: 5.2 MB, max 4.4 MB); on iOS when loading: „Die Datei ist 5,2 MB groß, das Trägerbild fasst 4,4 MB. Bitte ein größeres Trägerbild oder eine kleinere Datei wählen.“ (the file is 5.2 MB, the carrier holds 4.4 MB; choose a larger carrier or a smaller file)	Content plus framing data (header, salt, nonce, tag, filename, AI trailer) larger than the carrier capacity	Choose a larger carrier image or smaller content
„Falsches Passwort oder beschädigte Daten“ (wrong password or corrupted data)	GCM authentication tag check failed: wrong password or modified data	Check the password; use the unmodified original
„Kein erkanntes Steganographie-Format (PV1 oder StegSuite) in diesem Bild.“ (no recognised steganography format in this image)	No Stego-Lab identifier in the LSB layer: not an encoded image, or the image was saved lossily (JPEG, HEIC), resized or edited after embedding	Use the unmodified PNG/TIFF original

Error message	Cause	Remedy
„Header beschädigt — möglicherweise kein kodiertes Bild.“ / „Payload abgeschnitten — Bild möglicherweise bearbeitet.“ (header corrupted / payload truncated; macOS only)	Implausible length in the header, or image cropped or edited after embedding	Use the unmodified original
„Bild konnte nicht geladen werden.“ / „Datei konnte nicht gelesen werden oder ist kein gültiges Bild.“ (image could not be loaded / file unreadable or not a valid image)	File is not a readable image	Use PNG, JPEG, TIFF or HEIC
„Datei ist zu groß (312.0 MB). Maximal 250 MB erlaubt.“ / „Bild hat zu viele Pixel (340 MP). Maximal 300 MP erlaubt — möglicherweise eine präparierte Datei.“ (file too large, max 250 MB / too many pixels, max 300 MP, possibly a crafted file)	Protection limits against crafted files exceeded	Use a smaller file
„PNG-Export fehlgeschlagen“ or „TIFF-Export fehlgeschlagen“ (PNG/TIFF export failed; iOS); a system message on macOS	Image could not be written to the destination	Check the location and access rights

Glossary

CHAPTER 15

Term	Description
AES-256-GCM	Advanced Encryption Standard, 256-bit key, GCM mode (AEAD). FIPS 197, NIST SP 800-38D.
AI-DISCLOSURE-Trailer	Optional, unencrypted PV2 companion data (prefix "AIDISC1:") appended after the payload – carry the AI labelling invisibly in the image data (Pro, batch). They survive metadata removal but not lossy recompression (see 9.6).
Alpha channel	Fourth image channel (R,G,B+A) for transparency (0–255).
Chi-Squared	Statistical steganalysis method for LSB detection.
Digital Source Type	IPTC NewsCodes field that machine-readably states an image's origin, e.g. "trained algorithmic media" or "composite with trained algorithmic media"; the central building block of Art. 50 labelling.
DNG	Open RAW format (Adobe) for sensor data – uncompressed, losslessly or (since DNG 1.4) lossily compressed.
EU AI Act Art. 50	Transparency obligations of the EU AI Act: providers must label AI output in a machine-readable way (para. 2); deployers must disclose, for deep fakes, that the content has been artificially generated or manipulated (para. 4). Applicable since 2 August 2026.
EXIF 2.32	Standard for metadata in image files (CIPA DC-X008).
GCM	AES mode of operation (Galois/Counter Mode): encrypts and authenticates in one step (AEAD). In Stego-Lab the 16-byte authentication tag secures the integrity of the ciphertext; no separate HMAC is needed.
HEIC	Image file in the HEIF container (ISO/IEC 23008-12) with HEVC coding, adopted by Apple as its default format. Usually lossy.
HMAC-SHA256	Keyed message authentication code based on SHA-256 (RFC 2104). Stego-Lab uses it solely as the pseudo-random function inside PBKDF2, not for integrity checking; that is done by the GCM authentication tag.
LSB	Least significant bit. ± 1 color difference — invisible.
Nonce	Random 12-byte starting value for AES-GCM; ensures ciphertext uniqueness.
PBKDF2-HMAC-SHA256	Key derivation from password + salt + 600,000 iterations (OWASP 2023).
PNG	Lossless raster format. ISO/IEC 15948. Primary format in Stego-Lab.
PSNR	Image quality metric in dB. > 51 dB = change invisible.

Term	Description
PV1	Development format of Stego-Lab (31 March to 2 April 2026, key derivation HKDF-SHA256 with a fixed salt), never released. The app only reads it; new content is written exclusively as PV2.
PV2	Frame format of Stego-Lab since version 1.0: 10-byte header (identifier “PV2”, version, length, flags), when encrypted salt, nonce and AES-256-GCM ciphertext with tag, optionally an unencrypted AI trailer (see 14.2).
RAW	Camera raw format. Lossless, manufacturer-specific.
RS analysis	Regular-Singular Analysis — statistical LSB detector.
Salt	Random value for PBKDF2. Prevents rainbow table attacks.
Steganography	Hiding information within a carrier medium.
TIFF 6.0	Lossless image format. LZW/ZIP/RAW compression. Adobe specification.
UTF-8	Universal encoding. 1–4 bytes. ASCII-compatible.
Watermark	Visible or invisible authorship mark in digital media.

Market Positioning

CHAPTER 16

16.1 Feature Comparison

Feature	Stego-Lab	Online service	Desktop tool	Mobile app
LSB steganography	YES	Partly	YES	Rarely
AES-256	YES	Rarely	Partly	No
Visible watermark	YES	YES	YES	YES
Invisible watermark	YES	Rarely	Partly	No
EXIF editing	YES	Partly	YES	Partly
Removing GPS	YES	Partly	YES	Partly
Batch	YES	Rarely	YES	Rarely
Fully offline	YES	NO	YES	Partly
Native on iOS/iPadOS	YES	Browser	NO	YES
Native on macOS	YES	Browser	YES	NO
No account required	YES	Often NO	YES	Often NO

16.2 Target Audiences

Target group	Primary value
Photographers	EXIF management, GPS protection, batch watermarking
Content creators	Copyright watermarks, locally and fast
Privacy-conscious users	Local processing, GPS removal
IT security professionals	Stego demos, steganalysis exercises
Journalists	Encrypted embedding of evidence
Archivists	Bulk EXIF editing, metadata upkeep
Teachers and students	Hands-on demonstrations of the concepts
Private users	Discreet archiving of sensitive documents

Workflow Reference

CHAPTER 17

17.1 Encode Workflow (Pro)

Step	Action	Note
1	Load a carrier ≥ 8 MP (PNG/TIFF recommended; JPEG, HEIC and RAW are loaded as well)	Check capacity
2	Select content type: Text / Image / File	Capacity indicator
3	Enter a password (at least 12 characters, better a passphrase)	Recommended; the app warns below 8 characters
4	Output: PNG or TIFF	PNG = smaller
5	Start Encode	Progress
6	Verify in the Decode area	Quality check
7	Save or share	Keep the original!

17.2 EXIF Cleanup

Step	Action
1	Load image in the EXIF area
2	Check the GPS fields ("Breitengrad", "Längengrad", "Höhe" — latitude, longitude, altitude)
3	Remove GPS coordinates: clear the three fields (Pro) — other GPS tags remain; complete removal only with "Alle Metadaten löschen" (delete all metadata, Free)
4	Set copyright: "© 2025 Name" (Free)
5	Set the "Künstler" (Artist) field (Free)
6	"Änderungen anwenden" (apply changes; Mac: "Metadaten anwenden"), then export
7	Verify EXIF
or	"Alle Metadaten löschen" (Free), then "Änderungen anwenden"

17.3 Batch for Social Media (Pro)

Step	Setting
1	Select all photos in the batch
2	Visible WM: "@Username © 2025"
3	Position: "Unten M" (bottom centre) or "Unten R" (bottom right)
4	Opacity: 25–40 % ("Deckkraft" slider, range 15–100 %)
5	Export format: JPG (batch mode "Sichtbar" (visible) only; quality not adjustable)
6	Enable the "Alle Metadaten löschen" (delete all metadata) toggle — GPS is removed with it
7	Start batch → verify

17.4 Label an AI Image

Step	Action	Note
1	Load image in Encode or Batch	PNG/TIFF recommended
2	Open the "KI-Kennzeichnung" (AI labelling) section, enable the "Als KI-Bild kennzeichnen" (label as AI image) toggle	Check the status light
3	Choose variant: "vollständig KI-generiert" (fully AI-generated) / "KI-bearbeitet" (AI-edited)	Pre-filled label text
4	Optionally enter the generator (e.g. "Midjourney v7")	Appears only in metadata
5	Check the position of the visible label in the 3×3 grid	Default: bottom left
6	Batch/Pro: optionally enable "Robuste unsichtbare Schicht (Pro)" (robust invisible layer)	Forces PNG/TIFF
7	Start Encode/Batch, check that the status light is green	Green = all components set

Security Assessment

CHAPTER 18

18.1 Threat Model

Threat	Without password	With AES-256	Assessment
Visual inspection	Safe	Safe	OK
RS analysis	Detectable	Channel visible, content not	Acceptable with password
Chi-squared	Detectable	Channel visible, content not	Acceptable with password
Brute force (content)	Readable	Practically impossible	Safe with password
JPEG conversion	Destroyed	Destroyed	Keep PNG/TIFF!
Cloud interception	N/A	Not relevant	No cloud

18.2 Password Recommendations

Requirement	Recommendation	Rationale
Minimum length	12 characters	Below 8 characters attackable even with PBKDF2
Character set	Upper + lower + digits + symbols	Exponentially larger search space
Dictionary words	Avoid	Dictionary attacks are effective
Reuse	Never	One compromised password endangers all
Storage	Password manager	Kept securely encrypted

18.3 Internal Pre-Release Security Review

Between April and September 2026 the Stego-Lab source code was reviewed internally for security vulnerabilities several times, most recently before the submission of version 1.1. The reviews were AI-assisted; the developer evaluated and implemented the results. These are internal technical reviews, not an independent audit and not a certification. The report of the first review of 10 April 2026 is available in the public repository as SECURITY-AUDIT.md. None of the reports identified a critical finding; the findings rated high were fixed before submission. Known limitations are listed in the table; a few low-rated items are documented and remain open.

Area	Aspects inspected	Status
Cryptography	AES-256-GCM; PBKDF2-HMAC-SHA256 with 600,000 iterations; 16-byte random salt; 12-byte nonce; keys are never stored or transmitted	reviewed

Area	Aspects inspected	Status
Passwords in memory	Passwords and derived keys are not explicitly overwritten in memory by the app (Swift/SwiftUI limitation; finding S-01/N-02, severity low)	known limitation
Legacy formats (read only)	PV1 (development format, never released), Stegosuite and F5: weaker or outdated methods, reading only, warning shown in the app	known limitation
EXIF / metadata	Metadata sanitising; no unintended carry-over of GPS and camera data	reviewed
File access	Path traversal protection, sanitising of external filenames, size and pixel limits, App Sandbox (App Store build)	reviewed
Pro gating	Lock in the user interface and, for hiding and batch processing, additionally in the application logic; verification of StoreKit transactions	reviewed
Privacy	No network connections of its own, no telemetry, no cloud (exceptions: purchase handling via Apple's StoreKit, links opened in the browser)	reviewed
Watermarks	Combined mode: visible first, then invisible; hidden data are preserved	reviewed
Platforms	Shared security layer (StegoEngine) for iOS, iPadOS and macOS	reviewed

■ The report of the review of 10 April 2026 is available in the public repository as SECURITY-AUDIT.md; a summary of the later reviews can be found in CHANGELOG.md. The detailed later review reports are kept internally in the project archive and are not part of the published documentation. For security inquiries please reach out via the support email.

Support & Contact

CHAPTER 19

19.1 Support Channels

Channel	Description
App Store Bewertungen	Public feedback and ratings
App Store Support-Link	Direct support link
In-App Info	Quick reference in app
App-Updates	Regular updates with bug fixes

19.2 Known Limitations

Einschränkung	Detail
JPEG bei Steganographie	Not supported — use PNG/TIFF
Animierte GIFs	First frame only
RAW-Ausgabe	Input only; output PNG/TIFF
Batch-Stego	Same content for all images
Shortcuts-App	Planned
Passwort-Recovery	Not possible

List of Tables

CHAPTER 20

No.	Title	Chapter
T-01	App fact sheet	Cover
T-02	System requirements	2.1
T-03	Installation steps	2.2
T-04	Main navigation	3.1
T-05	Encode area	3.2
T-06	Decode area	3.3
T-07	LSB specification	4.2
T-08	Text capacities	4.3
T-09	Encryption params	4.4
T-10	Watermark params	5.1
T-11	Positioning options	5.3
T-12	EXIF readable fields	6.2
T-13	EXIF editable fields	6.3
T-14	GPS fields	6.4
T-15	Batch parameters	7
T-16	Input formats	8
T-17	Output formats	8
T-18	Format recommendations	8
T-19	AI labelling fields written	9.3
T-20	Privacy overview	10.1
T-21	Encryption steps	10.2
T-22	Steganographic security	10.3
T-23	Performance data	11.1
T-24	RAM requirements	11.2

No.	Title	Chapter
T-25	Algorithms	11.3
T-26	Use case 12.1	12
T-27	Use case 12.2	12
T-28	Use case 12.3	12
T-29	Learning objectives	12.4
T-30	FAQ (15 entries)	13
T-31	Error handling	14.3
T-32	Glossary (25 entries)	15
T-33	Feature comparison	16.1
T-34	Target audiences	16.2
T-35	Workflows (4×)	17
T-36	Threat model	18.1
T-37	Password recommendations	18.2
T-38	Known limitations	19.2

Version History

CHAPTER 21

Version	Date	Changes
1.0	2026-04-16	Initial release for iPhone and iPad (App Store): Encode, Decode, EXIF, Batch, Info
1.0	2026-04-20	Initial release for Mac (Mac App Store): Verstecken, Entdecken, Batch, EXIF, Info
1.0	2026-04	Minimum versions: iOS 16 / iPadOS 16 / macOS 13 (Mac: Apple Silicon + Intel)
1.0	2026-04	Password encryption AES-256-GCM (authenticated, 16-byte tag); key derivation PBKDF2-HMAC-SHA256 with 600,000 iterations and a 16-byte random salt (PV2 format)
1.0	2026-04	Read path for images from pre-release development builds (PV1 format, HKDF-SHA256) — decryption only, never shipped for encryption
1.0	2026-04	Mac: reading Stegosuite PNG images (decryption only: AES-128-CTR, PBKDF2-HMAC-SHA1, MD5 position generator)
1.0	2026-04	Mac: password dictionary attack in the Decode area with a user-supplied word list
1.0	2026-04	Batch (Pro): visible and invisible watermarks, hidden content optionally encrypted
1.0	2026-04	Carrier images: PNG, TIFF, JPEG, HEIC, various RAW formats
1.0	2026-04	Fully local — no cloud
1.1	2026	Minimum versions raised: iOS 17 / iPadOS 17 / macOS 14 (previously iOS 16 / iPadOS 16 / macOS 13) with the switch to Xcode 27; version 1.0 remains usable on older devices
1.1	2026	Encryption of new content unchanged from 1.0 (AES-256-GCM, PBKDF2-HMAC-SHA256, 600,000 iterations, PV2 format)
1.1	2026	Dark-mode picker (System/Light/Dark) on iOS/iPadOS
1.1	2026	iPhone/iPad: password dictionary attack in the Decode area; on both platforms now with the bundled Nmap word list (Mac: attack since 1.0)
1.1	2026	Consolidated iOS file picker with sheet coordinator
1.1	2026	Security hardening: AES parity, EXIF sanitization, Pro gating
1.1	2026	iPhone/iPad: external donation links removed; Mac: follows the system light/dark mode
1.1	2026	iPhone/iPad: reading Stegosuite PNG images (decryption only) — on Mac since 1.0

Version	Date	Changes
1.1	2026	F5/JPG decoder: legacy Stegosuite 0.8 JPGs (DCT-domain steganography) can be opened — including images carrying both a message AND a hidden file in one carrier
1.1	2026	Decode progress indicator for compute-intensive legacy paths (Stegosuite/F5)
1.1	2026	Advisory banner for weaker legacy crypto formats (PV1 / Stegosuite) — read-only by design
1.1	2026	Hide a photo straight from the camera: capture and embed in one step — no detour through the photo library (Mac: memory only; iPhone/iPad: temporary file in the app container), with quality slider; macOS supports Continuity Camera
1.1	2026	iOS camera: EXIF orientation is burned straight into the pixels on embed — captures are embedded correctly oriented
1.1	2026	Adopt a photo as the carrier straight from the share / “Open With” menu (“Copy to Stego-Lab” on iPhone/iPad, “Open With Stego-Lab” on Mac)
1.1	2026	Encode area: more compact, icon-only buttons for a cleaner interface
1.1	2026	AI labelling per EU AI Act Art. 50: preset in Encode/Batch, badge in Decode/EXIF, manual chapter
1.1	2026	Delete all metadata in one click (EXIF tab, Free) and in batch (Pro)

Planned Features

Feature	Status
Shortcuts app integration	Planned
Share Extension	Deliberately rejected (security: would need an App Group)
2-LSB mode	Idea
Steganalysis detector	Planned
IPTC metadata	Implemented (1.1)
XMP metadata	Implemented (1.1)

In-App Purchase Stego-Lab Pro

CHAPTER 22

Stego-Lab is available free of charge on the App Store. There is exactly one in-app purchase: “Stego-Lab Pro”. It is paid once, has no subscription and no expiry date, and permanently unlocks the features listed below. There are no further packs or extensions. The current price is shown in the App Store.

22.1 Free and Pro at a Glance

Unlocking runs exclusively through the App Store — no license key, no serial number, no involvement of the developer. Encryption is not sold separately: password protection is part of Stego-Lab Pro. The following table reflects the actual behaviour of the app.

Area	Free	Additionally with Stego-Lab Pro
Encode	Load a carrier image; apply the visible AI label and AI metadata and save the image	Embed text, image or file invisibly (LSB); optional password protection (AES-256-GCM, PBKDF2-HMAC-SHA256, 600,000 iterations); direct capture with the camera (Mac: Continuity Camera)
Decode	Show, copy and share unencrypted embedded text	Embedded images and files; password-protected content, including Stegosuite PNG and F5 JPG images (read-only); dictionary attack with the bundled Nmap list or your own list
EXIF	View all metadata; edit description, artist, copyright and software; delete all metadata in one click	Edit camera, lens, date, exposure, comment and GPS fields
Batch	Load images, check settings and preview; export a single image with a visible text or logo watermark	Export several images in one run and whole image series (processed one after another): visible, invisible and combined watermarks, AI labelling with robust invisible layer, deletion of all metadata
Info	Quick reference, version details, Pro window	—

22.2 Purchase and Restore

The purchase is made in the app’s Pro window (reachable from the Info area or from any Pro notice) with the “Stego-Lab Pro freischalten” (unlock Stego-Lab Pro) button; payment and processing are handled by the App Store. At launch the app checks the purchase status on its own, so on further devices with the same Apple Account the purchase is usually picked up automatically. If not, the “Käufe wiederherstellen” (restore purchases) button in the Pro window restores it.

✓ Stego-Lab Pro is a one-time purchase — no subscription, no expiry date, no license key. Purchase and payment are handled exclusively by Apple; restoring is done in the app via “Käufe wiederherstellen” (restore purchases) in the Pro window.

About the Developer / Imprint

CHAPTER 23



Stego-Lab

Developed by

Ing. Christian Raith, BEd

OE3LCR · Wiener Neustadt, Austria



Stego-Lab was developed by Ing. Christian Raith (BEd). He teaches computer workshop, electrical engineering and electronics at HTL Mödling, a higher technical college. His focus areas are IT security, security analysis and audits, programming (primarily Arduino and Raspberry Pi), developing apps for security, watermarking, copyright protection and photo processing, as well as technical documentation and teaching materials. He is also a licensed amateur radio operator (callsign OE3LCR, grid JN87ct) and chairman of the Triestingtal Amateur Radio Club Berndorf.

23.1 Developer Details

Property	Value
Name	Ing. Christian Raith, BEd
Location	Wiener Neustadt, Lower Austria, Austria
Ham radio	OE3LCR · Grid JN87ct
Support e-mail	stego-lab@proton.me
Website / project	www.craith.cloud/stego-lab
GitLab	gitlab.com/RaithChr/stego-lab
Profession	Teacher at HTL Mödling — computer workshop, electrical engineering, electronics
Specializations	IT security · security audits · Arduino/Raspberry Pi · app development · technical documentation

23.2 Other Projects by the Developer

Project	Description	Link
OE3LCR Ham Radio Dashboard	Live web application with solar data, band conditions, DX cluster, satellite tracking and space weather.	www.craith.cloud

Project	Description	Link
Frequency measurement (RP2350)	Mains frequency meter built on a Raspberry Pi Pico 2 with an ST7735 TFT display and PC817 optocoupler isolation.	gitlab.com/RaithChr/Frequenztmessung-PICO2-50HZ
Stego-Lab (this app)	Steganography, watermarking, EXIF and batch image processing for iPhone, iPad and Mac.	gitlab.com/RaithChr/stego-lab

23.3 GitLab Repository

The source code of Stego-Lab is publicly available on GitLab (gitlab.com/RaithChr/stego-lab). The version history is kept in CHANGELOG.md and docs/VERSION-HISTORY.de-en.md; the procedure for reporting security vulnerabilities is described in SECURITY.md.

GitLab Repository

gitlab.com/RaithChr/stego-lab

23.4 Website

Project Website

www.craith.cloud/stego-lab

Legal & Privacy

CHAPTER 24

24.1 Privacy Policy (Summary)

Stego-Lab processes all user data exclusively locally on the user's device. The app does not collect, store, transmit, or process any personal data on external servers.

Data category	Processing	Transmitted externally
Images	Local only	NO
Embedded content	Local only	NO
Passwords	In RAM only, never persisted	NO
EXIF metadata	Local only	NO
Usage statistics	Not collected	NO
Crash reports	Disabled	NO
Advertising data	None	NO
Location data	Never requested	NO

24.2 License and Terms of Use

The source code of Stego-Lab is published under the MIT License (gitlab.com/RaithChr/stego-lab, file LICENSE). The app obtained from the App Store is additionally governed by the Apple Media Services Terms and Conditions and Apple's standard End User License Agreement.

Provision	Details
License	The source code of Stego-Lab is licensed under the MIT License (© 2026 Christian Raith; file LICENSE in the repository). Third-party components (see 24.3) are subject to their own licenses. The name and word mark "Stego-Lab", the logo, the app icon and the manual text are not covered by the MIT License; all rights reserved.
App Store version	The app obtained from the App Store is governed by the Apple Media Services Terms and Conditions and Apple's standard End User License Agreement. Pro features are unlocked by in-app purchase — without a license key (see chapter 22).
Permitted use	Stego-Lab is intended for lawful purposes. The user is responsible for complying with applicable law, in particular the copyrights of third parties.
Disclaimer	The software is provided without warranty (see MIT License). A forgotten password cannot be recovered; keeping backups is explicitly recommended.

Provision	Details
Changes	The file LICENSE in the repository in its current version and Apple's license agreement are authoritative; the exceptions for the trademark, logo, app icon, manual text and third-party components remain unaffected.

24.3 Frameworks and Third-Party Components

Stego-Lab links no third-party program libraries. Cryptography, image processing and the user interface come from Apple's frameworks; they are part of the operating system and are not shipped with the app. One third-party component is bundled: the password list of the Nmap project. The read functions for legacy Stegosuite and F5 images were ported to Swift from the Java source code of Stegosuite 0.8 (GPL).

Component	Use in Stego-Lab	Origin / license
CryptoKit	AES-256-GCM (encryption and decryption); HKDF-SHA256 only for reading PV1 images; MD5 and SHA-1 only for reading Stegosuite and F5 images	Apple, part of the operating system
CommonCrypto	PBKDF2-HMAC-SHA256 (600,000 iterations); PBKDF2-HMAC-SHA1 and AES-128-CTR only for decrypting Stegosuite images	Apple, part of the operating system
SecRandomCopyBytes (Security framework)	16-byte random salt per encryption; the 12-byte nonce is generated by CryptoKit	Apple, part of the operating system
Compression	zlib decompression when reading Stegosuite and F5 images	Apple, part of the operating system
ImageIO, CoreGraphics, CoreText	Reading and writing images, pixel access for LSB, EXIF, IPTC and XMP metadata, font of the AI label	Apple, part of the operating system
CoreImage	iPhone/iPad only: orientation of camera captures	Apple, part of the operating system
AVFoundation	Direct capture with the camera, on Mac via Continuity Camera	Apple, part of the operating system
Photos, PhotosUI	iPhone/iPad only: import from and saving to the photo library	Apple, part of the operating system
StoreKit	In-app purchase "Stego-Lab Pro"	Apple, part of the operating system
SwiftUI, UIKit, AppKit, Foundation, Combine, UniformTypeIdentifiers, CoreTransferable	User interface, file selection, file type detection	Apple, part of the operating system
Nmap password list (passwords.lst)	Bundled list of about 5,000 entries for the dictionary attack in the Decode area	© 1996–2022 Nmap Software LLC; Nmap Public Source License (nmap.org/npsl)

Component	Use in Stego-Lab	Origin / license
Stegosuite/F5 read compatibility	Read-only: extracting and decrypting images created with Stegosuite 0.8 (PNG and JPEG with F5), including the Java pseudo-random generator SHA1PRNG ported to Swift (SHA-1 from CryptoKit)	Ported to Swift from the Java source code of Stegosuite 0.8 (GPL-3.0); F5 method by A. Westfeld, Huffman decoder by Westfeld/Breslin

24.4 Copyright & Trademarks

Stego-Lab, the Stego-Lab logo, and all associated graphics are property of Ing. Christian Raith. Apple, iPhone, iPad, Mac, iOS, iPadOS, macOS, App Store, and the Apple logo are registered trademarks of Apple Inc.

Resource	URL
GitLab repository	gitlab.com/RaithChr/stego-lab
Project page	www.craith.cloud/stego-lab/en/
Privacy policy	www.craith.cloud/stego-lab/en/privacy-policy.html
Support	www.craith.cloud/stego-lab/en/support.html
Support e-mail	stego-lab@proton.me
Apple privacy policy	apple.com/legal/privacy

Stego-Lab · Version 1.1 · © 2026 Ing. Christian Raith, BEd
Software: MIT License · Manual text, trademark & logo: all rights reserved
iOS · iPadOS · macOS · Apple Silicon & Intel x86_64



Stego-Lab

Version 1.1 · iOS · iPadOS · macOS

© 2026 Christian Raith

Ing. Christian Raith, BEd · OE3LCR

Software: MIT License · Manual & trademark: all rights reserved

gitlab.com/RaithChr/stego-lab · www.craith.cloud/stego-lab

stego-lab@proton.me