Testing & Evaluation Sheet	
MAT2	
1. Tool Overview	
Name:	Mat2
Category:	Data and Metadata Redaction
Purpose:	Mat2 (Metadata Anonymization Toolkit) is designed to remove metadata from files and documents to protect users' privacy. It strips metadata such as author names, software version, file history, location, and other personal information that can be embedded in files.
Date Tested	4/29/25
Status:	[Deployed / Not Deployed (Launched or not)] ☑ Operational - Actively running/maintained ☐ In Testing - Currently being evaluated or piloted ☐ Inactive/Deprecated - No longer maintained or functional
Deployment Architecture:	☑ A standalone software - Runs entirely locally (e.g., runs on computer and doesn't depend on external server) ☐ A locally hosted service with separate server and client component - Run both backend/frontend yourself (e.g., backend could be on a local network, or self-hosted on cloud) ☐ A service with a local client that's hosted by a third party - You install a client on your device, but it connects to and depends on a remote server (e.g., Signal: install app (client), but Signal's servers handle message relaying, etc.) ☐ A service that is hosted by a third party but can also be self-hosted
Version:	0.12.2
2. Installation & Setup	
OS Compatibility	Linux, macOS, Windows (via WSL or native support for certain components)
Installation Manual:	Yes (not friendly for Windows users)

Installation Steps:	If you are using Linux or macOS: ● Follow the official installation steps here: mat2 Installation guide If you are using Windows: mat2 is designed for Linux-based systems, so to use it on Windows, follow these steps: ● Install a Linux environment on Windows, using one of the following: ○ Windows Subsystem for Linux (WSL) (recommended) ■ Install instructions: https://learn.microsoft.com/en-us/windows/wsl/install ○ Or use a virtual machine like VirtualBox with a Linux distro (e.g., Ubuntu) ● Once inside the Linux terminal run, ○ **sudo apt update** ○ **sudo apt install mat2**
Mention if command-line setup or special configurations are needed	Mat2 is entirely command-line based (Graphical User Interface available using a Linux OS)
Common Installation Issues & Fixes:	● mat2: command not found ○ Add $HOME/.local/bin to PATH if installed via pip --user ● Missing dependencies (python3-gi, gir1.2-glib-2.0, etc.) ○ Install with sudo apt install python3-gi python3-magic gir1.2-glib-2.0 ● pip permission errors ○ Use --user flag or a virtual environment (python3 -m venv venv) ● GLib file errors ○ Ensure the file is readable/writable, or try sudo mat2 file.pdf ● pyproject.toml not found ○ Clone the full repo and run pip install . inside the mat2 folder ● On Windows: "mat2 only works on Linux" ○ Use WSL, install Ubuntu, then run sudo apt install mat2
User Documentation:	Yes (although not very extensive)

Required Technical Knowledge	Intermediate - Requires basic knowledge of command line usage and python dependencies.

3. Testing & Evaluation

Category	Details	Score
Operational Functionality:	**Functionality** ● Test Steps: Verify the tool's core features by using all major functions, tracking any failures or bugs. ☐ The tool is mostly non-functional with many broken features and bugs. ☐ Several broken features or bugs ☐ Minor bugs or issues ☐ Mostly functional with few bugs or no bugs ☑ Fully functional with no bugs **Internet Dependence:** ● **Offline use:** Fully functional offline. No internet needed for removing metadata. **Localization & Language Support** ● **Languages available:** Limited. Primarily English; minor support for other languages via translations. ● **East Asian languages:** No complete translations for Chinese, Japanese, or Korean as of now. **Mobile Accessibility** ● No official mobile app (not designed for mobile use — MAT2 is CLI-based and typically installed on desktop Linux environments).	
Usability for Non-Technical Users	**Ease of Installation & Deployment** ● 1–2 steps on most Linux and macOS distributions. Installation typically involves a single package installation command (for Windows, installation involves several steps, including downloading and installing Python, dependencies, and MAT2 manually or via pip). ● Requires command-line knowledge. MAT2 is a command-line tool, and installation is done via package managers (e.g., apt install mat2 on Debian-based systems). There is no GUI or one-click installer. ● Yes. The documentation is available on the official MAT2 GitHub and is reasonably maintained and updated with releases.	

	● Under 5 minutes on Linux if using a supported package manager. It may take longer (e.g., 10–15 minutes) if building from source. **User Onboarding Experience** ● No. MAT2 does not offer tool-tips or in-app guidance for new users. As a command-line tool, users are expected to refer to the manual (man mat2) or help command (mat2 --help) for guidance. **Technical Experience Level Required** ● MAT2 is not beginner-friendly for users with no programming or technical background. ○ On Linux, a non-programmer might manage with step-by-step instructions using a package manager. ○ On Windows, setup requires installing Python and using terminal commands, which is not suitable for non-technical users without assistance/research. ● Heavily relies on terminal commands. MAT2 is a command-line-only tool. There is no GUI, no visual interface, and no menu-driven interaction. Users must enter commands manually in a terminal to anonymize files.	
Security & Privacy Strength	**Encryption Standards** ● N/A — MAT2 is not an encryption tool. It focuses on metadata removal, not data encryption. It does not encrypt files. ● MAT2 does not use end-to-end (E2E) encryption or other security features. It simply strips metadata from files and outputs a cleaned version. It does not encrypt or transmit data. **Known Strength resilience** ● MAT2 can be used in regions with heavy censorship and surveillance because: ○ It runs entirely offline. ○ It doesn't require any server communication. ○ It can help reduce digital traces by stripping metadata. ● MAT2 does not include tools like VPNs, Tor, or proxy integration. It's strictly for metadata removal. ● It only removes known metadata types; embedded or proprietary metadata (e.g., custom application-specific data) can still remain. ● It does not analyze file content, so sensitive information embedded as visible or invisible text will not be removed.	

	• No real-time monitoring — users must manually process files. **Comparison with Known Standards** • MAT2 follows good privacy practices (e.g., local processing, no telemetry, and open-source transparency). • While not certified by any formal data protection standard (like ISO/IEC 27001 or GDPR compliance badges), it aligns with privacy-first principles and is used by security-conscious organizations like Tails OS. **Data Minimization** • Does not collect any data as it runs entirely locally (does not collect or transmit any user data). **Privacy Policy Accessibility and Clarity** • MAT2 does not have a traditional privacy policy because it does not handle any data externally. • The source code and behavior are publicly auditable on its GitLab page. • Privacy practices are implicitly handled by its offline, open-source design.	

Maintenance/Sustainability	**Community support**
	• It is moderately easy to get help, ask questions, or find solutions from other users or developers.
	  ○ MAT2 has an active issue tracker and discussions on its official GitLab repo and GitHub repo.
	  ○ Being a privacy-focused niche tool, it does not have a large community, but the developer is responsive to issues.
	  ○ Support is also available via communities like Tails OS and privacy-focused forums.
	Development active status
	• Actively maintained.
	  ○ The project receives multiple updates per year, with recent commits as of 2025.
	  ○ Updates include bug fixes, support for new file formats, and dependency improvements.
	• The development team is highly responsive.
	  ○ The lead developer, Julien Voisin, is active in maintaining the tool and responds to issues regularly on GitLab.
	  ○ Feature requests and bug reports are typically acknowledged within a few days.
	Funding and Sponsorship
	• Indirectly funded through its use in larger projects.	

	○ Tails OS (a privacy-focused live operating system) integrates MAT2 and receives NGO and foundation funding, which indirectly supports MAT2's development. ○ Supported by the Tails ecosystem which is supported by organizations like Tor Project, Access Now, and others. ● The software seems financially stable. ○ While MAT2 is a small project with no direct revenue model, it is maintained by a dedicated developer and used in well-funded projects like Tails. ○ Its open-source nature and low resource requirements make its continued maintenance viable.	
Performance / Effectiveness & Reliability	**Testing Environment Setup:** ● **Device:** Dell XPS 15 ● **OS:** Windows ● **Network:** WiFi **User Experience Observations** ● MAT2 is very lightweight; the command-line interface responds instantly. No GUI, so no visual lags. It felt smooth and responsive. ● Processing files is fast—even on low-end systems. **Speed & Responsiveness:** ● MAT2 is extremely fast and responsive — it launches instantly, processes most files in under 1 second, and gives immediate feedback. It's efficient even on low-end systems, with minimal delay or system impact, making it ideal for quick use in both local and remote environments. **Resource Usage:** ● **CPU/Memory**: Extremely low. Verified using **htop** on Linux—uses <1% CPU during use, minimal RAM. ● No background services, so no idle resource use. **Network Performance:** ● No bandwidth usage during normal use. ● MAT2 operates **entirely offline** (no ping, no network calls), making it ideal for air-gapped or low-connectivity environments. **Reliability** ● **Consistency**: Runs reliably across repeated file tests with 0% failure.	

	• **Uptime**: As a local tool, uptime depends only on system availability. • **Audits**: While no formal third-party audits found, the open-source nature allows peer verification and security reviews.	

Deployment Considerations:	**Open Source & Transparency:**
	• Yes, MAT2 is fully open-source, hosted on GitLab. Anyone can inspect, audit, or modify the code.
	Cloud vs. Local Deployment:
	• No cloud infrastructure needed.
	• MAT2 is a local command-line tool. It runs entirely on a local system — no servers, cloud services, or internet connection required.
	Dependencies:
	• Yes:
	  ○ Python 3.7+
	  ○ Required Python libraries (e.g., mutagen, PyPDF2, etc.)
	  ○ Optional: ExifTool for some metadata support.
	  ○ Dependencies are clearly documented in the README.md.
	Post-Deployment Maintenance
	• It's a lightweight tool with few dependencies, making it easy to update or maintain with pip or a package manager.
	• The codebase is clean, Pythonic, and modular. Developers familiar with Python will find it relatively easy to fork and customize.
	Merge/Sustainability:
	• The maintainer actively encourages contributions via merge requests (MRs) on GitLab.
	• Contributors can fork the repo, make changes, and submit a merge request through GitLab. The maintainer typically reviews MRs and issues promptly.	

4. Testing Scenarios		

- **Scenario 1**
  Installing MAT2 on Windows with Ubuntu Terminal

```
[redacted]:~$ sudo apt update
Hit:1 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://security.ubuntu.com/ubuntu jammy-security InRelease [129 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [128 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [127 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [2511 kB]
Get:6 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1200 kB]
Fetched 4095 kB in 4s (927 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
[redacted]:~$ sudo apt install mat2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  htop libnl-3-200 libnl-genl-3-200
Use 'sudo apt autoremove' to remove them.
Suggested packages:
  ffmpeg
The following NEW packages will be installed:
  mat2
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 68.8 kB of archives.
After this operation, 230 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy/universe amd64 mat2 all 0.12.2-1.1 [68.8 kB]
Fetched 68.8 kB in 1s (71.3 kB/s)
Selecting previously unselected package mat2.
(Reading database ... 52523 files and directories currently installed.)
Preparing to unpack .../mat2_0.12.2-1.1_all.deb ...
Unpacking mat2 (0.12.2-1.1) ...
Setting up mat2 (0.12.2-1.1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for man-db (2.10.2-1) ...
[redacted]:~$ mat2 --version
mat2 0.12.2
```

  - Above are images of the installation process of mat2. Also can verify if it is installed with **mat2 --version**.

- **Scenario 2**
  Using MAT2 with flags

```
[redacted]:~$ mat2 --help
usage: mat2 [-h] [-V] [--unknown-members policy] [--inplace] [--no-sandbox] [-v] [-l] [--check-dependencies] [-L | -s]
            [files ...]

Metadata anonymisation toolkit 2

positional arguments:
  files                 the files to process

options:
  -h, --help            show this help message and exit
  -V, --verbose         show more verbose status information
  --unknown-members policy
                        how to handle unknown members of archive-style files (policy should be one of: abort, omit,
                        keep) [Default: abort]
  --inplace             clean in place, without backup
  --no-sandbox          Disable bubblewrap's sandboxing
  -v, --version         show program's version number and exit
  -l, --list            list all supported fileformats
  --check-dependencies  check if mat2 has all the dependencies it needs
  -L, --lightweight     remove SOME metadata
  -s, --show            list harmful metadata detectable by mat2 without removing them
[redacted]:~$ mat2 -s "/mnt/c/Users/[redacted]/OneDrive/Pictures/Camera Roll/Alishan.JPG"
[+] Metadata for /mnt/c/Users/npson/OneDrive/Pictures/Camera Roll/Alishan.JPG:
    ApplicationRecordVersion: 2
    CodedCharacterSet: UTF8
    ColorSpace: sRGB
    CreateDate: 2025:04:29 15:49:01
    CurrentIPTCDigest: 3d3927fc8de05704713e9ed992f67888
    DateCreated: 2025:04:29
    DateTimeCreated: 2025:04:29 15:49:01+08:00
    DateTimeOriginal: 2025:04:29 15:49:01
    DigitalCreationDate: 2025:04:29
    DigitalCreationDateTime: 2025:04:29 15:49:01
    DigitalCreationTime: 15:49:01
    ExifByteOrder: Big-endian (Motorola, MM)
    ExifImageHeight: 2048
    ExifImageWidth: 1536
    IPTCDigest: 3d3927fc8de05704713e9ed992f67888
    ModifyDate: 2025:04:29 15:49:01
    OffsetTime: +08:00
    OffsetTimeDigitized: +08:00
    OffsetTimeOriginal: +08:00
    SubSecCreateDate: 2025:04:29 15:49:01.000+08:00
    SubSecDateTimeOriginal: 2025:04:29 15:49:01.000+08:00
    SubSecModifyDate: 2025:04:29 15:49:01.000+08:00
    SubSecTime: 000
    SubSecTimeDigitized: 000
    SubSecTimeOriginal: 000
    TimeCreated: 15:49:01+08:00
```

	• To see what flags are available to use with mat2, use command **mat2 --help**. • To see what metadata is in a file or image without removing it use the **-s** flag as shown above. • To remove metadata from a file using MAT2, simply run the tool without specifying any flags

5. Insights & Recommendations

Key Findings	**Strengths:** • **Comprehensive Metadata Removal:** MAT2 supports a wide variety of file types, including PDFs, audio files, video files, and images, ensuring that metadata is thoroughly stripped from all formats. • **Privacy-Focused:** The primary goal of MAT2 is to enhance privacy by anonymizing metadata, which often contains sensitive information. It's perfect for users who need to ensure their files don't contain traceable details. • **Easy to Use:** MAT2 operates via a simple command-line interface, making it easy to use even without advanced technical knowledge. You just need to run the tool without any additional flags for basic metadata removal. • **Open Source & Transparent:** MAT2 is open-source and actively maintained, meaning it can be independently verified, modified, and improved by anyone. This adds a level of transparency and trust, especially for privacy-conscious users. • **Lightweight and Efficient:** MAT2 is a lightweight tool with minimal system requirements, making it quick to install and run, and it doesn't burden system resources. **Weaknesses:** • **No Graphical User Interface (GUI):** MAT2 is strictly a command-line tool, which may be a barrier for users who prefer a graphical interface. This can be challenging for less technical users or those unfamiliar with terminal commands. • **Limited to Local Environments:** MAT2 works locally, meaning it doesn't provide cloud integration or online features for easy file management across multiple devices. Users need to manually handle files on their local machine. • **Limited Customization Options:** While MAT2 is excellent at stripping metadata by default, it doesn't offer much customization for advanced users who might want more granular control over the metadata removal process.

	• **Dependence on External Tools:** MAT2 requires additional tools like ExifTool for certain types of metadata removal, which can add complexity for new users and increase setup time. • **Only Python-Based:** MAT2 relies on Python, which may not be ideal for users who prefer tools built in other languages or for environments that do not easily support Python.
Suggested Improvements	• **Add a GUI**: A simple graphical interface would make it more user-friendly for non-technical users. • **Expand Customization**: Offer more control over which metadata to remove and allow custom rules. • **Batch Processing**: Enable processing of multiple files or entire directories at once. • **Support More Formats**: Add support for more file types like MKV, OGG, or spreadsheets. • **Automate Metadata Removal**: Integrate with cloud services to strip metadata automatically during uploads. • **Improve Documentation**: Provide more tutorials, use-case examples, and an expanded FAQ. • **Add Encryption**: Offer encryption options for users who want both metadata removal and file security. • **Integrate with Other Tools**: Support integration with privacy tools like Tor or Tails OS. • **Cross-Platform Support**: Add official support for Windows and macOS for wider compatibility.
Alternative Tools:	ExifTool, ImageOptim, FOTO, MKVToolNix, Shred
License	GNU Lesser General Public License (LGPL-3.0)
Cost/Resource Implications	**Total Cost of Ownership:** • MAT2 is free to use, as it is an open-source tool licensed under the GNU Lesser General Public License (LGPL-3.0). ○ Does not have any premium tiers or paid features. It's entirely open-source and free to use with no hidden costs for its basic functionality. • Minimal maintenance required for updates. • Third-party integrations may incur additional management effort. • Updates are free, but manual installation may be needed. • No license cost. • Installation requires basic technical knowledge.

	● Maintenance and troubleshooting may need time but are manageable with documentation/research.
Why is this useful to civil societies in authoritarian environments?	MAT2 is valuable for civil society organizations (CSOs), especially in authoritarian environments, where digital surveillance is common. ● It removes hidden metadata (like GPS, author info) from files, helping protect the identity of activists, journalists, or sources. ● For example, a CSO documenting abuse can use MAT2 to clean photos before sharing them, preventing location or device data from exposing individuals. ● It works fully offline and doesn't depend on the cloud, making it safer in regions with censorship or surveillance. ● While downloading it might be blocked in some countries (e.g., GitHub access restricted), MAT2 can be shared and run without internet once installed. Extra Note: If the current metadata only shows when you received or saved the image (like file system timestamps), but you're looking for original metadata like when the photo was taken, camera info, or GPS, you need to inspect the embedded EXIF metadata, which might be lost or stripped already. ● Use exiftool: ○ Install: **sudo apt install libimage-exiftool-perl** ○ Run: **exiftool path/to/image.jpg** ○ Look for: ■ Date/Time Original ■ Make / Model (camera) ■ GPS Latitude / Longitude ■ Software (used to edit) ■ Orientation, Shutter Speed, etc. --- 2. Alternative: Use mat2 in simulate mode mat2 -s image.jpg This will list metadata it would remove, but doesn't show full detail like exiftool. --- If you see only: ● File Modification Date/Time ● File Access Date/Time Then: ● The original EXIF data is likely missing or already stripped (possibly by the sender or during editing/sharing).

	• Some platforms (e.g., WhatsApp, Facebook, some email clients) automatically strip EXIF data for privacy. --- ✅ If EXIF is present, you'll see: Date/Time Original : 2022:03:05 15:42:01 Make : Apple Model : iPhone 13 Pro GPS Latitude : 40 deg 44' 54.12" N GPS Longitude : 73 deg 59' 8.52" W