



現實世界的 開源隱私工具

公民團體、獨立記者與個人隱私防護的實際運用

COSCUP 2026 | 匿名網路社群軌 | 08/08 10:50-11:20

三個場景，三種答案

公民團體與 NGO

人多、流動快、常跨國協作，可能已經被盯上。

需要：對外加密通訊、內部協作、接受匿名投遞、記錄封鎖

獨立記者

核心是保護消息來源，而非保護自己。

需要：清掉檔案裡的痕跡、保護靜態資料、隔離高風險作業

個人隱私防護

沒有特定對手，但每天被追蹤、被外洩。

需要：擋掉追蹤、取回裝置控制權、管好密碼

三個場景不約而同選開源工具，理由與代價下一頁說明。

上一場給了三個問題，這一場給對號之後的清單。**投影片會釋出，各位不用抄。**

■ 今天回去就能裝 ■ 要看一下說明 ■ 要花一個週末學，或需要換裝置 **第一步** 每個場景先做這兩個就好

為什麼選開源，以及它的代價

你得到什麼

可以被檢查。不必相信廠商的說法，可以自己找人看，或參考別人做過的稽核。

資料位置由你決定。自架時，伺服器在哪個法域、資料留多久，是你的選擇。

服務不會說收就收。廠商倒了、被併購、改政策，程式碼還在，你至少有機會找人接手。

你要付出什麼

沒有客服電話。多數專案人力吃緊，更新可能慢，正體中文介面未必完整。出問題時你面對的是 issue tracker（公開的問題回報系統），不是客服。

自架是持續成本。需要有人長期更新、備份、處理故障，不是一次性的工作。

學習曲線通常較陡。商業產品花錢買掉的那段設定與整合，開源要自己走完。

開源是一條不同的路，不保證比較好。它適合的情境很具體：你手上的資料外洩會有人受害、你需要知道資料放在哪裡、你不能接受服務說收就收。如果組織裡沒有工程師，先選公開實例與免安裝的工具，把自架那一類整批跳過。缺人手的時候，社群本身也是一個資源。

台灣的現況：你現在用的東西

LINE

一對一與群組聊天預設有 Letter Sealing 端對端加密，但要雙方的裝置與版本都支援才會生效，所以仍需逐台確認，不涵蓋所有功能與情境。
聊天備份存到 iCloud 或 Google 雲端，那一份不受 Letter Sealing 保護。
誰跟誰聯繫、何時、多頻繁，這層 metadata 一直都在。
營運公司在日本，資料治理與調取請求適用日本法。

Google Workspace

內容沒有端對端加密，服務方有能力讀取（企業版的用戶端加密（金鑰留在你手上、Google 讀不到）需要較高階方案與額外設定）。
美國公司，受 CLOUD Act（美國政府可要求美國業者交出資料，不論存在哪一國）規範。**關鍵不在公司國籍，在服務方手上有沒有可讀的內容**：Signal 與 OnionShare 同樣來自美國，但它們讀不到你的內容，收到調取令也交不出東西。
可用性、備份與災難復原做得比多數組織自架好，那是真實的優點。
整個組織綁在同一套帳號體系，一個帳號被盜影響範圍很大。

不必全部換掉，先分級。外洩會有人受害的（受訪者名單、檢舉內容、個案資料）搬走，行政與對外文宣留著。

今天就能做的：LINE 確認 Letter Sealing 已開、關掉雲端備份、敏感內容不進 LINE。Google Workspace 全員開兩步驟驗證、敏感檔案改放 CryptPad 或加密容器、離職流程確實收回權限。

場景一 公民團體：通訊與協作



Signal iOS / Android / 桌面 **第一步**

端對端加密通訊，個人與小群組的預設選擇。但註冊需要手機號碼，對想隱藏身分的來源就是風險。



CryptPad 瀏覽器，可自架 **第一步**

共筆、表格、投影片、白板都在瀏覽器裡做，伺服器端看不到內容。有公開實例可以直接用，不必自架。



Matrix / Element 全平台，可自架

去中心化的加密通訊網路。自架就能決定伺服器在哪個法域、資料留多久，也能跟外部組織互通。



Delta Chat 全平台

用既有的 email 帳號當傳輸層，對方不必註冊。加密是機會式的，雙方的軟體都支援自動金鑰交換才會端對端加密，否則是明文郵件。



Etherpad 瀏覽器，可自架

輕量即時共筆。沒有端對端加密，適合非敏感的公開協作。



Nextcloud 自架

檔案同步、行事曆與協作平台。資料放自己機器上，代價是要有人長期維運。

真實情境：跨境協作的團隊自架 Matrix，把伺服器放在對他們有利的法域。

沒有 IT 人力的組織：先用 Signal 與 CryptPad 的公開實例，自架那幾項跳過。

場景一 接受投遞與記錄封鎖

📶 OnionShare 桌面 第一步

一次性的匿名檔案分享、收件與聊天，走 Tor，不用另外架伺服器（你的電腦在傳輸期間就是那台服務）。

📶 OONI Probe 手機 / 桌面 第一步

實測某網站在某國、某電信商底下是否被擋，資料公開。用來佐證封鎖確實發生。

📶 GlobaLeaks 自架

開源檢舉平台，比 SecureDrop 好上手，適合中小型組織與公部門。

📶 SecureDrop 自架，走 Tor

Freedom of the Press Foundation 做的匿名投稿系統，設計上讓收件組織也不知道投稿者是誰。需要專職維運。

沒有 IT 人力的組織，要怎麼收匿名檢舉？ SecureDrop 與 GlobaLeaks 都需要專職維運，OnionShare 適合一對一傳檔、不適合當長期窗口。務實的做法有兩條。一是公開一個附 PGP 公鑰的信箱當最低限度管道，並誠實公告你們能提供的保護到哪裡。二是跟已經架好 SecureDrop 的媒體建立轉介關係，由他們收件。**不要自己架一個沒人維護的系統，那比沒有更危險。**

安全管道的最後一哩路

OnionShare 最容易失效的地方：那串 onion 網址本身要怎麼交給對方？貼進 LINE 或明文 email，前面做的匿名防護就白費了。

1 當面交換

掃 QR code 或寫在紙上。最安全，也最常被忽略。

2 拆成兩條管道

OnionShare 會產生網址與私密金鑰兩樣東西。分開走不同管道，對手要兩樣都取得才有用。

3 設定自動停止

傳完即關、或設停止時間。網址就算外洩，後果也有時限。

4 用既有的加密管道

雙方已經有 Signal 就用它傳網址。OnionShare 的價值在於檔案不限大小、不經過第三方伺服器。

工具不會消滅安全管道的問題，只會把它搬到別的地方。所以上一場才要先問「我要防誰」，再判斷這條管道對那個對手夠不夠用。

場景一 被擋住的時候：繞過封鎖

OONI 告訴你被擋了，接下來這一排負責讓你連得出去，但台灣平常用不到。出國、跨境協作、協助境外夥伴時，它就是必需品。

Tor Browser 內建橋接 桌面 / Android **第一步**

直連被擋時的第一步。設定裡選「使用橋接」即可，不需要另外找資源。

Snowflake Tor 傳輸外掛

靠世界各地志願者的瀏覽器當臨時代理，用瀏覽器視訊通話那種即時連線技術。取得最容易，也最難被整批封鎖。

obfs4 Tor 傳輸外掛

讓 Tor 流量看起來像隨機雜訊，躲過「比對流量長相」的偵測方式。需要另外取得橋接位址。

WebTunnel Tor 傳輸外掛

把 Tor 流量偽裝成一般 HTTPS 網站流量，在深度封包檢測（逐筆分析連線、判斷要不要放行的技術）下較不顯眼。

Briar Android

不依賴伺服器的通訊，平常走 Tor。斷網或封鎖期間可用藍牙、區域網路直接同步，專為抗議與災難情境設計。

架一個橋接給別人用 自架

有一台閒置的雲端主機就能做，成本很低。你架的橋接會給你認識的人用。

場景二 記者：檔案會說話，靜態資料要加密

mat2 命令列 **第一步**

清掉檔案內嵌的 metadata。照片的 GPS 與相機型號、文件的建檔者與修改歷程，一行指令移除。

Metadata Cleaner 桌面 GUI **第一步**

mat2 的圖形介面版本，拖進去就清掉，給不想碰終端機的同事。

ExifTool 命令列

檢視與編輯 metadata 的標準工具。清之前先用它確認裡面到底有什麼。

VeraCrypt 桌面

建立加密容器或加密整顆磁碟，跨平台。裝置被扣走時，加密是唯一真正有效的防線。

LUKS Linux

Linux 的標準全碟加密，安裝系統時就能開啟。

安全刪除的真相 觀念

SSD 會把寫入分散到不同區塊，你以為蓋掉的資料可能還留在碰不到的地方。真正的解法是全碟加密，需要銷毀時銷毀金鑰。

清完 metadata 之後，原始檔案還在硬碟裡。加密那一層是給「裝置離開你的手」那一刻用的。

一張照片裡藏了什麼

爆料者傳來的原檔 `mat2 --show demo.jpg`

Make	Apple
Model	iPhone 14 Pro
DateTimeOriginal	2026:08:07 14:32:11
GPSLatitude	25.0330 N
GPSLongitude	121.5654 E
BodySerialNumber	(單眼相機會有)

清乾淨之後 `mat2 demo.jpg`

No metadata found

圖還是那張圖，畫質沒變，原檔不動。
不碰終端機的同事用 Metadata Cleaner 拖進去就好。

時間是拍攝當下，能對上那天誰在現場。**GPS**貼進地圖就是地址。**機身序號**把同一台相機拍過的所有照片串成一組，包含來源以前公開發表的作品。Word 與 PDF 一樣帶 Author、Company、LastModifiedBy、修訂紀錄與 Creator，有時比照片更完整。

但清乾淨不等於安全，只處理了檔案本身。來源用什麼管道交給你、你們怎麼聯絡、手機裡還留著什麼對話，那些才決定他會不會曝光。內容可以否認，關係不行。

場景三 個人：瀏覽與網路

uBlock Origin 瀏覽器擴充 **第一步**

擋廣告與追蹤器，投報率最高，裝完就有效。

Mullvad Browser 桌面

反指紋追蹤但不走 Tor，日常速度正常。指紋是網站靠螢幕尺寸、字型等特徵認出你的方法。

Tor Browser 桌面 / Android

匿名瀏覽，同時擋掉指紋追蹤。

Pi-hole / AdGuard Home 自架

網路層過濾，家中所有裝置一次生效。

WireGuard 全平台

用一台雲端主機架自己的出口，在咖啡廳、機場這種不信任的網路上把流量帶出去，也可以反過來連回自家網路。

這一排門檻最低，而且對每個人都有效。uBlock Origin 裝完就生效，不需要任何設定。

場景三 個人：帳號、裝置與資料

 **Bitwarden / Vaultwarden** 全平台，

可自架 **第一步**

跨裝置密碼管理，Vaultwarden 是自架版。

 **KeePassXC** 桌面

純本機密碼管理，資料庫就是一個檔案。

 **Aegis Authenticator** Android

開源兩步驟驗證碼，可加密備份。

 **F-Droid** Android

自由軟體 app 商店，有追蹤或廣告會被標記。

 **GrapheneOS / LineageOS** Android

取回手機控制權。GrapheneOS 僅支援 Pixel。

 **Thunderbird** 桌面

郵件用戶端，內建 OpenPGP 加密。

 **Joplin** 全平台

加密同步的筆記，比雲端記事本安全。

如果今天只做一件事，選密碼管理器。多數人真正遇到的第一個資安事件是帳號被盜，而非被國家監控。

威脅升級之後：先做的三件事

收到律師函、發現被跟蹤、設備可能被扣。這種時候最不該做的，是花一個週末學一套新的作業系統。

1 什麼都先別刪

刪除紀錄本身可能構成滅證，而且對方手上多半已經有副本。先保存現狀，把螢幕截圖與信件原檔留好。第二場提到的「兩端定期刪除對話」是平時的習慣，走到這一步就停下來。

2 盤點範圍，不是換工具

誰已經知道你在做這件事、紀錄留在哪些系統、哪些是你刪得掉的。工具只能處理最後一項。

3 找人，而且先找律師

法律問題找律師。數位安全可找 Access Now 的求助專線 (accessnow.org/help)，24 小時、多語言、對公民團體與記者免費。台灣的社群也可以問。

設備不是你的（公司配發、沒有安裝權限）：優先選免安裝與可攜的做法。Tor Browser 有免安裝版，Tails 從 USB 開機不需要動到主機系統。不確定設備乾不乾淨時，把敏感作業整個移到那台之外。

等狀況穩下來再學這幾個：Tails（USB 開機、關機不留痕跡，但開了持續性儲存就有但書）、Qubes OS（不同工作分到不同虛擬機）、Whonix（強制所有連線走 Tor）。它們是重建工作環境時的選項，不是急救工具。

從哪一個開始：兩個不同的問題

你自己裝得多快

十分鐘：Signal、uBlock Origin、Bitwarden、KeePassXC、Aegis、Tor Browser、Mullvad Browser、OnionShare、CryptPad、Etherpad、Delta Chat、Metadata Cleaner、OONI Probe、F-Droid、Thunderbird、Briar、Tor Browser 內建橋接

一個下午：mat2、ExifTool、Matrix / Element、VeraCrypt、LUKS、Joplin、Pi-hole / AdGuard Home、WireGuard、Snowflake、obfs4、WebTunnel

一個週末或換裝置：Tails、Qubes OS、Whonix、GrapheneOS、LineageOS、Nextcloud、GlobaLeaks、SecureDrop、自架橋接

能不能交辦給不懂技術的同事

可以直接交辦：Signal、Bitwarden、uBlock Origin、Metadata Cleaner、Tor Browser

需要一次教學：CryptPad、Etherpad、OnionShare、Aegis、KeePassXC、VeraCrypt、Delta Chat、Joplin

需要有人固定維運：Matrix、Nextcloud、Pi-hole / AdGuard Home、WireGuard、GlobaLeaks、SecureDrop、Vaultwarden、自架橋接

組織導入順序：一次換一項，從對外通訊開始。一次全換必失敗，是最常見的失敗原因。

左欄是「我做得嗎」，右欄是「團隊做得嗎」。兩者不同，多數導入失敗只算了左邊。

如果你是工程師

服務只要接受使用者上傳檔案，就該預設清掉 metadata。使用者不必知道背後做了什麼，**預設保護比事後教育使用者有效得多。**

```
# 上傳流程裡加一步，mat2 也可以直接 import 成 Python 模組
subprocess.run(["mat2", "--inplace", upload_path], check=True)
```

EXIF 的 orientation 欄位決定直向照片顯示時要不要轉正。**全部清光但沒先把圖轉正，使用者的直向照片會躺著顯示**，很容易在 code review 被抓到。先依 orientation 轉正，再清 metadata。

也可以貢獻：翻譯介面與文件、回報問題、送 PR。多數開源專案人力吃緊，一個認真的 issue 就是真的貢獻。

還有：剛剛那幾張表就是一份需求清單，每一格都是某一群人真實遇到、而現有工具還沒解好的問題。

接下來，以及 Q&A

剛剛談過通訊、檔案與裝置。
還有兩種東西更難擺脫：**身分與金流**。

11:30 Denken Chen：身分隱私 **13:00** 教室 TR-511 ETHTaipei 合辦：匿名支付

這麼多工具，該從哪個開始？

回上一場的三個問題，先對號再挑。真要給一個答案：先裝密碼管理器。

自架會不會反而更危險？

會，如果沒有人長期負責更新。沒有專職人力就直接跳過所有自架選項，用信任的公開實例，並確認營運者是誰、資料在哪個法域。

我們組織要導入，從哪裡下手？

先確認威脅模型，再從對外通訊一項開始。一次換一項。

anoni.net/docs/

工具圖示取自 Simple Icons (CC0)，商標權仍屬各專案所有。未收錄者以功能性圖示替代。