Secrets Management Field Guide

1980-01-01T00:00:00Z

Scenarios

1. Someone close to you has died or been incapacitated, and pointed you here. You are in the right place. They were following a protocol — one that anticipated this exact moment.

- If you are familiar with this protocol, jump straight to Protocols.
- If you don’t know what this is, don’t worry. Just keep reading; it will tell you what kind of system this is and route you to what you need. Take your time. The protocol is designed to wait for you.

2. You’re handling a contingency — a possibly compromised password, a lost master password, an upcoming trip, or any other moment the system was built for. Go straight to Protocols for the step-by-step.

For everyone else

This is a guide to managing digital secrets — but probably not the kind you’ve seen before.

For better or worse, the digital world keeps stopping you in your tracks with screens that say “Set a PIN”, “Choose your password”, “Continue with Google?”, or “Please back up your 2FA recovery codes.” And whatever little ritual you have for getting past that screen probably comes with a flash of anxiety — “I must not forget this”, or “could a hacker get this?”, or “I should write it down somewhere so my wife can find it.” But like most people, you push those thoughts aside. You’ll do it “properly” later. Or maybe never. What are the odds anyway?

This guide exists because the answer, eventually, stops being reassuring.

Or perhaps that sketch isn’t you at all. You have a password manager, hardware keys, maybe even an offline backup. You’ve solved for attack and for mistakes. But there is a third failure mode that almost no system accounts for: what happens the day you cannot be there to maintain it?

Maybe you’ve thought of that, too. There’s an envelope somewhere, a lawyer briefed, the right person told. But a plan that has never been tested is just a plan you hope will work. And could a patient bad guy use your plan to your disadvantage?

Most approaches to this problem put you in one of two positions: hold everything yourself, or hand it to a corporation. This guide is built on a third option — one that most people already have and almost nobody has formalised. It is less about technology than you might expect. And it changes how you think about the problem entirely.

It is opinionated, practical, and free. Start with the Introduction →

Replace this page with your own bio.

Who I am

A brief introduction: who you are, what you do, why you are qualified or motivated to write this.

Why I wrote this

The motivation behind this project. What problem were you trying to solve? Who were you writing it for?

Contact

How readers can reach you. Email, a contact form, or a social profile — whichever you are comfortable with.

If you have found an error, want to suggest a correction, or want to contribute, the best way to reach me is [your preferred contact method].

This site has no ads, no paywalls, and no sponsored content. It is written in spare time and published freely.

If it has been useful to you — saved you time, helped you make a better decision, or given you something to send to a friend — a small contribution helps keep it going.

Ways to support

Ko-fi — one-off contributions, no account required. ko-fi.com/yourname

GitHub Sponsors — monthly or one-off, if you prefer to keep it in the GitHub ecosystem. github.com/sponsors/yourname

Buy me a coffee — another simple one-off option. buymeacoffee.com/yourname

Other ways to help

Money is not the only way to support this work:

- Share it. Send a page to someone who needs it. Word of mouth is the most useful thing.
- Report errors. If something is wrong or out of date, get in touch.
- Link to it. If you write online and reference something here, a link helps other people find it.

Thank you.

Content license

The written content on this site — all text in the manual, reference, and posts sections — is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0).

You are free to:

- Share — copy and redistribute the material in any medium or format
- Adapt — remix, transform, and build upon the material for any purpose, including commercially

Under the following terms:

- Attribution — you must give appropriate credit, provide a link to the license, and indicate if changes were made

Code license

The Hugo theme and templates that power this site are published under the MIT License.

MIT License

Copyright (c) [year] [your name]

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Attribution

If you republish content from this site, please credit it as:

> [Title], [Your Name], [URL], [date accessed]. Licensed under CC BY 4.0.

Introduction

Most privacy guides assume you are either alone with your secrets or you trust a corporation with them. This guide is built on a third option — and it changes everything.

Self-custody, third-party custody, and the option nobody talks about

When it comes to managing digital secrets, most people end up in one of two places.

The first is self-custody — you hold your own passwords, keys, and sensitive information, and nobody else has access. This is the gold standard for sovereignty. It is also, for most people, practically unachievable. One forgotten master password, one house fire, one stroke, and everything is gone. The burden is real and the failure modes are catastrophic.

The second is third-party custody — you hand your secrets to a corporation. A password manager in the cloud. A service that promises to store your digital estate. A platform you hope will still exist when your family needs it. This is convenient. It is also a fundamental betrayal of the thing you were trying to protect.

The Bitcoin protocol has been wrestling with exactly this tension, and the most elegant answer to emerge is a concept called second-party custody — pioneered by the Fedimint protocol. The insight is simple: between the corporation and the individual, there is a third option. People you actually know. People you can look in the eye. People who are embedded in your life and cannot disappear with your keys without consequence. Not a bank. Not a platform. A circle of trust — people who will notice if something goes wrong, and who have something to lose if they betray it.

This guide applies that insight to personal secrets management in its broadest sense — passwords, encryption keys, private documents, digital accounts, and the problem that almost nobody has solved: what happens to all of it when the bus hits you.

Who this is for

This guide is for everyone in the sense that everyone deserves control over their digital life. It is technical in the sense that some of what it describes requires a technical person to implement. That tension is resolved by the second-party model itself.

If you are reading this, you are probably the technical person. The computer wiz, as your grandmother would say. This guide is not just for you to help yourself — it is a tool for helping the people you care about. In some cases that means walking them through a process. In most cases it means acting as a trusted custodian of their digital privacy and security. The guide is written for that relationship.

If you are the non-technical person without a technical friend at hand, you can still use this guide. In the age of language models, a surrogate computer wiz is available to anyone. Paste the URL of this guide into an LLM and start asking questions. This is not the ideal arrangement — a human being who knows you and your family is always preferable — but it is a real starting point. The guide functions as a stable source of truth that you or a future technical person can return to and reason from.

The guide is also for people with genuinely high stakes — journalists, activists, people operating under repressive conditions. The protocol it describes is not designed specifically for them, but it scales in that direction. If that is you, read the manual first and then seek additional specialist guidance — this guide is a foundation, not a ceiling.

What this guide is not

It is not a corporate security awareness training programme. You know the kind — soul-destroying slide decks designed to generate compliance rather than understanding. This is the opposite of that.

It is not a James Bond manual. The romantic idea of the self-sovereign individual — fully independent, trusting no one, cloak-and-dagger techniques to outsmart every adversary — is both naive and dangerous. We are interdependent creatures. Trust is not a vulnerability. It is the foundation of every functioning society. The reason this guide exists is not that trust is bad. It is that sh!t happens, we make mistakes, and sometimes some people try to steal from us. We build systems that account for those people without pretending everyone is one of them.

If you currently have a cloak-and-dagger system, ask yourself one question: what happens when you have a stroke and lose half your memories? What happens when you are in hospital for a month and cannot maintain it? A system that only works while you are present, alert, and alive is not a system. It is a single point of failure.

It is not a learning resource. If you want to understand the tools and concepts in this space, Privacy Guides is excellent. Tools in this guide are not explained from first principles — the reference section points outward to better explanations than this guide could provide. The exception is where no good external explanation exists.

What this guide is

A highly opinionated, step-by-step implementation playbook for managing secrets within a circle of trust — typically a family and group of friends.

Opinionated not because this is the only way, or even the best way, but to relieve the burden of choice. The single greatest barrier to personal secrets management is not technical complexity. It is the overwhelming number of options. This guide removes that burden by making the decisions for you. The protocol described here is good enough for most people. It has been chosen for resilience, simplicity, and longevity — not for elegance or technical sophistication.

The guide is designed to solve for three failure modes: attack — someone actively trying to access your secrets, including under duress — where you are physically present and under pressure to hand over access; mistakes — human error in using or managing them; and loss — secrets becoming inaccessible because of death, incapacity, or forgotten credentials.

A protocol designed only for remote attacks is incomplete. The moment someone can compel you in person — a robbery, a border crossing, a coercive authority — the technical sophistication of your system becomes irrelevant unless the protocol has accounted for that scenario from the start.

Throughout, the emphasis is on the why as much as the how. A protocol you understand is a protocol you will maintain. A protocol you follow blindly is a protocol that will decay.

The rituals are the protocol

Most security guides treat the technical layer as the whole system. This guide takes the human layer equally seriously — and that means being concrete about what maintaining it requires.

The tools — the password managers, the encrypted vaults, the secret sharing schemes — are scaffolding. The load-bearing structure is human. It is the annual review where everyone checks in. It is the named person whose job it is to notice when something has changed. It is the simple habit that proves the system is still alive.

A circle of trust with no maintenance ritual is just a very well-intentioned arrangement that will quietly decay until the moment it is needed, at which point it will fail. The rituals prevent that. They are not administrative overhead. They are the product.

The manual makes this concrete. But it is worth knowing from the start what kind of guide this is — one that takes the human layer as seriously as the technical one.

The circle of trust is not a security weakness. It is the security model.

Structure

This site is a structured field guide. It is not a wiki, not a traditional blog, and not a documentation site — though it borrows something from each. The website format is for easy navigation and cross-referencing, but it reads like a book — and like a book, it will demand your attention. It’s long because the problem deserves it.

It is built around three kinds of content that serve different reading modes.

The manual

The manual is for reading in order. Each page builds on the last. Start at the beginning if you want to understand the whole subject from the ground up, or navigate to a specific chapter if you already know what you need.

The manual is opinionated and linear by design. It makes choices so you don’t have to.

The reference section

The reference section is for looking things up. Entries are self-contained: a comparison table, a glossary definition, a template you can copy. Dip in when you need something specific; you do not need to have read the manual first.

Posts

Posts are where the thinking happens before it is ready for the manual. Notes, updates, things that don’t yet have a permanent home. Loosely structured, dated, filtered by tag.

Where to start

If you are new here, start with the manual:

- If you want context first, begin at the beginning.
- If you have a specific problem, use the reference section.
- If you want to follow along as this develops, subscribe via RSS or check the posts.

More

Visitor tracking

This site does not employ any kind of visitor tracking. No Google Analytics. No Umami. No cookies. This is deliberate.

Engagement as a number is not an important metric here. Engagement as a relationship is. If something in this guide helped you, or if something is wrong, or if you have built on it in some way — that is worth far more than a pageview. Get in touch.

How this site is built

Built with Hugo, a static site generator. No JavaScript frameworks, no tracking, no cookies. The source is plain Markdown files. See the License page for terms of use.

About the author

Written pseudonymously — not out of secrecy, but because the guide should stand on its own merits. If the ideas are sound, they are sound regardless of who wrote them. If they are flawed, they should be challenged on those grounds.

The author is the technical person in their circle. This guide is what they built for themselves, and then could not find a good reason to keep private.

AI transparency

AI plays three roles in this project:

Editor. The primary reason I write this manually is the pure joy of it. But English grammar is not my strongest suit, and AI has been a useful second pair of eyes for catching bad spelling, clumsy sentences, and awkward transitions.

Critic. I use AI-as-a-judge to cross-reference this work against well-established sources — a first-pass agent for punching holes in the security reasoning before humans do.

Developer. I’m a software engineer by day, but I don’t hand-craft infrastructure the way I used to. I used Claude to build the Hugo-based site that pulls content from the corresponding Obsidian notebook — a workflow I can highly recommend.

This is the guided path through the manual. Start with the first chapter and move forward when each idea feels settled—there is no prize for rushing.

The introduction set out the framing — second-party custody, the circle of trust. This page is about the stakes, and about why this guide exists at all.

The risks

Poor secrets management is one of those problems where the odds are low but the impact is often catastrophic. The risks cluster into three real categories.

Someone may try to steal your accounts — at best a financial headache, at worst identity theft that follows you for years.

You may lock yourself out — a forgotten master password, a recovery code typed into the wrong box (and yes, it happens to careful people too).

Or you may be incapacitated or die, leaving the people closest to you unable to access anything that matters. The legal processes for handling that last scenario — even with a death certificate in hand — are routinely broken. Preparation is the only reliable alternative.

Why this guide exists

With the morbid picture painted above, this is not an attempt to scare you — life insurance ads do enough of that. And the fact that this guide is free is the first proof of that.

So what’s the actual hidden agenda?

I’m an avid advocate for individual freedom and sovereignty — and even more so for true, trusting relationships between people. That’s the bedrock of society. Both are quietly eroded when we hand our most sensitive information to corporations we barely understand. Secrets management done well is not just a security practice. It’s an act of care toward the people in your circle, and I’m using this very practical problem as a gateway to make that case.

Publishing it openly also means the community can challenge it. The more holes people punch in it, the better it becomes.

So yes — this is a passion project.

This section gives and end-to-end overview of the entire proposal so you can get a feel for it. After that we deep dive into each component

The core challenge

We need a way to store passwords in such a way that

1. We can easily use them on a day-to-day basis conveniently. This means that it is easy to access them and also “cockpit proof” meaning that the system as a whole is robust against lapses in concentration (or at least damage is limited to a small subset)
2. Impossible for other people to get access to them. (Note that impossible includes “not worth the trouble”)
3. Possible for trusted people only to get access to them after you die in a secure way
4. Robust against forgotten or lost element.

The solution to this challenge is the design of what we’ll call “the vault”.

The vault consist of tiered and partitioned components designed to allow you to make the choice of convenience VS security on a per-item basis as well as to limit the blast radius of mistakes.

The vault is split into tiers. Offline, Online, Hardware, Memory and Paper.

The Offline Tier serves as the root of the system and it is the least frequently accessed. This is a KeePass vault that lives on USB drives. It is offline in the sense that the vault itself is never in contact with the internet (or “air-gapped”) and only opened on a trusted machine with an operating system that has no access to any network (TailsOS). The Offline Vault contains the passwords to unlock the Online vaults.

The Online Tier consists of multiple online password managers. This guide assumes Bitwarden. These will be for day-to-day use. The are partitioned by “accounts”. So you have a vault for every email account that you have (say “personal” and “work”).

The Hardware tier is for passwords bound to a specific hardware device such as a Yubikey or the secure element in your phone or laptop.

The memory tier are those passwords that you carry around in your head. Importantly, these passwords are also on one of the other physical media. But they are in memory for convenience.

The paper tier are for passwords or “partial passwords” as we’ll see that is written on a durable physical medium such as paper or metal.

The sections detail exactly how to configure these and which kinds of passwords to store where. For now it is enough to understand that there is a vault design that if well organised will give you the benefits mentioned above.

A well-organised vault is also an inventory: a complete record of what accounts and credentials exist and where they live. This turns out to be as important as the credentials themselves — family members left to manage an estate often know that accounts exist, but not which ones matter, or where to find them.

The trust network

The vault is not worth anything if (1) you lose access to it due to error or forgetfulness or (2) those you want access to it after you die lose access to it because well.. you’re not there to help.

So the second component to the system is the social structure — what the introduction calls the circle of trust. This guide assumes that there are people in your life that you trust. If your circle is small right now, that is also something this guide can help you think through — the system scales down. Even one person is meaningfully better than none.

Here we will distinguish between two types of trusted people using the names “kith” and “kin”. (These are used as technical terms — not the common phrase “kith and kin” meaning all one’s friends and family together.) Importantly this is not different levels of trust. It is merely a case of logistics as the guide will show in detail. The basic distinction is that “kin” are those people that you actively work with to organise and review your vault and your risk profile. Kith are merely people that can help recover your vault in the event of loss or death. A two key element that this guide solves is

1. to eliminate the risk the kin and kith carry. All the information any one individual has is not able to compromise the vault This means they cannot be held under duress to do so. 2. After your death it is very difficult for a patient bad actor to access your vault

Protocols

These are a set of processes and rituals that you and your trust network adhere to ensure the security and recoverability of your vault.

They include things like a yearly review with a kin member. And of course the protocol for your kith to recover your vault if they wish after your death.

Checklists

Finally the guide gives a set of quick reference checklists that you can follow for given scenarios to give you peace of mind that you are not making mistakes. They included everything form deciding where a password must live, what to do if you suspect a password has been breached, what to take with on and overseas trip and so on.

Normal

Sending a password securely

- Bitwarden Send

Advanced techniques (for the techies)

- SSH using OpenPGP and a secure element: impossible to steal your SSH key

- Privacy Guides — practical, opinionated privacy tooling and concepts (mentioned in the draft introduction).
- Jameson Lopp
- Jolly Roger’s Security Guide
- Smart Custody (by Christopher Allen and Shannon Appelcline)
- The OffcierCia GitHub repo (Crypto-OpSec-SelfGuard-RoadMap)

Metadata is data about data. In the context of communications, it is everything except the content of a message: who you contacted, when, for how long, from where, and how often.

You can encrypt words and still leak patterns.

The former NSA director Michael Hayden: “We kill people based on metadata.” The point is not hyperbole — it is that metadata reveals patterns of behaviour, relationships, and intent more reliably than content.

What metadata reveals

- Social graph: who you know and how frequently you interact with them
- Location: where you were when you sent or received a message
- Behavioural patterns: sleep schedule, work hours, relationships, health concerns (from the timing and frequency of communications with certain contacts)
- Device fingerprint: what hardware and software you use

Where metadata leaks

Channel	Metadata exposed
Email	To, From, Subject, IP addresses, timestamps, mail servers
SMS / phone calls	Numbers, duration, cell tower location, carrier
Signal	Minimal — phone number, timestamp of last connection to server
HTTPS traffic	Destination domain (via DNS and SNI), data volume, timing
Encrypted messaging with cloud backup	Full message graph to backup provider

Reducing metadata exposure

- Use Signal for sensitive communications (minimal metadata)
- Use a VPN or Tor on untrusted networks to mask destination metadata — see vpn-comparison
- Be aware that reference/concepts/end-to-end-encryption protects content, not metadata

Signal is an end-to-end encrypted messaging app maintained by the Signal Foundation, a non-profit. It is the baseline recommendation for secure one-to-one and group messaging.

Key settings

Setting	Recommended value	Why
Note to Self	Enable	Encrypted personal clipboard across devices
Screen lock	On	Prevents shoulder-surfing
Screen security	On	Hides previews in the app switcher
Incognito keyboard	On (Android)	Prevents keyboard from learning message content
Registration lock	On	Prevents SIM-swap account takeover

Disappearing messages

Set a default timer on all new conversations. A sensible default for most people is one week; adjust per relationship. Disappearing messages do not protect against the other party screenshotting, but they reduce the value of device seizure over time.

Safety numbers

Before sending anything sensitive to a new contact, verify safety numbers out of band — by phone call, in person, or via a QR code scan. A safety-number change notification means a new device or reinstall; verify before continuing.

Limitations

Signal requires a phone number to register, which links your identity to a carrier. This is a metadata weakness. For higher-threat models, consider a VoIP number or a dedicated SIM. See the vpn-comparison entry for related network-layer considerations.