Branches and Gateways

Friends, services, employers, and the connections between them

February 2026

Real friends, online “friends”

The simplest way to understand how the vine works is to look at two kinds of relationships you already have: a real friend and a social media contact. Where each one lives on your vine tells you everything about how branches work.

A real friend

Your friend Sarah is a real person you trust. You scan a QR code, exchange profiles, both accept. Sarah’s vault connects directly to your vault at the root of your vine. This is a first-class connection: vault to vault, end-to-end encrypted, with mutual consent. You can message her securely. Share your location with her. Verify each other’s identity to third parties. The connection is between two people with nothing in between.

Sarah sees your real name, your verified email, whatever you chose to share. You see hers. There is no platform mediating the relationship. No service storing your messages on its servers. No algorithm deciding what you see. The connection is direct, private, and yours.

An online “friend”

Now consider someone you follow on a social media platform. You do not have a vault-to-vault connection with this person. You both have connections to the platform — the platform is a branch on your vine, and the social experience happens inside the platform’s service.

You might not even use your real identity on this platform. Maybe you authenticate with the service using your vault — proving you are a real person, not a bot — but present a pseudonym. The platform knows you are legitimate. Your followers see your chosen name. Your vault never shared secrets with the platform. It never needed to.

The people you interact with on the platform are not on your vine. They are on the platform. The platform provides the timeline, the interactions, the content, the social graph. Your vault provided the authentication. That is the right scope for that relationship.

> *A real friend connects to your root because the relationship is real, direct, and trusted. An online contact lives behind a service branch because the relationship is mediated by the platform. The structure of your vine reflects the actual trust topology of your life.*

This distinction is important because it shows that ZKT does not force you to share your real identity everywhere. It lets you choose. A real friend gets your real identity at the root. A social platform gets proof of legitimacy and whatever persona you choose to present. The vine accommodates both because the vine reflects how trust actually works: different relationships deserve different levels of access.

But what if the platform went further?

The gateway version of social media is the starting point: the platform authenticates you through your vault, but it still owns the experience, the content, and the social graph. Your posts live in their database. Your followers are their data. If you leave, you leave empty-handed.

A full ZKT social media platform would look different. You hold the encryption keys for your content. Your social graph is a set of vault connections that you own and can take with you. Your posts are encrypted at rest, and the platform needs your key to serve them. If you leave, you revoke the platform’s access to your keys. Your content goes dark on their system, but it is still yours — you hold the keys, and you can authorize a different platform to serve it. Your followers who connected through their vaults can follow you to the next platform because the connections are on the vine, not in the platform’s database.

That is the range. A gateway gives a platform vault-based authentication today with no other changes. A full implementation gives users portable identity, portable content, and portable relationships. Most platforms will start at the thin end. The architecture supports the full range whenever they are ready.

Gateways and vaults: two ways to join the vine

The social media example introduces something important: the platform does not need to run a full vault with hardware-isolated enclaves. It just needs to connect to the vine, verify users, and serve its experience. This is the difference between a gateway and a vault, and it is critical for understanding how the ecosystem grows.

Gateways: the on-ramp

A gateway — sometimes called a service-vault — is a bridge between an existing service and the vine. It handles connection management, authentication, and data requests. It does not perform cryptographic operations in hardware isolation. It does not hold user secrets. It is an adapter that lets existing services participate in the vine without rebuilding their infrastructure.

A social media platform runs a gateway. A retail checkout system runs a gateway. A content streaming service runs a gateway. These services do not need to handle your private keys or sign transactions on your behalf. They need to verify your identity (or pseudonym), serve their experience, and perhaps request a few pieces of personal data — your shipping address, your payment preference — through the connection when needed.

> **What a gateway does:** Connects to user vaults over the vine's messaging layer. Verifies user connections. Requests profile data or personal information per the accepted contract. Serves the service's own UI and experience. Adds a ring to the user's keychain for service-specific data (like order history or preferences) if the contract allows it.

Building a gateway is relatively lightweight. It is the low-barrier entry point for the ecosystem. Any service that can integrate an API can connect to the vine as a gateway.

Vaults: the real thing

A proper vault runs inside a secure enclave and performs sensitive operations in complete privacy. Your personal vault is this. It holds your keys, signs your transactions, manages your secrets, and enforces your policies — all inside hardware-isolated memory that even the vault provider cannot access.

But proper vaults are not just for individuals. An organization can run a vault that manages employee access, handles document signing, and enforces organizational security policies. A financial institution can run a vault that co-signs transactions in hardware isolation. A healthcare system can run a vault that manages patient consent and record access with full cryptographic enforcement.

What a proper vault does: Everything a gateway does, plus: performs cryptographic operations inside a hardware-isolated enclave, holds and manages secrets, provides attestation that proves the code running inside is genuine, and enforces policies that even the vault operator cannot override.

The distinction matters for adoption. Most services start as gateways. Some graduate to full vaults as their needs grow and as the vine connection deepens. A retailer that starts with gateway authentication might eventually run a vault that handles payment credentials in an enclave. The gateway-to-vault progression is the provider's version of the growth path — just as users start thin and grow, so do the services on the other end.

The keychain: a service adds a ring, not a filing cabinet

When a service connects to your vault, the mental model is not a filing cabinet where the service gets a drawer to stuff with data. The mental model is a keychain. Each service connection adds a ring to your keychain — a credential, a capability, something you hold and carry.

In the current model, the service holds everything: your account credentials, your preferences, your history, your payment data. It is all in the service's database. You are an entry in their system. In the vine model, the service adds a ring to your keychain. The ring might hold a loyalty membership, an encryption key for your records, a set of preferences, or a credential that proves your relationship with the service. But the ring is on your keychain. You hold it. You see it. You control it.

What the rings look like: A retailer's ring might carry a loyalty credential and a pointer to your order history. A streaming service's ring might hold the encryption key for your watchlist and preferences. A healthcare provider's ring might carry encryption keys for your medical records and a consent manifest. The actual data — the orders, the watchlist, the records — lives where it lives. The ring holds the key that unlocks it and the pointer that finds it. No bulk data in the vault. Just the credentials, keys, and references that give you control.

Not all rings are the same. A service connection can add two kinds of data to your keychain. Some rings carry data and secrets that you own and control — your encryption keys, your credentials, your consent records. These are transparent to you: you can see them, manage them, and take them with you. Other rings carry data and secrets that the service uses for its own operations — session tokens, internal references, service-specific keys. These are opaque to you: the service needs them, you hold them, but you do not need to understand them. What lands on your keychain depends on what the service needs, and the contract spells out which is which.

The keychain inverts the control relationship. Today, the service holds your credentials and your data, and you hope it manages both responsibly. With the keychain, you hold the keys. The data — your order history, your medical records, your watchlist — still lives in the service's systems. But the encryption key that unlocks it, the credential that proves your relationship, the pointer that locates it — those are on your keychain. If you revoke the connection, the service still has encrypted data in its database, but it can no longer ask your vault for the key to read it. You hold the keys. The service holds the data. That is the right division.

This is why the keychain matters for the growth path. On day one, a service adds a thin ring — just an authentication token. Over time, the ring accumulates more: a preference pointer, then an encryption key, then a credential. Each addition deepens the connection without the service ever building a database of user data that it has to protect, report on, and answer for when things go wrong.

> *Gateways are on-ramps. Vaults are the real thing. The ecosystem needs both. A gateway lets any existing service join the vine with minimal changes. A vault lets organizations handle secrets with the same cryptographic rigor as a user's personal vault. Most services start as gateways. Some grow into vaults. The vine supports both patterns.*

Bring your own identity: the work example

The most natural example of a branch — and the one enterprise security teams will immediately understand — is work. Today, your employer gives you an identity. They provision an account in their directory, issue you credentials, and manage your access to internal systems. When you leave, they deprovision the account and your access disappears.

In the vine model, you bring your own identity. Your vault is your identity provider. Your employer's vault or directory service accepts it.

How it works

Connect: You join a company. HR initiates a connection between the organization's vault and your personal vault. You review the organization's profile and contract: what data they need (name, email, role), what capabilities the connection will provide (access to internal tools, expense submission, document signing), and what policies apply. You accept. The organization accepts. The branch is established.

Access: The organization's vault acts as a branch on your vine. Through that branch, the org grants you connections to internal resources: Slack, GitHub, project management tools, internal documentation. You do not create separate accounts in each of these systems. The org's vault manages that layer. You authenticate once through your vault, and the org's vault handles the routing to its systems.

Leave: When you leave the company, the organization severs the connection. In one action, your access to everything behind that branch disappears. Slack, GitHub, internal tools, project systems — all gone. The org does not need to deprovision accounts in a dozen systems. The branch is cut, and everything that hung off it goes with it.

But here is what does not disappear: your identity. Your vault. Your other connections. Your personal vine is untouched. You brought your identity to work. You took it with you when you left. The employer never held your credentials in the first place.

Notice that the organization’s vault is a proper vault, not just a gateway. It manages employee access policies, enforces security rules, handles document signing with organizational keys, and may perform operations in a hardware-isolated enclave. It is a first-class participant on the vine with its own connections, its own policies, and its own security guarantees.

> *You do not get an identity from your employer. You bring one. The org’s vault is a branch on your vine that grants access to organizational resources. When you leave, the branch is severed. Your identity stays with you.*

How requests find you: the long branch

The vine is not transitive. You cannot see what is behind your connections. Connecting to your employer does not give you access to the employer’s payroll provider. Connecting to an exchange does not let you browse the exchange’s blockchain connections.

But when anything on the vine needs your secrets, the request finds you. This is the property that makes the vine work at scale: you see your connections, not the vineyard, but nothing that involves your keys can happen without passing through your vault.

Here is how that works across a long branch.

The NDA that finds you

You work for a consulting firm. The firm has a client. The client is working with an external legal service. The legal service needs you to sign an NDA for a specific project. The chain looks like this:

You → Your employer’s vault → Client project system → External legal service

You have no connection to the client project system. You have no connection to the legal service. You have never heard of the legal service. But the legal service has a document that needs your signature, and your signature requires your private identity key, and your private identity key lives in your vault.

Step 1: The legal service sends a signing request to the client project system through their connection.

Step 2: The client project system forwards the request to your employer’s vault through their connection.

Step 3: Your employer’s vault routes the request to your personal vault through your employment connection.

Step 4: Your vault receives the request. Your app shows you: your employer connection is requesting a document signature. The document is an NDA for Project X. You see the document. You see who is asking. You see what key will be used.

Step 5: You approve. Your vault signs the document inside the hardware-isolated enclave using your identity key, zeros the key from memory, and returns the signature back up the chain. The legal service has what it needs.

You never had a connection to the legal service. You never needed one. The request traveled through existing connections until it reached the vault that holds the required key. Each vault along the chain made its own routing decision based on its own policies. The legal service asked the client system. The client system asked the employer. The employer asked you. Nobody bypassed anyone. Nobody escalated privileges. The request found you because it needed your key, and no one can use your key without your authorization.

Your vault's audit log records the entire interaction: at this time, through this connection, this document was signed with this key, and you approved it. The trail is complete because the architecture makes it impossible for it to be incomplete — nothing involving your secrets happens without passing through your vault.

> *You cannot see through the vine. But the vine can reach you. Every request for your secrets travels through existing connections until it arrives at your vault, where you see it, review it, and decide. The vine is opaque outward and transparent inward.*

One vine, many patterns

The examples in this document are not separate features. They are different expressions of the same architecture. A real friend, a social media platform, an employer, a legal service three hops away — they all connect to the same vine, governed by the same principles, managed from the same control plane. What changes is the type of connection and the depth of trust.

The shape of your vine

Picture a single user's vine at a typical moment in their life.

At the root: A handful of real friends. Sarah, Marcus, your sister. Direct vault-to-vault connections with real identities, end-to-end encrypted. No platform in between. These are the people you trust with your actual name, your actual phone number, your actual location when you choose to share it.

On the social branch: A gateway connection to a social media platform. You authenticated with your vault but present a pseudonym. The platform provides the social experience — the timeline, the content, the interactions. Your hundreds of online contacts live inside the platform's service, not on your vine. The gateway is thin: it verified you are real and it carries your preferences on your keychain. It has no access to your secrets.

On the work branch: A proper vault connection to your employer's organization. You brought your identity. The org's vault granted you access to internal tools, project systems, and collaboration platforms. Behind the org's vault are connections you never see: payroll, HR, legal, IT infrastructure. You interact with the branch. The branch manages everything behind it.

On the financial branch: A vault connection to your bank and a gateway connection to a retailer. The bank runs a proper vault that can co-sign transactions in hardware isolation. The retailer runs a gateway that verifies your identity at checkout and requests payment credentials from your vault on demand. Same branch of your life, two different connection types, each appropriate to the relationship.

All of these coexist on one vine. Your vault is the root. The control plane is unified. The audit trail is complete across every connection. But the depth, the trust, and the type of connection are different for each — because the relationships are different.

Services grow too

The growth path applies to both sides of the vine. Users start with a vault and grow their connections. Services start with a gateway and can grow into vaults.

A retailer begins with a gateway: authenticate users at checkout, request payment credentials on demand, add a ring to the user's keychain for order history. That is enough to eliminate credential storage and reduce breach exposure. As the retailer matures, it might move to a proper vault that handles payment operations in a hardware-isolated enclave. The gateway was the on-ramp. The vault is the destination. But the on-ramp delivered value from day one.

An enterprise begins with a gateway for SSO: accept employee vaults as identity providers, route access to internal tools. Over time it deploys a proper organizational vault that enforces security policies, manages document signing, and handles sensitive operations in an enclave. The initial integration was one API. The full deployment is a security infrastructure. But the path between them is continuous, not a cliff.

This is why the ecosystem can grow. The barrier to entry is a gateway — lightweight, API-level integration. The ceiling is a full vault with hardware-isolated security. Every service chooses where it sits on that spectrum today and can move along it as the vine connection deepens.

> *The vine accommodates everything from a pseudonymous social media account to a hardware-isolated financial vault to a three-hop legal signing request — all governed from one control plane, all logged in one audit trail, all controlled by one person. That is the architecture. The branches are just the shape it takes in your life.*

The vine is the connections

The vine is not a single kind of connection. It is a family of patterns that share the same principles: mutual consent, self-describing profiles, scoped visibility, and cryptographic enforcement. What varies is the shape.

A friend at the root is the simplest expression: two vaults, direct, nothing in between. A social media gateway is the thinnest: authentication and a pseudonym, no secrets shared. An employer's vault is a proper branch: identity federation, resource access, organizational policies, with an entire network of connections behind it that you never see. A three-hop signing request is the vine at its most extended: a request that originates somewhere you have no connection to, travels through existing connections, and arrives at your vault because it needs your key and nobody else's.

These are not different systems. They are different shapes of the same vine. The gateway and the vault are two points on the same spectrum. The friend at the root and the legal service three hops away are two distances on the same architecture. What holds it all together is the profile — the self-describing contract that every node carries — and the vault, the control plane where your authorization decisions are made.

The vine grows the way relationships grow: some are direct and personal, some are mediated by institutions, some are deep, some are thin, and all of them can change over time. The architecture does not prescribe the shape. It provides the structure that lets any shape work — securely, privately, and under your control.

> *A friend, a platform, an employer, a legal service you have never heard of. Different relationships. Different connection types. Different depths of trust. One vine. One control plane. Yours.*