A complete guide on how to use Next.ID for your #Web2 & #Web3 Digital Identity! 🌈

What is Next.ID? 🆔

Next.ID is the world's first protocol offering **Decentralized-Identity-as-a-Service (DIaaS)**, is a decentralized identity aggregation protocol that integrates all **Web2** and **Web3** digital identities, offering a comprehensive and verifiable identity database for open-source developers and projects to innovate and build dApps on.

Next.ID creates an identity infrastructure that securely aggregates users' identities into Avatars, making it the primary touch point for interfacing with any app or dApp.

Built by www.hlabz.cc
Github: github.com/HLABZcc
ETH: **0xbff3ee7d3648ce6b7de82dea427c3a1629aaf671**

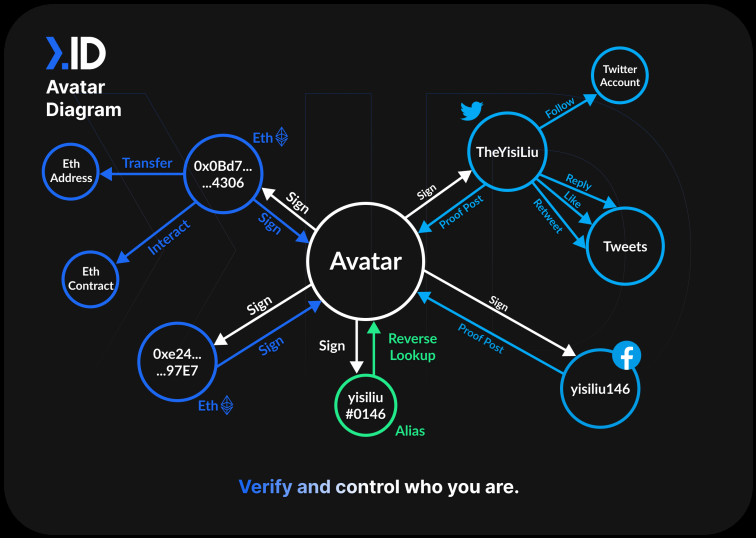
INTRODUCTION — What is Next.ID? 🏴

This is how Next.ID works:

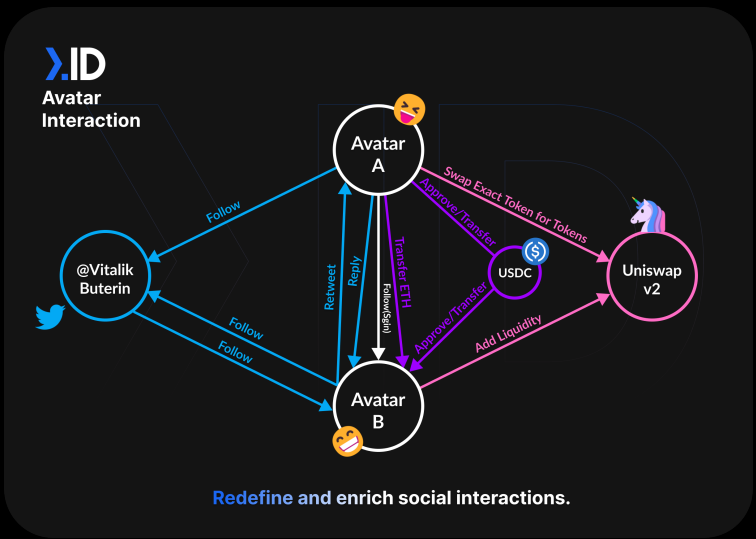
Next.ID's backend workflow is simple and streamlined. Next, let's review the basic concepts of **Next.ID** system design. This is also a guide for creating your first app with **Next.ID**.

Avatar in Next.ID

You can create avatars to perform actions (such as signing follow-ups) for on-chain and off-chain activities via **Next.ID**. **Next.ID**'s relational aggregation can map all these operations:



How the avatars of two users interact:



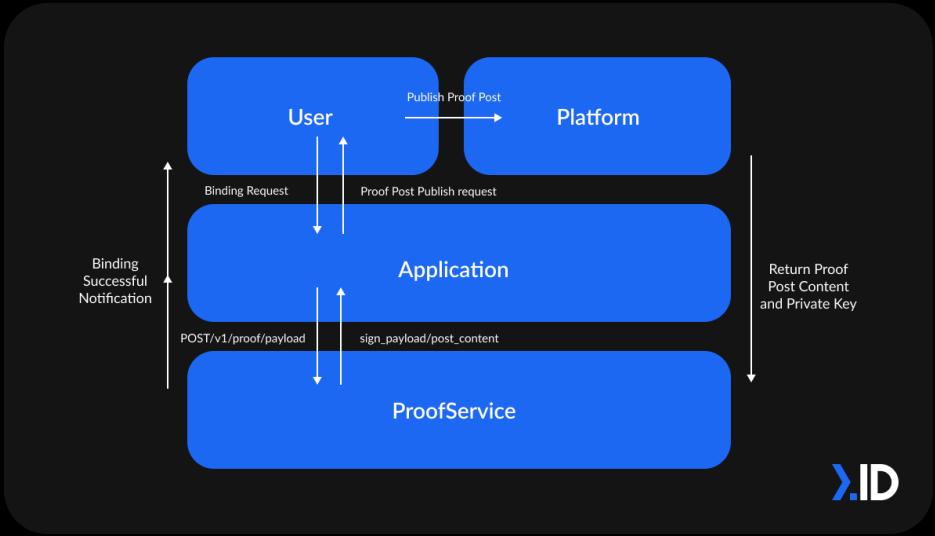
Proof of service

BIND THE AVATAR TO THE APPLICATION
In this case, the user requests an avatar to be bound to the application, which allows the avatar to be tied to the user account within the application. ProofService returns sign_payload and post-content based on application's POST /v1/proof/payload.

Then the app requests the private key of the user's avatar to generate a signature based on sign_payload. The user then publishes the proof to the target platform. Proof post link and ID with uuid and created_at are sent to ProofService on request and validated against application post content.

After the ProofService verifies the existence of the proof item and verifies the private key, it returns a binding success notification to the application and the user.

Example of a user's appearance: A user requests that an account (such as @nextdotid on Twitter) be tied to their avatar. Users sign transactions to verify ownership of Twitter accounts. The avatar then generates a public tweet with the signature. When a user tweets, the tweet along with the signature independently verifies the link between the user and the avatar.



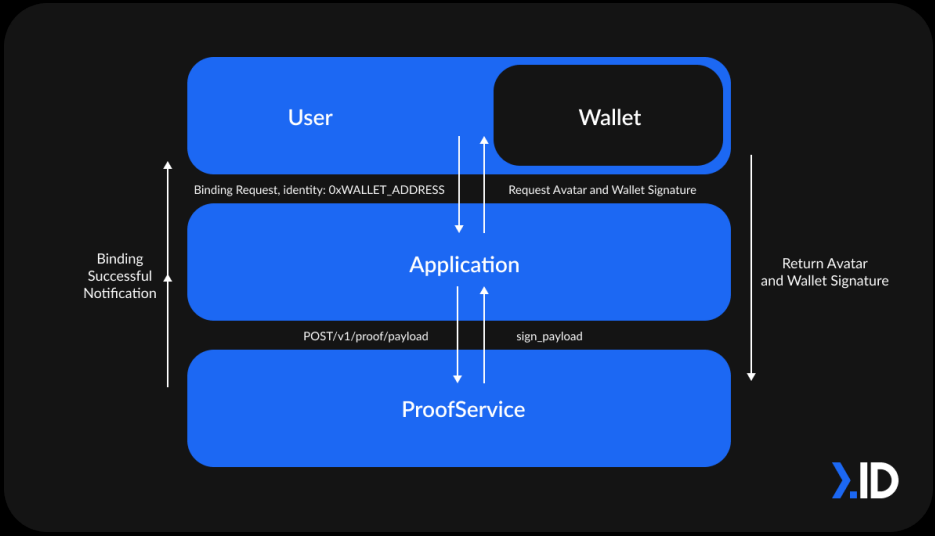
Bind the avatar to the application via the Ethereum wallet

In this case, the user requests an avatar to bind to an application with identity 0xWALLET_ADDRESS. ProofService returns the avatar's sign_payload based on the application's POST /v1/proof/payload.

Then the app requests the user's avatar private key to generate a signature based on Avatar sign_payload. Afterwards, the app requests the user's wallet private key to generate a signature based on Wallet sign_payload.

ProofService verifies uuid and created_at with sign_payload, then verifies avatar and wallet signature.

Afterwards, it returns a notification to the application and the user that the binding was successful.



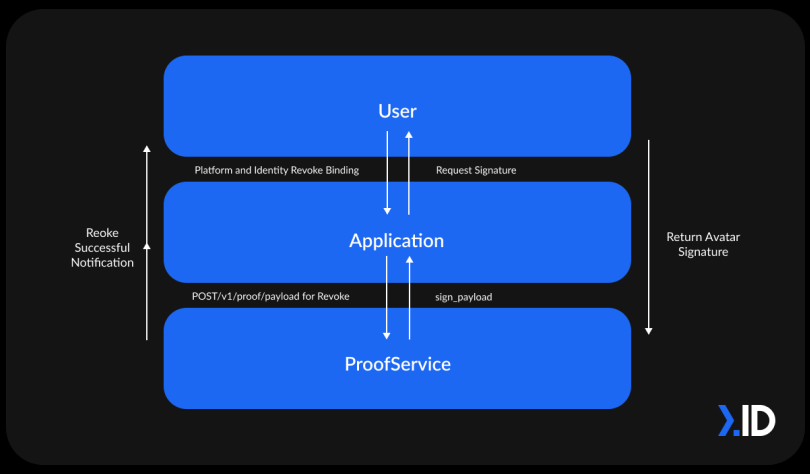
Revoke binding

In this case, the user revokes the binding previously created with the application.

ProofService returns the avatar's sign_payload based on the application's POST /v1/proof/payload as an undo operation. (Revoke action)

Then the app asks the user's avatar private key to generate a signature based on sign_payload.

After ProofService verifies signature with uuid and created_at from sign_payload, it verifies avatar and wallet signature for revocation. Upon completion, a successful revocation notification is forwarded to the application and user.

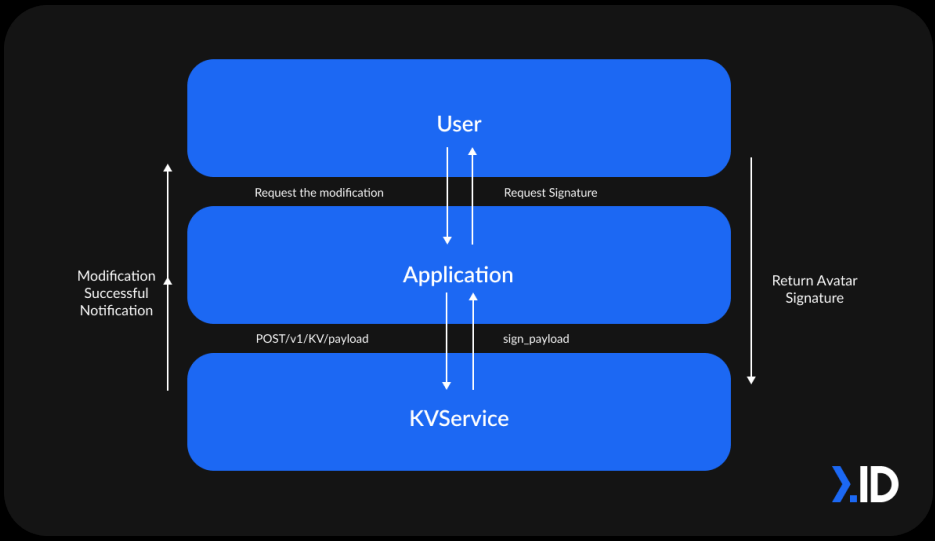


KV service

WRITE DATA
In this case, the user requests a change to the application.
KVService returns sign_payload according to the application's POST /v1/KV/payload.

Then the app requests the private key of the user's avatar to generate a signature based on sign_payload.

After KVService verifies signature with uuid and created_at from sign_payload, it verifies avatar signature for write data operation. Afterwards, the success notification is forwarded to the application and the user.



How tu use Next.ID? ✨

Now that you understand the value of using **Next.ID** for your **Digital Identity**, let's list some benefits of using it.

SOME BENEFITS WHEN USING NEXT.ID:

- A user requests that an account (such as @nextdotid on Twitter) be tied to their avatar. Users sign transactions to verify ownership of Twitter accounts. The avatar then generates a public tweet with the signature. When a user tweets, the tweet along with the signature independently verifies the link between the user and the avatar.
- ...

How can developers start building with Next.ID?

Next.ID is based on the **Decentralized Identifier (DID)**, a new type of identifier that allows users complete control over their digital identities.

DIDs are stored on a decentralized ledger like the blockchain, making them immutable and allowing users to prove their identity without the need for a central authority.

Developers can start using **Next.ID** by setting up a development environment and installing the necessary tools. **Next.ID** provides extensive documentation and resources for developers, including tutorials, code samples, and API reference.

To integrate **Next.ID** into a project, developers can use the **Next.ID SDK** or the **Next.ID REST API**. The **SDK** provides a set of libraries and tools for building applications that use **Next.ID**, while the **REST API** allows developers to interact with Next.ID using any programming language.

How can developers start building with Next.ID and what should the community expect in the long run to gain with the growth of the protocol?

Next.ID is a **decentralized identity aggregation protocol** that provides **decentralized identity as a service (DIaaS)**. The protocol integrates **#Web2 & #Web3 digital identities** and provides a comprehensive and verifiable identity database for open source developers and projects to use when building dApps.

First, developers can visit the protocol's website and access the documentation and resources available there. This may include tutorials and guides on how to use the **protocol API**, as well as sample code and other tools to help developers get started. Once developers have a basic understanding of how it works and how to use its **API**, they can start experimenting with building **dApps** that leverage the protocol's decentralized identity capabilities. This may involve creating test environments or sandboxes to try out different ideas and approaches, and iterating on those ideas until the **dApp** is ready for wider release.

The success of a protocol or platform is often determined by the strength of its community and the value it provides to its users. If it continues to provide valuable and useful services to developers and users, the community can expect continued growth and adoption of the protocol.

A potential growth area for **Next.ID** and the broader decentralized identity space is the development of more user-friendly and intuitive dApps. As decentralized technologies become more mainstream, there is a growing demand for dApps that are easy to use and understand, while providing users with a seamless and convenient experience. By providing a comprehensive and verifiable identity database, **Next.ID** and other decentralized identity protocols can facilitate the development of these types of dApps, driving the growth of decentralized applications (dApps).

CONCLUSION — Should you use Next.ID? 🤔

In conclusion, **Next.ID** is a revolutionary technology designed to bring new levels of security and convenience to the world of online identity verification.

Using blockchain technology and biometric data, users can easily and securely verify their identity without the need for traditional forms of identification. **Next.ID** not only makes it easier for individuals to prove their identity online, it also helps businesses and organizations simplify the verification process and reduce the risk of fraud and identity theft.

Overall, **Next.ID** is a promising solution for the future of authentication, and we can't wait to see how it develops and improves over the next few years.

The **Next.ID** community can look forward to continued development and growth of the protocol. **Next.ID** has an active roadmap outlining future planned features and partnerships.

The team is committed to building a decentralized digital identity ecosystem that is both secure and user-friendly. The community is also expected to play a key role in the development of **Next.ID**.

The protocol is open-source and community-driven, and the team encourages user input and feedback. By participating in the **Next.ID** community, developers and other stakeholders can help shape the direction of the protocol and ensure it meets the needs of users.

There are many other decentralized identifier (DID) protocols and projects on the market, but **Next.ID** stands out because of its focus on self-sovereign digital identities. While other **DID protocols** may focus on specific use cases or industries, **Next.ID** is designed to be a universal protocol that anyone can use for any purpose. **Next.ID** also stands out by offering developers a user-friendly interface and a comprehensive set of tools. The **Next.ID SDK** and **REST API** make it easy for developers to integrate **Next.ID** into their projects, and the extensive documentation and resources provided by the **Next.ID** team make it easy for developers to get started.

In summary, **Next.ID** is a protocol for creating and managing self-sovereign digital identities, giving users control over their personal data. Developing with **Next.ID** allows developers to create applications that enable users to use multiple identities for different purposes and share only the information they choose.

The **Next.ID** community can look forward to continued development and growth of the protocol, and developers can easily integrate with Next using the **Next.ID SDK and REST API.**

Ressources:
- Notion: https://nextid.notion.site/Next-ID-Home-0fb4ba9200d6458ab6c4fa81778f6a7b
- Docs: https://docs.next.id/
- GitHub: https://github.com/nextdotid/community
- Twitter: https://twitter.com/nextdotid
- Telegram: https://t.me/NextDotIDofficial
- Discord: https://discord.gg/tepa9K7E55