INTERNXT

White Paper

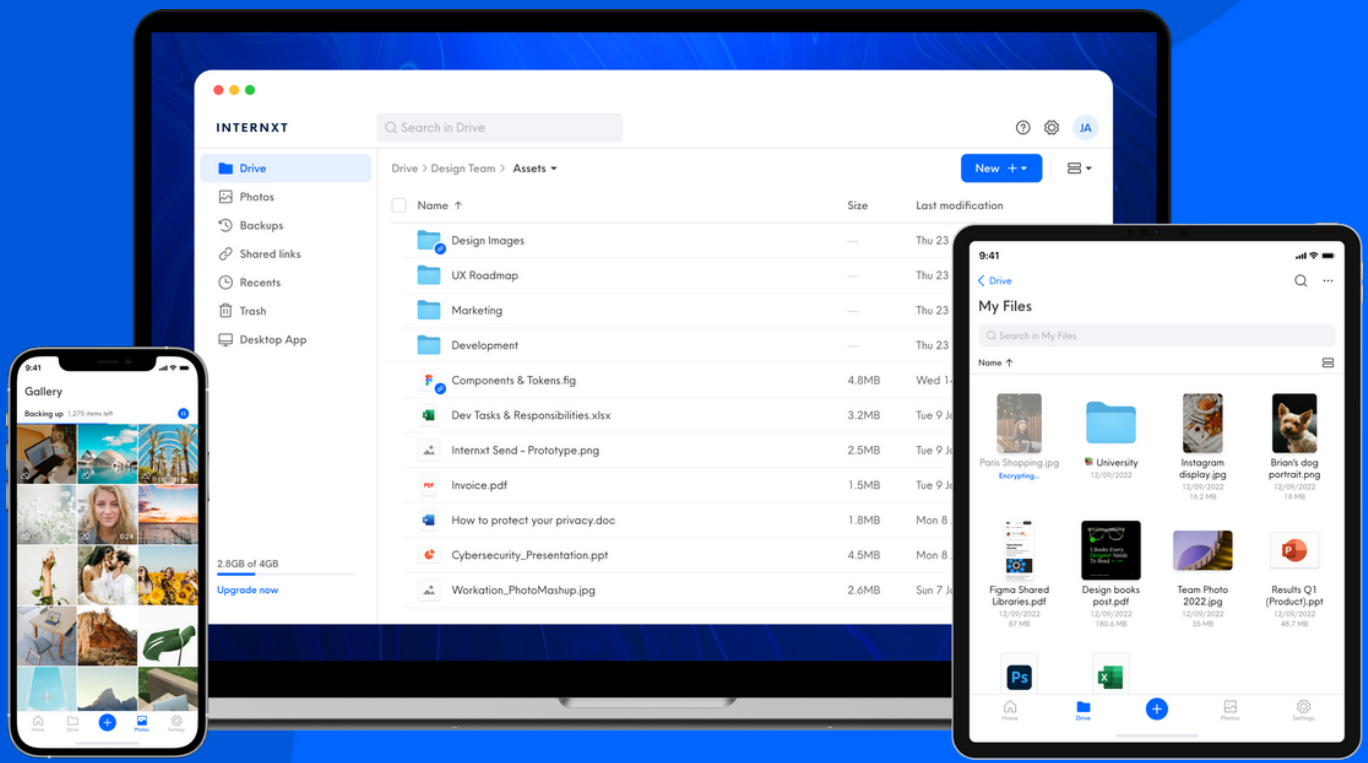
Last Update: October, 2023

Index

1. What is Internxt?

Internxt Drive is a state-of-the-art suite of cloud storage and data-sharing services that stands out as a cutting-edge solution offered by Internxt. As part of our mission to empower users and prioritize their privacy and security, Internxt Drive is designed to put the user back in control of their data.

At the core of Internxt Drive is a commitment to safeguarding user files and data, while respecting their fundamental right to privacy. By leveraging advanced technologies and implementing stringent security measures, Internxt Drive offers a highly secure and efficient platform for storing and sharing files. Internxt Drive has been meticulously built from the ground up with an unwavering focus on safety and security.



1.1 End-to-end encryption and zero-knowledge

In contrast to the majority of other cloud storage services, Internxt ensures that only the user has the ability to access the data stored on its platform. Since the beginning, Internxt was built to give users full control through zero-knowledge encryption. This means that data is encrypted directly on the user's device before it is sent to the platform.

The user exclusively holds the encryption keys necessary to access the data; not even Internxt has any access to them. If a user wants to share their data, they will encrypt the required encryption key using the recipient's public key before sharing it. This meticulous process guarantees complete data ownership and privacy, ensuring users have total control over their data.

1.2 Data storage

Internxt Drive revolutionizes cloud storage and data sharing by combining multiple cutting-edge technologies to deliver an exceptionally transparent, secure, and private suite of cloud storage and data sharing services.

Even if one server is compromised, or even the entire network, the information would not be accessible as the protocol to encrypt/decrypt requires the client's information.

In addition, Internxt Drive employs an end-to-end AES-256 encryption mechanism that guarantees files are encrypted on the user's device and can only be decrypted by the intended recipient. This means that not

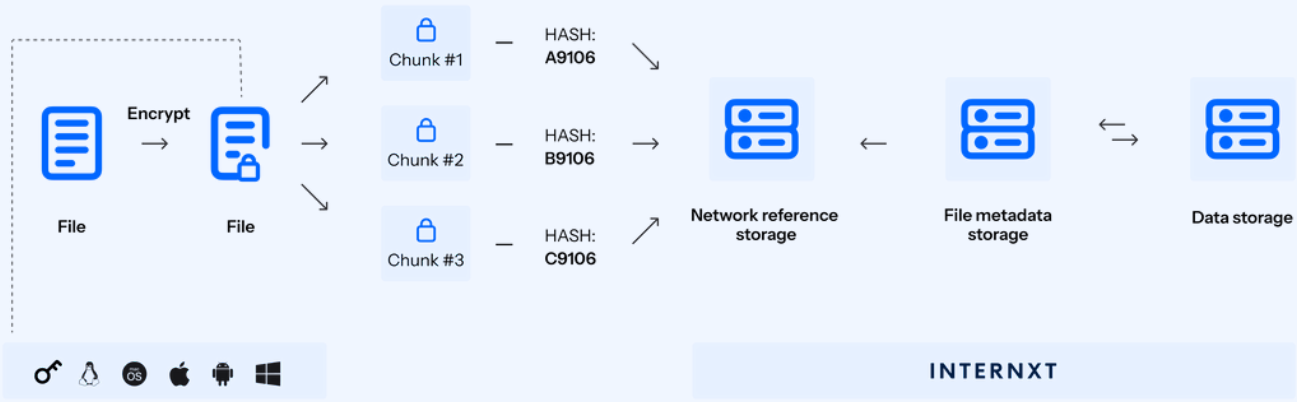
even Internxt has access to the decrypted data, providing unprecedented data privacy and protection.

By adopting a zero-knowledge system, we ensure that users have exclusive control over their encryption keys, preventing any third party, including Internxt, from accessing their data without authorization.

1.3 Open source

Internxt Drive's open-source code exemplifies our commitment to providing full visibility into our operations. By making our code open-source, users can scrutinize and verify the integrity of our system, ensuring that no hidden vulnerabilities or backdoors are compromising the security of their data.

The transparency of open-source software builds trust and confidence, as users benefit from having complete confidence in the security measures implemented within Internxt Drive. Also, we use the latest security tools implemented by edge-cutting technologists like GitHub and Sonar to scan the codebase for possible vulnerabilities daily and update the code with the necessary changes.



User's device

Windows, macOS, Linux, Android or iOS

Internxt internal API

Product API (Drive, Photos, Send)
Network API

OVHcloud network & storage

Erasure coding technique ensures files integrity when sent to OVH for storage

1.4 Transparency

All the necessary cryptographic operations occur directly on the user's device. Internxt ensures transparency in its implementation by making the complete and up-to-date source code of its applications publicly available. You can find links to the source code on the main homepage for easy access.

1.5 Privacy

Internxt prioritizes privacy through its implementation of zero-knowledge end-to-end encryption. This unique approach ensures that user data remains accessible only to the user, except when encryption keys are willingly shared.

While Internxt retains certain transactional metadata, such as user email addresses, it enforces privacy through its policy guidelines. You can review Internxt's comprehensive privacy policy on the main homepage for further details.

1.6 Compliance

Internxt is operated with a strong commitment to achieving the highest level of compliance with regulatory requirements. The services provided by Internxt are subject to governance under Spanish law. While Internxt ensures that it cannot view the data stored on its platform, it does take swift action to remove copyrighted content when it is reported.

Internxt users have the capability to share data via the platform using public links. These public links automatically include the necessary decryption key within the URL hash when users choose to create them for a file or folder. In the event that Internxt receives abuse reports or

notices, it acts promptly by removing or disabling access to the specific offending file or files, including folders, depending on the nature of the request. These actions are taken in accordance with the Terms of Service agreed upon by every registered user.

2. Technology and innovation

We have implemented the protocol on React Native through a new library, as no-one has written before an AES-256 (CTR mode) library for React Native using Android & Swift source code exposed to React Native. The use of the Advanced Encryption Standard (AES) algorithm with a 256-bit key length is employed to encrypt data during transmission and storage on Internxt Drive on react Native.

Through end-to-end encryption, data is encrypted on the client-side before being uploaded to Internxt servers. It remains encrypted until the authorized user downloads and decrypts it, ensuring that only the authorized user can access the unencrypted data. Internxt's implementation of AES-256 encryption guarantees a high level of security for stored data, effectively safeguarding it against unauthorized access, interception, and data breaches. Encryption keys are never stored on Internxt servers, further enhancing data security and reducing the risk of data breaches. This feature plays a crucial role in preserving user privacy, protecting sensitive information, and ensuring compliance with data protection laws such as GDPR.

3. Registration and log in

3.1 Registration process

When registering for an account, the user must input their email address and password. It's important to note that Internxt does not store passwords in their raw form. Instead, we generate:

- A unique random key, 256-bit long (mnemonic), using the bip39 protocol used as the seed to derive any new encryption key, which is unique per file.
- A pair of keys (private/public) using the ed25519 algorithm

We encrypt the mnemonic with the user's password and then hash the password, also, we encrypt the private user's key with the mnemonic, everything on the client. Then, we send that information to the backend.

The password hashing process is as follows:

- A cryptographically secure random 256-character value is generated as a salt.
- The hash algorithm used is SHA-1.
- Iterations are set to 10,000.
- The bit length is configured as 256.
- The PBKDF2 function is applied to the password, salt, iterations, hash, and bit length, which is then converted into base64.

3.2 Login process

The login procedure operates as outlined below:

1. The user is required to input their email address and password via the client interface.
2. The email address is transmitted to the API.
3. If a matching record with the provided email address is found in the database, the API will respond by supplying the user's salt.
4. In cases where the email address is not located in the database, the API will respond with a randomly generated salt. This step serves to safeguard against brute force attacks aimed at email addresses.
5. Subsequently, the client can calculate the user's master key and the hashed authentication key, following the same process as described in the registration phase.
6. After computing the hashed authentication key, the client forwards both the email address and the hashed authentication key to the API. The API will then respond with the user's API key if the authentication process is successful.
7. In the event of a user's initial login, the client encrypts their master keys and transmits them to the API.

Additionally, an RSA-OAEP keypair (with a 4096-bit modulus and SHA-512 hash) is created, encrypted using the user's master key (AES-GCM 256-bit encryption), and sent to the API. It's important to note that Internxt never stores unencrypted keys.

These encrypted keys are used to ensure that other clients, which the user may utilize, can decrypt data. For instance, during a password change, Internxt appends the newly derived master key from the new password to the old master key. Internxt refers to this as "master key chaining," simplifying the process of password changes.

4. Cloud Drive encryption and decryption

4.1 Upload encryption

Every individual file is assigned its own encryption key which is used to encrypt not only the file data but also its name, metadata, and folder names.

To encrypt the files, a secure and random 256-bit key is generated. Each file is divided into 10-50 MB segments, which are then encrypted using AES-GCM with a 256-bit encryption key. This encryption method also includes authentication to guarantee the integrity of the data. These encrypted file segments are subsequently uploaded to the API

4.2 Downloads

When a client initiates a file download, it follows a process of downloading the encrypted file segments, decrypting them, and then sequentially streaming the decrypted segments into the user's local file system or the designated browser download directory.

For folder downloads, the procedure is similar, except that all folder contents are zipped on the client side before downloading. It's important to note that due to browser memory limitations, the size of folder downloads is restricted.

INTERNXT

Stand for privacy, switch to Internxt