bitwarden | Help Center Article

SECURITY

Encryption

View in the help center:
https://bitwarden.com/help/what-encryption-is-used/

Encryption

Bitwarden uses AES-CBC 256-bit encryption for your vault data, and PBKDF2 SHA-256 or Argon2 to derive your encryption key.

Bitwarden **always** encrypts and/or hashes your data on your local device before anything is sent to cloud servers for storage. **Bitwarden servers are only used for storing encrypted data.** For more information, see Storage.

All vault data is encrypted by Bitwarden before being stored anywhere. To learn how, refer to the Bitwarden Security Whitepaper. Bitwarden is a zero knowledge encryption solution, meaning you are the only party with access to the keys required to decrypt the vault data.

> **Tip**
>
> If you'd like to learn more about how these encryption keys are used to protect your vault, you can also check out our Security Whitepaper.

AES-CBC

AES-CBC (cipher block chaining), used to encrypt vault data, is a standard in cryptography and used by the US government and other government agencies around the world for protecting top-secret data. With proper implementation and a strong encryption key (your master password), AES is considered unbreakable.

PBKDF2

PBKDF2 SHA-256 is used to derive the encryption key from your master password, however you may choose Argon2 as an alternative. Bitwarden salts and hashes your master password with your email address **locally**, before transmission to our servers. Once a Bitwarden server receives the hashed password, it is salted again with a cryptographically secure random value, hashed again, and stored in our database.

The default iteration count used with PBKDF2 is 600,001 iterations on the client (client-side iteration count is configurable from your account settings), and then an additional 100,000 iterations when stored on our servers (for a total of 700,001 iterations by default). The organization key is shared via RSA-2048.

> **Tip**
>
> The number of default iterations used by Bitwarden was increased in February, 2023. Accounts created after that time will use 600,001, however if you created your account prior to then you should increase the iteration count. Instructions for doing so can be found in the following section.

The utilized hash functions are one-way hashes, meaning they **cannot be reverse engineered** by anyone at Bitwarden to reveal your master password. Even if Bitwarden were to be hacked, there would be no method by which your master password could be obtained.

Changing KDF iterations

Bitwarden uses a secure default, as mentioned above, however you can change the iteration count from the **Settings → Security → Keys** menu of the web vault.

Changing the iteration count can help protect your master password from being brute forced by an attacker, however should not be viewed as a substitute to using a strong master password in the first place. Changing the iteration count will re-encrypt the protected symmetric key and update the authentication hash, much like a normal master password change, but will not rotate the symmetric encryption key so vault data will not be re-encrypted. See here for information on re-encrypting your data.

Setting your KDF iterations too high could result in poor performance when logging into (and unlocking) Bitwarden on devices with slower CPUs. We recommend that you increase the value in increments of 100,000 and then test all of your devices.

When you change the iteration count, you'll be logged out of all clients. Though the risk involved in rotating your encryption key does not exist when changing KDF iteration count, we still recommend exporting your vault beforehand.

Argon2id

Argon2, the winner of the 2015 Password Hashing Competition, is available as an alternative to PBKDF2 (learn more). There are three versions of the algorithm, and Bitwarden has implemented Argon2id as recommended by OWASP. Argon2id is a hybrid of other versions, using a combination of data-depending and data-independent memory accesses, which gives it some of Argon2i's resistance to side-channel cache timing attacks and much of Argon2d's resistance to GPU cracking attacks (source).

By default, Bitwarden is set to allocate 64 MiB of memory, iterate over it 3 times, and do so across 4 threads. These defaults are above current OWASP recommendations, but here are some tips should you choose to change your settings:

- Increasing **KDF iterations** will increase running time linearly.
- The amount of **KDF parallelism** you can use depends on your machine's CPU. Generally, Max. Parallelism = Num. of Cores x 2.

> **Note**
>
> Argon2id users with a KDF memory value higher than 48 MB will receive a warning dialogue every time iOS autofill is initiated or a new Send is created through the Share sheet. To avoid this message, adjust Argon2id settings or enable unlock with biometrics.

Invoked crypto libraries

Bitwarden does not implement any cryptographic primitives. Bitwarden only uses cryptographic primitives from popular and reputable crypto libraries that are written and maintained by cryptography experts. The following crypto libraries are used:

- JavaScript (Web vault, browser extension, desktop, and CLI)
  - Web crypto
  - Node.js crypto
  - Forge
- C# (Mobile)
  - CommonCrypto (iOS, Apple)
  - Javax.Crypto (Android, Oracle)
  - BouncyCastle (Android)