Diceware Lists and BIP39 Words: Different Jobs

October 30, 2026 · Document #413 · bip39-phrase.com

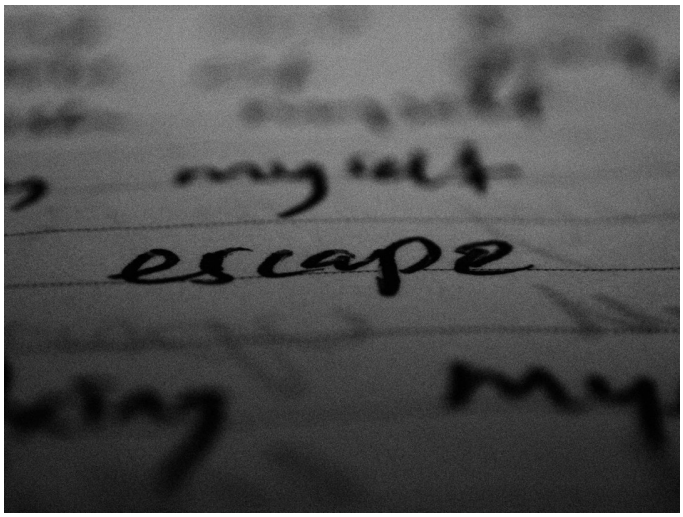


Photo by Ali Shah Lakhani on Unsplash

Diceware generates passwords; BIP39 encodes wallet entropy. Learn why dice-selected words suit the optional passphrase, not recovery words.

Contents

Diceware selects words for passwords	2
BIP39 encodes entropy and a checksum	2
Why Diceware fits the separate passphrase	2
Generate and verify the two inputs separately	3
FAQ	3
Sources	4
Sample sequences of BIP39 words	4
Reading a BIP39 phrase	5
See also	6

A Diceware passphrase is not a substitute for a wallet’s BIP39 recovery phrase. Both use readable words, but Diceware selects a password, while BIP39 encodes random data with a checksum. Diceware words can instead supply the optional BIP39 passphrase, a separate input used alongside the recovery phrase in wallets that support it. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

Diceware selects words for passwords

A traditional Diceware list contains `7,776` entries because five six-sided dice have `6^5 = 7,776` ordered outcomes. Each outcome identifies one entry. Roll five dice once, or one die five times, for each word. Read the results in a fixed order; adding the faces together would discard most possible outcomes. The original list includes words, abbreviations and short character strings. ([theworld.com](https://theworld.com/~reinhold/diceware.html))

EFF's Long Wordlist uses the same five-dice selection method. Its purpose is to produce memorable passwords through random selection. The words do not encode a wallet format, and the final word has no checksum role. EFF also publishes shorter lists using four dice, so check which list you downloaded before starting. ([eff.org](https://www.eff.org/dice))

With fair, independent rolls, each selection from a 7,776-entry list contributes approximately `12.925` bits of entropy. Six words provide about `77.55` bits; eight provide about `103.40` bits. These figures follow from `n × log2(7776)` and assume the words remain distinguishable in the final string. Choosing attractive words yourself or rearranging them into a sentence invalidates that calculation for the resulting passphrase. ([theworld.com](https://theworld.com/~reinhold/diceware.html))

BIP39 encodes entropy and a checksum

BIP39 uses a `2,048`-entry wordlist for the chosen language, where each word represents an 11-bit index. Its permitted mnemonic lengths are `12`, `15`, `18`, `21` and `24` words. Generation starts with `128` to `256` random bits, in increments of `32`, then appends checksum bits derived from SHA-256 before converting the result into words. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

For 12 words, `128` entropy bits plus `4` checksum bits occupy `132` bits. For 24 words, `256 + 8 = 264` bits. Consequently, only `1/16` of uniformly selected 12-word sequences from a BIP39 wordlist pass the checksum; for 24 words, the fraction is `1/256`. The checksum catches most random errors, but not all, and adds no secret entropy. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

This explains why ordinary Diceware selection is not a BIP39 generation procedure: it uses a different vocabulary and does not construct the checksum. Accidental validity is possible if every selected word belongs to the chosen BIP39 list, the length is permitted and the checksum matches. That coincidence does not establish that the phrase was generated with uniformly random BIP39 entropy. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

Why Diceware fits the separate passphrase

The optional BIP39 passphrase has no required wordlist or word count. Diceware therefore fits this input: its randomly selected words become a text string. They remain separate from the numbered recovery words. The nickname "25th word" can mislead because this input may contain several words, and BIP39 recovery phrases need not contain 24 words. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

Recovery requires the same recovery phrase and passphrase, with compatible wallet settings. Capitalization and spaces matter; BIP39 applies Unicode NFKD normalization before deriving the seed. A typo that changes the normalized passphrase derives a different seed. BIP39 itself provides no wrong-passphrase check, although

wallet software may offer confirmation prompts or warnings. Keep a secure written backup; memorization alone creates an avoidable recovery dependency.
([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki), [trezor.io](https://trezor.io/guides/backups-recovery/advanced-wallets/what-is-a-passphrase))

Adding a passphrase derives a separate wallet rather than moving existing funds automatically. Before transferring substantial funds, test that you can leave the wallet, enter the passphrase again and recover access to the same receiving address. Losing that passphrase can permanently prevent access even when your recovery words remain intact.
([trezor.io](https://trezor.io/guides/backups-recovery/advanced-wallets/what-is-a-passphrase), [trezor.io](https://trezor.io/guides/trezor-suite/using-a-passphrase-wallet-in-trezor-suite))

Generate and verify the two inputs separately

Use this workflow when combining a BIP39 wallet with a dice-generated passphrase:

1. Generate the recovery phrase through the wallet's documented BIP39 procedure. Keep its words in their original order. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))
2. Obtain EFF's Long Wordlist and physical dice. Check your wallet's passphrase support and input limits, then choose your word count before rolling; EFF recommends at least six words from this list. ([eff.org](https://www.eff.org/dice), [trezor.io](https://trezor.io/guides/backups-recovery/advanced-wallets/what-is-a-passphrase))
3. Record five ordered die results per word, look up each entry and retain the resulting sequence. Do not substitute personally meaningful words. ([theworld.com](https://theworld.com/~reinhold/diceware.html))
4. Separate the words consistently, record the complete string and verify that it fits your wallet's character and length limits. Do not truncate it to make it fit. ([eff.org](https://www.eff.org/dice), [trezor.io](https://trezor.io/guides/backups-recovery/advanced-wallets/what-is-a-passphrase))
5. Enter that string into the wallet's separate passphrase field. Securely back it up and test that re-entering it opens the same wallet before funding it. ([trezor.io](https://trezor.io/guides/trezor-suite/using-a-passphrase-wallet-in-trezor-suite))

FAQ

Can I choose recovery words from the BIP39 list myself?

List membership alone is insufficient. Secure BIP39 generation requires random entropy and the prescribed checksum construction. Choosing words for their meaning does not provide that randomness. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

Must my optional passphrase use BIP39 words?

No. Diceware words can supply it, subject to the wallet's input limits. Preserve the complete text, including capitalization and separators.
([trezor.io](https://trezor.io/guides/backups-recovery/advanced-wallets/what-is-a-passphrase))

Can physical dice help generate a BIP39 recovery phrase?

Yes, through a procedure that converts rolls into unbiased entropy and applies BIP39 encoding and its checksum. Direct Diceware word lookup omits those steps. Use a documented procedure supported by your wallet rather than inventing a conversion. ([github.com](https://github.com/bitcoin/bips/blob/master/bip-0039.mediawiki))

Sources

- BIP 39: Mnemonic code for generating deterministic keys (github.com)
- The Diceware Passphrase Home Page (theworld.com)
- EFF Dice-Generated Passphrases (www.eff.org)

> Continue reading on bip39-phrase.com: Is a Seed Phrase Case Sensitive? BIP39 Words vs Passphrase
> More on this topic: Seed Phrases

Sample sequences of BIP39 words

Randomly generated sequences of words from the BIP39 English wordlist. None of them passes the BIP39 checksum, so no wallet will accept them: they show the format only. Never store funds with a phrase you found in a document.

Example 1		
1 husband	2 craft	3 crush
4 bar	5 square	6 march
7 column	8 soft	9 laptop
10 engage	11 direct	12 pretty

Example 2		
1 hint	2 need	3 bless
4 abstract	5 summer	6 print
7 memory	8 shock	9 panda
10 wisdom	11 later	12 hunt

Example 3		
1 defy	2 vanish	3 thing
4 ahead	5 then	6 sunset
7 miss	8 brand	9 crater
10 tornado	11 beef	12 ill

Example 4		
1 riot	2 rely	3 promote
4 promote	5 width	6 example
7 valve	8 object	9 motion
10 laugh	11 believe	12 coyote

Example 5		
1 filter	2 peasant	3 galaxy
4 furnace	5 village	6 balcony
7 lucky	8 view	9 rocket
10 bounce	11 divorce	12 ski

Example 6		
1 renew	2 trouble	3 inquiry
4 tortoise	5 code	6 sister
7 pipe	8 reform	9 stumble
10 shoulder	11 ski	12 leisure

Reading a BIP39 phrase

Every word stands for an 11-bit number: its position in the 2048-word list, counted from 0. Here is Example 1 decoded word by word. Each word links to its page on bip39-phrase.com, where the list is numbered from 1.

#	Word	Index (0-2047)	Word no.	11 bits
1	husband	895	896	01101111111
2	craft	400	401	00110010000
3	crush	424	425	00110101000
4	bar	147	148	00010010011
5	square	1691	1692	11010011011
6	march	1086	1087	10000111110
7	column	366	367	00101101110
8	soft	1650	1651	11001110010
9	laptop	1000	1001	01111101000
10	engage	594	595	01001010010
11	direct	500	501	00111110100
12	pretty	1361	1362	10101010001

The 12 words carry 132 bits: 128 bits of entropy followed by a 4-bit checksum. The checksum must equal the first 4 bits of the SHA-256 hash of the entropy.

```
Entropy (hex): 6fe640d4093d370f8b76727d0948fa55
```

This sequence ends in the checksum bits 0001, but SHA-256 of its entropy starts with 1000, so wallets reject it as an invalid phrase. That is the point of the checksum: about 15 of every 16 random or mistyped sequences fail it.

See also

- Coins Sent to Mistyped or Wrongly Restored Wallets · October 15, 2026
- SeedQR: Hand-Drawn QR Backups for Seed Phrases · October 29, 2026
- Picking a Seed Phrase Executor: Trust Before Skill · October 27, 2026
- Fire Safes Can Trap Humidity Around a Seed Backup · October 26, 2026

Educational document. The example sequences are random and fail the BIP39 checksum: they are not wallets and must never hold funds.
Security notes: bip39-phrase.com/security/