

### Create/List key pairs

|                          |                                                 |
|--------------------------|-------------------------------------------------|
| dotenvx init             | Create .env and .env.keys in current directory  |
| dotenvx -p test init     | Create .env.test and .env.keys for test profile |
| dotenvx init -g          | Create \$HOME/.env.keys for global profiles     |
| dotenvx keypair          | Validate and print key pair for .env            |
| dotenvx -p test keypair  | Validate and print key pair for .env.test       |
| dotenvx keypair --all    | List all global key pairs                       |
| dotenvx keypair --import | Import private key                              |

Please use 'cargo binstall dotenvx-rs' to install dotenvx or download it from <https://github.com/linux-china/dotenvx-rs/releases>

### encrypt and decrypt for .env files

|                          |                                                                  |
|--------------------------|------------------------------------------------------------------|
| dotenvx encrypt          | Encrypt .env file                                                |
| dotenvx -p test encrypt  | Encrypt .env.test file                                           |
| dotenvx encrypt --sign   | Encrypt and sign .env file                                       |
| dotenvx verify           | Verify the .env file with public key and sign header in the file |
| dotenvx decrypt          | Decrypt .env file                                                |
| dotenvx -p test decrypt  | Decrypt .env.test file                                           |
| dotenvx decrypt --stdout | Read .env file and print the decrypted items to stdout           |
| dotenvx decrypt --dump   | Decrypt .env file and output entries to stdout with JSON format  |

You can sign an .env file. If somebody modify the .env file, and verification will be failed.

### SDKs & CLIs with Dotenvx

|                               |                                                  |
|-------------------------------|--------------------------------------------------|
| dotenvx link .venv/bin/python | Generate python interpreter with Dotenvx support |
| dotenvx link bin/mysql        | Generate mysql cli with .env setup               |
| dotenvx link bin/psql         | Generate psql cli with .env setup                |
| dotenvx link bin/mongo        | Generate mongo cli with .env setup               |

### SDKs & CLIs with Dotenvx (cont)

|                            |                                        |
|----------------------------|----------------------------------------|
| dotenvx link bin/redis-cli | Generate redis-cli with .env setup     |
| dotenvx link bin/duckdb    | Generate duckdb with .env duckdb setup |

Environment variables from .env:

- MYSQL\_URL, POSTGRES\_URL, DATABASE\_URL
- MONGODB\_URL
- REDIS\_URL

### .env file example

```
# ---
# uuid: f7580a c5- 0b2 4-4 385 -b3 ff- 819 225 -
# b687f3
# name: input your name here
# group: demo
# ---
DOTENV _PUBLIC_KEY= " 02b 497...."
# Environment variables. MAKE SURE to ENCRYPT
them before committing to source control
HELLO= encrypted :base64_text
```

### .env.key file example

```
# ---
# uuid: 8499c5 c3- cee 3-4 c94 -99 a4- 9c8 6b2 -
# ed5dd9
# name: input your name here
# group: demo
# ---
# Private decryption keys. DO NOT commit to source
control
DOTENV_PRIVATE_KEY =9e 701 8-h ex-text
DOTENV_PRIVATE_KEY_EXAMPLE= a3d 15- hex -
text
```

### GitHub Actions setup

```
jobs:
  dot env x-demo:
    run s-on: ubuntu -latest
    steps:
      - uses: action s/c hec kout@v4
      - uses: linux- chi na/ set up- dot env -
x@main
      - run: npm install
      - run: $HOME/.ca rgo /bi n/d otenvx run
-- node index.js
    env:
      DOT ENV _PR IVA TE_KEY: ${{
secret s.D OTE NV_ PRI VAT E_KEY }}
```

Please add DOTENV \_PR IVA TE\_KEY secret to the Repository secrets first.

If you use act for local GitHub Actions test, please use `act -j dot env x-demo --secret-file .env.keys`.



By [linux\\_china](#)

[cheatography.com/linux-china/](https://cheatography.com/linux-china/)

Published 24th July, 2025.

Last updated 19th September, 2025.

Page 1 of 3.

Sponsored by [Readable.com](#)

Measure your website readability!

<https://readable.com>

### Get/Set item from .env file

|                                   |                                        |
|-----------------------------------|----------------------------------------|
| dotenvx get admin                 | Get ADMIN value from .env file         |
| dotenvx get                       | GET all key/value pairs from .env file |
| dotenvx set admin Jackie          | SET ADMIN to Jackie in .env file       |
| dotenvx set admin -               | SET ADMIN from input                   |
| dotenvx set key - < ./private.pem | Set key from text of private.pem file  |
| dotenvx set <key> --clipboard     | Set key from clipboard                 |

Please press Ctrl+D on Linux/macOS or Ctrl+Z on Windows to finish input.

### Switch profile

|                               |                                          |
|-------------------------------|------------------------------------------|
| dotenvx -p test               | Change profile to test                   |
| dotenvx encrypt -f .env.test  | Change profile to test by .env file name |
| NODE_ENV=test dotenvx encrypt | Change profile to test by environment    |

profile's environment variable names: NODE\_ENV, RUN\_ENV, APP\_ENV, SPRING\_PROFILES\_ACTIVE.

### Run command with .env support

|                                  |                                                       |
|----------------------------------|-------------------------------------------------------|
| dotenvx run -- ./demo.sh         | Inject env variables from .env and run ./demo.sh      |
| dotenvx -f prod run -- ./demo.sh | Inject env variables from .env.prod and run ./demo.sh |
| dotenvx run -- echo '\$HELLO'    | Inject env variables from .env to run command line    |
| dotenvx -c 'echo \$HELLO'        | Inject env variables from .env and run command line   |

Please use single quote to wrap variables from .env file.

### Rotate key pair for project

|                             |                                    |
|-----------------------------|------------------------------------|
| dotenvx rotate              | Rotate key pair for .env file      |
| dotenvx -p prod rotate      | Rotate key pair for .env.prod file |
| dotenvx rotate -f .env.prod | Rotate key pair for .env.prod file |

### Global private key management

|                       |                                                                                |
|-----------------------|--------------------------------------------------------------------------------|
| dotenvx init --global | Generate global private keys in \$HOME/.env.keys                               |
| dotenvx -- seal       | Encrypt \$HOME/.env.keys, \$HOME/.dotenvx/.env.key.json with your password     |
| dotenvx -- unseal     | Decrypt \$HOME/.env.keys.aes, \$HOME/.dotenvx/.env.keys.json.aes with password |

### Misc

|                                            |                                                            |
|--------------------------------------------|------------------------------------------------------------|
| eval \$(dotenvx decrypt -- export)         | Export .env items to environment variables                 |
| eval \$(dotenvx get hello -- format shell) | Export HELLO from .env items as environment variable       |
| dotenvx ls                                 | List all .env file from current directory recursively      |
| dotenvx -f test ls                         | List all .env.test file from current directory recursively |
| dotenvx diff key1,key2                     | Display keys' values between all .env files                |
| dotenvx diff key1,key2 -- format csv       | Display keys' values between all .env files by csv         |
| dotenvx completion -- shell zsh            | Output auto-completion script for bash/zsh/fish/powershell |

**Tips:** if you use direnv, and you can add "eval \$(dotenvx decrypt -- export)" to the .envrc file to automatically load .env as the environment variables when you enter the directory.

### Node.js(Volat), Bun, Deno with Dotenvx

```
#!/bin/bash
# load .env by dotenvx
eval $( dotenvx decrypt --stdout --format shell )
# Execute bun command with arguments
$HOME/.bun/bin/bun " $@"
#$HOME/.deno/bin/deno " $@"
```

**Volta Dotenvx edition:** <https://github.com/linux-china/volta/releases/tag/v2.0.2-dotenvx20250830>

Node.js, Bun/Deno script wrapper with Dotenvx: nodew, bunw, denow.

You can create wrapper script for Lua, PHP, Perl with same way.