

# token create

The `token create` command creates a new token that can be used for authentication. This token will be created as a child of the currently authenticated token. The generated token will inherit all policies and permissions of the currently authenticated token unless you explicitly define a subset list policies to assign to the token.

A `ttl` can also be associated with the token. If a `ttl` is not associated with the token, then it cannot be renewed. If a `ttl` is associated with the token, it will expire after that amount of time unless it is renewed.

Metadata associated with the token (specified with `-metadata`) is written to the audit log when the token is used.

If a role is specified, the role may override parameters specified here.

## Examples

Create a token attached to specific policies:

```
$ vault token create -policy=my-policy -policy=other-policy
Key          Value
---          -
token        95eba8ed-f6fc-958a-f490-c7fd0eda5e9e
token_accessor 882d4a40-3796-d06e-c4f0-604e8503750b
token_duration 768h
token_renewable true
token_policies [default my-policy other-policy]
```

Create a periodic token:

```
$ vault token create -period=30m
Key          Value
---          -
token        fdb90d58-af87-024f-fdcd-9f95039e353a
token_accessor 4cd9177c-034b-a004-c62d-54bc56c0e9bd
token_duration 30m
token_renewable true
token_policies [my-policy]
```

## Usage

The following flags are available in addition to the standard set of flags (`/docs/commands/index.html`) included on all commands.

### Output Options

- `-field (string: "")` - Print only the field with the given name. Specifying this option will take precedence over other formatting directives. The result will not have a trailing newline making it ideal for piping to other processes.
- `-format (string: "table")` - Print the output in the given format. Valid formats are "table", "json", or "yaml". This can also be specified via the `VAULT_FORMAT` environment variable.

## Command Options

- `-display-name (string: "")` - Name to associate with this token. This is a non-sensitive value that can be used to help identify created secrets (e.g. prefixes).
- `-explicit-max-ttl (duration: "")` - Explicit maximum lifetime for the token. Unlike normal TTLs, the maximum TTL is a hard limit and cannot be exceeded. This is specified as a numeric string with suffix like "30s" or "5m".
- `-id (string: "")` - Value for the token. By default, this is an auto-generated 36 character UUID. Specifying this value requires sudo permissions.
- `-metadata (k=v: "")` - Arbitrary key=value metadata to associate with the token. This metadata will show in the audit log when the token is used. This can be specified multiple times to add multiple pieces of metadata.
- `-no-default-policy (bool: false)` - Detach the "default" policy from the policy set for this token.
- `-orphan (bool: false)` - Create the token with no parent. This prevents the token from being revoked when the token which created it expires. Setting this value requires sudo permissions.
- `-period (duration: "")` - If specified, every renewal will use the given period. Periodic tokens do not expire (unless `-explicit-max-ttl` is also provided). Setting this value requires sudo permissions. This is specified as a numeric string with suffix like "30s" or "5m".
- `-policy (string: "")` - Name of a policy to associate with this token. This can be specified multiple times to attach multiple policies.
- `-renewable (bool: true)` - Allow the token to be renewed up to it's maximum TTL.
- `-role (string: "")` - Name of the role to create the token against. Specifying `-role` may override other arguments. The locally authenticated Vault token must have permission for "auth/token/create/".
- `-ttl (duration: "")` - Initial TTL to associate with the token. Token renewals may be able to extend beyond this value, depending on the configured maximumTTLs. This is specified as a numeric string with suffix like "30s" or "5m".
- `-use-limit (int: 0)` - Number of times this token can be used. After the last use, the token is automatically revoked. By default, tokens can be used an unlimited number of times until their expiration.