





One use case for this manual setting of token access is to manage Batch actions, that can be run based on cron jobs set on external servers. See more information in [Batch](#)

### Using a token

#### A token looks like this

```
http://demo.tiki.org/trunk/tiki-index.php?TOKEN=937d83bc9dc6ede58d247df505011t43
```

### Limitations

You cannot edit already created tokens.

The token param requested in the URL must always be in uppercase letters: `TOKEN=...`.

The URL used to generate the new token should not contain any URL encoded values, like `%2C`, and any multivalue array field params with square brackets, like `[listfields[]]`. For example using URL like this will fail:

```
https://yoursite/tiki-ajax_services.php?controller=tracker&action=export_items&trackerId=3&encoding=UTF-8&separator=%2C&delimiterL="&delimiterR="&CR=%25%25%25&listfields[]=2&listfields[]=3&recordsMax=-1
```

You will get an error message "**Token Error:** Your access to this page has expired" when you try to access that page using the token.

Instead you need to use URL like this to create the token:

```
https://yoursite/tiki-ajax_services.php?controller=tracker&action=export_items&trackerId=3&encoding=UTF-8&separator=,&CR=%%&recordsMax=-1
```

Then accessing the URL using

[https://yoursite/tiki-ajax\\_services.php?controller=tracker&action=export\\_items&trackerId=3&encoding=UTF-8&separator=,&CR=%%&recordsMax=-1&TOKEN=...](https://yoursite/tiki-ajax_services.php?controller=tracker&action=export_items&trackerId=3&encoding=UTF-8&separator=,&CR=%%&recordsMax=-1&TOKEN=...) will work.

### Related

- [PluginGetAccessToken](#)
- [Batch](#)
- [Tokens](#)

- [PluginArchiveBuilder](#)

aliases

---

[Token](#) | [TokenAccess](#) | [Tokens](#) | [Security Token](#)