

---

# **generalwords Documentation**

*Release latest*

January 11, 2015



<b>1</b>	<b>Goals</b>	<b>3</b>
<b>2</b>	<b>Word Source</b>	<b>5</b>
<b>3</b>	<b>Usage Example</b>	<b>7</b>
<b>4</b>	<b>Not Done Yet</b>	<b>9</b>



Often, when building an application, I've had occasions where I need to provide a nonce of some sort to a user. A nonce is some thing, whether number, or otherwise, that is used only once.



### Goals

---

The over-arching goal was for this to be simple to use, and reasonably fast. Next, I wanted something that the average user would be able to enter manually with minimal risk. This ruled out a lot of the crazy mixed-case alphanumeric strings that are often used. Finally, a goal was to have a reasonably large number of permutations available so that the nonce, while not cryptographically strong, was strong enough for uses like registration, password reset, etc. For example, with the default [word source](#), and a selection of three words, there are [almost 19 billion permutations](#) available. Presuming you're doing any form of brute-force-attack prevention, this should be more than adequate.



---

### Word Source

---

The list of words comes from the [New General Service List v1.01](#). The original list, distributed in Excel format, is covered under the [Creative Commons Attribution 3.0 Unported License](#). More information can be found on the website. From the base data, the following changes have been made:

1. The list was converted from Excel to a simple one-word-per-line text file format;
2. Only headwords were retained for clarity;
3. Supplemental words, such as months and numbers, were added; 3. Words less than 4 characters were removed.

Combined, a total of 2,668 headwords were retained.



---

## Usage Example

---

The library is trivially easy to use:

```
>>> from generalwords import *
>>> get_word()
'tire'
>>> get_word()
'offense'
>>> get_words(n=3)
('climb', 'repair', 'force')
```

In the future, there may be more options, but not right now.



---

**Not Done Yet**

---

While the library will never be massively complicated, there's a few things I'd like to have in the future:

- Expand to other languages
- Add additional word sources