

## Navigation

- [index](#)
- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [14. File Formats](#) »

## 14.4. netrc — netrc file processing¶

New in version 1.5.2.

The [netrc](#) class parses and encapsulates the netrc file format used by the Unix **ftp** program and other FTP clients.

```
class netrc.netrc(file)¶
```

A [netrc](#) instance or subclass instance encapsulates data from a netrc file. The initialization argument, if present, specifies the file to parse. If no argument is given, the file `.netrc` in the user's home directory will be read. Parse errors will raise [NetrcParseError](#) with diagnostic information including the file name, line number, and terminating token.

```
exception netrc.NetrcParseError¶
```

Exception raised by the [netrc](#) class when syntactical errors are encountered in source text. Instances of this exception provide three interesting attributes: `msg` is a textual explanation of the error, `filename` is the name of the source file, and `lineno` gives the line number on which the error was found.

### 14.4.1. netrc Objects¶

A [netrc](#) instance has the following methods:

```
netrc.authenticators(host)¶
```

Return a 3-tuple (`login`, `account`, `password`) of authenticators for *host*. If the netrc file did not contain an entry for the given host, return the tuple associated with the 'default' entry. If neither matching host nor default entry is available, return `None`.

```
netrc.__repr__()¶
```

Dump the class data as a string in the format of a netrc file. (This discards comments and may reorder the entries.)

Instances of [netrc](#) have public instance variables:

```
netrc.hosts¶
```

Dictionary mapping host names to (`login`, `account`, `password`) tuples. The 'default' entry, if any, is represented as a pseudo-host by that name.

```
netrc.macros¶
```

Dictionary mapping macro names to string lists.

Note

Passwords are limited to a subset of the ASCII character set. Versions of this module prior to 2.3 were extremely limited. Starting with 2.3, all ASCII punctuation is allowed in passwords. However, note that whitespace and non-printable characters are not allowed in passwords. This is a limitation of the way the `.netrc` file is parsed and may be removed in the future.

### [Table Of Contents](#)

[14.4. netrc — netrc file processing](#)

- [14.4.1. netrc Objects](#)

Previous topic

[14.3. robotparser — Parser for robots.txt](#)

Next topic

[14.5. xdrlib — Encode and decode XDR data](#)

This Page

- [Show Source](#)

Navigation

- [index](#)

- [modules](#) |
- [next](#) |
- [previous](#) |
- [Python v2.6.4 documentation](#) »
- [The Python Standard Library](#) »
- [14. File Formats](#) »

© [Copyright](#) 1990-2010, Python Software Foundation.

The Python Software Foundation is a non-profit corporation. [Please donate.](#)

Last updated on Feb 26, 2010. Created using [Sphinx](#) 0.6.3.