

14.3. `netrc` — netrc file processing

Source code: [Lib/netrc.py](#)

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. If no argument is specified on a POSIX system, the presence of passwords in the `.netrc` file will raise a `NetrcParseError` if the file ownership or permissions are insecure (owned by a user other than the user running the process, or accessible for read or write by any other user). This implements security behavior equivalent to that of `ftp` and other programs that use `.netrc`.

Changed in version 3.4: Added the POSIX permission check.

`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.

14.3.1. 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. 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.