NAME

pam_userdb − PAM module to authenticate against a db database

SYNOPSIS

pam_userdb.so db=*/path/database* [debug] [crypt=[crypt|none]] [icase] [dump] [try_first_pass] [use_first_pass] [unknown_ok] [key_only]

DESCRIPTION

The pam_userdb module is used to verify a username/password pair against values stored in a Berkeley DB database. The database is indexed by the username, and the data fields corresponding to the username keys are the passwords.

OPTIONS

crypt=[crypt|none]

Indicates whether encrypted or plaintext passwords are stored in the database. If it is **crypt**, passwords should be stored in the database in **crypt**(3) form. If **none** is selected, passwords should be stored in the database as plaintext.

db=*/path/database*

Use the /path/database database for performing lookup. There is no default; the module will return **PAM_IGNORE** if no database is provided. Note that the path to the database file should be specified without the .db suffix.

debug

Print debug information.

dump

Dump all the entries in the database to the log. Don't do this by default!

icase

Make the password verification to be case insensitive (ie when working with registration numbers and such). Only works with plaintext password storage.

try_first_pass

Use the authentication token previously obtained by another module that did the conversation with the application. If this token can not be obtained then the module will try to converse. This option can be used for stacking different modules that need to deal with the authentication tokens.

use_first_pass

Use the authentication token previously obtained by another module that did the conversation with the application. If this token can not be obtained then the module will fail. This option can be used for stacking different modules that need to deal with the authentication tokens.

unknown_ok

Do not return error when checking for a user that is not in the database. This can be used to stack more than one pam_userdb module that will check a username/password pair in more than a database.

key_only

The username and password are concatenated together in the database hash as 'username−password' with a random value. if the concatenation of the username and password with a dash in the middle returns any result, the user is valid. this is useful in cases where the username may not be unique but the username and password pair are.

MODULE TYPES PROVIDED

The **auth** and **account** module types are provided.

RETURN VALUES

PAM_AUTH_ERR

Authentication failure.

PAM_AUTHTOK_RECOVERY_ERR

Authentication information cannot be recovered.

PAM_BUF_ERR
: Memory buffer error.

PAM_CONV_ERR
: Conversation failure.

PAM_SERVICE_ERR
: Error in service module.

PAM_SUCCESS
: Success.

PAM_USER_UNKNOWN
: User not known to the underlying authentication module.

EXAMPLES

```
auth  sufficient pam_userdb.so icase db=/etc/dbtest
```

SEE ALSO

crypt(3), **pam.conf**(5), **pam.d**(5), **pam**(7)

AUTHOR

pam_userdb was written by Cristian Gafton >gafton@redhat.com<.