Apache Module mod_authz_dbm

Description:	Group authorization using DBM files
Status:	Extension
Module Identifier:	authz_dbm_module
Source File:	mod_authz_dbm.c
Compatibility:	Available in Apache 2.1 and later

Summary

This module provides authorization capabilities so that authenticated users can be allowed or denied access to portions of the web site by group membership. Similar functionality is provided by `mod_authz_groupfile`.

Topics

Directives

See also

- `Require`
- `Satisfy`

AuthDBMGroupFile Directive

Description:	Sets the name of the database file containing the list of user groups for authentication
Syntax:	`AuthDBMGroupFile` *`file-path`*
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Extension
Module:	mod_authz_dbm

The `AuthDBMGroupFile` directive sets the name of a DBM file containing the list of user groups for user authentication. *File-path* is the absolute path to the group file.

The group file is keyed on the username. The value for a user is a comma-separated list of the groups to which the users belongs. There must be no whitespace within the value, and it must never contain any colons.

Security

Make sure that the `AuthDBMGroupFile` is stored outside the document tree of the web-server. Do **not** put it in the directory that it protects. Otherwise, clients will be able to download the `AuthDBMGroupFile` unless otherwise protected.

Combining Group and Password DBM files: In some cases it is easier to manage a single database which contains both the password and group details for each user. This simplifies any support

programs that need to be written: they now only have to deal with writing to and locking a single DBM file. This can be accomplished by first setting the group and password files to point to the same DBM:

```
AuthDBMGroupFile /www/userbase
AuthDBMUserFile /www/userbase
```

The key for the single DBM is the username. The value consists of

```
Encrypted Password : List of Groups [ : (ignored) ]
```

The password section contains the encrypted password as before. This is followed by a colon and the comma separated list of groups. Other data may optionally be left in the DBM file after another colon; it is ignored by the authentication module. This is what www.telescope.org uses for its combined password and group database.

AuthzDBMAuthoritative Directive

Description:	Sets whether authorization will be passed on to lower level modules
Syntax:	`AuthzDBMAuthoritative On|Off`
Default:	`AuthzDBMAuthoritative On`
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Extension
Module:	mod_authz_dbm

Setting the `AuthzDBMAuthoritative` directive explicitly to `Off` allows group authorization to be passed on to lower level modules (as defined in the `modules.c` file) if there is no group found for the the supplied userID. If there are any groups specified, the usual checks will be applied and a failure will give an Authentication Required reply.

So if a userID appears in the database of more than one module; or if a valid `Require` directive applies to more than one module; then the first module will verify the credentials; and no access is passed on; regardless of the `AuthAuthoritative` setting.

A common use for this is in conjunction with one of the auth providers; such as `mod_authn_dbm` or `mod_authn_file`. Whereas this DBM module supplies the bulk of the user credential checking; a few (administrator) related accesses fall through to a lower level with a well protected `.htpasswd` file.

By default, control is not passed on and an unknown group will result in an Authentication Required reply. Not setting it thus keeps the system secure and forces an NCSA compliant behaviour.

Security

Do consider the implications of allowing a user to allow fall-through in his .htaccess file; and verify that this is really what you want; Generally it is easier to just secure a single `.htpasswd` file, than it is to secure a database which might have more access interfaces.

AuthzDBMType Directive

Description:	Sets the type of database file that is used to store passwords
Syntax:	`AuthzDBMType default|SDBM|GDBM|NDBM|DB`
Default:	`AuthzDBMType default`
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Extension
Module:	mod_authz_dbm

Sets the type of database file that is used to store the passwords. The default database type is determined at compile time. The availability of other types of database files also depends on compile-time settings[1].

It is crucial that whatever program you use to create your password files is configured to use the same type of database.

URI References

[1] http://httpd.apache.org/docs-2.1/install.html#dbm