

Apache Module mod_auth_digest

Description:	User authentication using MD5 Digest Authentication.
Status:	Experimental
Module Identifier:	auth_digest_module
Source File:	mod_auth_digest.c

Summary

This module implements HTTP Digest Authentication. However, it has not been extensively tested and is therefore marked experimental.

Topics

Using Digest Authentication	1
URI References	4

Directives

AuthDigestAlgorithm	2	AuthDigestNonceLifetime	3
AuthDigestDomain	2	AuthDigestProvider	3
AuthDigestNcCheck	2	AuthDigestQop	4
AuthDigestNonceFormat	3	AuthDigestShmemSize	4

See also

- [AuthName](#)
- [AuthType](#)
- [Require](#)
- [Satisfy](#)

Using Digest Authentication

Using MD5 Digest authentication is very simple. Simply set up authentication normally, using `AuthType Digest` and `AuthDigestProvider` instead of the normal `AuthType Basic` and `AuthBasicProvider`. Then add a `AuthDigestDomain` directive containing at least the root URI(s) for this protection space.

Appropriate user (text) files can be created using the `htdigest`¹ tool.

Example:

```
<Location /private/>
  AuthType Digest
  AuthName "private area"
  AuthDigestDomain /private/ http://mirror.my.dom/private2/

  AuthDigestProvider file
  AuthUserFile /web/auth/.digest_pw
  Require valid-user
</Location>
```

Note

Digest authentication provides a more secure password system than Basic authentication, but only works with supporting browsers. As of November 2002, the major browsers that support digest authentication are Opera², MS Internet Explorer³ (fails when used with a query string),

Amaya⁴, Mozilla⁵ and Netscape⁶ since version 7. Since digest authentication is not as widely implemented as basic authentication, you should use it only in controlled environments.

AuthDigestAlgorithm Directive

Description:	Selects the algorithm used to calculate the challenge and response hashes in digest authentication
Syntax:	AuthDigestAlgorithm MD5 MD5-sess
Default:	AuthDigestAlgorithm MD5
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Experimental
Module:	mod_auth_digest

The `AuthDigestAlgorithm` directive selects the algorithm used to calculate the challenge and response hashes.

MD5-sess is not correctly implemented yet.

AuthDigestDomain Directive

Description:	URIs that are in the same protection space for digest authentication
Syntax:	AuthDigestDomain <i>URI</i> [<i>URI</i>] ...
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Experimental
Module:	mod_auth_digest

The `AuthDigestDomain` directive allows you to specify one or more URIs which are in the same protection space (*i.e.* use the same realm and username/password info). The specified URIs are prefixes, *i.e.* the client will assume that all URIs "below" these are also protected by the same username/password. The URIs may be either absolute URIs (*i.e.* including a scheme, host, port, etc) or relative URIs.

This directive *should* always be specified and contain at least the (set of) root URI(s) for this space. Omitting to do so will cause the client to send the Authorization header for *every request* sent to this server. Apart from increasing the size of the request, it may also have a detrimental effect on performance if `AuthDigestNcCheck` is on.

The URIs specified can also point to different servers, in which case clients (which understand this) will then share username/password info across multiple servers without prompting the user each time.

AuthDigestNcCheck Directive

Description:	Enables or disables checking of the nonce-count sent by the server
Syntax:	AuthDigestNcCheck On Off
Default:	AuthDigestNcCheck Off

Apache Module mod_auth_digest

Context:	server config
Status:	Experimental
Module:	mod_auth_digest

Not implemented yet.

AuthDigestNonceFormat Directive

Description:	Determines how the nonce is generated
Syntax:	AuthDigestNonceFormat <i>format</i>
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Experimental
Module:	mod_auth_digest

Not implemented yet.

AuthDigestNonceLifetime Directive

Description:	How long the server nonce is valid
Syntax:	AuthDigestNonceLifetime <i>seconds</i>
Default:	AuthDigestNonceLifetime 300
Context:	directory, .htaccess
Override:	AuthConfig
Status:	Experimental
Module:	mod_auth_digest

The `AuthDigestNonceLifetime` directive controls how long the server nonce is valid. When the client contacts the server using an expired nonce the server will send back a 401 with `stale=true`. If *seconds* is greater than 0 then it specifies the amount of time for which the nonce is valid; this should probably never be set to less than 10 seconds. If *seconds* is less than 0 then the nonce never expires.

AuthDigestProvider Directive

Description:	Sets the authentication provider(s) for this location
Syntax:	AuthDigestProvider On Off <i>provider-name</i> [<i>provider-name</i>] ...
Default:	AuthBasicProvider On
Context:	directory
Status:	Experimental
Module:	mod_auth_digest

The `AuthDigestProvider` directive sets which provider is used to authenticate the users for this location. Setting the value to `On` will choose the default provider (`file`). Since the `file` provider is implemented by the `mod_authn_file` module, you have to make sure, that the module is present in the server.

See `mod_authn_dbm` and `mod_authn_file` for providers.

The value `off` clears the provider list and sets it back to the default.

AuthDigestQop Directive

Description:	Determines the quality-of-protection to use in digest authentication
Syntax:	<code>AuthDigestQop none auth auth-int [auth auth-int]</code>
Default:	<code>AuthDigestQop auth</code>
Context:	directory, <code>.htaccess</code>
Override:	<code>AuthConfig</code>
Status:	Experimental
Module:	<code>mod_auth_digest</code>

The `AuthDigestQop` directive determines the *quality-of-protection* to use. `auth` will only do authentication (username/password); `auth-int` is authentication plus integrity checking (an MD5 hash of the entity is also computed and checked); `none` will cause the module to use the old RFC-2069 digest algorithm (which does not include integrity checking). Both `auth` and `auth-int` may be specified, in which the case the browser will choose which of these to use. `none` should only be used if the browser for some reason does not like the challenge it receives otherwise.

`auth-int` is not implemented yet.

AuthDigestShmemSize Directive

Description:	The amount of shared memory to allocate for keeping track of clients
Syntax:	<code>AuthDigestShmemSize size</code>
Default:	<code>AuthDigestShmemSize 1000</code>
Context:	server config
Status:	Experimental
Module:	<code>mod_auth_digest</code>

The `AuthDigestShmemSize` directive defines the amount of shared memory, that will be allocated at the server startup for keeping track of clients. Note that the shared memory segment cannot be set less than the space that is necessary for tracking at least *one* client. This value is dependant on your system. If you want to find out the exact value, you may simply set `AuthDigestShmemSize` to the value of 0 and read the error message after trying to start the server.

The *size* is normally expressed in Bytes, but you may let the number follow a `K` or an `M` to express your value as KBytes or MBytes. For example, the following directives are all equivalent:

```
AuthDigestShmemSize 1048576
AuthDigestShmemSize 1024K
AuthDigestShmemSize 1M
```

URI References

Apache Module mod_auth_digest

- [1] <http://httpd.apache.org/docs-2.1/programs/htdigest.html>
- [2] <http://www.opera.com/>
- [3] <http://www.microsoft.com/windows/ie/>
- [4] <http://www.w3.org/Amaya/>
- [5] <http://www.mozilla.org>
- [6] <http://channels.netscape.com/ns/browsers/download.jsp>