NAME

Digest::HMAC – Keyed-Hashing for Message Authentication

SYNOPSIS

```
# Functional style
use Digest::HMAC qw(hmac hmac_hex);
$digest = hmac($data, $key, \&myhash);
print hmac_hex($data, $key, \&myhash);

# OO style
use Digest::HMAC;
$hmac = Digest::HMAC->new($key, "Digest::MyHash");

$hmac->add($data);
$hmac->addfile(*FILE);

$digest = $hmac->digest;
$digest = $hmac->hexdigest;
$digest = $hmac->b64digest;
```

DESCRIPTION

HMAC is used for message integrity checks between two parties that share a secret key, and works in combination with some other Digest algorithm, usually MD5 or SHA-1. The HMAC mechanism is described in RFC 2104.

HMAC follow the common `Digest::` interface, but the constructor takes the secret key and the name of some other simple `Digest::` as argument.

The *hmac()* and *hmac_hex()* functions and the Digest::HMAC->*new()* constructor takes an optional `$blocksize` argument as well. The HMAC algorithm assumes the digester to hash by iterating a basic compression function on blocks of data and the `$blocksize` should match the byte-length of such blocks.

The default `$blocksize` is 64 which is suitable for the MD5 and SHA-1 digest functions. For stronger algorithms the blocksize probably needs to be increased.

SEE ALSO

Digest::HMAC_MD5, Digest::HMAC_SHA1

RFC 2104

AUTHORS

Graham Barr <gbarr@ti.com>, Gisle Aas <gisle@aas.no>