



**Full credit is given to the above companies including the Operating System (OS) that this PDF file was generated!**

***Rocky Enterprise Linux 9.2 Manual Pages on command 'Crypt::Mac::PMAC.3pm'***

***\$ man Crypt::Mac::PMAC.3pm***

Crypt::Mac::PMAC(3pm)      User Contributed Perl Documentation      Crypt::Mac::PMAC(3pm)

NAME

Crypt::Mac::PMAC - Message authentication code PMAC

SYNOPSIS

### Functional interface:

```
use Crypt::Mac::PMAC qw( pmac pmac_hex );
```

```
# calculate MAC from string/buffer
```

```
$pmac_raw = pmac($cipher_name, $key, 'data buffer');
```

```
$pmac_hex = pmac_hex($cipher_name, $key, 'data buffer');
```

```
$pmac_b64 = pmac_b64($cipher_name, $key, 'data buffer');
```

```
$pmac_b64u = pmac_b64u($cipher_name, $key, 'data buffer');
```

### OO interface:

```
use Crypt::Mac::PMAC;
```

```
$d = Crypt::Mac::PMAC->new($cipher_name, $key);
```

```
$d->add('any data');
```

```
$d->addfile('filename.dat');
```

```
$d->addfile(*FILEHANDLE);
```

```
$result_raw = $d->mac;    # raw bytes
```

```
$result_hex = $d->hexmac; # hexadecimal form
$result_b64 = $d->b64mac; # Base64 form
$result_b64u = $d->b64umac; # Base64 URL Safe form
```

## DESCRIPTION

Provides an interface to the PMAC message authentication code (MAC) algorithm.

## EXPORT

Nothing is exported by default.

You can export selected functions:

```
use Crypt::Mac::PMAC qw(pmac pmac_hex );
```

Or all of them at once:

```
use Crypt::Mac::PMAC ':all';
```

## FUNCTIONS

### pmac

Logically joins all arguments into a single string, and returns its PMAC message authentication code encoded as a binary string.

```
$pmac_raw = pmac($cipher_name, $key, 'data buffer');
```

#or

```
$pmac_raw = pmac($cipher_name, $key, 'any data', 'more data', 'even more data');
```

### pmac\_hex

Logically joins all arguments into a single string, and returns its PMAC message authentication code encoded as a hexadecimal string.

```
$pmac_hex = pmac_hex($cipher_name, $key, 'data buffer');
```

#or

```
$pmac_hex = pmac_hex($cipher_name, $key, 'any data', 'more data', 'even more data');
```

#### pmac\_b64

Logically joins all arguments into a single string, and returns its PMAC message authentication code encoded as a Base64 string.

```
$pmac_b64 = pmac_b64($cipher_name, $key, 'data buffer');
```

#or

```
$pmac_b64 = pmac_b64($cipher_name, $key, 'any data', 'more data', 'even more data');
```

#### pmac\_b64u

Logically joins all arguments into a single string, and returns its PMAC message authentication code encoded as a Base64 URL Safe string (see RFC 4648 section 5).

```
$pmac_b64url = pmac_b64u($cipher_name, $key, 'data buffer');
```

#or

```
$pmac_b64url = pmac_b64u($cipher_name, $key, 'any data', 'more data', 'even more data');
```

## METHODS

#### new

```
$d = Crypt::Mac::PMAC->new($cipher_name, $key);
```

#### clone

```
$d->clone();
```

#### reset

```
$d->reset();
```

#### add

```
$d->add('any data');
```

#or

```
$d->add('any data', 'more data', 'even more data');
```

addfile

```
$d->addfile('filename.dat');
```

#or

```
$d->addfile(*FILEHANDLE);
```

mac

```
$result_raw = $d->mac();
```

hexmac

```
$result_hex = $d->hexmac();
```

b64mac

```
$result_b64 = $d->b64mac();
```

b64umac

```
$result_b64url = $d->b64umac();
```

SEE ALSO

? CryptX

? <[https://en.wikipedia.org/wiki/PMAC\\_%28cryptography%29](https://en.wikipedia.org/wiki/PMAC_%28cryptography%29)>

perl v5.34.0

2022-02-06

Crypt::Mac::PMAC(3pm)