



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 'APR::SockAddr.3pm'

\$ man APR::SockAddr.3pm

libapache2-mod-perl2-2.0.12::doUserConlibapache2-mod-perl2-2.0.12::docs::api::APR::SockAddr(3pm)

NAME

APR::SockAddr - Perl API for APR socket address structure

Synopsis

```
use APR::SockAddr ();
```

```
my $ip = $sock_addr->ip_get;
```

```
my $port = $sock_addr->port;
```

Description

"APR::SockAddr" provides an access to a socket address structure fields.

Normally you'd get a socket address object, by calling:

```
use Apache2::Connection ();
```

```
my $remote_sock_addr = $c->remote_addr;
```

```
my $local_sock_addr = $c->remote_local;
```

API

"APR::SockAddr" provides the following functions and/or methods:

"ip_get"

Get the IP address of the socket

```
$ip = $sock_addr->ip_get();
```

obj: \$sock_addr ("APR::SockAddr object")

ret: \$ip (string)

since: 2.0.00

If you are familiar with how perl's "Socket" works:

```
use Socket 'sockaddr_in';
```

```
my ($serverport, $serverip) = sockaddr_in(getpeername($local_sock));
```

```
my ($remoteport, $remoteip) = sockaddr_in(getpeername($remote_sock));
```

in apr-speak that'd be written as:

```
use APR::SockAddr ();
```

```
use Apache2::Connection ();
```

```
my $serverport = $c->local_addr->port;
```

```
my $serverip = $c->local_addr->ip_get;
```

```
my $remoteport = $c->remote_addr->port;
```

```
my $remoteip = $c->remote_addr->ip_get;
```

"port"

Get the IP address of the socket

```
$port = $sock_addr->port();
```

obj: \$sock_addr ("APR::SockAddr object")

ret: \$port (integer)

since: 2.0.00

Example: see "ip_get()"

Unsupported API

"APR::SockAddr" also provides auto-generated Perl interface for a few other methods which aren't tested at the moment and therefore their API is a subject to change. These methods will be finalized later as a need arises. If you want to rely on any of the following methods please contact the the mod_perl development mailing list so we can help each other take the steps necessary to shift the method to an officially supported API.

"equal"

META: Autogenerated - needs to be reviewed/completed

See if the IP addresses in two APR socket addresses are equivalent. Appropriate logic is present for comparing IPv4-mapped IPv6 addresses with IPv4 addresses.

```
$ret = $addr1->equal($addr2);
```

obj: \$addr1 ("APR::SockAddr object")

One of the APR socket addresses.

arg1: \$addr2 ("APR::SockAddr object")

The other APR socket address.

ret: \$ret (integer)

since: subject to change

The return value will be non-zero if the addresses are equivalent.

See Also

mod_perl 2.0 documentation.

Copyright

mod_perl 2.0 and its core modules are copyrighted under The Apache Software License,

Version 2.0.

Authors

The mod_perl development team and numerous contributors.

perl v5.34.0

libapache2-mod-perl2-2.0.12::docs::api::APR::SockAddr(3pm)