

NAME

Net::DNS::RR::OPT – DNS OPT resource record

SYNOPSIS

```

use Net::DNS;
$packet = new Net::DNS::Packet( ... );

$packet->header->do(1);                # extended flag

$packet->edns->size(1280);              # UDP payload size

$packet->edns->option( COOKIE => $cookie );

$packet->edns->print;

;; EDNS version 0
;;     flags:  8000
;;     rcode:  NOERROR
;;     size:   1280
;;     option: DAU      => ( 8, 10, 13, 14, 15, 16 )
;;             DHU      => ( 1, 2, 4 )
;;             COOKIE => ( CLIENT-COOKIE => 7261776279746573,
;;                       SERVER-COOKIE => )

```

DESCRIPTION

EDNS OPT pseudo resource record.

The OPT record supports EDNS protocol extensions and is not intended to be created, accessed or modified directly by user applications.

All EDNS features are performed indirectly by operations on the objects returned by the `$packet->header` and `$packet->edns` creator methods. The underlying mechanisms are entirely hidden from the user.

METHODS

The available methods are those inherited from the base class augmented by the type-specific methods defined in this package.

Use of undocumented package features or direct access to internal data structures is discouraged and could result in program termination or other unpredictable behaviour.

version

```
$version = $rr->version;
```

The version of EDNS used by this OPT record.

size

```

$size = $packet->edns->size;
$more = $packet->edns->size(1280);

```

size() advertises the maximum size (octets) of UDP packet that can be reassembled in the network stack of the originating host.

rcode

```

$extended_rcode = $packet->header->rcode;
$incomplete_rcode = $packet->edns->rcode;

```

The 12 bit extended RCODE. The most significant 8 bits reside in the OPT record. The least significant 4 bits can only be obtained from the packet header.

flags

```
$edns_flags = $packet->edns->flags;
```

```
$do = $packet->header->do;
$packet->header->do(1);
```

16 bit field containing EDNS extended header flags.

options, option

```
@option = $packet->edns->options;
```

```
$octets = $packet->edns->option($option_code);
```

```
$packet->edns->option( COOKIE => $cookie );
$packet->edns->option( 10      => $cookie );
```

When called in a list context, **options()** returns a list of option codes found in the OPT record.

When called in a scalar context with a single argument, **option()** returns the uninterpreted octet string corresponding to the specified option. The method returns undef if the specified option is absent.

Options can be added or replaced by providing the (name => string) pair. The option is deleted if the value is undefined.

When **option()** is called in a list context with a single argument, the returned array provides a structured interpretation appropriate to the specified option.

For the example above:

```
%hash = $packet->edns->option(10);

%hash = (
    'CLIENT-COOKIE' => 'rawbytes',
    'SERVER-COOKIE' => ''
);
```

For some options, an array is more appropriate:

```
@algorithms = $packet->edns->option(6);
```

Similar forms of array syntax may be used to construct the option value:

```
$packet->edns->option( DHU => [1, 2, 4] );
$packet->edns->option( 6   => (1, 2, 4) );

$packet->edns->option( COOKIE => {'CLIENT-COOKIE' => $cookie} );
$packet->edns->option( 10     => ('CLIENT-COOKIE' => $cookie) );
```

COPYRIGHT

Copyright (c)2001,2002 RIPE NCC. Author Olaf M. Kolkman.

Portions Copyright (c)2012,2017 Dick Franks.

All rights reserved.

Package template (c)2009,2012 O.M.Kolkman and R.W.Franks.

LICENSE

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of the author not be used in advertising or publicity pertaining to distribution of the software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED,

INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SEE ALSO

perl, Net::DNS, Net::DNS::RR, RFC6891, RFC3225