

## NAME

Net::Config - Local configuration data for libnet

## SYNOPSIS

```
use Net::Config qw(%NetConfig);
```

## DESCRIPTION

Net::Config holds configuration data for the modules in the libnet distribution. During installation you will be asked for these values.

The configuration data is held globally in a file in the perl installation tree, but a user may override any of these values by providing their own. This can be done by having a `.libnetrc` file in their home directory. This file should return a reference to a HASH containing the keys described below. For example

```
# .libnetrc
{
    nntp_hosts => [ "my_preferred_host" ],
    ph_hosts   => [ "my_ph_server" ],
}
__END__
```

## METHODS

Net::Config defines the following methods. They are methods as they are invoked as class methods. This is because Net::Config inherits from Net::LocalCfg so you can override these methods if you want.

requires\_firewall HOST

Attempts to determine if a given host is outside your firewall. Possible return values are.

```
-1  Cannot lookup hostname
 0  Host is inside firewall (or there is no ftp_firewall entry)
 1  Host is outside the firewall
```

This is done by using hostname lookup and the `local_netmask` entry in the configuration data.

## NetConfig VALUES

nntp\_hosts

snpp\_hosts

pop3\_hosts

smtp\_hosts

ph\_hosts

daytime\_hosts

time\_hosts

Each is a reference to an array of hostnames (in order of preference), which should be used for the given protocol

inet\_domain

Your internet domain name

ftp\_firewall

If you have an FTP proxy firewall (**NOT** an HTTP or SOCKS firewall) then this value should be

set to the firewall hostname. If your firewall does not listen to port 21, then this value should be set to "hostname:port" (eg "hostname:99")

#### ftp\_firewall\_type

There are many different ftp firewall products available. But unfortunately there is no standard for how to traverse a firewall. The list below shows the sequence of commands that Net::FTP will use

```
user      Username for remote host
pass      Password for remote host
fwuser    Username for firewall
fwpass    Password for firewall
remote.host The hostname of the remote ftp server
```

```
0      There is no firewall

1      USER user@remote.host
        PASS pass

2      USER fwuser
        PASS fwpass
        USER user@remote.host
        PASS pass

3      USER fwuser
        PASS fwpass
        SITE remote.site
        USER user
        PASS pass

4      USER fwuser
        PASS fwpass
        OPEN remote.site
        USER user
        PASS pass

5      USER user@fwuser@remote.site
        PASS pass@fwpass

6      USER fwuser@remote.site
        PASS fwpass
        USER user
        PASS pass

7      USER user@remote.host
        PASS pass
        AUTH fwuser
        RESP fwpass
```

#### ftp\_ext\_passive

#### ftp\_int\_passive

FTP servers can work in passive or active mode. Active mode is when you want to transfer data you have to tell the server the address and port to connect to. Passive mode is when the server provide the address and port and you establish the connection.

With some firewalls active mode does not work as the server cannot connect to your machine (because you are behind a firewall) and the firewall does not re-write the command. In this case you should set `ftp_ext_passive` to a *true* value.

Some servers are configured to only work in passive mode. If you have one of these you can force `Net::FTP` to always transfer in passive mode; when not going via a firewall, by setting `ftp_int_passive` to a *true* value.

#### `local_netmask`

A reference to a list of netmask strings in the form "134.99.4.0/24". These are used by the `requires_firewall` function to determine if a given host is inside or outside your firewall.

The following entries are used during installation & testing on the libnet package

#### `test_hosts`

If true then `make test` may attempt to connect to hosts given in the configuration.

#### `test_exists`

If true then `Configure` will check each hostname given that it exists