

**NAME**

**ng\_deflate** - Deflate PPP compression (RFC 1979) netgraph node type

**SYNOPSIS**

```
#include <sys/types.h>
#include <netgraph/ng_deflate.h>
```

**DESCRIPTION**

The **deflate** node type implements the Deflate sub-protocols of the Compression Control Protocol (CCP).

The node has two hooks, *comp* for compression and *decomp* for decompression. Only one of them can be connected at the same time, specifying node's operation mode. Typically that hooks would be connected to the `ng_ppp(4)` node type hook of the same name. Corresponding `ng_ppp(4)` node hook must be switched to `NG_PPP_DECOMPRESS_FULL` mode to permit sending uncompressed frames.

**HOOKS**

This node type supports the following hooks:

*comp* Connection to `ng_ppp(4)` *comp* hook. Incoming frames are compressed (if possible) and sent back out the same hook.

*decomp* Connection to `ng_ppp(4)` *decomp* hook. Incoming frames are decompressed (if they are compressed), and sent back out the same hook.

Only one hook can be connected at the same time, specifying node's operation mode.

**CONTROL MESSAGES**

This node type supports the generic control messages, plus the following:

**NGM\_DEFLATE\_CONFIG (config)**

This command resets and configures the node for a session (i.e., for compression or decompression). This command takes a *struct ng\_deflate\_config* as an argument:

```
struct ng_deflate_config {
    u_char    enable;                /* node enabled */
    u_char    windowBits;           /* log2(Window size) */
};
```

The *enabled* field enables traffic flow through the node. The *windowBits* specify compression windows size as negotiated by the Compression Control Protocol (CCP) in PPP.

**NGM\_DEFLATE\_RESETRREQ (resetreq)**

This message contains no arguments, and is bi-directional. If an error is detected during decompression, this message is sent by the node to the originator of the NGM\_DEFLATE\_CONFIG message that initiated the session. The receiver should respond by sending a PPP CCP Reset-Request to the peer.

This message may also be received by this node type when a CCP Reset-Request or Reset-Ack is received by the local PPP entity. The node will respond by flushing its compression state so the sides can resynchronize.

**NGM\_DEFLATE\_GET\_STATS (getstats)**

This control message obtains statistics for a given hook. The statistics are returned in *struct ng\_deflate\_stats*:

```
struct ng_deflate_stats {
    uint64_t  FramesPlain;
    uint64_t  FramesComp;
    uint64_t  FramesUncomp;
    uint64_t  InOctets;
    uint64_t  OutOctets;
    uint64_t  Errors;
};
```

**NGM\_DEFLATE\_CLR\_STATS (clrstats)**

This control message clears statistics for a given hook.

**NGM\_DEFLATE\_GETCLR\_STATS (getclrstats)**

This control message obtains and clears statistics for a given hook.

**SHUTDOWN**

This node shuts down upon receipt of a NGM\_SHUTDOWN control message, or when hook have been disconnected.

**SEE ALSO**

netgraph(4), ng\_ppp(4), ngctl(8)

J. Woods, *PPP Deflate Protocol*, RFC 1979.

W. Simpson, *The Point-to-Point Protocol (PPP)*, RFC 1661.

**AUTHORS**

Alexander Motin <*mav@alkar.net*>

**BUGS**

Due to nature of netgraph PPP implementation there are possible race conditions between data packet and ResetAck CCP packet in case of packet loss. As result, packet loss can produce bigger performance degradation than supposed by protocol.