

**HOST LINKS**



***GPROXY***<sup>TM</sup>

***Network Manager***  
***&***  
***SNMP proxy agent***

<http://www.gar.no/hostlinks/>



Microsoft, Windows, MS, MS-DOS are registered trademarks of Microsoft Corp.  
IBM and PC are registered trademarks of IBM Corp.  
UNIX is a registered trademark in the United States and other  
countries, licensed exclusively through X/Open Company, Ltd.

Any other product names are trademarks of their respective owners.

Version 6.6  
© Gallagher & Robertson as 1990-2013  
All Rights Reserved

**GALLAGHER & ROBERTSON AS**, Kongens gate 23, N- 0153 Oslo, Norway  
Tel: +47 23357800  
www: <http://www.gar.no/>  
e-mail: [support@gar.no](mailto:support@gar.no)

# Contents

---

<b>Host Links Gproxy .....</b>	<b>1</b>
Installation .....	1
Host Links Product Overview .....	1
Terminal environment .....	1
Server environment .....	2
Scope of the Product .....	3
Run-time licenses .....	4
<b>Load balancing.....</b>	<b>5</b>
Enabling management statistics .....	6
SSL connections .....	7
Port numbers .....	7
Ggate .....	8
Ggate port number .....	8
Client configuration .....	9
Glink.....	9
Glink for Java .....	10
Gweb .....	10
GwebS, the G&R web server.....	10
Port number .....	11
Other web servers .....	11
Port number .....	12
Client configuration .....	12
Glink for Java servers.....	13
Port number.....	13
Client configuration .....	13
<b>License sharing .....</b>	<b>15</b>
Initializing the license server.....	15
Requesting a license .....	15
License back up.....	16
Restoring license limits .....	16
License communities .....	17
Using ‘backup only’ licenses .....	17

## Contents

<b>Alarms.....</b>	<b>19</b>
The alarm text library .....	19
The alarm log .....	21
<b>SNMP support.....</b>	<b>23</b>
<b>Startup/Configuration.....</b>	<b>25</b>
Parameters .....	25
Command line examples .....	29
<b>Gproxy input data .....</b>	<b>30</b>
Type 0 – alarm record .....	30
Type 1 - connection event record.....	30
Type 2 - disconnection event record .....	30
Type 3 – Connection failures .....	30
Type 4 – Statistics record.....	31
Type 5 - Route state event record .....	31
Type 6 – Glink for Java Logon request record .....	31
Type 7 – Gweb statistics record .....	31
Type 8 – License statistics record .....	31
<b>Gproxy reports.....</b>	<b>32</b>
The System report .....	33
The Summary report.....	36
The Detail report .....	38
The Ggate Gateway report .....	40
The Load balancing reports.....	42
The Glinkj server report .....	44
The License server report.....	45
The Route report.....	46
The User report.....	48
The Session report .....	50
The MIB report.....	51
The ‘HEX report’ .....	53
<b>Administration.....</b>	<b>54</b>
Gmanager operator commands.....	54
Monitoring Gproxy .....	55
Gproxy logging .....	55

**Appendix: Host Links Manuals..... 56**

**Appendix: Host Links Server Administration..... 58**

**Appendix: Host Links DSA Utilities ..... 60**

    Gcname ..... 60

    Gerror ..... 61

    Glnode..... 61

    Gmacfix..... 61

    Gping..... 61

    Grnode..... 62

    Gtrace ..... 62

    Gtsupd ..... 62

**Appendix: Host Links Trace ..... 64**

Trace activation ..... 64

Trace types ..... 64

Structure ..... 65

Tracing Ggate..... 66

Examples - G&R products ..... 66

CPI-C and Gweb trace files..... 68

**Appendix: Error Codes..... 70**

OSI/DSA error codes ..... 70

Windows Sockets error Codes ..... 82

**Appendix: G&R MIB extension..... 85**

**Appendix: Formatted SNMP trap trace..... 91**

# Contents

# Host Links Gproxy

---

## Installation

The G&R emulations and gateways are independent programs, but part of the *G&R Host Links* product set available on all major UNIX/Linux platforms. Many of the products are also available for Windows servers. For details on platforms supported, software delivery and installation refer to the *Host Links Installation and Configuration* manual.

## Host Links Product Overview

### Terminal environment

Host links products that run on UNIX or Linux servers with a terminal driven user interface include emulators and concentrators, as well as various utilities.

- **G3270** provides synchronous IBM3270 functionality. G3270 emulates IBM LU type 2, including base and extended colour together with extended highlighting.
- **Qsim** provides synchronous Questar terminal functionality. Qsim simulates all Questar models, including the DKU7007, DKU7107, DKU7105 and DKU7211 (Mono, four colour A/B and seven colour modes are supported). It also simulates the VIP7760 and the VIP7700.
- **V78sim** provides Bull VIP78xx (BDS) functionality. V78sim emulates all models of the VIP7800 family; the actual reference is the BDS7. All visual attributes including colour are supported.
- **Pthru** provides transparent VIP7800 visibility to Bull mainframes for users with asynchronous VIP7800 terminals or emulators. The terminals are used in text or forms mode.

### ***Server environment***

Host Links products that run on UNIX, Linux or Windows servers.

- **Ggate** is a transparent gateway to the Bull native network. It avoids all need for Front-ends (MainWay/Datanet) or other gateways. It can be used to connect G&R/Glink (for Windows or Java) emulators or any of the emulators, concentrators, network printer emulators and file transfer clients/servers in the Host Links product set. It also supports third party clients using the TNVIP, TN3270, TN3270E and standard asynchronous Telnet protocols.
- **Gweb** provides a web browser interface to any host application that is otherwise accessible using the *Host Links Qsim, V78sim, or G3270* emulations.
- **Gspool** is designed to run as an unattended process and accept transparent print output from any type of host application (GCOS8, GCOS7, GCOS6, IBM) that normally sends print data to network printers (ROPs), or to a remote spooling system (DPF8-DS). On the Gspool system the print may be directed to a physical printer or to the local spooling system. Gspool operates in different modes, Connect mode, Terminal Writer mode, DPF8 mode, SNM mode, IBM mode, TN3270 mode and TN3270E mode.
- **GUFT** is a G&R implementation of the Bull UFT file transfer protocols. It enables transfer of data files between Host Links and GCOS systems over a DSA network.
- **Gproxy** is a network management program used for supervision, management, load balancing and license sharing of G&R *Host Links* applications. Gproxy can be set up as a freestanding monitor program and/or report generator in a small network, or play a bigger role in a larger network.
- **Gsftp** is a transparent gateway between two different File Transfer protocols: FTP (RFC 959) and SFTP (the SSH File Transfer Protocol). The purpose is to present a seamless integration between the two protocols, with automatic conversion.



# Scope of the Product

Gproxy is a network management program used for supervision, management, load balancing and license sharing of G&R *Host Links* applications. Gproxy can be set up as a freestanding monitor program and/or report generator in a small network, or play a bigger role in a larger network. It can also act as an SNMP 'proxy agent' and report to one or more SNMP management systems. Even though Gproxy can in principle be used by anybody who has access to a Host Links installation, the typical Gproxy user is an administrator or an operator in a network control center.

One of Gproxy's most important features is load balancing for G&R gateway and server applications, ensuring that connections from *Host Links* clients or from *Glink* users are evenly spread.

Another important task is acting as a license server in a shared license configuration, ensuring that if a *Host Links* gateway or server system is temporarily out of business, the licenses belonging to the failing system will be automatically taken over by the remaining *Host Links* gateway or server systems.

Gproxy receives and acts on information in a 'network management UDP record' that is optionally generated and broadcast by G&R *Host Links* applications. These broadcasts contain either statistical type information (e.g. gateway load) or network event type information, such as detailed information about connections, disconnections, error events and backup-route switching.

On UNIX/Linux systems, Gproxy can run in the foreground on a user screen (in 'VDU presentation mode'), or it can be set up to run unattended as a background process. In the latter case Gproxy can optionally generate HTML reports (in 'HTML presentation mode') that can be viewed by anyone with a web browser. A choice of reports is available in both presentation modes that enable the administrator to monitor Host Links user activities and resource usage.

If Gproxy is used in an SNMP environment, Gproxy can be set up to forward information in SNMP 'trap' format to up to 4 SNMP manager stations located somewhere in the network.

Gproxy can optionally log network administration information to a log-file that can be viewed at any time.

## Gproxy

Gproxy is delivered as a single executable. No data- or configuration files are necessary, but the command line arguments can optionally be included in a configuration file.

## Run-time licenses

In order to run Gproxy, the following license keys must be present in your licenses file:

basic	For the base G&R run-time system
gproxy	For Gproxy

If SSL is used, you also need the following license key:

ssl	For the SSL versions of the DSA/DIWS line handler
-----	---

The licenses file is located:

<b>Windows Server</b>	<code>\gar\config\licenses</code>
<b>UNIX/Linux</b>	<code>/usr/gar/config/licenses</code>

The licenses file identifies the G&R distributor, the owner of the license and the licensed products. The license key for a product will normally state for how many simultaneous sessions the product is licensed. If a limitation is specified in the license, only the licensed number of sessions can be active at any time.

# Load balancing

---

An important Gproxy feature is the ability to balance the load of gateways and server programs. Currently Gproxy can perform load balancing for *G&R/Ggate* gateways, *G&R/Gweb* servers and *G&R/Glinkj* servers. A subset of the load balancing functionality is also supported for other (i.e. non *G&R*) web servers. Gproxy can perform load balancing for all of these servers simultaneously if desired.

Gproxy running on a single system can load balance for several servers running on different systems. Two copies of Gproxy on different systems provide robust load balancing that functions when one Gproxy system is down, provided the clients are configured to try both systems, or if the DNS name server is configured to return both systems when the client does a look-up for the IP-address. If more Gproxy copies are to be used for load balancing, then the DNS configuration needs to be more advanced.

If Gproxy is to also provide license-sharing services, then it is a prerequisite that Gproxy runs on all systems where a server runs. When license sharing is in force it overrides load balancing, so that clients are not directed to a server that is operative, but has no licenses available (a back-up server).

Load balancing of the *G&R* products is based on statistical information that is optionally distributed in network management UDP records by the various Host Links server components. In load balancing mode, Gproxy maintains various load-tables that contain current, highest and maximum usage counters as well as various other statistical information reported by these servers.

Since this statistical information is sent out at regular intervals, Gproxy detects any non-responsive or terminated server by the absence of its UDP statistics record. If no statistical information has been received from a given server within the time limits (configurable, default 20 seconds), Gproxy assumes that the server has been terminated. This ensures that clients are always directed to an operative, as well as the lightest loaded, server program.

## Gproxy

In load balancing mode, Gproxy takes over the TCP/IP socket of the server in question and 'listens' for connection requests coming in from the clients e.g. requests for connection to *G&R/Ggate* or *G&R/Gweb*. When a connection request comes in from a client, Gproxy sends back a 'redirect record'. This redirect record contains the network address of the server with the lowest load. The client then automatically issues a new connect request using the suggested network address. The suggested server can be located on the same system as Gproxy or on some other *Host Links* system in the network. This initial redirection dialog is handled automatically by the client application and is transparent to the user.

Gproxy optionally generates load-balancing reports in either 'VDU' or HTML format (see the *Gproxy reports* chapter). The reports are updated every time load statistics are received from the servers and show the current, highest and maximum load of the server program.

Load balanced servers can be manually disabled and re-enabled by means of simple operator commands (see the *Administration* chapter).

Load balancing is a background task that takes place independently of other activity, such as report generation.

## ***Enabling management statistics***

The statistical information, which is a prerequisite for the load balancing functionality, must be explicitly enabled by command line parameter to the server or gateway. The information can be either broadcast (i.e. made available to all systems) or directed to up to 4 specific Gproxy addresses.

All server programs support the following parameters:

<code>-NA xxx yyy zzz</code>	Send statistics to Gproxy network administration stations (up to 4 IP-addresses may be given)
<code>-NB on/OFF</code>	Enable broadcasting to all Gproxy stations

e.g. send statistics from a *G&R/Ggate* gateway to Gproxy on 'saturn' and 'jupiter':

```
gg_tcp -na saturn jupiter -lp 30843 .....
```

## SSL connections

Gproxy supports connections from native Ggate SSL clients and can load-balance these in exactly the same way as ordinary (non-SSL) connections. In the case of a SSL configuration, you need a separate version of the Gproxy program, *GproxyS*. There are no other functional differences between these two versions of the program than SSL support; the SSL version supports an extra set of common SSL command line parameters and uses other port numbers as outlined in the paragraph below.

For a general description of SSL support in Host Links programs, and the command line parameters, see the *G&R/SSL* manual.

## Port numbers

When Gproxy is load balancing a server, it takes over the TCP/IP socket (i.e. the port number) that is normally used by the server for client connections. The table below shows the default port numbers used by Gproxy when load balancing the various servers, and the Gproxy command line parameter that can be used if you want to use a non-default port (note that if you change the default port, you must also change the client configurations).

Server	Default port	Parameter
Ggate	30841	-LBGGA NNNNN
Ggate SSL	30851	-LBGGA NNNNN
GlinkJS	30842	-LBGLJ NNNNN
GlinkJS SSL	30852	-LBGLJ NNNNN
Gwebs	80	-LBGWB NNNNN
Gweb SSL	443	-LBGWB NNNNN
Other web server	80	-LBWEB NNNNN
Other web server SSL	443	-LBWEB NNNNN

# Ggate

*Ggate* load balancing is by default enabled. e.g.

```
gproxy -pt html -rt gga
```

Gproxy would run in default background mode, doing default load balancing for *Ggate*. It would generate a HTML report for *Ggate* load balancing.

In load balancing mode, Gproxy will open the *Ggate* TCP/IP socket and listen for connection requests coming in from *Glink*, from *Host Links* clients or from user-written *GIAPI* applications. Gproxy will use the default *Ggate* port number (see ***Ggate port number***) unless another port number has been given with the `-LBGGA` command line parameter. When a connection request is received on the socket, Gproxy starts up a separate process that takes over the initial dialog with the client. When the client parameters are received and validated, a redirect command is sent to the client redirecting it to the *Ggate* system with lowest load factor. Gproxy then terminates the connection to the client. The client automatically issues a new connect request using the suggested network address.

If the balanced systems are of comparable size, the load factor is simply the number of concurrent sessions reported by *Ggate*. If Gproxy is load balancing *Ggate* systems of different sizes (most importantly as regards the physical link capacity i.e. link types and numbers), a 'load threshold' value can be used by *Ggate*; in which case the load factor is calculated based on the threshold percentage.

## ***Ggate port number***

The default port used by *Ggate* is 30841. Gproxy takes over this port when load balancing, so *Ggate* must be started using another port. There are two ways of achieving this:

Use the *Ggate* `-gga` command line parameter e.g.

```
gg_tcp -gga 30843 -nb
```

Configure the *Ggate* port number in the TCP/IP ‘services’ file. This file is located in the following directory:

<b>Windows server</b>	%SystemRoot\System32\Drivers\Etc
<b>UNIX/Linux</b>	/etc

add a line to the file e.g.

```
ggate 30843/tcp
```

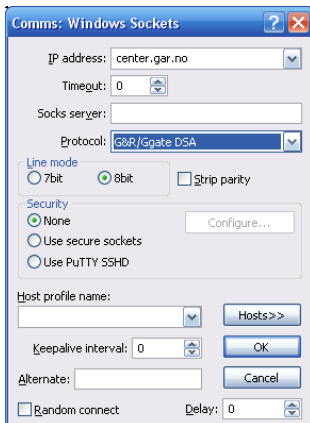
Alternatively Gproxy can be configured to use another connection port. This is done in the same way using either the ‘services’ file or the Gproxy `-LP` parameter. This is not the preferred alternative since it would require that all client (e.g. Glink) configurations would need to be configured to use a non-standard port number.

If an attempt is made to start *Ggate* and Gproxy using the same port number, the second program will not start. An error message will be returned to the caller e.g. if *Ggate* was the last program to start:

*Bind Ggate socket failed (Ggate executing on same system?)*

## Client configuration

### Glink



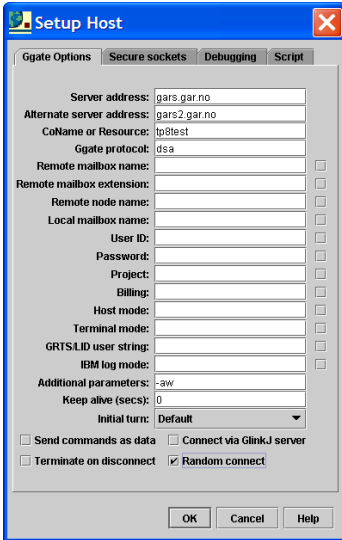
The **IP-address** remains the same, but Gproxy is running on the target system, and has taken over the Ggate port.

An **Alternate** IP-address is configured in case the first connect fails. This is a second Gproxy system. The result is robust load balancing, which functions even if one Gproxy system is down.

The delay of zero means that connects are sent simultaneously to both Gproxy systems.

## Gproxy

### *Glink for Java*



The **Server address** remains the same, but Gproxy is running on the target system, and has taken over the Ggate port.

An **Alternate server address** is configured in case the first connect fails. This is a second Gproxy system. The result is robust load balancing, which functions even if one Gproxy system is down.

The **Random connect** check box causes Glink for Java to try the first connect to the Server or Alternate server in a random manner.

## *Gweb*

### *GwebS, the G&R web server*

You enable load balancing for the *Gweb* server (*GwebS*) using the Gproxy command line parameter `-LBGWB [port number]`. The port number is optional, and defaults to 80 (the standard *GwebS* port) e.g.

```
gproxy -nlb -lbgwb -pt html -rt gw
```

Gproxy runs in default background mode, doing load balancing for *GwebS*, and generating a *Gweb* load balance report in HTML. Load balancing for *Ggate* is suppressed. Note that port 80 is a restricted TCP/IP port. This is no problem for the Windows version of the server program, but in the UNIX/Linux case Gproxy must be launched from the `root` user and the `s-bit` must be set on the executable file.



When load balancing for *GwebS*, Gproxy takes over the *GwebS* port (see ***GwebS port number***) and listens for *Gweb* connection requests coming from the client browsers. When a connection is received, a standard HTTP redirect command is sent back to the browser containing the IP-address and port number of the *GwebS* system with the lowest load.

## ***Port number***

The default port used by *GwebS* is the standard web server port (port number 80). This is taken over by Gproxy when load balancing is enabled, so *GwebS* must be started using another port e.g. using the *GwebS* `-LP` command line parameter e.g.

```
gwebs -lp 8000 -nb
```

## ***Other web servers***

Gproxy offers a subset of load balancing functionality for other web servers. This is enabled by the command line parameters:

```
-LBWEB [port number] -WEBSPORT port number
```

```
gproxy -nlb -lbweb -websport 8000
```

Gproxy must run on every system that runs a web server to be load balanced, and must be started with the `-lbweb` parameter, which causes it to check local *Gweb* usage, and broadcast web server statistics on behalf of the non-*G&R* web server. These broadcasts enable the load balancing function.

Gproxy takes over the web server port given in the `-LBWEB` parameter (defaults to 80) and listens for *Gweb* connection requests from the client browsers. When a connection is received a standard HTTP redirect command is sent back containing the IP-address of the web server with the lowest current load, and the non-standard port number on which the web servers are listening (given to Gproxy in the `-WEBSPORT` parameter).

## **Gproxy**

There are restrictions when load balancing for other web servers; there is no communication between Gproxy and the non-*G&R* web server on who's behalf Gweb is generating web server statistics, and therefore no way Gproxy can be sure of the state of the web server. If the *Gweb* usage count is zero, this can be due to the fact that the web server is not executing. If the web server has for some reason stopped, a manual intervention is necessary (e.g. restart Gproxy without the `-LBWEB/-WEBSPO` parameters on that system).

### ***Port number***

The default connection port number used by web servers is 80. This is taken over by *GPROXY* when load balancing is enabled, so the non-*G&R* web server in question needs to be started using another port.

### ***Client configuration***

Web browsers have standard functionality for connecting to an alternative address if the first connect fails. URLs for connection to a particular web server are always symbolic, and must be looked up in a DNS server. If there are two web servers that are equivalent in functionality (i.e. both have Gweb running on the system), then DNS for the system can be configured to return both addresses, and browsers automatically try the second if the first is down.

## ***Glink for Java servers***

You enable load balancing with the Gproxy command line parameter `-LBGLJ` [`port number`]. The port number is optional, and defaults to 30842 (the standard GlinkJS port) e.g.

```
Gproxy -nlb -lbglj -pt html -rt glj
```

In the example, Gproxy does load balancing for GlinkJS, and generates a GlinkJS load balance report in HTML. It does not do load balancing for Ggate.

When doing load balancing for *Glinkjs*, Gproxy takes over the *Glinkjs* port (see ***Port number***), and listens for client connection requests. When it receives a connection it sends a redirect command containing the IP-address and port number of the *Glinkjs* system having the lowest current load.

When *Glinkjs* is started with the `-NA/-NB` parameter it reports load information in a statistical network management record that is sent out at regular intervals to Gproxy.

### ***Port number***

The default connection port used by *Glinkjs* is 30842. This is taken over by Gproxy when load balancing is enabled so *Glinkjs* must be started using another port using the *Glinkjs* server's `-LP` command line parameter e.g.

```
glinkjs -lp 30844 -nb
```

### ***Client configuration***

The port number in the client configuration remains the same, but the client connects to Gproxy. Gproxy then redirects the client to the Glinkjs with least load. You change `glink.ini` to specify the Gproxy system and port and when load balancing Gproxy would normally be running on two systems.

```
configserver=myserver.acme.com
backupserver=mybackup.acme.com
```



# License sharing

---

In a multiple system *Host Links* configuration, Gproxy by default operates as a license server for *Host Links* programs if the licenses have a site limit in addition to the system limit. In such a configuration, Gproxy must run on all *Host Links* systems sharing licenses.

When operating in license server mode, Gproxy is responsible for allocating and de-allocating licenses for the local applications, and distributing license statistics information periodically to other Gproxy license servers. The license statistics allow Gproxy to know the state and current usage of the shared licenses on all the *Host Links* systems, and in the event that one or more of the systems should fail, makes it possible for the remaining servers to take over the licenses belonging to the failing systems.

Gproxy can generate a license server HTML report. It shows the status and license usage of all the license servers in the configuration and Gproxy updates it each time it receives a license statistics broadcast.

## Initializing the license server

When Gproxy starts, it reads the local license file looking for shared license entries. A shared license is one for which there is a site limit in addition to the system limit. Currently Gproxy handles license sharing for *Ggate*, *Gweb*, *Glink for Java*, *GLapi* and UNIX/Linux-based emulator licenses (*Qsim*, *V78sim*, *G3270*), up to a maximum of eight different products. If Gproxy finds shared licenses, it starts a license-server child process. The child initializes license usage structures and license server pipe-files for all shared license entries. It then starts its normal license routine, listening for license statistics from other license servers and broadcasting its own license tables.

## Requesting a license

*Host Links* maintains all license entries in a local license pool. The first time any Host Links application starts on a system it builds the license pool in the system's shared memory.

## **Gproxy**

*Host Links* applications check their license when they start. They request unshared licenses from the local license pool. They request shared licenses by initiating a connection to the license server over the license server pipe-file. The Gproxy license server process receives this license request and returns the first available license it finds in the license pool, or rejects the request if no licenses are available. When the *Host Links* application terminates, it calls the license server to release the license.

## ***License back up***

A Gproxy license server knows the maximum and current license load of the systems at the same site from the license statistics that are periodically (default every 10 seconds) broadcast by each Gproxy operating in license sharing mode. If a Gproxy license server stops sending license statistics, the others assume that the system on which it is running has failed, and automatically take over the failing system's licenses. In a configuration with multiple active license servers, the remaining license servers share the failing system's licenses.

When a Gproxy license server starts, it expects other Gproxy license servers at the same site to have reported their license statistics within an initialization period (currently set to 5 minutes). If the sum of the license limits reported by the other systems does not reach the configured 'site limit' within this initialization period, Gproxy assumes that one or more systems must be inoperative and this triggers the license backup routine as described above.

## ***Restoring license limits***

Whenever Gproxy license servers detect a new Gproxy license server, they adjust their license tables to reflect their share of the site limit. If this new system is a known (previously active) system that comes back on line, the license servers restore their original license limits and resume normal processing.

## ***License communities***

A license server operates in a 'license community' if a license entry has a 'community number'. In such cases, the license server will only maintain license information reported by servers in the same community. If the license community number reported in a broadcast is not identical to the one in the license entry, the license server ignores the license broadcast. This makes it possible to divide an intranet of production systems into several license 'subnets' or to mix test systems and production systems without confusing the license server routines.

## ***Using 'backup only' licenses***

A shared license entry can have a license limit of zero. This license entry is ignored until one or more systems in that license configuration is not functioning i.e. it is treated as a 'backup-only' system. In a backup situation (if one or more systems stop working), the backup-only site license is activated and the system takes its share of the shared license. The backup-only system deactivates when the license network is operational, i.e. when the failing system is back on line.





# Alarms

---

Gproxy optionally generates a stack of ‘Alarms’. This stack is in HTML format and contains error messages generated by *Host Links* applications. *Host Links* applications call the alarm routines whenever an event has taken place that might need some sort of manual intervention.

The alarm stack makes it easier for the administrator to investigate possible abnormalities and simplifies the task of finding significant error messages which otherwise might be located in various log and trace files.

There are 4 Gproxy alarm severity levels:

```
6 - information
4 - low/warning
3 - high/error
2 - critical
```

Gproxy takes a command line parameter `-ALM N`, where `N` sets the Gproxy severity level i.e. the lowest severity (highest number) to be included in the HTML alarm stack e.g. `-ALM 3` results in ‘error’ and ‘critical’ alarms only.

Gproxy takes a command line parameter `-ALMSTK NNN`, where `NNN` is the number of entries in the circular alarm stack.

The Alarm routines are implemented in *Gspool*, *GUFTsrv*, *Listener* and in *Ggate*. Alarm routines are also called from some of the central subroutines that are used by most *Host Links* server programs.

## The alarm text library

The alarm text library is located in the ‘misc’ subdirectory:

<b>Windows server</b>	<code>\gar\misc\alarm</code>
<b>UNIX/Linux</b>	<code>/usr/gar/misc/alarm</code>

## Gproxy

It is a plain text file and can be modified or translated to suit the installation. The severity levels can also be changed, but the record format must be correctly maintained.

The following is an extract of the alarm library:

```
# Format: NNNN M,text string
# NNNN = 4 digit alarm number
# M = 1 digit alarm level, 6=info, 4=warning, 3=error, 2=fatal
# e.g. Error 1234 with severity 3 and text 'Warning' is coded:
# 1234 3,Warning
#####
#
# 100 - Listener texts
#
0101 2,Could not initialize Gmonitor information
0102 3,The Gstart DLL could not be loaded
0103 3,Failed to allocate memory for listening application
0104 3,Refusing incoming connection request
#
# 200 Ggate
#
0201 2,Accept client socket failed
0202 2,Invalid Ggate license
0203 2,Could not open Ggate server structure
0204 2,Could not open Log file
0210 3,Insufficient Ggate client licenses
0211 4,Operator requested DISConnect of all clients
#
# 300 Gspool
#
0304 3,Macro not defined
0305 3,Gspool failed to open and write to file
0306 3,Gspool will now abort DPF8 connection
0307 3,DPF8 report count reset to 0, report files removed
0308 4,DPF8 STOP command received
0309 3,Printdata received for unknown printer
```

# The alarm log

The following is an example of an Alarm log:

**FAST LINKS**  
**GPROXY™**

## Alarm log

Seq: 001, Time: Mon Aug 11 16:18:20 , Program: Guftsrv 620, Address: arldxp.gar.no  
-> **Severity: Info (6), 401 : Client session disconnected**

Seq: 002, Time: Mon Aug 11 16:18:35 , Program: Gl\_dsa 620, Address: arldxp.gar.no  
-> **Severity: High (3), 0 : Refusing connection from mbx:D21205 mode:IS2C to mbx MYPRTY, reason 0082 (UNKNOWN)**

Seq: 003, Time: Mon Aug 11 16:18:46 , Program: Gspool 620, Address: arldxp.gar.no  
-> **Severity: Critical (2), 0 : \$\$\$DSA: Session error: Request denied, mailbox in use by another process**

Seq: 004, Time: Mon Aug 11 16:19:14 , Program: Gspool 620, Address: arldxp.gar.no  
-> **Severity: Critical (2), 356 : Spooler/print-device check failed, session aborted**

Seq: 005, Time: Mon Aug 11 16:19:34 , Program: Guftsrv 620, Address: arldxp.gar.no  
-> **Severity: Low (4), 408 : Could not assign local file**

Seq: 006, Time: Mon Aug 11 16:19:38 , Program: Guftsrv 620, Address: arldxp.gar.no  
-> **Severity: Info (6), 401 : Client session disconnected**

Seq: 007, Time: Mon Aug 11 16:19:53 , Program: Ggate 620, Address: arldxp.gar.no  
-> **Severity: Low (4), 211 : Operator requested DISConnect of all clients**

Seq: 008, Time: Mon Aug 11 16:20:17 , Program: Ggate 620, Address: arldxp.gar.no  
-> **Severity: High (3), 210 : Insufficient Ggate client licenses**

Seq: 009, Time: Mon Aug 11 16:20:38 , Program: Gspool 620, Address: arldxp.gar.no  
-> **Severity: Critical (2), 356 : Spooler/print-device check failed, session aborted**

Opening page http://arldxp:8090/cgi-bin/nph-push.exe/gproxy/almdat.ssr?3f37a636...



# SNMP support

---

Gproxy can act as an SNMP ‘proxy’ agent. SNMP support is enabled by the `-mipN` command line parameter. In SNMP mode, Gproxy sends SNMP ‘enterprise specific traps’ to SNMP network management systems. Up to 4 SNMP management systems can be addressed. SNMP traps can be filtered using the ‘exclusive filter’ parameter, `-efX`. Generation and distribution of SNMP traps is a background task that takes place independent of other (output-) activity.

The SNMP trap format is available as an enterprise specific MIB extension. Two trap formats are used: ‘event trap format’ (used by enterprise specific trap 1,2 and 3) and ‘statistic trap format’ (used by enterprise specific trap no.4). See appendix for a detailed description of the G&R MIB extension. See also the formatted dump of the SNMP traps.

Gproxy does not support SNMP ‘get’, ‘getnext’ and ‘set’ commands.



# Startup/Configuration

---

## Parameters

Gproxy takes parameters from the command line and/or from a configuration file, `gproxy.cfg` located in the standard Host Links configuration directory i.e.

```
/usr/gar/config/default/gproxy.cfg
```

Parameters from the command line override parameters from the configuration file. The configuration file expects a single parameter (with qualifier if appropriate) per line. Any text to the right of a [`* ; #`] character is regarded as a comment.

Gproxy accepts the following parameters:

Parameter	Description
<code>-alm N</code>	Enable Alarm HTML report and set the severity level N: 6=info, 4=low, 3=high, 2=critical. All levels less than N are also reported. N=3 reports high and critical
<code>-almstk nnnn</code>	Size of the circular alarm stack in entries
<code>-bg ON/off</code>	Sets Gproxy in background mode and runs as a server
<code>-ci 10</code>	Check interval in seconds. How often Gproxy checks for inactive gateways and updates status information
<code>-co name</code>	SNMP community name used by the receiving SNMP entity
<code>-dbg on/OFF</code>	Activate Gproxy tracing
<code>-efX</code>	Exclude forwarding SNMP traps of broadcast record type X. This parameter is positional i.e. must follow the <code>-miPn</code> parameter to which it applies
<code>-erX</code>	Exclude reporting of broadcast record type X (1-8)

## Gproxy

Parameter	Description
-gga addr	Monitor the Ggate gateway running on the specified IP address (necessary for Ggate report mode, was -gwa)
-glj addr	Monitor the GlinkJ server running on the specified IP address (necessary for GlinkJS report mode)
-hp path	HTML path. Default SYSDIR/html/gproxy
-idt GUID/IP	Client identification type. Default IP
-ipn ON/off	Use numeric/symbolic IP addresses (reports)
-lbgga ON/off /portno	Load balance Ggate. Default ON with port 30841
-lbglj on/OFF /portno	Load balance GlinkJS. Default OFF. Default port 30842
-lbgbw on/OFF /portno	Load balance GwebS. Default OFF. Default port is 80
-lbweb on/OFF /portno	Load balance other web server. Default port is 80
-lic on/off	Enable license server activity. Default enabled if license file contains site limits
-ln scid	Local DSA node name. Defaults to the local node from the DSA configuration file
-log ON/off	Activate logging. Writes a summary record to the log file for each broadcast received. When in 'server mode', the log file will be located in the server directory /usr/gar/servers/SCID.gpr/ Otherwise the log file will be opened in the working directory
-logsz KB	Sets the limit of the log file. Default 64KB
-mipN addr	IP address of the SNMP management system. Up to 4 management stations are supported (i.e. N = 1-4). If no SNMP IP address is given, SNMP activities are disabled



Parameter	Description
-myip addr	IP address of the system on which Gproxy is running. This should not normally be needed, but Gproxy can fail to find its own address on systems with multiple network cards (or an incorrect network configuration)
-pt vdu html all	Presentation mode. Set to VDU, HTML or ALL if Gproxy should generate reports. Defaults to no reporting. Setting VDU forces background OFF
-rt type types	If presentation type (-pt) is set. Possible report types: SUM – summary report DET – detail report GGA – Ggate load balance report GLJ – GlinkJ load balance report GWB - Gweb load balance server report LIC - license server report MIB – MIB report RTE - route report SES – session report USR – user report HEX – hex dump report ALL – all reports except ‘summary’  Several -RT parameters can be supplied
-si secs	Time before an inactive server is considered defunct
-snf IPmask	Gproxy should redirect Glink connection to Ggate systems in the same subnet first. Sets mask to be used to check for the same subnet
-sno IPmask	If Gproxy should redirect Glink connection to Ggate systems in the same subnet only. Sets mask to be used to check for the same subnet
-ssi ssi	SSI (Server Side Include) extension used for HTML pages that need server-side update. Default ‘SSI’
-uc 512	Max user count. Number of users monitored in ‘user’ report mode. Maximum 4096. Defaults to 512
-websport	Port used by non-G&R web servers for which Gproxy is doing load balancing.

## **Gproxy**

For SSL parameters, see the ***G&R/SSL*** manual.

## Command line examples

`gproxy`

Gproxy runs in background mode doing load balancing for Ggate, logging of all incoming broadcasts in the G&R server directory and license server processing if the license contains a site limit.

```
gproxy -nlb on -lbglj 30845 -pt html -rt glj -glj hlpC
```

Gproxy runs in background mode doing load balancing for a Glinkj server on port 30845, and generating a report showing usage of the Glink for Java server on the system having the IP-address `hlpC`. It does license server processing if the license contains a site limit. Load balancing is disabled for Ggate.

```
gproxy -pt html -rt sum -rt lic -rt gga -er4
```

Gproxy runs in background mode with load balancing for Ggate. It does license server processing if the license contains a site limit. A summary report, load balance report and a license server HTML report are generated. Record type 4 is ignored (on the summary report).

Given the following `gproxy.cfg` file:

```
*
* Parameters file for Gproxy on 192.111.222.24
*
-co secret
-mip1 saturn           ; Manager on 'Saturn'
-ef2                   ; Ignore disconnects
-mip2 plut011        ; Manager on 'Plut011'
```

and this command line:

```
gproxy -pt html -rt gga
```

Gproxy runs in background mode showing Ggate load balancing statistics in a HTML report. SNMP support is enabled and Gproxy communicates with the SNMP 'community' `secret`. It forwards all broadcasts as SNMP traps to IP-address `saturn` and forwards all broadcasts except type2 (disconnects) to `pluto11`. It does Ggate load balancing (default).

# ***Gproxy input data***

---

Gproxy receives and acts on network related administrative data records that are optionally broadcast by Host Links components (most importantly line handlers and gateways) in the network. They report statistical information or details of connections, disconnections, error events and backup-route switching. The following record types are defined:

## ***Type 0 – alarm record***

These records contain Alarm information e.g. severity and error text. Host Links server programs send alarms. The record is used for the Alarm log.

## ***Type 1 - connection event record***

These records contain data on a DSA connection event. It includes addresses and names of the client, the application and the host, and a list of all the Gline parameters given (and default) for the session. The line handler (`gl_dsa`) generates this record type when `-NA` or `-NB` is among the Gline parameters.

## ***Type 2 - disconnection event record***

These records contain information about normal user or application-initiated disconnects. The line handler (`gl_dsa`) generates this record when a client disconnects and `-NA` or `-NB` is among the Gline parameters.

## ***Type 3 – Connection failures***

These records contain information about unsuccessful connection attempts. Both the error-code and error-text are supplied, as well as the parameters used in the failing request. The line handler (`gl_dsa`) generates this record when a client connection request fails and `-NA` or `-NB` is among the Gline parameters.

## **Type 4 – Statistics record**

These records contain statistical information about gateway load and resource usage. *Ggate* and the *Glink for Java* server generate them for use in load balancing and load reports.

## **Type 5 - Route state event record**

These records contain information about network events that cause a change in a network route state. The state is changed either automatically due to an unsuccessful network connection, or as a result of an operator update command (using either the `gtsupd` utility or an internal Gproxy service command).

## **Type 6 – Glink for Java Logon request record**

These records contain information about a client registering itself with the Glink for Java server. Used for the *GlinkJ* server report.

## **Type 7 – Gweb statistics record**

These records contain statistical information about *Gweb* load and resource usage. *Gweb* generates them for use in load balancing and load reports.

## **Type 8 – License statistics record**

These records contain information about shared license resource usage. Gproxy generates them when license sharing is enabled.

# Gproxy reports

---

Administrative records broadcast from the various Host Links components will, as long as the event is relevant for the chosen report and not explicitly excluded by the `-erX` parameter, cause reports to be generated or updated if reporting is enabled (`-pt`). The report types generated or updated can be specified individually (several `-rt`) or collectively by means of `-rt ALL` (note that ALL does not include the 'summary report').

Gproxy supports two presentation modes, VDU and HTML. In VDU presentation mode the event will immediately be visible in the current report on the user's terminal. VDU mode requires that Gproxy is run in the foreground. In HTML mode a new HTML page is generated and will be sent to the browser using a 'push-technique' without any interaction from the browser user. This documentation will focus on HTML mode.

Several report types are supported and enabled by command line arguments at Gproxy startup time. The HTML reports are displayed in separate browser windows. The administrator can display all reports that are of particular interest, or can choose to have the summary report active and select the appropriate report whenever an interesting network event occurs in the summary report.

The following is a description of the report types generated by Gproxy.

## The System report

The system report is a collection of valuable information otherwise found on several other reports. It is generated automatically whenever the `-PT HTML` parameter is set and is the natural choice of report in a daily monitoring situation. The report is updated every 'check interval' (default 10 seconds) and contains the following:

### System information:

IP address	Local IP address
Node name	Local node name (DSA SCID)
System ID	Local system ID (computer name)
Time	Last time stamp
Base directory	Host Links base directory (SYSDIR)
Started	Gproxy start time

### Static license information:

Installation	Customer name
Platform	Type of platform (e.g. Windows)
Release	Current release number
SSL Mode	If operating in SSL Mode
Customer no	Customer number
Ref. No	License reference number
Community	License community number

## Gproxy

### License server information (one line per shared license entry)

License name	The name of the license entry
Used	Number of licenses currently in use
Max	Current maximum number of licenses (current limit)
Org	Original maximum license limit
Site	The site limit
Corp	Corporate limit (for future use)

### Load balance information (one line for Ggate, web server and GlinkjS)

Port	Port number used for client connection to server
Servers	Number of servers in this network
Best	Current 'best' server
Load factor	Load factor (normally current number of connections)
Redirects	Number of redirects since startup

**Event information** (one set for each record type: license stats, gateway stats, connect events, connection failures, GlinkjS stats, Alarms, GwebS stats, disconnect events, route events and IO counts):

Number	Number of records received (since last update)
Total	Number of records received (since startup)



## Example System report:



**System report**

**System information**  
 IP addr.: [arldxp.gar.no](#) Node name: LAB1 System ID: [arldxp](#)  
 Time: 10:30:39.266 Base dir.: d:\gar Started: Wed Aug 06 10:01:19 2003

**Static license information**  
 Installation: G&R A/S Platform: \* Release: SSL mode: No  
 Customer no: 0001 Reference no: 0001 Community: 0

**License server information**  
 License name Ggate Used: 0022 Max: 0200 Org: 0100 Site: 0200 Corp.: 0500  
 License name Gweb Used: 0024 Max: 0100 Org: 0050 Site: 0100 Corp.: 0100

**Load balance information**  
 Ggate port: 30841 Servers: 2 Best: [hobby.gar.no](#) Loadfactor: 0 Redirects: 0  
 Webserv port: 8080 Servers: 4 Best: [jim2.gar.no](#) Sessions: 0 Redirects: 0  
 Glinkjs port: 30842 Servers: 2 Best: [hobby.gar.no](#) Loadfactor: 0 Redirects: 0

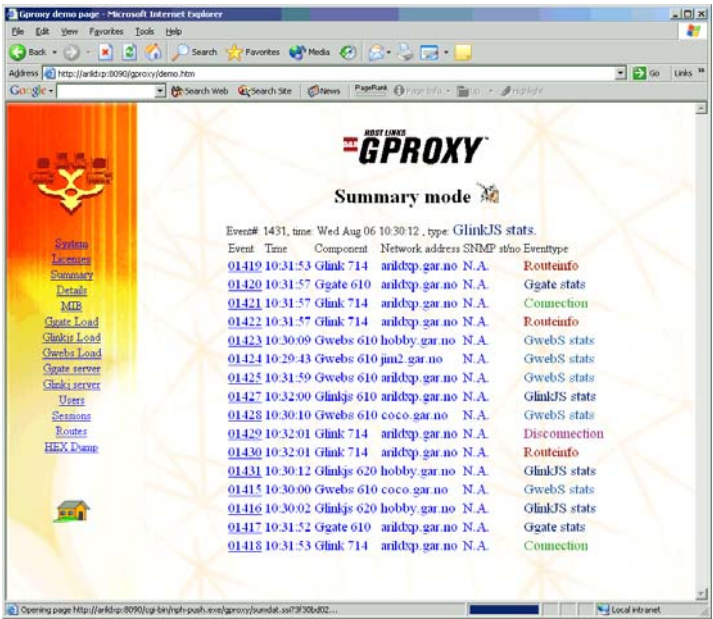
**Event information**  
 License stats events: 001 Total: 245 Alarm events: 000 Total: 001  
 Gateway events: 003 Total: 389 Gwebs stat. event: 004 Total: 706  
 Connect events: 000 Total: 002 Disconnect events: 000 Total: 002  
 Connection failure events: 000 Total: 000 Route info. events: 000 Total: 004  
 GlinkJ events: 000 Total: 000 IO counts: 000 Total: 000

# The Summary report

Each broadcast network event generates a single line of output in the summary report, and the report covers the 16 most recent events. Please note that generating the summary report is resource intensive, since every event generates 16 individual 'detailed reports' that you can access from the Event link on the summary report line.

For each of the events the following fields are shown:

Event	Gproxy internal event sequence number
Time	The time Gproxy received the information
Component	The name of the reporting program
Address	The network (IP) address of the reporting program
SNMP status	Indicates if the event was reported as an SNMP trap
Event type	Event (record-) type



Click the 'event' link for a closer look. A new browser window will be opened with a detailed view of the chosen entry.

# The Detail report

In the detail report a network event generates a full screen of data. It displays all elements of the administrative record. This always includes names, addresses, time-stamp and event serial number of the reporting program, as well as the name and address of the client (i.e. in the case of an event reported by a gateway application). In the case of host connection requests, all the parameters used by the line handler (including defaults) are shown. In the case of a failing host connection, any error text returned by the host or generated by the line handler or the underlying communication stack is also shown.

The following is an overview of the fields reported:

**Statistics records:**

Event no	The serial number of the reported event
Time	The time the event was reported
SNMP status	Values: ok, failed, Ign, N.A.
Program	The name of the reporting Host Links program
Network address	The IP address of the reporting component
Serial no	The number of the Gproxy event
Uptime	Seconds since startup (of the reporting component)
Sent host	Number of bytes sent upstream
Sent client	Number of bytes sent downstream
Sessions	Total no. of sessions through this gateway
High	Highest no. of simultaneous sessions
Max	Maximum (set by license)

**Route records:**

Event no	The serial number of the reported event
Time	The time the event was reported
SNMP status	Values: ok, failed , Ign., N.A.
Program	The name of the reporting Host Links program

**Connects, disconnects and error records:**

Event no	The serial number of the reported event
----------	---

Time	The time the event was reported
SNMP status	Values are: ok, failed, Ign., N.A.
Program	The name of the reporting Host Links program
Client	Name (userid) of the client
Address	Client network address (if gateway type)
Host name	The DSA (SCID) name of the host system
Coname	Name of <code>dsa.config</code> 'coname' entry if used
Network address	The IP-address of the reporting component
Serial no	The number of the Gproxy event
Uptime	Seconds since startup (of the reporting component)
Sent host	Number of bytes sent upstream
Sent client	Number of bytes sent downstream

Example Detail mode report:

The screenshot shows a web browser window titled "Gproxy - Detail mode page - Microsoft Internet Explorer". The address bar contains the URL: `http://arldxp:8090/cgi-bin/nph-push.exe/gproxy/005dat.ssi?method=nopush`. The main content area displays the Gproxy logo and the text "Detail mode". Below this, the following event details are shown:

```

Event# 1623, time: Wed Aug 06 10:35:41 , type: Connection
Prog name: Glink 714 Prog addr: arldxp.gar.no Local time: 10:35:51
Local evt: 00001 Uptime: 1 SNMP status: Disabled
Client name: arld Client addr: arldxp.gar.no Pid: 3040
Hostname: IS2C Coname: MISFLD
Params: -li DSA -ll 6144 -nmn D03040 -cs ascii -dn IS2C -da
: MISFLD -brk atpglt -ii 0 -lhm dps8 -na 193.216.112.24
: -tm DKU7107

```

The browser's status bar at the bottom indicates "Done" and "Local intranet".

# The Ggate Gateway report

In gateway mode Gproxy monitors the usage of a specific Ggate gateway. The address of the gateway can be given on the command line (using the `-gga` parameter) or can be entered interactively using the Gproxy command interface. In gateway monitor mode, key information from the most recent event reported by the monitored gateway is shown in the first few lines. Then for each user connected through the gateway follows the process ID (PID), the users IP-address and the host name to which the user is connected. In VDU presentation mode, if there are more users connected through the gateway than can be shown on a single screen, you can 'navigate' by means of the page-up and page-down keys on the keyboard.

The following information is shown:

**The headings:**

Program	The name and version of the gateway being monitored
Serial number	Serial number of the last event reported by this gateway
Uptime	The 'uptime' (in seconds) for this gateway
KB Up	The number of K bytes moved upstream (to host)
KB Down	The number of K bytes moved downstream (to client)
Sessions	The current number of host sessions
Hi	The highest number of host sessions registered
Max	Maximum number of session (allowed per license)
Users	The current number of host users
Hi	Highest number of users registered
Max	Maximum number of users (allowed per license)

**Per session:**

PID	Process ID of the line handler
Username	The network address (IP address) of the user
UID	User identifier (if supplied by the client)

## Example Gateway mode report:

**Ggate gateway mode**

Event#: 76, time: Wed Aug 06 13:19:31 , type: [Ggate stats](#).

Program: [Ggate 620](#) Serial: 00031 Uptime: 74 KB Up: 0 KB Down: 0

Sessions: 6    H: 6    Max: 0    Users: 3 H: 3 Max: 0

PID	Useraddr	Hostname	UID
4540	aridxp.gar.no	RS2C	BFC04DA5-9893-4395-9180-C71982DE32CD
6060	arid2.gar.no	RS2C	5FFF6054-348B-4696-9160-8CB56D92310A
1980	arid2.gar.no	RS2C	5FFF6054-348B-4696-9160-8CB56D92310A
6124	lion.gar.no	RS2C	591D9CBF-3CB7-44AA-9FDE-DACAEC9CC466
5864	lion.gar.no	RS2C	591D9CBF-3CB7-44AA-9FDE-DACAEC9CC466

Navigation links on the left: [System](#), [Licenses](#), [Summary](#), [Details](#), [MIB](#), [Ggate Load](#), [Glinks Load](#), [Gwebs Load](#), [Ggate server](#), [Glinkj server](#), [Users](#), [Sessions](#), [Routes](#), [HEX Dump](#)

Browser status bar: <http://aridxp:8090/cgi-bin/nph-push.exe/gproxy/gtwdst.ssi> Local intranet

## ***The Load balancing reports***

Gproxy produces a load balancing report mode for each server type (*Ggate*, *Glinkjs*, *Gwebs*). Each report shows load information for every server of that type known to Gproxy (i.e. configured to send statistical information to Gproxy). The report reflects the current state of the 'load tables' used for load balancing purposes. The 'best' (lowest load) gateway is indicated i.e. this is the gateway that will be suggested in the redirect record for the next client connection.

The following information is shown per gateway:

Gateway address	The network address of the gateway
Port	The port number
Load	Current number of sessions
Fact	The load factor (used in calculation of 'best' gateway)
Tres	Threshold value (max) set for this gateway
High	The highest number of session registered
Max	The maximum (set by license)
Updated	Time stamp last update
Remarks	Indicates 'best' gateway or 'disabled' gateway



Example load balancing report:

[System](#)  
[Licenses](#)  
[Summary](#)  
[Details](#)  
[MTB](#)  
[Ggate Load](#)  
[Glinks Load](#)  
[Gwebs Load](#)  
[Ggate server](#)  
[Glinks server](#)  
[Users](#)  
[Sessions](#)  
[Routes](#)  
[HEX Dump](#)

**GATEWAY LINKS**  
**GPROXY™**  
**Ggate load balance mode**

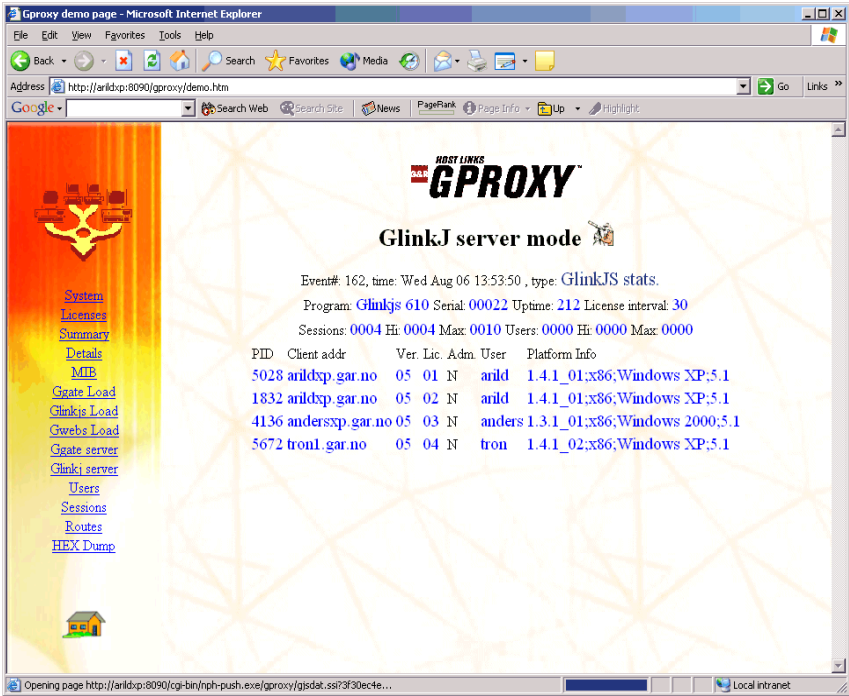
Event# 1964, time: Wed Aug 06 10:42:13, type: Ggate stats.  
 Gateway address Port Load Fact Tres High Max: KB up KB down Uptime Remarks  
 arldxp.gar.no 31111 3 0 5 0 0 9 2465  
 hobby.gar.no 30841 0 0 0 0 100 0 0 941 best

Opening page http://arldxp:8090/cgi-bin/nph-push.exe/gproxy/toddatt.ssi?3f30bf66... Local intranet

# The Glinkj server report

In GlinkJ server reporting mode Gproxy maintains a table of clients connected to a given GlinkJ server. The address of the GlinkJ server must be given in the `-glj` command line parameter at Gproxy start time. The table is updated each time Gproxy receives a 'logon record' broadcast from the GlinkJ server when a client connects. The report is only available in HTML format.

Example Glinkj server report:



# The License server report

In license reporting mode Gproxy gathers license statistics from all Gproxy systems known to it, including itself if license server activity is enabled by parameter (-lic) or default. All shared licenses for all servers are shown. The license name, current usage, the original maximum license count and the maximum site limitations are reported.

The example below shows a configuration with 2 known Gproxy systems acting as license servers.

The second one, hobby.gar.no, is no longer reporting its counters and Gproxy on arildxp.gar.no has assumed that hobby has failed, and taken over its shared licenses.

Example License server report:

**HARD LINKS GPROXY™**

**License server mode**

IP address	License name	Ref. no.	Community	Lic. used	Lic. max	org Lic.	Lic. site	Corporate	Updated
arildxp.gar.no	Ggate	0001	0001	0022	0200	0100	0200	0500	10:27:39
arildxp.gar.no	Gweb	0001	0001	0024	0100	0050	0100	0100	10:27:39
hobby.gar.no	Gweb	0256	0001	0000	0000	0050	0100	0100	10:26:00
hobby.gar.no	Ggate	0256	0001	0000	0000	0100	0200	0500	10:26:00

Navigation menu:

- [System](#)
- [Licenses](#)
- [Summary](#)
- [Details](#)
- [MIB](#)
- [Ggate Load](#)
- [Glinkjs Load](#)
- [Gwebs Load](#)
- [Ggate server](#)
- [Glinkj server](#)
- [Users](#)
- [Sessions](#)
- [Routes](#)
- [HEX Dump](#)

Status bar: Opening page http://arildxp:8090/cgi-bin/nph-push.exe/gproxy/licdat.ssi?3f30bbfd... Local intranet

## The Route report

In route mode Gproxy collects information about the state and usage of all the configured transport routes configured in the TS directives in the `dsa.cfg` configuration file. Each time a host connection is established or terminated, the line module will update the count for the TS route used and a route statistics record will be forwarded to Gproxy. In a backup route configuration, an unsuccessful connection attempt causes *Gline* to set the TS route in question in a disabled state. Such state changes are also detected and reported on the output report.

The following fields are shown:

TS route name	The route name i.e. the name configured in the TS directive in <code>dsa.cfg</code> .
NSAP	The network address as configured in the <code>-NS</code> parameter of the TS directive.
State	The route state (possible values: down, lock, disa, enbl and used)
Sessions	# of session - number of session using this route

Example Route mode report:

The screenshot shows a web browser window with the following content:

- Navigation Menu (Left):**
  - System
  - Locations
  - Summary
  - Details
  - MIB
  - Gate Load
  - Client's Load
  - Gate's Load
  - Gate server
  - Client server
  - Users
  - Sessions
  - Routes
  - HEX Dump
- Page Header:**
  - ANY LINKS
  - GPROXY**
  - Route mode
- Event Information:**
  - Event# 179, time: Wed Aug 06 13:21:22, type: **Route event**
- Table:**

Transport route name	Network address (NSAP)	Current state	Active sessions
rs2c	gars.gar.n	USED	0002

# The User report

In user report mode Gproxy maintains a table of all Host Links users and the number of sessions in the network. The definition of a ‘user’ depends on the setting of the `-uid` command line parameter and is either a unique IP-address or a unique UserID (GUID). The number of users monitored is set using `-uc N` (N defaults to 512, maximum 4096) The table is updated every time Gproxy registers a connect, disconnect or error broadcast.

The following fields are shown:

## Heading info

Users	The current number of users
HiUser	Highest number of users allowed
HiReg	Highest number of users registered
Max	Maximum number of sessions (allowed per license)
Sessions	The current number of sessions
HiSess	The highest number of sessions registered

## Per user

User IP	IP-address
Sessions	Number of active sessions this user
KB Up	The number of K bytes moved upstream (to host)
KB Down	The number of K bytes moved downstream (to client)
UserID	GUID, if supplied by the client application

Example User mode report:

**GPROXY**  
User mode

Event#: 431, time: Wed Aug 06 13:04:27, type: Disconnection  
Users: 01 HUser: 04 HReg: 04 Max: 16 Sessions: 01 HsSess: 07

User IP	Sessions	KB Up	KB down	UserID/GUID
aridxp.gar.no	0	0	1	BFC04DA5-9893-4395-9180-C71982DE32CD
arid2.gar.no	0	0	1	5FFF6054-348B-4696-9160-8CB56D92310A
andersxp.gar.no	1	0	1	D408EFD4-A172-4F4D-BBB6-4AFFC2AE321
lion.gar.no	0	0	1	591D9CBF-3CB7-44AA-9FDE-DACAEC9CC466

System  
Licenses  
Summary  
Details  
MIT  
Gate Load  
Glinks Load  
Gwebs Load  
Gate server  
Glinks server  
Users  
Sessions  
Routes  
HEX Dump

Opening page http://aridxp:8090/cgi-bin/gh\_push.exe?gproxy/uidst.s973f30e0bc... Local intranet

# The Session report

In session reporting mode Gproxy maintains a table of all Host Links sessions. The table is updated ever time Gproxy registers a connect, disconnect or error broadcast. The report is only available in HTML format.

Example Session report:





# The MIB report

In MIB reporting mode all relevant information is displayed in SNMP MIB format. Each field is prefixed with its SNMP 'object ID', 'object name' and its value.

The following fields are shown:

## Statistics:

progno	internal number of the reporting component
progrname	the name of the reporting component (Host Links program)
progaddr	the network address (normally the IP-address)
pid	the program ID
serialno	the number of the Gproxy event
sessionno	total no. of session through this gateway
max	maximum number of sessions this interval
min	minimum number of sessions this interval
event no	the serial number of the reported event
upstreams	number of bytes sent to host
downstreams	number of bytes sent to client

## Connections, disconnections and errors:

progno	internal number of the reporting component (Host Links program)
progrname	name of the reporting component
progaddr	network address (normally the IP address)
pid	program ID
serialno	number of the Gproxy event
username	applicable to GWDNTD gateways only
useraddr	network address (IP-address) of the user
session number	applicable to gateways
hostname	name of the host (normally DSA remote session ID)
retcode	return code (applicable to connect errors)

# Gproxy

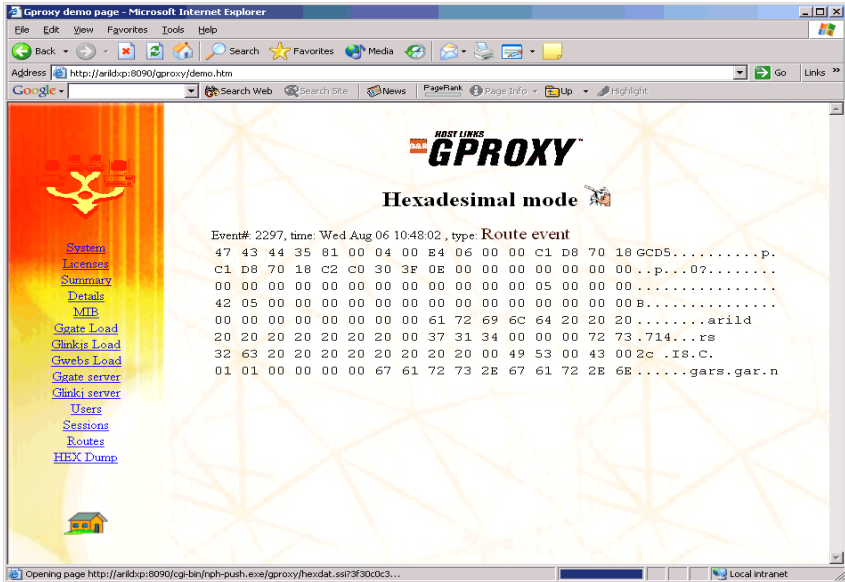
timeofday	of the event
error text	(if connect failure)
coname	correspondent name (from <code>dsa.cfg</code> )
parameters	all given and generated parameters (connects and errors)

## Example MIB report:



# The 'HEX report'

In HEX mode the UDP record is displayed 16 characters per line in hexadecimal and printable ASCII as shown in the following:



# Administration

---

## Gmanager operator commands

When operating in background server mode, commands must be given through the command interface in *Gmanager* (see the *Host Links server administration* appendix below). Select the Gproxy server (position by means of the cursor up/down keys) and click 'send server command' (F3 in the character based version of *Gmanager*). The following commands can be given:

Command	Parameter	Description
DOWN		Terminates Gproxy
STATUS		Request that status information be reported to the log file
PARAM		Brings up a dialog box or command line that allows you to feed any parameter to Gproxy. Please note that some parameters might not work when given interactively i.e. they can only be given at server start time
DEBUG	ON/OFF	Toggles on and off tracing and debugging interactively
NUMI	ON/OFF	Toggles numeric IP format on and off i.e. switches between use of numeric and symbolic (host name) IP-address format
DISABLE gga/gwb/glj	IPaddress	Disables a server for load balancing e.g. DISA gga gars.gar.no
ENABLE gga/gwb/glj	IPaddress	Enables a server for load balancing e.g. ENAB gga gars.gar.no
SILENT		Stop generation of output reports
TSUPD	route state	Set a transport route to a given state. See the gtsupd utility for parameters

# Monitoring Gproxy

When Gproxy is started with the `-sv` (server) parameter it executes in the background. None of the Gproxy VDU output screens are generated and there is no command interface. As all other Host Links server programs, Gproxy reports its current status to the *Gmanager* database. This can be viewed using the UNIX, DOS or Windows version of *Gmanager*. If the G&R system directory on a file server is shared by several Host Links platforms, *Gmanager* shows the status of all Host Links servers on all the Host Links platforms sharing the G&R system directory. *Gmanager's* active server list shows you the most recent status message from any Gproxy sharing the G&R system directory, and you can view the log files at the touch of a key or a mouse click. You can also terminate Gproxy by sending it the DOWN command from *Gmanager*.

## Gproxy logging

If the log file is enabled (`-log`, default ON), essential information from the administrative records received is written to the log file. If Gproxy is configured to run as a G&R server, the log file will be created in the G&R server directory, otherwise the log file will be located in the current directory. The following is an extract from a Gproxy log file.

```
[2003/09/11 15:00:17] G&R/Gproxy 6.1.0/m32pc Jul 19 2003 started
[2003/09/11 15:00:17] Thu Sep 11 15:00:17 2003 15ac357f:
    1 G&R A/S:G&R A/S:*:1:1 for Gproxy:*:0:0:0:0:*
[2003/09/11 15:00:17] 1x 2519MHz Intel(R) Pentium(R) 4 CPU 2.53GHz 83MB
[2003/09/11 15:00:17] Command line:
-dbg -alm 6 -lbgwb 8080 -lbgsa 30841 -pt html -rt all
[2003/09/11 15:00:17] CreateLicChild
[2003/09/11 15:00:18] BrdCst: GWBstat Prog: Gwebs 611, Addr: raider.gar.no
[2003/09/11 15:00:18]   Req. cmplt:42, running:0, sess. cmplt:3, running:0
[2003/09/11 15:00:21] BrdCst: GGAstat Prog: Ggate 610, Addr: arildxp.gar.no
[2003/09/11 15:00:21] LicDialog, granted Ggate license 1 to pid=5932
[2003/09/11 15:00:21]   Curr user: 0, max: 100, high: 0 Curr sess: 0 high:0
[2003/09/11 15:00:22] BrdCst: GWBstat Prog: Gwebs 610, Addr: andersh2.gar.no,
Serial: 59374
[2003/09/11 15:00:22]   Req. cmplt:43, running:0, sess. cmplt:6, running:11
[2003/09/11 15:00:27] BrdCst: GWBstat Prog: Gwebs 610, Addr: jim2.gar.no
[2003/09/11 15:00:27]   Req. cmplt:0, running:0, sess. cmplt:0, running:0
[2003/09/11 15:00:28] BrdCst: GWBstat Prog: Gwebs 611, Addr: raider.gar.no
[2003/09/11 15:00:28]   Req. cmplt:42, running:0, sess. cmplt:3, running:0
[2003/09/11 15:00:32] BrdCst: GWBstat Prog: Gwebs 610, Addr: andersh2.gar.no
[2003/09/11 15:00:32]   Req. cmplt:43, running:0, sess. cmplt:6, running:11
[2003/09/11 15:00:35] BrdCst: GGAstat Prog: Ggate 610, Addr: arildxp.gar.no
[2003/09/11 15:00:35]   Curr user: 0, max: 100, high: 1 Curr sess: 0 high:1
```

# Appendix: Host Links Manuals

---

Below you find a complete list of all available Host Links manuals:

<b>Installation</b>	
Host Links Servers	Installation and Configuration on UNIX/Linux
Host Links Emulators	Installation and Configuration on UNIX/Linux
Host Links	Installation and Configuration on Windows
<b>Line handling</b>	
Gline	Line Handler and DSA/OSI Configuration
Ggate	Transparent Gateway
Gproxy	Network Manager & SNMP Proxy Agent
G&R SSL	Using SSL for security in G&R products
GIAPI	Application Programming Interfaces
Gsftp	Gateway between FTP and SFTP
<b>Emulations</b>	
Gspool	Network Printer Emulation
GUFT	Unified File Transfer
G3270	Emulating IBM 3270 Terminals
G5250	Emulating IBM 5250 Terminals
Pthru	Gateway to the Bull Primary Network
Qsim	Emulating Questar DKU7107-7211 & VIP7700-7760
V78sim	Emulating VIP7801 & VIP7814
Gweb	Web Browser Front-end for DKU, VIP7700-7760, VIP7800, IBM3270 and IBM5250 Emulations



# Appendix: Host Links Server Administration

---

Gmanager is the Host Links administration tool. It can be used to control, configure and monitor all the G&R Host Links server programs.

The dialog and interaction between the server programs and Gmanager is based on information located in a database file `_active.srv` that is located in the Host Links `servers` directory. The first time a Host Links server program starts up it registers itself in this ‘active’ file. Thereafter the server program updates this database with status information whenever the server is active.

The Gmanager program is available in 2 different versions – a Windows GUI based version `gmanw.exe` and a character based subset `gman` (UNIX/Linux binary) or `gman.exe` (PC console application).

The basic functionality of the two versions is the same, but the Windows version interfaces directly to other Windows-only Host Links administrative tools (*Gconfig*, *Gservice*), and can also start the browser directly to view HTML reports produced by Gproxy, if enabled, or to view the HTML pages associated with a *Gweb* or *Glink for Java* installation.

The *Gproxy* reports, *Gweb* and *Glink for Java* web pages are of course available to administrators of UNIX/Linux Host Links systems, and can be viewed by starting a browser manually, and connecting to the appropriate URLs:

```
http://mysite.mydomain.com/Gproxy  
http://mysite.mydomain.com/Gweb  
http://mysite.mydomain.com/GlinkJ
```

Gmanager can be used to perform the most common Host Links administrative tasks. A summary of the available functions follows. The Windows-only functions are marked.



- View the last reported status information from the servers
- Stop or restart all servers, start a new server, stop, restart or delete a server
- Send a command to a server
- Load the DSA configuration into an editor, compile the DSA configuration
- Start the server configuration program or the configuration wizard (Windows)
- Load the *Gservice* configuration into an editor (Windows)
- Edit the product specific configuration files
- View a server log file, a server trace file or the server configuration file
- View program version numbers, program link information (Windows)
- View license info and license usage (Windows)
- View Host Links environment information, the 'VMAP' (Windows)
- Start Gdir directory administrator, Ggate monitor, Gspool monitor
- Gather all traces and logs for trouble-shooting by support
- Gping a DSA node to check the connection, use Gerror to explain error code
- Set a transport route state (down, enbl, lock, used)
- Check if a printer is on-line, request a list of bins
- Connect directly to the *Gproxy*, *Gweb*, *GlinkJ* HTML pages(Windows)

The commands that are accepted by all servers are:

- DOWN - terminates the server
- STATUS - reports server-specific status information to the log file
- PARAM - allows the operator to give a command line parameter to the server. Note that some parameters do not work when given interactively i.e. they can only be handled at server startup time
- DEBUG ON/OFF - toggles on and off tracing interactively

Additionally, the server in question might support other interactive commands. For a description of the supported commands, check the server-specific documentation.

# Appendix: Host Links DSA Utilities

---

The Gline package includes a set of Gline communication utilities. These are used when testing and debugging connection problems. The utilities are delivered as part of the Gline package and can be used without any additional configuration. The nodes to be tested must of course be configured in the `dsa.cfg` file.

## Gconame

Lists the parameters generated from a given CONAME. The utility works for both CONAME and RESOURCE e.g.:

```
gconame tnviptm

Checking 'dsa.cfg' for coname 'tnviptm'
Coname: tnviptm, type TM, parameters:
-DA misfld
-S_
-D_
-CODE 0000
-CODE 1000
-CODE 1800
-TEXT Remote SCID?:
-CODE 4700
-TEXT Remote application?:
-CODE 1400
-CODE 1600
-TEXT Password?:
```

## Gerror

Shows the text message associated with a DSA reason code. Only the most common codes are supported i.e. the ones related to network, transport and session communication layers. Errors generated by the OSI-stack on the Host Links platform are not covered by this utility; please refer to the documentation from the vendor of the stack e.g.:

```
gerror 0109
Reporting component: Session control (01) 0109, Dialog
protocol error or negotiation failed (wrong logical record).
```

For a detailed description of all reason codes, please consult the Bull manual *OSI/DSA Network System Messages and Return codes* (39A2 26DM).

## Glnode

List and verify the communications parameters of the local node e.g.:

```
glnode
Local node name : GRDL
Local session control id : GRDL
DSA200 address (area:tsm): 54:60 (36:3C)
```

## Gmacfix

When you connect to FCP cards on Bull mainframes via an Ethernet port on the LAN-Extender the mainframe address is given in Ethernet (LLC) format. If you connect to an FDDI adapter you must convert the MAC address to SMT. e.g.:

```
gmacfix 080038000fab
MAC address 080038000fab = 10001c00f0d5
```

## Gping

Connects to a remote system using the Gline parameters set on the command line. If successful it returns 'connected to application', otherwise it shows the error code returned e.g.:

```
gping -li dsa -dn b7dl -da iof -du jim -pw mydogsname
Gping - $$DSA: Connected to application
```

## Gproxy

### Grnode

Return the parameters (in `dsa.cfg`) and the state of a remote node e.g.:

```
grnode b6dl
Checking 'dsa.cfg' for node 'b6dl'
Session control id : B6DL
DSA200 address (area:tsm) : 1:5 (1:5)
Inactivity interval : 0
Route 0
Load balance percentage : 0
TP class : 2
TP expedited : 0
TPDU size : 0
Network address : 130405
```

### Gtrace

Same as `gping` but writes the DSA communication trace on the user's terminal (applicable to UNIX versions) e.g.:

```
gtrace -li dsa -dn ln40 -da snml51
D6:Application event @ 14:17:17.6003. tokenitem = 00
D6:Application event @ 14:17:17.6082. tokenitem = 00
D6:Connect request called, node = LN40
D6:OurBufferSizes. ApplMaxXmit = 511, ApplMaxRecv = 500
Rec:4000 0002 s:2
Rec:506B 0010 s:16
etc etc
Gtrace - line trace ending.
Gtrace - $$DSA: Connected to application.
```

### Gtsupd

Update the state of a transport route. Transport routes can be set automatically in a disabled state if a backup route is configured. When such a state change occurs the route will be set back to the enabled state after a configurable timer has expired. The default is 15 minutes. You can reset the state of such a route with `gtsupd ts-name enbl/used/down/locked` e.g.:

```
gtsupd gars_rfc enbl
TS-entry 'gars_rfc', new state = enbl
```



# Appendix: Host Links Trace

---

If you experience any kind of problem when using a Host Links application, the application trace file and/or the line handler trace file will provide useful documentation of the problem.

## Trace activation

The Host Links products automatically create sub-directories in the debug directory when debug is activated: at product level using the `-dbg` parameter, or at line level using the `-d_` or `-s_` parameters to the line module.

<b>Windows server</b>	<code>gspool -id gsl -dbg -ps \\SERVER\LEXMARK -li dsa -da tptst -d_ on</code>
<b>UNIX Linux</b>	<code>gspool -id gsl -dbg-pc lp -li dsa -da tptst -d_ on</code>

Most G&R products include a facility for setting product or line parameters dynamically. It is therefore generally possible to turn on debug or trace without modifying the command line or configuration of a production system.

## Trace types

All Host Links products accept a parameter `-dbg`, which starts an application level trace of internal events. This is useful when investigating malfunctions or looking closely at product behaviour.

All Gline line handlers accept a parameter `-d_` to turn on a data trace. It records data and enclosure level being exchanged with the line handler. It is useful when documenting product malfunction e.g. an emulation error, because it records exactly what the host sends and what the G&R application replies. It can be used to simulate a customer situation, reproduce a problem and to verify that a correction fixes the documented problem.

All Gline line handlers accept a parameter `-s_` to turn on a session trace. It records the raw data being exchanged between the line module and the underlying transport layer (e.g. OSI Transport, or TCP socket), as well as internal events and protocol states. It is useful when investigating protocol failures such as unsuccessful connect attempts or abnormal disconnections.

## Structure

The Host Links file structure includes a debug directory to collect the trace and debug files in one location where the permissions can be adjusted as required for security. By default only the Host Links administrator can access the directory. The debug directory is created by the initialization procedure and located (by default) in:

<b>Windows server</b>	<code>\gar\debug</code>
<b>UNIX Linux</b>	<code>/usr/gar/debug</code>

If the application is a client type of application, a debug sub-directory with the same name as the user (UNIX username or PC login name) is created and all debug files are located there. This includes the line level trace except in the special case where the client application connects via Ggate and the line level trace is written on the Ggate system using the Ggate DSA node name as a debug sub-directory.

If the application is a server type of application, then a sub-directory will be created using the DSA node name on behalf of which the server application is executing. If the server does not use DSA the default local session control name is still used if there is a `dsa.cfg` file. If there is no `dsa.cfg` file then the system's UNIX or Windows communications node name is used. You can find this name using the command `uname -n` on UNIX systems, or the Network section of the control panel on Windows systems. This covers situations where several instances of a server are executing on the same system and accepting incoming calls to different DSA node names, or where several Host Links systems using the same server product share a file system.

# Tracing Ggate

When Glink, a Host Links client or a customer application based on GI-API connects through Ggate to the application, the line handler trace is generated on the Ggate system, with the name and location shown in the table:

<b>Windows server</b>	<code>\gar\debug\NODE\ggaNN-PPPP.dbg</code>
<b>UNIX Linux</b>	<code>/usr/gar/debug/NODE/ggaNN-PPPP.dbg</code>

NODE is the local DSA node name used by the Ggate system.

The trace file name consists of the prefix `ggaNN-` followed by the IP-address of the client, suffixed by `.dbg` for a terminal session or `-dbg` for a printer session. The following is a trace file name for Ggate session sequence number 5 executing on Host Links system GRDL initiated from a Glink client on IP-address `jim.gar.no`:

```
gga05-jim.gar.no.dbg
```

This file, and possibly also a Glink debug file and a Glink communication trace file activated by the `/J` command line parameter will be needed by the support engineer investigating any problem.

To enable a line handler trace through Ggate the product's start-up command or configuration file would look like this:

```
-LI YYY:ZZZZ -S_ -D_
```

(YYY =line handler identification, i.e. DSA or DIWS)

(ZZZZ =IP-address of the system running Ggate)

## Examples - G&R products

Examples of directory and file names in the debug structure are:

<code>/usr/gar/debug/jim</code>	<b>Debug directory for user 'jim'</b>	
<code>qsm.dbg</code>	Qsim emulator debug file	<code>-dbg</code>



qsm-gli.dbg	Qsim host line trace	-li dsa -s_
pth-glit.dbg	Pthru terminal line trace	-term -s_
pth-glih.dbg	Pthru -host line trace	-li dsa -s_
g32.dbg	G3270 emulator debug file	-dbg
g32-gli.dbg	G3270 host line trace	-s_
<b>/usr/gar/debug/mike</b>	<b>Debug directory for user 'mike'</b>	
v78.dbg	V78sim emulator debug file	-dbg
v78-gli.dbg	V78sim host line trace	-li dsa -s_
guf.dbg	GUFT client debug file	-dbg
guf-gli.dbg	GUFT client host line trace	-li dsa -s_
<b>/usr/gar/debug/en01</b>	<b>Debug directory for node 'en01'</b>	
guf.def	GUFT server debug file	-dbg
guf-gli.def	GUFT server host line trace	-li dsa -s_
gli-gli.dsa	DSA listener host line trace	-s_
gli-gli.diw	DIWS listener host line trace	-s_
gsp.def	Gspool (default -id) debug file	-dbg
gsp-gli.def	Gspool (default -id) host trace	-li dsa -s_
gga01-mike.gar.no.dbg	Ggate line trace, first Glink	-s_
gga02-mike.gar.no.dbg	Ggate line trace second Glink	-s_
<b>/usr/gar/debug/en02</b>	<b>Debug directory for node 'en02'</b>	
gsp.abc	Gspool (-id abc) debug file	-dbg
gsp-gli.abc	Gspool (-id abc) host trace	-li dsa -s_
gspc-gli.def	Gspool DPF8 command trace	-li tcp -s_
gspd-gli.def	Gspool DPS8 data trace	-li tcp -s_

## Gproxy

gsp._00	Gspool started on demand debug	-dbg
gsp-gli._00	Gspool started on demand trace	-li dsa -s_

## ***CPI-C and Gweb trace files***

Gweb uses the CPI-C libraries so the Gweb debug structure is exactly the same as for CPI-C, except that Gweb inserts its own product identifier into the file name structure. CPI-C applications use the 'client' style of debug and create a debug directory with the UNIX username or PC login name used by the process that started them.

The application level debug (-dbg) and line trace (-s\_ and -d\_) are set in the `cpic.cfg` file. The line trace goes to the debug directory, with the name built up as follows:

```
<product_id><session_id>-<process_id>.<debug_type>
```

<b>product_id</b>	<i>Value</i>	<i>Comment</i>
	cpi	CPI-C API
	cp3	CPI-C 3270
	cp7	CPI-C 7800
	cpd	CPI-C DKU
	gw3	Gweb3270
	gw7	Gweb7800
	gwd	Gwebdku
<b>session_id</b>	(nn)	If multi-session application, 1-63
<b>process_id</b>	n (n n n...)	Varies by platform
<b>debug_type</b>	dgb	Application level debug
	gli	Line trace

Example:

\gar\debug\system		debug directory for user "system"
cpi-16.dbg	CPI-C single session debug	-dbg
cpi-16.gli	CPI-C single session line trace	-li dsa -s_
cpi2-123.dbg	CPI-C session 2 application debug	-dbg
gw7-20172.gli	Gweb7800 host line trace	-li dsa -s_

# Appendix: Error Codes

---

## OSI/DSA error codes

Below is a list of OSI/DSA error codes and the corresponding description. These are the same descriptions that the G&R/Geterror utility will display when given the DSA code as a parameter.

code	Description
<b>00xx</b>	<b>General Errors</b>
0001	Open Failure in LC - Reject for unknown reason
0002	Open Failure in LC - Acceptor customer node inoperable
0003	Open Failure in LC - Acceptor customer node saturated.
0004	Open Failure in LC - Acceptor mailbox unknown.
0005	Open Failure in LC - Acceptor mailbox inoperable.
0006	Open Failure in LC - Acceptor mailbox saturated.
0007	Open Failure in LC - Acceptor application program saturated
0008	Connection refused. Transport protocol error or negotiation failed.
0009	Open Failure in LC - Dialog protocol error or negotiation failed
000A	Open Failure in LC - Presentation protocol error or negotiation failed
000B	Open Failure in LC / Connection refused lack of system resources
000C	Open Failure in LC / Connection refused from GCOS7 duplicate user
000D	Open Failure in LC, Duplicate implicit LID / Q class not started
000E	Open Failure in LC, Duplicate GRTS Id / lack of memory resources
000F	Open Failure in LC, No Logical line declared for DACQ / 7 connection refused
0010	Open Failure in LC, GCOS 8 GW Missing translation / Incorrect device length in ILCRL.
0011	Open Failure in LC, DAC connection not initialized / Too many jobs executing
0012	Open Failure in LC, No binary transfer / impossible to start the IOF job

0013	Open Failure in LC, connection is not negotiated in FD mode / impossible to start the IOF job
0014	Disconnection - Timeout resulting from absence of traffic.
0016	Option missing for an RBF mailbox.
0017	Connection refused - Incorrect access right for MB.
0018	Connection refused - Incorrect access rights for the application.
0019	Connection refused - Unknown pre-negotiated message path
001A	Connection refused - Security validation failed.
001B	Connection refused - Unknown acceptor mailbox extension.
001C	Connection refused - Inoperable acceptor mailbox extension.
001D	Connection refused - Invalid Message group number.
001F	Disconnection - no more memory space.
0020	Connection refused - Unknown node.
0021	Connection refused - inaccessible node or Host down.
0022	Connection refused - saturated site.
0023	Connection refused - inoperable mailbox.
0024	(X.25) Packet too long. Problem with packet size. / Connection block already used.
0030	Syntax Error - option not known (received on close VC).
0031	(X.25) No response to call request packet - timer expired.
0033	(X.25) Timer expired for reset or clear indication.
0039	Disconnection - transport protocol error (MUX).
003C	Presentation Control Protocol Error
003E	The application has not the turn
003F	Message group closed
0040	(X.25) Facility code not allowed. / Connection refused - unknown node
0041	Connection refused - path not available.
0042	Connection refused - Duplicate USER ID / Facility parameter not allowed
0044	(X.25) Invalid calling address.
0045	(X.25) Invalid facility length.
0047	(X.25) No logical channel available.
004F	DNDC: (X.25) Invalid call packet length.
0050	Normal disconnection (GCOS3/8)
0051	Error or Event on LC initiated by GW
0052	Error or Event on LC initiated by GW.
0053	Error or Event on LC initiated by GW. TCall
0054	Error or Event on LC initiated by GW. DIA in LOCK State
0055	Error or Event on LC initiated by GW. DIA error

## Gproxy

0056	Error or Event on LC initiated by GW. GW has no known explanation.
0057	Error or Event on LC initiated by GW. Reject mailbox permanent
0058	Error or Event on LC initiated by GW. No more input lines in DACQ
0059	Time-out on GCOS 3/8 gateway.
005A	Error or Event on LC initiated by GW. Disconnect from terminal without reason
005B	Error or Event on LC initiated by GW. Wrong letter or wrong record
005C	Error or Event on LC initiated by GW. Forbidden letter received
005D	Error or Event on LC initiated by GW. Forbidden letter received
005E	Error or Event on LC initiated by GW. No buffer for secondary letter
005F	Error or Event on LC initiated by GW. No buffer for fragmented letter
0060	Error or Event on LC initiated by GW. Disconnect on end of phase record
0061	Error or event on LC initiated by GW. No buffer for control letter.
0062	Error or event on LC initiated by GW. Mailbox in closing phase
0064	Error or event on LC initiated by GW. Flow control error.
0065	Error or event on LC initiated by GW. CH locked by operator.
0066	Error or event on LC initiated by GW. Disconnect with a normal TMG F2 exchange.
0067	Error or event on LC initiated by GW. Teletel rerouting error from DACQ
0068	Error or event on LC initiated by GW. Teletel routing error from DACQ
0069	Error or event on LC initiated by GW. Teletel rerouting error from TM
006A	Error or event on LC initiated by GW. Teletel rerouting error from TM
006B	Syntax error - text too long.
006C	Syntax error - illegal object in a GA command.
006D	Syntax error - unknown node Id.
0078	Syntax error - illegal command for this object.
0079	Syntax error - illegal date.
007F	(X.25) No route available for X.25 switching.
0081	No more network routes available for switching.
0082	(X.25) Hop count reached for X.25 switching.
0083	(X.25) Flow control negotiation error.
0085	(X.25) Frame level disconnection.
0086	(X.25) Frame level connection.

0087	(X.25) Frame level reset.
0090	Frame level not set.
0092	(X.25) X.25 Echo service in use.
0093	(X.25) Incorrect password for PAD connection.
0094	(X.25) No more PAD connections allowed.
0096	(X.25) TS SX25 or NU X25 objects locked.
009C	(X.25) Invalid packet header. X.25 protocol error.
009D	(X.25) Incompatible header. X.25 protocol error.
009E	(X.25) Logical Channel Number too high.
009F	(X.25) Incorrect packet type.
00B2	Use of invalid password through PAD
00B6	Unknown mailbox selection for PAD connection using the PAD password.
00C0	(X.25) Normal disconnection.
00D7	(X.25) TS image (of type DSA or DIWS) in LOCK state.
00DE	(X.25) NS RMT or NR SW in LOCK state.
00E1	Connection refused. Mailbox is not in ENBL state.
00E6	QOS not available permanently.
<b>01xx</b>	<b>Session Control</b>
0100	Logical connection accepted or normal termination
0101	Rejection for unknown reason or abnormal termination
0102	Acceptor node inoperable.
0103	Acceptor node saturated. When a node has no available resources
0104	Acceptor mailbox unknown.
0105	Acceptor mailbox inoperable.
0106	DNS: Acceptor mailbox saturated.
0107	DNS: Acceptor application program saturated.
0108	Transport protocol error or negotiation failed (DSA 200 only).
0109	Dialog protocol error or negotiation failed. (Wrong logical record).
010A	Time-out on session initiation / unknown LID
010B	Acceptor mailbox extension unknown.
010C	Acceptor mailbox extension inoperable.
010D	Invalid Session Number.
010E	Unknown node.
010F	System error. System generation error or insufficient memory space
0110	Application abnormal termination. Subsequent to an abnormal occurrence in the dialogue
0111	Normal terminate rejected.
0112	Protocol not supported.
0113	Session control service purged by user.
0115	Disconnection Time-out on message group initiation.

# Gproxy

0117	Incorrect Access Right for MB
0118	Incorrect Access Right for the Application
0119	Pre-negotiated Message Path Descriptor unknown
011A	Security validation failed
011E	Incorrect object status
011F	Not enough memory space available.
0120	Node unknown.
0121	The channel object (CH) is in LOCK state
0122	Saturation - no plug available
0123	Object status = LOCK
0124	Connection block (TSCNX) already used
0125	Disconnection already running
0126	The connection block (TSCNX) is disconnected (or not connected)
0127	Change Credit value < 0
0128	Ineffective Change Credit ( delta = 0 )
0129	No more deferred letters
012B	"Reinitialization" Request
012C	"Reinitialization" in progress
012D	"Reinitialization" in progress, letters are dropped
012E	Close virtual circuit. Either no mapping exists between PA/NR or CL and VC/NS
012F	Null connection object index.
0130	Undefined function at Sysgen time.
0131	Letter too large with respect to the negotiated size.
0132	The received letter is longer than the size which was
0133	Disconnection of the session control user
0134	Interface error on EOR (End-Of-Record) processing.
013C	Presentation control protocol error.
013E	You do not have the turn.
013F	Message group closed.
0140	Session is closed.
0151	Request refused, no system buffers available.
0152	Incorrect addressing record.
0153	No presentation record in the ILCAL or ILCRL
0154	Negotiation failed on session mode
0156	Negotiation failed on resynchronization.
0157	Negotiation failed on END to END ACK
0158	No presentation record in the connection letter
0159	Negotiation failed on session mode
015A	Negotiation failed on letter size (in the Logical Connection record).



015B	Negotiation failed on resynchronization (in the Logical Connection record).
015C	Negotiation failed on end-to-end ACK (Logical Connection record).
015D	No support of the "letter" interface because Multirecord is not negotiated.
0160	Incorrect TSPACNX table.
0161	Protocol error on letter reception.
0162	Negotiation failure.
0163	Record header length error.
0164	Protocol error.
0165	Protocol error reception of control letter.
0166	Type or length error on interrupt letter.
0167	Protocol error on reception of data letter.
0168	Dialog protocol error.
0169	Unknown event.
016A	Protocol error on data transfer.
016B	Invalid status for a disconnection request.
016C	Invalid status for a recover
016D	Invalid status for a suspend/resume request.
016E	Negotiation failure.
016F	Unknown command.
0170	Error in presentation protocol
0171	Letter header length error in
0172	ILCAL is not DSA 200 protocol.
0173	Error in session record.
0174	Normal disconnection, without complementary reason code.
0175	Letter is not in ASCII or EBCD.
0176	Connection protocol letter header
0177	Letter header protocol error.
0178	Record header protocol error.
0179	Record header length error.
017A	Mbx record header length error.
017B	Error on buffer transfer.
017C	DSA 200 record header protocol
017D	DSA 300 record header protocol
017E	Unsupported connection options.
017F	Character error in ASCII string.
0180	No segmented record size.
0181	Invalid mailbox object index.
0182	Mapping error for a remote connection.
0190	No more buffers.

# Gproxy

0191	Byte count is greater than GP.
0192	Byte count is greater than GP.
0193	Byte count is greater than GP.
0194	Byte count is greater than GP.
0195	Byte count is greater than GP.
0196	Byte count is greater than GP.
0197	Byte count is greater than GP.
0198	No more buffers.
0199	Byte count is greater than GP.
019A	Byte count is greater than GP.
019B	Byte count is greater than GP.
019C	Byte count is greater than GP.
019D	Byte count is greater than GP.
019E	Byte count is greater than GP.
019F	Byte count is greater than GP.
01A0	Invalid transfer state.
01A1	Suspend protocol running.
01A2	Suspend protocol running.
01A3	Recover protocol running.
01A4	Forbidden function in write request. (\$WRITE)
01A5	Conflicting parameters for segmented record. (SWBREC)
01A6	Protocol conflict - suspend/recover.
01A7	Protocol not supported - letter/end-to-end ACK. (SWBLET)
01A8	Multi-record letter in progress.
01A9	Interrupt request forbidden.
01AA	Send control record request forbidden. (SCTROL)
01AB	Forbidden for TWA session - turn is here. (SREAD)
01AC	Termination forbidden - suspend or recover in progress. (STERM)
01C0	No space available for downstream connection request. (SMECNX)
01C1	No space available for upstream connection request. (SMUCNX)
01C2	No space available for upstream SCF connection. (SMRCNX)
01C3	No space available for session context. (\$SCTX)
01E0	Enclosure or data length error for a write request. (\$WRITE)
01E1	Enclosure or data length error for a write segment record request. (SWBREC)
01E2	Enclosure error for 'give turn' request. (SGVTRN)
01E3	Interrupt request is not demand turn, attention/data attention, or purge record.
01E4	Input status for a send control letter is not permitted.
01E8	Write request without turn.
01E9	Write segmented record request without turn.

01EA	Write segmented letter request without turn.
01EB	Send control letter request without turn.
01EC	Disconnection request without turn.
<b>02xx</b>	<b>Presentation Control</b>
0201	Protocol level not supported
0202	Application designation protocol error.
0203	Character encoding error. TM cannot support the proposed encoding.
0204	Character set error. TM cannot support the proposed character set.
0205	Character subset error. TM cannot support the proposed character subset.
0206	Incorrect record encoding.
0207	Incorrect parameter encoding.
0230	Data presentation control error. The presentation control proposed for this session cannot be used
0231	Device type is incompatible with the configuration.
0232	TM control protocol is incorrect.
0233	Device-sharing attributes are invalid.
0234	Initiator or acceptor configuration is not correct.
0235	Logical device index error.
0236	Number of logical devices is incompatible with the configuration.
0237	TM protocol record not supported.
<b>03xx</b>	<b>Terminal Management</b>
0300	Sysgen error WARNING. There is no mapped object; some objects will be spare.
0301	Operator requested session abort or logged.
0302	Idle time run out after secondary network failure.
0303	Idle time run out for no traffic.
0304	Form not found.
0305	Operator requested suspension.
0306	Destructive attention send on the session.
0307	Unknown TX addressed in this session. TM is unable to a the session.
030A	Protocol error. A record was received which did not comply with current standards
0310	Insufficient resources. The receiver cannot act on the request because of a temporary
031E	Incorrect value for Retry or Wait parameters on UP LL command.
0320	Function not supported.
0321	Parameter error. This can result
0322	Resource not available. The

## Gproxy

0323	Intervention required (on principal device).
0324	Request not executable.
0325	EOI required.
0326	Presentation space altered, request executed.
0327	Presentation space altered, request not executed.
0328	Presentation space integrity lost.
0329	Device busy. The device is busy and cannot execute the request.
032A	Device disconnected.
032B	Resource not configured.
032C	Symbol set not loaded.
032D	Read partition state error.
032E	Page overflow.
0330	Subsidiary device temporarily not available.
0331	Intervention required at subsidiary device.
0332	Request not executable because of subsidiary device.
0340	TM cannot accept a new connection.
0341	Object status incorrect.
0342	The TM configuration is not correct.
0343	Unknown TX addressed on this session.
0344	Data presentation protocol error.
0345	Device type is incompatible with the configuration, or is not supported.
0346	TM control protocol incorrect.
0347	Device shareability attributes are invalid.
0348	Initiator or acceptor configuration is not correct.
0349	Logical device index error.
034A	Number of logical devices incompatible with the configuration.
0350	Disconnection of TM after reinitialization of the network.
0360	File not found. (Welcome and Broadcast Messages)
0361	Site not found. (Welcome and Broadcast Messages)
0362	NASF error. (Welcome and Broadcast Messages)
0370	No-session timeout. Device disconnected.
0371	No-input timeout. Device disconnected.
0372	No-output timeout. Device disconnected.
0373	Timeout due to no backup session being initiated.
0374	Timeout due to no backup session being established.
0375	Connection refused because of late activation of back up session.
0376	Disconnection of current session to switch to backup session.
0380	AUTOCN parameter not declared.
0381	Mixed ETB in data sent by VIP screen and cassette
0382	Data header sent by the terminal incorrect.

0383	Desynchronization in the exchange of data.
0384	KDS block count error.
038C	Remote terminal is not connected
0390	Unknown mailbox.
0391	No call packet to return.
0392	No "Possibility" command to return Protocol error
03C0	Slave device disconnection.
<b>17xx</b>	<b>Network Layer</b>
1701	PAD connection refused.
1702	Flow control error.
1706	Logical channel number not zero in restart packet.
1707	Illegal packet length or use of D-bit forbidden.
1708	Illegal header.
1709	Illegal Logical Channel Number.
1710	Invalid packet type for the automaton state. Protocol error
1711	Incorrect packet type.
1712	Inconsistent network parameters in the generation file.
1713	No more space.
1714	DSAC network layer object not usable.
1717	USED/ENBL transition. Transport station is locked.
1718	USED/ENBL transition. This is a back-up NR.
1719	USED/ENBL transition. Dynamic close due to load.
171A	USED/ENBL transition. Transfer time-out has elapsed.
171B	USED/ENBL transition. This is a back-up NR.
171C	USED/ENBL transition. Transport station is idle.
171E	USED/ENBL transition. NR object is locked.
171F	ENBL/LOCK transition. NR HDLC has no more memory space.
1721	Remote station is inaccessible via the configured network. Check
1723	Incorrect PAD password.
1724	Virtual circuit already in use. LCN (Logical Channel Number) too high.
1725	Invalid virtual circuit.
1726	Packet too short. Protocol error for the equipment directly connected to the Bull Datatnet.
1727	Incompatibility between the generation parameters of two communicating systems on window or packet size.
1729	Packet size in communicating systems not the same.
1731	Timer runs out while waiting for call confirmation.
1732	Timer runs out while waiting for clear confirmation.
1733	Timer has run out while waiting a reset confirm.
1740	Call setup or call clearing problem.

## Gproxy

1741	Open failure on virtual circuit. No flow control on this NS.
1742	Incorrect facility. Protocol error for the equipment directly connected to the Bull Datanet.
1744	Unknown subscriber.
1745	End of time-out on reset confirm. Invalid facility length. Protocol error for the equipment directly
1747	No logical channel available.
1749	End of time-out on call confirm.
174F	Incorrect packet length. Protocol error for the equipment directly connected to the Bull Datanet.
1755	Flow control, window, packet size or reset error.
1760	Frame disconnection.
1770	Frame connection.
1771	Frame reset.
1781	No more network routes available for X.25 switching.
1782	Maximum of 15 switches have been used,
1783	Flow control negotiation error.
1785	Frame level disconnection.
1786	Frame level connection.
1787	Frame level reset.
1790	Frame level not established.
1791	No more logical paths available for the PAD.
1792	Echo service busy.
1793	Incorrect PAD password.
1794	All the PAD virtual circuits are used
1795	X.25 initialization not possible.
179B	LCN not null in restart packet
179D	Incompatible header (receive error: all VC of concerned NS
179E	LCN greater than NBVC in NS directive
179F	Incorrect packet type
17A0	Invalid facility.
17B0	Normal disconnection.
17B1	X.25 Echo in use.
17B2	No more logical channels available.
17B3	No more PAD connections allowed.
17B4	TS SX25 or NU X25 object locked.
17B5	Buffer capacity overflow.
17B6	Normal disconnection.
17B8	Unknown calling SNPA (Sub-Network Point of Attachment).
17B9	Internet problem.
17CB	Call collision on VC

17CC	Incompatible generations (NR object without mapping).
17CE	Invalid status NR locked.
17CF	Lack of space.
17D0	Unknown subscriber.
17D4	TSCNX already used for another connection. SCF internal error.
17D7	Transport station locked.
17DD	Proper NS locked.
17DE	Invalid status NR locked.
17DF	Lack of space.
17E0	Forbidden parameter or invalid value.
17E1	Invalid transition.
17E2	Upward-mapped object (TS) not locked.
17E3	No object mapped above.
17E4	NR not locked (MP NR -ADD/-SUB) or virtual circuit already open.
17E5	NR is last in list and the TS is not locked.
17E6	No object mapped above (UP NR -PRIO). NR not mapped on TS.
17E7	Upward mapped object not locked
17E9	Mix of datagram and connection network
17EB	Class inconsistent with NR.
17EE	Incompatible generations. NR object without mapping.
17FF	Wrong parameter in administrative CALL
<b>18xx</b>	<b>Transport Layer</b>
1800	Normal disconnection initiated by the correspondent
1801	Local saturation at connection request time.
1802	Failed negotiation at connection time.
1803	Duplicate connection. Two or more requests have been issued for the same connection.
1804	Redundant request.
1805	Retransmission Time-out at transport level.
1806	Survey time-out at transport level.
1807	Transport protocol error.
1808	Session Control specified is not available (inaccessible).
1809	Requested Session Control Id unknown by remote transport.
180A	Termination because of disconnection by administration.
180B	Session Control/Transport interface error.
180C	Connection request on non-sharable VC in case of ISO Transport. ISO: header or parameter length is invalid.
1817	Station in shut-down state.
181F	No memory space at connection time.
1821	Session Control inaccessible by configured session routes. ISO: Session entity not attached to TSAP.

1824	Collision between Close NC and Open TC.
182E	Remote station not configured.
182F	Resource saturation.
1831	ISO: No route for the called NSAP.
1832	ISO: Received NSAP addresses are wrong.
1833	Segmentation violation.
1834	ISO:QOS priority not available temporarily, due to a local condition (for example, lack of resources).
1835	ISO:QOS priority permanently unavailable locally (for example, due to an error in the system generation).
183A	ISO: Remote reason not specified.
183C	ISO: Remote transport entity congestion at connect request time.
1840	Server in terminating state. TC has been re-assigned on another NC.
18A1	An additional NC has been assigned to a TC.
18B0	NC has been re-assigned on another VC.
18EF	Disconnection at Transport level caused by reception of RESTART DSA during the transfer phase.

## **Windows Sockets error Codes**

Below is a list of Windows Sockets return codes and the corresponding description.

<b>Hex code</b>	<b>Windows Sockets Access Error name</b>	<b>Description</b>
2714	WSAEINTR	The (blocking) call was cancelled via WSACancelBlockingCall()
2719	WSAEBADF	The socket descriptor is not valid.
271E	WSAEFAULT	An invalid argument was supplied to the Windows Sockets API.
2726	WSAEINVAL	An invalid call was made to the Windows Sockets API.
2728	WSAEMFILE	No more file descriptors are available.
2733	WSAEWOULDBLOCK	The socket is marked as non-blocking and no connections are present to be accepted.
2734	WSAEINPROGRESS	A blocking Windows Sockets call is in progress.



2735	WSAEALREADY	The asynchronous routine being cancelled has already completed.
2736	WSAENOTSOCK	The descriptor is not a socket.
2737	WSAEDESTADDRREQ	A destination address is required.
2738	WSAEMSGSIZE	The datagram was too large to fit into the specified buffer and was truncated.
2739	WSAEPROTOTYPE	The specified protocol is the wrong type for this socket.
273A	WSAENOPROTOOPT	The option is unknown or unsupported.
273B	WSAEPROTONOSUPPORT	The specified protocol is not supported.
273C	WSAE SOCKTNOSUPPORT	The specified socket type is not supported in this address family.
273D	WSAEOPNOTSUPP	The referenced socket is not a type that supports connection-oriented service.
273E	WSAEPFNOSUPPORT	
273F	WSAEAFNOSUPPORT	The specified address family is not supported by this protocol.
2740	WSAEADDRINUSE	The specified address is already in use.
2741	WSAEADDRNOTAVAIL	The specified address is not available from the local machine.
2742	WSAENETDOWN	The Windows Sockets implementation has detected that the network subsystem has failed.
2743	WSAENETUNREACH	The network address can't be reached from this host. There is probably a problem in the way you have set up TCP/IP routing for your PC (most likely you have not defined a default router).
2744	WSAENETRESET	The connection must be reset because the Windows Sockets implementation dropped it.
2745	WSAECONNABORTED	The connection has been closed.
2746	WSAECONNRESET	
2747	WSAENOBUFS	Not enough buffers available, or too many connections.
2748	WSAEISCONN	The socket is already connected.

## Gproxy

2749	WSAENOTCONN	The socket is not connected.
274A	WSAESHUTDOWN	The socket has been shutdown.
274B	WSAETOOMANYREFS	
274C	WSAETIMEDOUT	Attempt to connect timed out without establishing a connection.
274D	WSAECONNREFUSED	The attempt to connect was forcefully rejected. The service on the other side is not available.
274E	WSAELOOP	Too many symbolic links were encountered in translating the path name.
274F	WSAENAMETOOLONG	
2750	WSAEHOSTDOWN	The host machine is out of service.
2751	WSAEHOSTUNREACH	The host machine is unreachable.
2752	WSAENOTEMPTY	
2753	WSAEPROCLIM	
2754	WSAEUSERS	
2755	WSAEDQUOT	
2756	WSAESTALE	
2757	WSAEREMOTE	
276B	WSASYSNOTREADY	Indicates that the underlying network subsystem is not ready for network communication.
276C	WSAVERNOTSUPPORTED	The version of Windows Sockets API support requested is not provided by this particular Windows Sockets implementation.
276D	WSANOTINITIALISED	A successful WSStartup() must occur before using this API.
2AF9	WSAHOST_NOT_FOUND	Authoritative answer host not found.
2AFA	WSATRY_AGAIN	Non-authoritative answer host not found, or SERVERFAIL.
2AFB	WSANO_RECOVERY	Non-recoverable errors, FORMERR, REFUSED, NOTIMP.
2AFC	WSANO_DATA	Valid name, no data record of requested type.

# Appendix: G&R MIB extension

---

```

GAR-MIB DEFINITIONS ::= BEGIN

--
• GAR-MIB: This MIB defines the G&R manageable objects.
--

IMPORTS
enterprises, Counter
FROM RFC1155-SMI
OBJECT-TYPE
FROM RFC-1212
TRAP-TYPE
FROM RFC-1215
gar                               OBJECT IDENTIFIER ::= { enterprises 827 }
garTrapEvt                         OBJECT IDENTIFIER ::= { gar 1 }
garTrapStat                       OBJECT IDENTIFIER ::= { gar 2 }

--
• garTrap Group
--
• The trap group contains information about traps sent by the
• various G&R-programs in the network.
• Trap TE: Contains all info available for a spec event
• Trap TS: Statistics trap, IO counters etc
--
• Currently all ServerX comms programs (using GL_DIWS) and all
• DNTD gateways broadcast information which is processed by the
• G&R proxy SNMP agent (Gproxy).
• Specific trap numbers used:
• 1 (host connect),
• 2 (host disconnect),
• 3 (failed host connect)
• 4 (statistics)

```

# Gproxy

- Event trap

```
garTEProgNo      OBJECT-TYPE
    SYNTAX        INTEGER{
gwnb6(1),
gwnw6(2),
gwtcp6(3),
gldiws(10),
glx25(11),
gltcp(12),
glsna(13),
qsim(20),
pthru(21),
v78sim(22),
g3270(23),
padcon(24),
gmailer(25),
gspool(26),
guft(27),
guftsrv(28)
    }
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Identifies the broadcasting application (program) type"
 ::= { garTrapEvt 1 }
garTEProgAddress OBJECT-TYPE
    SYNTAX        OCTET STRING
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Network address of broadcasting program"
 ::= { garTrapEvt 2 }
garTEPid         OBJECT-TYPE
    SYNTAX        OCTET STRING
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Pid (ServerX) or gatewayname"
 ::= { garTrapEvt 3 }
garTESerialNo   OBJECT-TYPE
    SYNTAX        INTEGER
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Broadcasting programs broadcast serial number"
 ::= { garTrapEvt 4 }
garTEProgVersion OBJECT-TYPE
    SYNTAX        OCTET STRING
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Version/release number of broadcasting program"
 ::= { garTrapEvt 5 }
garTEUserName   OBJECT-TYPE
    SYNTAX        OCTET STRING
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Name of (end-)user"
 ::= { garTrapEvt 6 }
garTEUserAddr   OBJECT-TYPE
    SYNTAX        OCTET STRING
    ACCESS        read-only
    STATUS        mandatory
DESCRIPTION "Network address of (end-) user (if gateway)"
 ::= { garTrapEvt 7 }
garTESessionNo  OBJECT-TYPE
    SYNTAX        INTEGER
```

```

ACCESS      read-only
STATUS      mandatory
DESCRIPTION "The session number in question.
Applicable for gateways only."
::= { garTrapEvt 8 }
garTEHostName OBJECT-TYPE
SYNTAX      OCTET STRING
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Name of host (TSAP/SC-ID or Insid)"
::= { garTrapEvt 9 }
garTERetCode OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Error code returned in case of connection failure"
::= { garTrapEvt 10 }
• Statistics trap

garTSProgNo  OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
::= { garTrapStat 1 }
garTSProgAddress OBJECT-TYPE
SYNTAX      OCTET STRING
ACCESS      read-only
STATUS      mandatory
::= { garTrapStat 2 }
garTSPid     OBJECT-TYPE
SYNTAX      OCTET STRING
ACCESS      read-only
STATUS      mandatory
::= { garTrapStat 3 }
garTSSerialNo OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
::= { garTrapStat 4 }
garTSSessNo  OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Current no of sessions"
::= { garTrapStat 5 }
garTSMaxNo   OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Max no. of sessions since last report"
::= { garTrapStat 6 }
garTSMinNo   OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Min no. of sessions since last report"
::= { garTrapStat 7 }
garTSUpstream OBJECT-TYPE
SYNTAX      INTEGER
ACCESS      read-only
STATUS      mandatory
DESCRIPTION "Number of octets sent to host"
::= { garTrapStat 8 }

```

## Gproxy

```
garTSDownstream    OBJECT-TYPE
    SYNTAX          INTEGER
    ACCESS          read-only
    STATUS          mandatory
DESCRIPTION "Number of octets sent to host"
 ::= { garTrapStat 9 }

END
```







# Appendix: Formatted SNMP trap trace

---

```

SNMP trap formatted ('Sniffer' format) dump:
----- Frame 1 -----
SUMMARY Delta T Destination Source Summary
M 1 ArildDell Arild Zenith ARP R PA=[192.150.211.25]
HA=0000C0F2A817 PRO=IP

ARP: ----- ARP/RARP frame -----
ARP:
ARP: Hardware type = 1 (10Mb Ethernet)
ARP: Protocol type = 0800 (IP)
ARP: Length of hardware address = 6 bytes
ARP: Length of protocol address = 4 bytes
ARP: Opcode 2 (ARP reply)
ARP: Sender's hardware address = WstDigF2A817, Arild Zenith
ARP: Sender's protocol address = [192.150.211.25]
ARP: Target hardware address = 00608CB3A339, ArildDell
ARP: Target protocol address = [192.150.211.12]
ARP:

ADDR HEX ASCII
0000 00 60 8C B3 A3 39 00 00 C0 F2 A8 17 08 06 00 01 .'.9. ....
0010 08 00 06 04 00 02 00 00 C0 F2 A8 17 C0 96 D3 19 .....
0020 00 60 8C B3 A3 39 C0 96 D3 0C 00 00 00 00 00 00 .'.9. ....
0030 00 00 00 00 23 21 79 17 02 00 00 00 .....#!y.....
----- Frame 2 -----
SUMMARY Delta T Destination Source Summary
2 0.0005 Arild Zenith ArildDell SNMP Trap Enterprise specific
enterprise.827.1.1 .. enterprise.827.1.9 (9 items)

SNMP: ----- Simple Network Management Protocol -----
SNMP:
SNMP: Version = 0
SNMP: Community = public
SNMP: Command = Trap
SNMP: Enterprise = {1.3.6.1.4.1.827}
SNMP: Network address = [192.150.211.12]
SNMP: Generic trap = 6 (Enterprise specific)
SNMP: Specific trap = 2
SNMP: Time ticks = 239500
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.1} (enterprise.827.1.1)
SNMP: Value = 2
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.2} (enterprise.827.1.2)
SNMP: Value = 00006E205FC5
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.3} (enterprise.827.1.3) SNMP: Value =
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.4} (enterprise.827.1.4)
SNMP: Value = 17

```

# Gproxy

```
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.5} (enterprise.827.1.5)
SNMP: Value =
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.6} (enterprise.827.1.6)
SNMP: Value = ARILD
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.7} (enterprise.827.1.7)
SNMP: Value = 00608CB3A339
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.8} (enterprise.827.1.8)
SNMP: Value = 0
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.9} (enterprise.827.1.9)
SNMP: Value = Ella Server
SNMP:
ADDR      HEX                                                                 ASCII
0000  00 00 C0 F2 A8 17 00 60 8C B3 A3 39 08 00 45 00  ......\...9..E.
0010  00 FF 00 10 00 00 3C 11 56 8B C0 96 D3 0C C0 96  ....<.V.....
0020  D3 19 78 77 00 A2 00 EB 20 FB 30 82 00 DF 02 01  ..xw.... .0....
0030  00 04 06 70 75 62 6C 69 63 A4 82 00 D0 06 07 2B  ...public.....+
0040  06 01 04 01 86 3B 04 04 C0 96 D3 0C 02 01 06 02  ....;.....0....+
0050  01 02 02 03 03 A7 8C 30 82 00 B2 30 0E 06 09 2B  ....0.....0....+
0060  06 01 04 01 86 3B 01 01 02 01 02 30 19 06 09 2B  ....;.....0....+
0070  06 01 04 01 86 3B 01 02 04 0C 30 30 30 30 36 45  ....;.....00006E
0080  32 30 35 46 43 35 30 0D 06 09 2B 06 01 04 01 86  205FC50.....+....
0090  3B 01 03 04 00 30 0E 06 09 2B 06 01 04 01 86 3B  ;....0...+.....;
00A0  01 04 02 01 11 30 0D 06 09 2B 06 01 04 01 86 3B  ....0...+.....;
00B0  01 05 04 00 30 12 06 09 2B 06 01 04 01 86 3B 01  ....0...+.....;
00C0  06 04 05 41 52 49 4C 44 30 19 06 09 2B 06 01 04  ...ARILD0...+...
00D0  01 86 3B 01 07 04 0C 30 30 36 30 38 43 42 33 41  ..;....00608CB3A
00E0  33 33 39 30 0E 06 09 2B 06 01 04 01 86 3B 01 08  3390.....+.....;
00F0  02 01 00 30 18 06 09 2B 06 01 04 01 86 3B 01 09  ...0...+.....;
0100  04 0B 45 6C 6C 61 20 53 65 72 76 65 72 53      ..Ella ServerS
- - - - - Frame 3 - - - - -
SUMMARY Delta T      Destination Source      Summary
3      1.1384 Arild Zenith ArildDell      SNMP Trap Enterprise specific
enterprise.827.1.1 .. enterprise.827.1.9 (9 items)

SNMP: ----- Simple Network Management Protocol -----
SNMP:
SNMP: Version = 0
SNMP: Community = publicSNMP: Command = Trap
SNMP: Enterprise = {1.3.6.1.4.1.827}
SNMP: Network address = [192.150.211.12]
SNMP: Generic trap = 6 (Enterprise specific)
SNMP: Specific trap = 1
SNMP: Time ticks = 239500
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.1} (enterprise.827.1.1)
SNMP: Value = 2
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.2} (enterprise.827.1.2)
SNMP: Value = 00006E205FC5
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.3} (enterprise.827.1.3)
SNMP: Value =
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.4} (enterprise.827.1.4)
SNMP: Value = 18
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.5} (enterprise.827.1.5)
SNMP: Value =
```

```

SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.6} (enterprise.827.1.6)
SNMP: Value = ARILD
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.7} (enterprise.827.1.7)
SNMP: Value = 00608CB3A339
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.8} (enterprise.827.1.8)
SNMP: Value = 0
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.1.9} (enterprise.827.1.9)
SNMP: Value = Ella Server
SNMP:
ADDR  HEX                                     ASCII
0000  00 00 C0 F2 A8 17 00 60 8C B3 A3 39 08 00 45 00 .....`...9..E.
0010  00 FF 00 11 00 00 3C 11 56 8A C0 96 D3 0C C0 96 .....<.V.....
0020  D3 19 78 77 00 A2 00 EB 1F FC 30 82 00 DF 02 01 ..xw.....0.....
0030  00 04 06 70 75 62 6C 69 63 A4 82 00 D0 06 07 2B ...public.....+
0040  06 01 04 01 86 3B 04 04 C0 96 D3 0C 02 01 06 02 .....?.....
0050  01 01 02 03 03 A7 8C 30 82 00 B2 30 0E 06 09 2B .....0.....+
0060  06 01 04 01 86 3B 01 01 02 01 02 30 19 06 09 2B .....;.....0.....+
0070  06 01 04 01 86 3B 01 02 04 0C 30 30 30 30 36 45 .....;.....00006E
0080  32 30 35 46 43 35 30 0D 06 09 2B 06 01 04 01 86 205FC50.....+
0090  3B 01 03 04 00 30 0E 06 09 2B 06 01 04 01 86 3B ;....0.....+.....;
00A0  01 04 02 01 12 30 0D 06 09 2B 06 01 04 01 86 3B .....0.....+.....;
00B0  01 05 04 00 30 12 06 09 2B 06 01 04 01 86 3B 01 .....0.....+.....;
00C0  06 04 05 41 52 49 4C 44 30 19 06 09 2B 06 01 04 ...ARILD0.....+...
00D0  01 86 3B 01 07 04 0C 30 30 36 30 38 43 42 33 41 .,;....00608CB3A
00E0  33 33 39 30 0E 06 09 2B 06 01 04 01 86 3B 01 08 3390.....+.....;..
00F0  02 01 00 30 18 06 09 2B 06 01 04 01 86 3B 01 09 ...0.....+.....;..
0100  04 0B 45 6C 6C 61 20 53 65 72 76 65 72 53 ..Ella ServerS
- - - - - Frame 4 - - - - -
SUMMARY Delta T Destination Source Summary
4 122.1291 Arild Zenith ArildDell SNMP Trap Enterprise specific
enterprise.827.2.1 .. enterprise.827.2.9 (9 items)

SNMP: ----- Simple Network Management Protocol -----
SNMP:
SNMP: Version = 0
SNMP: Community = public
SNMP: Command = Trap
SNMP: Enterprise = {1.3.6.1.4.1.827}
SNMP: Network address = [192.150.211.12]
SNMP: Generic trap = 6 (Enterprise specific)
SNMP: Specific trap = 4
SNMP: Time ticks = 252000
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.1} (enterprise.827.2.1)
SNMP: Value = 2
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.2} (enterprise.827.2.2)
SNMP: Value = 00006E205FC5
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.3} (enterprise.827.2.3)
SNMP: Value =
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.4} (enterprise.827.2.4)
SNMP: Value = 19
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.5} (enterprise.827.2.5)
SNMP: Value = 2
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.6} (enterprise.827.2.6)

```

# Gproxy

```
SNMP: Value = 2
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.7} (enterprise.827.2.7)
SNMP: Value = 0
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.8} (enterprise.827.2.8)
SNMP: Value = 188
SNMP:
SNMP: Object = {1.3.6.1.4.1.827.2.9} (enterprise.827.2.9)
SNMP: Value = 20092
SNMP:
```

ADDR	HEX	ASCII
0000	00 00 C0 F2 A8 17 00 60 8C B3 A3 39 08 00 45 00	.....\...9..E.
0010	00 E9 00 12 00 00 3C 11 56 9F C0 96 D3 0C C0 96	.....<.V.....
0020	D3 19 78 77 00 A2 00 D5 04 3C 30 82 00 C9 02 01	..xw.....<0.....
0030	00 04 06 70 75 62 6C 69 63 A4 82 00 BA 06 07 2B	...public.....+
0040	06 01 04 01 86 3B 04 04 C0 96 D3 0C 02 01 06 02	.....;.....
0050	01 04 02 03 03 D8 60 30 82 00 9C 30 0E 06 09 2B	.....'0...0...+
0060	06 01 04 01 86 3B 02 01 02 01 02 30 19 06 09 2B	.....;.....0...+
0070	06 01 04 01 86 3B 02 02 04 0C 30 30 30 30 36 45	.....;....00006E
0080	32 30 35 46 43 35 30 0D 06 09 2B 06 01 04 01 86	205FC50...+.....
0090	3B 02 03 04 00 30 0E 06 09 2B 06 01 04 01 86 3B	;....0...+.....;
00A0	02 04 02 01 13 30 0E 06 09 2B 06 01 04 01 86 3B	.....0...+.....;
00B0	02 05 02 01 02 30 0E 06 09 2B 06 01 04 01 86 3B	.....0...+.....;
00C0	02 06 02 01 02 30 0E 06 09 2B 06 01 04 01 86 3B	.....0...+.....;
00D0	02 07 02 01 00 30 0F 06 09 2B 06 01 04 01 86 3B	.....0...+.....;
00E0	02 08 02 02 00 BC 30 0F 06 09 2B 06 01 04 01 86	.....0...+.....
00F0	3B 02 09 02 02 4E 7C 53	;....N S