Dante, Module *Bandwidth*

Inferno Nettverk A/S
Oslo Research Park
Gaustadalléen 21
NO-0349 Oslo
Norway

Date: 2001/12/12 13:17:27

1 Description

The *Bandwidth* module gives you control over how much bandwidth the *Dante* server uses on behalf of the clients.

It can be used to limit bandwidth to non-work related web/ftp sites, or to prevent ftp-related traffic from impacting too much on interactive telnet/ssh traffic.

It can also be used to give more bandwidth to certain clients or for traffic to certain cites.

In addition, when using the *Dante bind extension*, it can be used to provide bandwidth control to networkservers (like e.g. webservers) that do not support bandwidth control internally.

2 Syntax

The syntax of the `bandwidth` statement is as follows:

`bandwidth: <bytes>`

`bytes` is the maximum bandwidth, measured in bytes, to use per second.

3 Semantics

The `bandwidth` statement integrates as a part of socks-rules.

The maximal `bandwidth` set for a rule will be shared by all clients matching that rule. The *Dante* server will distribute the bandwidth to the matching clients in a least-recently used fashion, trying to let all clients get a fair share.

4 Special notes

Sending the *Dante* server a `SIGHUP` signal forces a reload of the configuration file. It should be noted that this does not affect current sessions.

Changing e.g. a *pass* statement to a *block* statement, does not terminate the session of any existing client. Likewise, a reload of the configuration file does not let sessions created before the reload affect sessions created after the reload.

This means that after a reload of the configuration file, the bandwidth counter for new sessions will be reset, and the possibly confusing situation might arise where more than the configured number of bandwidth is used, as old sessions will use the bandwidth allocated to them before the reload, independently of new sessions.

Eventually the old sessions will finish however, and the maximal number of sessions will be the the number currently configured, until a new reload of the configuration-file occurs.

5 Examples

This section shows several examples of how one could use the *bandwidth* module.

5.1 Limiting web/http bandwidth

The below rule shows how one can limit the bandwidth used for webtraffic from the clients on the 10.0.0.0/24 net to a total of 10240 bytes, or 100 KiloBytes.

```
pass {
   from: 10.0.0.0/24 to: 0.0.0.0/0 port = http
   command: connect
   bandwidth: 102400
}
```

5.2 Increasing web/http bandwidth

The next rule, if placed before other bandwith-limiting rules, shows how one can increase the bandwidth used for webtraffic from the clients on the 10.0.0.0/24 net to certain sites.

In this case, the clients will be able to use 1024000 bytes, or one MegaByte, per second when going to the address work.example.com.

```
pass {
   from: 10.0.0.0/24 to: work.example.com port = http
   command: connect
   bandwidth: 1024000
}
```

5.3 Limiting ftp bandwidth

The next rule shows how one can limit the bandwidth used for ftp-data for the clients on the 10.0.0.0/24 net to a total of 10240 bytes, or 10 kB/s

This only works for *active* ftp, since for *passive* ftp we don't have fixed portnumbers.

```
pass {
   from: 0.0.0.0/0 port = ftp-data to: 10.0.0.0/24
   command: bindreply
   bandwidth: 10240
}
```

5.4 Limiting bandwidth provided by internal servers to the outside

The next rule shows how one could use the *Dante bind extension* together with the *Bandwidth* module to limit the amount of data provided by a internal server, in this case, a webserver called *our-webserver.example.com*, to a total of 10240 bytes, or 10 kB/s.

This requires the webserver to be socksified and the *bind extension* to be enabled on both the socksified client and on the *Dante* server.

```
pass {
   from: 0.0.0.0/0 to: our-webserver.example.com port = http
   command: bindreply
   bandwidth: 10240
}
```