Inferno Nettverk A/S

Barefoot Module Documentation

Session Module

May 15, 2012

1 Description

The session module makes it possible to set an upper limit on the number of simultaneous connections that clients will be allowed to create to given destinations.

The *Barefoot* server distributes the available sessions on a first come, first served basis.

This module can be used to e.g., limit the number of sessions going to certain internal servers, reducing the risk that they will be overloaded. When the session limit has been reached, future clients will not be able to create new sessions until some of the existing sessions have ended.

The *Barefoot* server first accepts any connection in order to determine the IP-address and any other credentials. Connections are then immediately closed if a session-limit has been reached.

Clients may re-try again later, and if one of the old clients has finished in the meantime, the new client will be granted access.

2 Syntax

The syntax is as follows:

`maxsessions: <sessions>`

`sessions` is the maximum number of sessions that can be active at any time.

3 Semantics

The `maxsessions` statement integrates as a part of the *rules*. See *barefootd.conf(5)* for more information about rules.

4 Examples

This section gives some examples on how this module can be used.

4.1 Limiting the number of clients accessing an internal mail server

Assume you have an internal mail server that does not handle overload conditions well.

The next rule shows how one can limit the number of clients connecting to the internal mailserver, reducing the chance that e.g., a virus outbreak or denial of service attack against your mailserver takes it down.

```
pass {
   from: 0.0.0.0/0 to: barefoot-server port = smtp
   bounce to: internal-mail-server
   maxsessions: 100
}
```

4.2 Limiting the number of clients from a specific country

Assume you want to limit the number of clients from China to your internal mail server, but not limit clients from other countries.

E.g., you have an internal mail server called ”internal-mail-server”, and due problems with clients from China overloading it, you want to limit connections from China to a maximum of 5 concurrent sessions.

Assume all clients from China will have a IP-address that resolves to a domain in China, i.e., it will end in ”.ch”.

Assume also that you have via e.g. the *srchost* keyword disallowed clients that do not have any IP-address registered, aswell as clients trying to ”fake” their domain-name, from using the *Barefoot* server.

The next two rules then shows how you could accomplish this.

```
# disallow clients that do not have a dns-record, as well as clients
# where there is a mismatch between what what the dns server for the
# IP-address says, and what the dns server for the domain-address says.
srchost: nodnsunknown nodnsmismatch

# clients from china will share a total of five sessions at most.
client pass {
   from: .ch to: barefoot-server port = smtp
   bounce to: internal-mail-server
   maxsessions: 5
}

# clients from anywhere else will not have any limits imposed on them
client pass {
   from: 0.0.0.0/0 to: barefoot-server port = smtp
   bounce to: internal-mail-server
}
```

4.3 SIGHUP

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

A reload of the configuration file only affects sessions created after the reload. It will not affect any of the existing sessions.

This means that changing e.g., a *pass* statement to a *block* statement, does not terminate the session of any existing client. Likewise, changing the limits set in a rule does not change the values for any existing session.

After a reload of the configuration file, old sessions will continue to operate in a separate space, using the old configuration, while new sessions will use the new configuration.