Gateway connections remain open because of firewall

Purpose

The purpose of this wiki is to explain why gateway connections could remain open when making a connection through a firewall.

Overview

The page will look at parameters that can be used to disconnect gateway connections and also reasons why firewalls may cause connections to remain on the gateway.

Parameter Settings

Parameters gw/gw_disconnect and gw/keepalive can be used to disconnect gateway connections:

Parameter : gw/gw_disconnect:

This parameter specifies the time period after which an inactive gateway connection is to be closed. Normally, connections that have been created to other gateways are held until the other gateway is closed.

If no active connections exist to this gateway and the time frame has expired, then the connection to this gateway is closed.

Value of 0 deactivates check

This check is executed a maximum of every gw/max_sleep seconds. Therefore, it does not make sense to set the value smaller than gw/max_sleep.

Parameter gw/keepalive:

Specifies the maximum time period (in seconds) before the system checks, using a ping, whether the partner is still alive when there is no data transfer across a CPIC connection.

A value of 0 deactivates this check.

Parameters gw/gw_disconnect and gw/keepalive will close connections that have been OPENED by that gateway, connections ACCEPTED by the gateway will not be closed by these parameters.

Also check parameter gw/close_routes:

Connections that run over a Wide Area Network cause high running costs. Connections of this type are realized in an SAP environment using a SAProuter. The time period with which inactive gateway connections are closed is 30 minutes by default (gw/gw_disconnect). This is too long for WAN connections of this type. Therefore, connections that use SAProuter are handled separately. If a connection that uses an SAProuter is not used, this connection is closed after a certain time. This period is determined by the value of this parameter. Value 0 deactivates the check of these connections. Connections between gateways in a Local Area Network (LAN), which do not usually run through an SAProuter, are not affected by this parameter.

This check is executed a maximum of every gw/max_sleep seconds. Therefore, it does not make sense to set the value smaller than gw/max_sleep.

Firewall issues

Gateway connections may also remain because of firewall problems. The design of such firewalls is as follows:
In other words, the firewall has two TCP connections. The TCP connection of gateway 1 ends in the firewall. The firewall opens up a second one to gateway 2 and directs the traffic internally (and therefore has influence on the traffic) to the correct TCP channels.

Gateway 1 can cancel the TCP1 connection because of the gw/gw_disconnect parameter. TCP1 is deleted. However, the firewall doesn't propagate the closing to the internally linked TCP 2 connection. In other words, the gateway 2 doesn't get informed that the remote gateway 1 closed the connection and therefore the TCP 2 remains in the gateway 2 display.

If now, a new connection has to be setup from gateway 1 to gateway2, a TCP 3 and TCP 4 are built. TCP 4 appears in the gateway 2 display. If gw/gw_disconnect applies, TCP3 will be deleted again, but TCP 4 remains in the display of gateway 2 and so on.

This particular issue would need to be checked by the local network administrators to check the timeout parameter on the firewall.

Related Content

Related Documents

Related Notes

743888  "GW: gw/so_keepalive profile parameter"
1578168  "gw/so_keepalive"
1095272  Redundant GW/GW connections in transaction SMGW
900132  Remaining user contexts, remaining locks
1261669  RFC connections are not closed
568271  Lifetime of an RFC connection
824722  Gateway parameter: gw/local_addr
1731719  Hanging RFC connections
1791528  GW: Hanging connections between J2EE and gateway