**NANC 301**

**NPAC Monitoring of SOA and LSMS Associations via NPAC TCP Level Heartbeat (transport layer)**

**Origination Date :**01/12/2000

**Originator:**LNPA WG

**Description:**

Same as NANC 299, but using the TCP Keepalive feature (transport layer) instead of an application level heartbeat.

The requested functionality of this change order *“NPAC Monitoring of SOA and LSMS Associations via Heartbeat”* can be accomplished using the TCP Keepalive feature.  Since no data flows across an idle TCP connection (i.e., between the two TCP modules), the TCP Keepalive feature can be used to poll the other end of an idle connection to make sure the connection is still available (active).  This will also alleviate any situations of a half-open connection.

With this change order, the NPAC SMS is being required to enable the TCP Keepalive feature. The NPAC SMS will serve as the server side, and the local system (SOA or LSMS) will serve as the client side. Optionally (but recommended), the Service Provider can enable the TCP Keepalive feature to ensure an active connection.  If only the NPAC SMS were to enable the TCP Keepalive feature, a situation exists where the local side may not detect an inactive connection (and therefore not try to re-associate). If the local system does enable the TCP Keepalive feature, they would also detect an inactive connection (and accordingly attempt to re-associate with a new bind request). In this example, the length of time that the Service Provider is un-available will be greater if the local side has not enabled the TCP Keepalive feature since they are relying on NPAC Personnel to contact them about an aborted association (rather than having their own system recognize the abort and initiate a new bind request to the NPAC SMS).

Additionally, the NPAC SMS needs to provide logging functionality (which is documented in change order NANC 219).  This is accomplished by enabling the TCP Keepalive feature of the OTS stack software such that it recognizes an inactive association and issues an abort to the application for a given association. Since an inactive connection will appear as a stack abort to the server (in this case the NPAC SMS), logging will be done as an abort from a client (in this case the local system).

In summary, the requested change is for the NPAC SMS to enable the TCP Keepalive feature on all TCP connections initiated from OTS/RFC1006.

The TCP Keepalive feature was introduced to the HP OTS stack software in November 1999 (patch PHNE\_17376). Jan 00, Jim loaded the stack patch on a test box at ESI.  He worked with Beth to test the patch. The TCP level heartbeat worked fine one-way (from NPAC to local system not supporting the patch).  Beth is planning on loading the patch on her local system and testing the other way as well.  It is believed that enabling the TCP Keepalive will solve many of the association control problems that have experienced in the production environment.

Feb 00, LNPA-WG meeting, the group proposes that we have a con call to discuss this further.  The desire is to have this functionality implemented prior to R4.

**Final Resolution:**

Pure Backwards Compatible:  YES

Utilizing the TCP Keepalive feature of the HP-UX OTS stack software (otsadm) involves starting the stack software with the -K option. With this feature enabled, all subsequent TCP connections initiated from OTS/RFC1006 will have the TCP\_KEEPALIVE option set, which allows TCP to inform OTS/RFC1006 of lower layer failures. With the TCP Keepalive feature turned on, the OTS stack software uses the tcp\_keepstart, tcp\_keepfreq, and tcp\_keepstop system tunables to execute the keep-alive message. The tcp\_keepstart tunable is the number of seconds that a TCP connection can be idle before keep-alive packets will be sent attempting to solicit a response. When a packet is received, keep-alive packets are not longer sent unless the connection is idle again for this period of time. The tcp\_keepfreq tunable is the interval in seconds at which keep-alive packets will be sent on a TCP connection once they have been started. The receipt of a packet will stop the sending of keep-alive packets. The tcp\_keepstop tunable is the number of seconds keep-alive packets will be sent on a TCP connection without the receipt of a packet after which the connection will be dropped and an abort message sent up the stack to the application. The setting of these tunables apply across a machine for all TCP connections.

The values of the tcp\_keepstart, tcp\_keepfreq, and tcp\_keepstop must be determined by the LNPA-WG. The default values are tcp\_keepstart=7200 secs, tcp\_keepfreq=75 secs, and tcp\_keepstop=600 secs. In order to detect an inactive associations as soon as possible, the following values have been suggested based on current testing efforts: tcp\_keepstart=60 secs, tcp\_keepfreq=60 secs, and tcp\_keepstop=60 secs.

New requirement:

Req 1 – NPAC SMS Monitoring of SOA and Local SMS Connections via a TCP Level Heartbeat

The NPAC SMS shall be capable of supporting a TCP Level Heartbeat via the TCP Keepalive Feature.

**Related Release:**

2.0.2

**Status:** Implemented