

DCMTK - Feature #245

Maximal Number of Associations vereinheitlichen

2006-03-14 00:00 - Jörg Riesmeier

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0:00 hour
Target version:		Compiler:	
Module:	diverse		
Operating System:			
Description			
siehe interne E-Mail-Kommunikation vom 2006-02-21 === Mail JR ===			
habe gerade zufällig beim Blick in den ppsmgr-Quelltext herausgefunden, daß die maximale Anzahl der Netzwerkverbindung bei einigen Tools sehr wohl begrenzt ist und tw. auch konfiguriert werden kann.			
• storescp: unbegrenzt, zumindest keine Beschränkung durch das Programm • wlmcpfs: Option --max-associations, Standardwert = 50 (Bereich: 1-2^31) • wlmcpdb: offensichtlich hart kodiert (!) auf den Wert 20 • wlmcpki: offensichtlich hart kodiert (!) auf den Wert 20 • ppscpfs: offensichtlich hart kodiert (!) auf den Wert 20 • ppscpdb: offensichtlich hart kodiert (!) auf den Wert 20 • ppsmgr: offensichtlich hart kodiert (!) auf den Wert 20			
Praktisch wird die Höchstgrenze natürlich auch bei wlmcpfs weit unterhalb von 2^31 liegen. Warum das allerdings bei einem Tool konfigurierbar ist und bei anderen nicht, weiß ich leider auch nicht. Meines Erachtens ist eine konfigurierbare Obergrenze sinnvoll, um z.B. "Denial of Service Attacks" zu unterbinden oder zumindest die damit verbundenen Probleme zu reduzieren.			
Beim imagectn/dcmqrscp kann man die maximale Anzahl der Netzwerkverbindungen übrigens in der Konfigurationsdatei einstellen.			
=== Mail ME ===			
Der Mechanismus, den der Worklist-SCP und der Query/Retrieve SCP unter Unix nutzen, um mehrere Clients gleichzeitig bedienen zu können, funktioniert grundsätzlich anders, als die Implementierung, die wir derzeit unter Windows im storescp und PPS-SCP haben:			
Worklist + Q/R: * fork() liefert die PID des Child-Prozesses * diese wird in eine Liste eingetragen * ein regelmäßiges wait3() / waitpid() liefert die PID eines Child-Prozesses, der terminiert ist. * diese wird aus der Liste wieder ausgelesen * Vor jedem fork() wird geprüft, ob die Liste die angegebene Maximalgröße hat und dann ggf. die Verbindung abgelehnt. Dies wiederum setzt voraus, dass Association Negotiation vom Parent-Prozess gemacht wird und nicht vom Child-Prozess.			
StoreSCP / PPS-SCP * Die von fork() gelieferte PID wird unter Unix ignoriert * Die von CreateProcess() vermutlich ebenfalls gelieferte PID auch * Association Negotiation wird vom Child durchgeführt, nicht vom Parent (eine Notwendigkeit, da der Child-Prozess unter Windows sonst nicht weiss, was ausgehandelt wurde) * waitpid/wait3 unter Unix räumt nur die Zombies ab, die PIDs werden ignoriert. * Unter Windows gibt es kein waitpid/wait3.			
Wenn man hier eine Kontrolle der Anzahl der Prozesse einführen wollte, müsste man in der Tat einen Mechanismus finden, mit dem man unter Windows feststellen kann, wieviele Child-Prozesse noch laufen oder - analog zu waitpid - welcher Prozess sich gerade beendet hat.			
Im übrigen werden wir den Worklist-SCP ja wohl auf das "StoreSCP-Modell" umstellen müssen, wenn der unter Windows mehrere Clients parallel bedienen soll. Dann geht die Kontrolle dort ebenfalls flöten. Ditto bei einer etwaigen Umstellung vom Q/R-SCP und dem Printserver.			

Eine andere Sache ist die hardcodierte Obergrenze von 20, die ist natürlich hässlich.