Continuous Computing
Create | Deploy | Converge™

Diameter 1.2 Performance Report

1/16/2007

This document presents the results for the performance testing completed for Diameter 1.2. The performance analysis has been done for specific hardware and software environments as described in this document. The performance depends on many factors, such as the system architecture, layer coupling, operating system, hardware organization, and compiler options. Also, the figures indicated correspond to performance of stand alone Diameter software with sample application running over it.

The performance of Continuous Computing's Diameter stack was measured in the hardware and software environments as described in Table 1 and Table 2. Therefore, the performance may vary from the performance described in this document depending on the environment.

Table 1 describes the hardware configuration used for the performance testing.

Table 1: Hardware configuration

Type	Description
Operating System	Red Hat Enterprise Linux
CPU	Single core CPU / 32 bit; 2.8 GHz, 240 MB Memory, 1 MB Cache
Processor	Pentium 4

Table 2 describes the software configuration used for the performance testing.

Table 2: Software configuration

Type	Description
Operating System	Linux, Kernel version 2.4.21
Compiler	GNU C Compiler gcc version 3.2.3
Compiler options	-O3 -ansi -Wall -Wno-comment -pipe -Wstrict-prototypes -Wshadow -Wcast-qual -Wmissing-prototypes –pedantic -Wimplicit –Wunused
Protocol layer	Diameter 1.2

The test setup is illustrated in Figure 1.

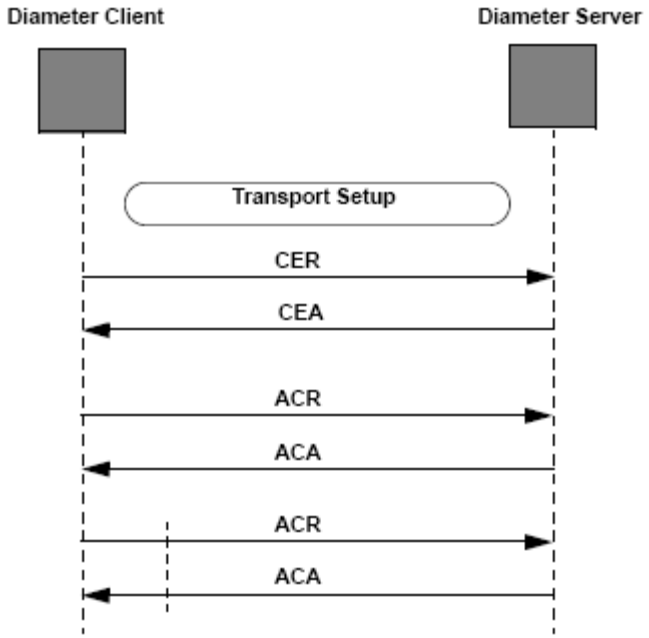

Figure 1: Test setup

The test procedure is as follows:

- Continuous Computing's Trillium Diameter stack is configured as Server with a thin sample application running over it.
- Another Diameter stack is configured as Client with a thin sample application running over it.
- Connection is established between Diameter Server and Client over TUCL.
- Once the CER and CEA are exchanged, ACR (Accounting Request) messages are sent from the Diameter Client.
- Diameter Server will in-turn process the ACR and send the ACA (Accounting Answer) in response.
- Around 4 million ACRs were sent and 4 million ACAs were received in response to the ACRs. The performance is calculated based on the time taken to process 8 million messages in tightly coupled mode.
- The performance results are shown in Table 3.

Table 3: Performance results

Sl. No.	Messages per second (Sent + Received)	CPU Utilization	Average message length (bytes)	Comments
1	13,480	45%	270	With Diameter message and AVP validation against DM and AVP dictionaries.
2	15,384	45%	270	With Diameter message validation against DM dictionary, but without AVP validation against AVP dictionary.