Managing Web Servers with SNMP

Karina Marcus, Christian Gigandet, Olivier Spielmann
Institut Eurécom

May 26, 2000

Web servers play a very important role in the E-services that are being offered nowadays. Several different HTTP servers exist, as Apache and Microsoft Site Server, and many offer their own proprietary management tool. We propose to manage a Web server using the SNMP framework. SNMP agents were primary designed to control network elements, but gradually they are stepping into the application area, including databases, mail servers, and other.

The objective of this Web server SNMP agent is to centralize all the information concerning the Web server and its host machine, in order to make it available to an SNMP manager, in the form of a simple browser, or a network management platform, as HPOV NNM.

The agent implements the `system` group of MIB-2 (RFC1213) and an HTTP MIB that is simpler than the "WWW Service MIB" (RFC2594), and is instrumented to monitor Apache Web servers. The MIB stores information about the configuration file of the Web server, about the performance of the host machine, and about the log files. A group has been designed to present a summary of the received protocol requests, and the produced errors. Moreover, one top N list concerning the most frequently visiting IP addresses and another on the most frequently requested pages are also maintained. Figure 1 shows the structure of the MIB.

Web server log files have usually an impressive size. In order to keep the information extracted from the log files updated, two schemes have been conceived and are compared in this work.

In the first one the access.log file is processed periodically, with the results being stored in an auxiliary file. Whenever an SNMP-GET request concern-

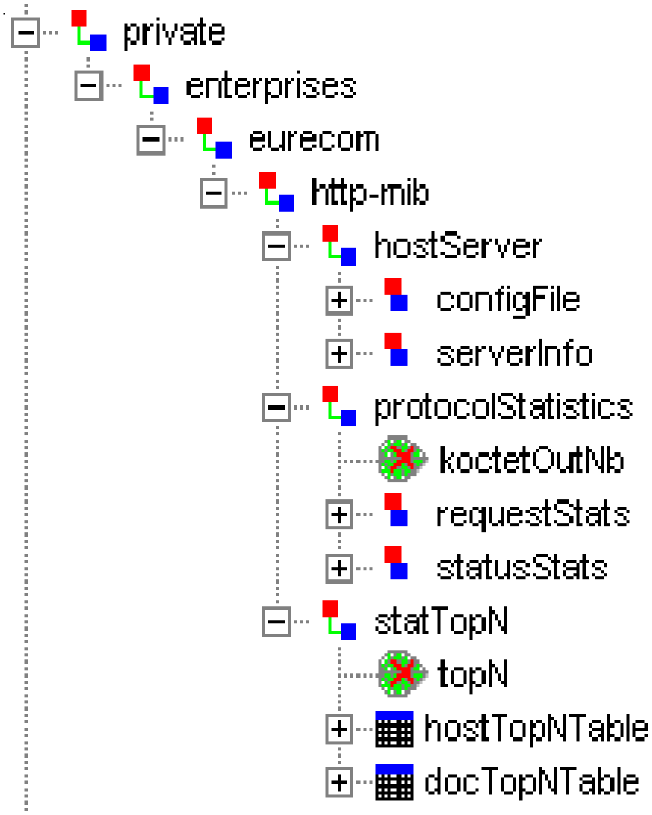

Figure 1: Structure of the implemented HTTP MIB

ing the protocol statistics or the Top N lists is received, then the agent uses the results file and the non-processed last part of the access log file to give the updated information.

A second method is also studied, where the results file is subdivided in three files, concerning the protocol statistics, the Top N visiting hosts, and the Top N visited HTML pages. As soon as a new line is inserted in the access log file, an independent daemon processes the information contained in this line, and updates the three resulting files.

Comparing the two methods we concluded that the obtained results are the same, but that the imposed CPU load pattern is quite different. The first scheme periodically demands a high CPU usage, but between the calculations the CPU is not used; on the other hand, the second method constantly uses CPU resources, but with no big punctual picks.