Rate Limiting in My Assembly Language Simulators

Overview

In the autumn of 2022, I started to notice that my hosting service was recording overloads from the sites I was running and taking them offline. As I started to look at logs more and implement overload detection, I saw some disturbing cases that (more or less) amounted to abuse. All of the simulators now have limits on the rate they can loaded and rejection of reload requests in the same second. There is also logging of all requests that exceed the limits and, where necessary, action is taken against specific IP addresses.

Some overloads have been traced to people using a management system to load a room full of computers at the same time. I ask you not to do this and get the students to do their own setup (obviously without deliberately synchronising). The average load over a day is insignificant, but artificial peaks cause problems. There is more discussion of this later.

I'm going to write more over time. So far, I have done a few sections about the changes happening now. Later I intend to add more about the technicalities and a look at some of the logs from 2022 and 2023 with examples of the overloads.

Cookie Use

The LMC simulators have been setting a cookie called RATELIMIT for over a year now. This is essential because it is the only way to tell that a single IP address represents a number of users (and how many). I am going to extend this to all the simulators over time and this note is being written to coincide with cookie introduction to the AQA simulator.

I will adapt the rate limiting algorithms over time. Initially the AQA simulator will keep its existing rate limit and rejection algorithms but there is scope for improvement. I will certainly use the cookie to detect classrooms that do bulk loads and put limits on their IP address.

One thing I may do is improve the service to someone presenting the cookie, because if they have kept the cookie, then they likely have the other files cached and will make no follow-on requests. (This highlights one of the problems that classroom computers rarely have any files cached.)

I don't ask about cookies because most sites that ask have already set several and keep at least one to remember the answer (on essential grounds). I regard the one cookie as essential to the operation of the site.

A user who has cookies disabled presents a problem. For the existing LMC implementation they are diverted to Google Login which achieves the same object (and ironically gives me more information than a cookie ever would). For the new AQA case, the user gets a "no cookie" page where they can click to continue without cookies. However they get a lower rate limit than users who allow cookies and do not get automated retries.

If you are a classroom with cookies disabled, please send me an email (plh256 at hotmail.com) with your external IP address so I can treat it appropriately. (E.g. known schools may get a rate limit algorithm designed to improve their service or suspected schools may get restrictions.)

Rate Limiting

I do adapt the rate limiting algorithms over time to respond to issues that I notice in the logs. I keep a log of all reject or delay events. A current very common action is to impose a 10 second delay (the user is told 15) if someone attempts a reload within the same second as the previous load they made. After that, 5 loads are permitted every two seconds.

If the user has to wait, then the system looks for several people waiting on the same IP address and issues a “classroom warning” together with a random 5 to 20 second delay before retry. If not a classroom, then a fixed 2 or 4 second delay before retry is done, depending on how busy the system is.

If a reload is attempted during a retry wait, there are various algorithms; the LMC diverts to Google Login, other cases just say “you have gone to the back of the queue”.

Analytics

I use a system called Matomo to collect basic information about users. It only monitors initial loads (and F5 reloads) and is run in a cookie-less mode. You could get almost the same information from the server logs, however using JavaScript automatically omits anyone just scanning the web (whether they declare it or not). When looking for issues, I use the IP addresses collected by Matomo and in the server logs but only statistics are ever shared with others (which might include usage by country, but no further).

F5 reloads are currently monitored to test the effectiveness of the change to internal restarts as compared to Browser reloads. Since the object is to reduce overheads, I will probably remove this at some point.

28/9/2024

Peter Higginson