

بسم الله الرحمن الرحيم

راهنمای developer فرایند load balancer

فصل ۱

۱-۱ نگارش

در جدول زیر تاریخچه نگارش های مختلف ماژول آورده شده است.

تاریخ	نسخه	توضیحات
مرداد ۹۷	۱,۰	نسخه اولیه document آماده شد.

۲-۱ توضیحات اولیه

این فرایند به جهت استفاده بهینه از منابع سیستم و جلوگیری از پر شدن منابع به طوری که تعادل بین پاسخگویی درخواست ها صورت پذیرد فراهم شده است. برای این کار هر ماژول به یکی از دسته های low, medium, high اختصاص داده شده اند و هر کدام از این دسته ها به صورت نسبی می توانند مقدار مشخصی از سیستم را در اختیار گیرند. در ادامه الگوریتم load balancer تشریح شده است.

۱-۳- الگوریتم load balancer

۱- مقداردهی اولیه: در شروع کار system api و اجرای فرایند load balancer یک حافظه ی حداقلی به سه فرایند process_h, process_m و process_l اختصاص داده می شود. این مقدار حداقلی برابر 900mb می باشد. این مقدار به شکل زیر محاسبه شده است:

فرض می شود مقدار فضای حافظه مورد نیاز برای ماژول های اجرایی توسط فرایند H برابر 300mb، M برابر 40mb و برای فرایند L برابر 10mb می باشد. به طور نسبی برای فرایند H یک سهمیه اجرای درخواست، برای فرایند M، $\text{round}(300/40)$ در خواست و برای فرایند L، $\text{round}(300/10)$ می باشد. این تعداد درخواست ها، حداقل تعداد درخواست ها در زمان شروع به کار سیستم می باشد، بنابراین به ازای هر یک از این سه فرایند 300mb از سیستم در شروع کار رزرو می شود. این مقادیر در مدل Setting ذخیره می شود و هنگامی که یک درخواست جدید به system api داده می شود، تعداد درخواست های در حال اجرا برای process مربوطه با مقداری که به عنوان تعداد درخواست های مجاز برای اجرا در Setting ذخیره شده است مقایسه می شود و در صورت مجاز بودن درخواست داده شده به حالت اجرا در می آید در غیر این صورت درون صف قرار می گیرد.

۲- به روزرسانی

الگوریتم load balancer طی یک interval مشخص روال زیر را برای به روز رسانی مقادیر ذخیره شده در Setting برای تعداد درخواست های مجاز، اجرا می کند:

۱- مقدار منابع آزاد سیستم با مقدار حداقلی که در قسمت ۱ گفته شد مقایسه می شود. اگر منابع آزاد بیشتر بود به ۲ می رود وگرنه به ۳ می رود.

۲-۱- اگر تعداد درخواست های مجاز برای اجرا از تعداد درخواست های اجرا حداقل به اندازه ی تعداد درخواست اولیه که در قدم اول مشخص شده است بیشتر بود باید عمل down روی تعداد درخواست های مجاز صورت گیرد. مقدار کاهش برابر است با اختلاف تعداد درخواست های مجاز و درخواست های در حال اجرا تقسیم بر مقدار اولیه. برای مثال اگر تعداد درخواست های مجاز 2k از تعداد درخواست های در حال اجرا بیشتر بود و k برابر مقدار اولیه ی درخواست های اجرایی بود، به اندازه 2k مقدار ذخیره شده در Setting کاهش می یابد.

۲-۲- اگر تعداد درخواست های مجاز برای اجرا از تعداد درخواست های اجرا بیشتر بود اما تفاوتشان به اندازه ی مقدار اولیه نبود تغییری صورت نمی گیرد.

۲-۳- اگر تعداد درخواست های در حال اجرا با درخواست های مجاز در دیتابیس برابر باشد، یک واحد به اندازه ی مقدار اولیه با دیتابیس اضافه می شود.

۳- اگر منابع آزاد سیستم از مقدار اولیه کمتر بود از فرایندی که بیشترین مقدار منابع را بخود اختصاص داده یک واحد به اندازه ی مقدار اولیه از مقادیر مجاز برای اجرا در دیتابیس کم می شود.

۱-۴- شبه کد load balancer

assuming that $H_{process}$ consume 300 memory units, $M_{process}$ 40, $L_{process}$ 10.

calculate init ratio: $H_{process} = 1, M_{process} = \frac{300}{40}, L_{process} = \frac{300}{10}$

While True:

IF available_resource:

for process:

$Allow_number = get_allow_number_req_for_process_from_database$

$Running_number = get_running_number_of_process$

if $running_process < allowed_process$:

if $allowed_number - running_process < init_ratio$:

do nothing

if $allowed_number - running_process > init_ratio$:

downsize allow_number to : $allow_number - (allowed_number - running_number) / init_ratio$

if running_process = allowed_process:

allowed_process+=1

else if not available_resource:

for process that hold max value of allowed_number:

downsize allowed_number