

GlideinWMS - Feature #7920

Log all curbs & limits hit to the user collector ads

02/23/2015 01:24 PM - Brian Bockelman

Status:	Closed	Start date:	01/13/2016
Priority:	Normal	Due date:	
Assignee:	HyunWoo Kim	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:	v3_2_12	Spent time:	0.00 hour
Stakeholders:	CMS, OSG		
Description			
<p>The curbs and limits are one of the most opaque behaviors in glideinWMS. Many times (>10) we've tried to debug mysterious lack of glideins only to find the issue was not site-related -- rather, they had to do with an unrelated global limit and a problem at the other site.</p> <p>We should be able to get the following into the glideresource ads:</p> <ul style="list-style-type: none">- Per entry point, any factory-side curbs or limits applied (max running / idle jobs per frontend, for example)- Per entry point, any frontend-side curbs or limits applied (max running / idle jobs per group, frontend, or globally, for example).- Any schedd exclusions applied. <p>Since some limits are frontend-wide, it may make sense to have a "frontendresource" ad in the user collector. Not necessary though.</p>			
Subtasks:			
Feature # 11418: Curbs and Limits for Factory			Closed
Related issues:			
Has duplicate GlideinWMS - Feature #7826: Need additional debugging informati...			Closed 02/10/2015

History

#1 - 04/13/2015 04:17 AM - Parag Mhashilkar

- Stakeholders updated

#2 - 07/22/2015 04:20 PM - Parag Mhashilkar

- Assignee set to HyunWoo Kim
- Target version set to v3_2_12

#3 - 08/12/2015 03:54 PM - HyunWoo Kim

I learned that Frontend composes glideresource classad based on the contents of glideclient and glidefactoryclient classads. So, I will just have to add new entries in Frontend's glideclient classad and Factory's glidefactoryclient classad and these new entries can be used for curbs and limits that the user wants to know about. I have identified the section of the relevant codes where I can add new classad entries, and I will find out how to gather relevant information on useful curbs and limits.

#4 - 08/14/2015 10:41 AM - HyunWoo Kim

- % Done changed from 0 to 50

In Frontend code:
glideinFrontendInterface.py
class MultiAdvertizeWork:
def createAdvertizeWorkFile should be modified to add new attributes for curbs and limits to glideclient classad

In Factory code:
glideFactoryEntry.py:
class Entry:
def writeClassadsToFile and gfi.MultiAdvertizeGlideinClientMonitoring should be modified to add new attributes for curbs and limits to glidefactoryclient classad

glideFrontendInterface.py
class ResourceClassad: has two methods
def setGlideClientMonitorInfo(self, monitorInfo): for glideclient classad
def setGlideFactoryMonitorInfo(self, info): for glidefactoryclient classad

The second method loops over all the attributes in glidefactoryclient classad which means our new attributes will be picked up by glideresource but the first method copies attributes using their names which means we will have to modify this method to explicitly specify new attributes to be copied.

#5 - 08/19/2015 04:26 PM - HyunWoo Kim

- % Done changed from 50 to 60

I confirmed that when I added new test attributes in glidefactoryclient classad, those were picked up automatically by glideresource classad. How glideclient classad is translated to glideresource classad is also understood. Now, I just need to find actual information to be advertized. Parag just showed me compute_glidein_min_idle method of class glideinFrontendElement which computes those limits that were hit. I will have to figure out how to provide that type of information to those codes that produce glideclient and glidefactoryclient classads.

#6 - 08/26/2015 12:23 PM - HyunWoo Kim

- % Done changed from 60 to 80

I have implemented current idea of how to take care of this ticket in my local GWMS instance and tested. I am now creating a new git branch for this ticket to ask other developers to review it.

#7 - 09/04/2015 10:51 AM - HyunWoo Kim

I am currently finalizing on what kind of information should be provided and how to display. Basic variables to be displayed are
- number of user jobs (running or idling),
- number of glideins(or startds) that are running user jobs or waiting(idling) for user jobs
- and limits(and curbs) that affect the number of glideins in each category (per entry, per group, per FE, or globally)

The following is a candidate solution that shows Frontend side. I am still working on Factory side information..

1. Information on Jobs
Total Idling Jobs in the Queue (matched or unmatched) = 0
Total Idling Jobs in the Queue matched for this Entry = 0
Total Running Jobs in the Queue matched for this entry = 2
2. Information on Entry-centric Glideins or Startds
Total(Idling+Running) Glideins/Startds per Entry(and its limit) = 2(10000)
All Idling-----Glideins/Startds per Entry(and its limit) = 0(100)
3. Information on Frontend-centric Glideins or Startds
Total(Idling+Running) Glideins/Startds per Group(and its limit) = 4(100000)
All Idling-----Glideins/Startds per Group(and its limit) = 0(1000)
Total(Idling+Running) Glideins/Startds per Frone(and its limit) = 4(100000)
All Idling-----Glideins/Startds per Frone(and its limit) = 0(1000)
Total(Idling+Running) Glideins/Startds per Globa(and its limit) = 4(100000)
All Idling-----Glideins/Startds per Globa(and its limit) = 0(1000)

#8 - 09/15/2015 07:32 PM - HyunWoo Kim

- % Done changed from 80 to 90

The above codes have been modified to advertize only those limits and curbs that have been exceeded. Changes have been pushed to the main repository for review.

#9 - 09/23/2015 10:31 PM - HyunWoo Kim

I did some more testings to see how the new code works

In /etc/gwms-frontend/frontend.xml, I modified the following 3 max

```
<frontend advertise_delay="
<config>
<running_glideins_total curb="90000" max="2"/> per frontend
<running_glideins_total_global curb="90000" max="3"/> per global

<lt;groups>&gt;
  <lt;group name="main" enabled="True"&gt;&gt;
    <lt;config>&gt;
      <lt;running_glideins_total curb="90000" max="4"/&gt;&gt; per group
```

My internal log shows

Total(Idle+Running) Glideins/Startds per Group(and its limit) = 4(4)
Total(Idle+Running) Glideins/Startds per Frone(and its limit) = 4(2)
Total(Idle+Running) Glideins/Startds per Globa(and its limit) = 4(3)

```
[root@fermicloud159 frontend]# condor_status -any -l -constraint 'MyType=="glideresource" | grep Dummy
TotalGlideinsPerGroup   _HasBeenTriggered_NotRequestingMoreGlideins = "Dummy"
TotalGlideinsPerFrontend _HasBeenTriggered_NotRequestingMoreGlideins = "Dummy"
TotalGlideinsGlobal     _HasBeenTriggered_NotRequestingMoreGlideins = "Dummy"
```

At this point, there were 4 running glideins and no more glideins were requested because the above 3 limits have been exceeded.

So, we can get some useful information from glideresource classad attributes..

I will talk with Parag tomorrow and proceed to "Feedback" status if this is satisfactory.

I will create a sub-ticket for a similar code for Factory..

#10 - 10/02/2015 03:36 PM - HyunWoo Kim

- Tracker changed from Bug to Feature
- Status changed from New to Feedback
- Assignee changed from HyunWoo Kim to Parag Mhashilkar

I think it is time to move this ticket to a feedback status

#11 - 10/09/2015 03:36 PM - Parag Mhashilkar

- Assignee changed from Parag Mhashilkar to HyunWoo Kim

Adding value "Dummy" to value is not useful and not a good practice. You can easily make this classad attribute more useful by assigning useful value.

Here is an example,

```
if (count_status['Total'] >= self.max_running):
    which_limits_triggered['TotalGlideinsPerEntry'] = 1
```

Change this to

```
if (count_status['Total'] >= self.max_running):
    limits_triggered['TotalGlideinsPerEntry'] = 'count=%i, limit=%i' % (count_status['Total'], self
.max_running)
```

Other comments

- Shorten classad attribute name
 - Example: foo*_HasBeenTriggered_NotRequestingMoreGlideins -> GlideResource_Limit_foo*
 - Example: Curb*_HasBeenTriggered_GlideinsRequestsReduced -> GlideResource_Curb_*. For values of Curbed attributes you can have 'count=%i, limit=%i, curbed='
- Remove extra spaces from the comparisons in the if statements

#12 - 10/26/2015 10:43 AM - Parag Mhashilkar

- Stakeholders updated

#13 - 11/05/2015 12:59 PM - HyunWoo Kim

- Status changed from Feedback to Resolved
- % Done changed from 90 to 100

I have merged this branch into branch_v3_2
I am closing this ticket.

#14 - 01/19/2016 10:09 AM - Parag Mhashilkar

- Status changed from Resolved to Closed

#15 - 06/22/2016 03:42 PM - Parag Mhashilkar

- Related to Feature #7826: Need additional debugging information in the glideresource classad added

#16 - 06/22/2016 03:43 PM - Parag Mhashilkar

- Related to deleted (Feature #7826: Need additional debugging information in the glideresource classad)

#17 - 06/22/2016 03:43 PM - Parag Mhashilkar

- Has duplicate Feature #7826: Need additional debugging information in the glideresource classad added