

Network Tools - Feature #1790

Add support for cron scheduling to modular input base class

03/17/2017 06:18 PM - Luke Murphey

Status:	Rejected	Start date:	03/17/2017
Priority:	Normal	Due date:	
Assignee:	Luke Murphey	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
Related issues:			
Related to File Meta-data Monitoring - Feature #1095: Use cron type frequency...		New	10/28/2015
Related to Network Tools - Feature #1731: Speedtest modular input		Closed	02/06/2017

History

#1 - 03/17/2017 06:18 PM - Luke Murphey

- Blocks Feature #1731: Speedtest modular input added

#2 - 03/17/2017 06:19 PM - Luke Murphey

Needs:

- Ability to parse cron schedule
- Ability to derive when the schedule indicates a run should happen

#3 - 03/17/2017 06:37 PM - Luke Murphey

Some libraries:

<https://github.com/josiahcarlson/parse-crontab>

- Two files
- Don't see a way to parse

<https://github.com/zibera/py-scheduler>

- Not a single file
- Can take a string in parse() and run the default function

<https://pypi.python.org/pypi/croniter>

- Allows the designation of the current time
- Used in REST API mod input

<https://github.com/dbader/schedule>

- Not a cron scheduler

<https://github.com/ahawker/crython>

- Seems a little heavy
- Not a single file

<http://code.activestate.com/recipes/577466-cron-like-triggers/>

- Might be a good fit
- Not sure if it needs a complete cron or just a time
- Not sure how to run missed inputs

<https://pypi.python.org/pypi/python-crontab>

#4 - 03/17/2017 06:41 PM - Luke Murphey

Could make the cron schedule readable with https://pypi.python.org/pypi/cron_descriptor/1.2.0

#5 - 03/24/2017 06:57 AM - Luke Murphey

This one looks good: <https://pypi.python.org/pypi/croniter/>

#6 - 03/24/2017 06:58 AM - Luke Murphey

- Related to Feature #1095: Use cron type frequency definition added

#7 - 04/14/2017 06:04 AM - Luke Murphey

- Target version changed from 0.8 to 0.10

#8 - 04/14/2017 06:51 AM - Luke Murphey

- Blocks deleted (Feature #1731: Speedtest modular input)

#9 - 04/14/2017 06:52 AM - Luke Murphey

- Related to Feature #1731: Speedtest modular input added

#10 - 05/04/2017 05:54 AM - Luke Murphey

- Target version changed from 0.10 to 1.0

#11 - 05/09/2017 07:50 PM - Luke Murphey

I think the algorithm could work like this:

1. Every minute, check to see if the input should run: `job.check_trigger()`
2. If so, execute

Problems to be solved:

- What happens when the search has missed execution
 - I think we should do nothing
- Should this only be used with multi-threaded, or multi-instance mode since otherwise a flooded system may skip inputs?
 - Probably
- Should the check time be made more frequent? 15 seconds means that only 4 inputs can be executed per minute.

#12 - 05/10/2017 05:33 PM - Luke Murphey

<https://splunkbase.splunk.com/app/1546/> uses croniter

#13 - 05/11/2017 07:23 AM - Luke Murphey

- Status changed from New to Rejected

- Target version deleted (1.0)

#14 - 05/16/2017 09:43 PM - Luke Murphey

Dependencies:

- crontab
 - croniter

- six
- dateutil