

## GlideinWMS - Support #22470

### Use of Python constructs when possible to help context managers to properly manage resources.

04/29/2019 05:26 PM - Lorena Lobato Pardavila

<b>Status:</b>	Feedback	<b>Start date:</b>	08/23/2019
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Lorena Lobato Pardavila	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	v3_6_1	<b>Spent time:</b>	0.00 hour
<b>Stakeholders:</b>			
<b>Description</b>			
<p>We can improve the performance (and reduce code) with the <b>with Statement</b> and Context Managers when files are being managed. Currently, close() is called on every file opened, and some times the file is in a function that may raise an exception or has multiple return paths, using for this try...except.</p> <p>We could use 'with' statement that clarifies code that previously would use try...finally blocks to ensure all resources we use are properly cleaned up, regardless of if the code returns or an exception is thrown: context managers.</p> <p>For example:</p>			
<pre># old file exists, check if same content fl = open(fname, 'r') try:     old_data = fl.read() finally:     fl.close()</pre>			
<p>We could do :</p>			
<pre>with open(fname,'r') as fl     fd.read()</pre>			
<p>and not needed to close the files and the block is calling it automatically.</p>			
<p>To DO: Review the code and do the replacement where this case is satisfied.</p>			
<p>Note: It works with open(). But not with os.open() as it is just a wrapper for the lower-level POSIX syscall. It returns the file descriptor (a number) that represents the opened file and not a file object: so we cannot use it in this case.</p>			
<b>Subtasks:</b>			
Support # 23166: Apply the TODOs from #22470 related to "with statement" compatibility			<b>New</b>

## History

### #1 - 04/30/2019 04:39 PM - Lorena Lobato Pardavila

- Target version set to v3\_5

### #2 - 05/20/2019 12:55 PM - Lorena Lobato Pardavila

- Target version changed from v3\_5 to v3\_5\_1

### #3 - 06/04/2019 01:59 PM - Lorena Lobato Pardavila

- Status changed from New to Work in progress

- Tracker changed from Idea to Support

### #4 - 07/01/2019 03:56 PM - Lorena Lobato Pardavila

Most of the changes were straightforward but there are two cases that we must have in consideration.

1. Some parts of the changes are using file() [built-in constructor] instead of open().

Example:

```
glidein_params_to_encrypt={}
```

```
fd=file(tmpname, "a")
nr_credentials=len(self.x509_proxies_data)
if nr_credentials>0:
    glidein_params_to_encrypt['NumberOfCredentials']="%s"%nr_credentials
```

Basically, works the same way and open has been the canonical way to open a file (and hence create a file instance) for a long time, even on Python 2.x. As the documentation states it's preferable to use open() instead of invoking this constructor directly. So, when doing the changes for "with statement", I propose to replace file() by open() at the same time.

1. The other part to have in consideration is when catching exceptions, especially they are nested.

#### **#5 - 07/10/2019 03:55 PM - Lorena Lobato Pardavila**

- Status changed from Work in progress to Feedback

it turned out that a lot of files have been changed requiring a deep review (some of the replacement are tricky and it might need discussion) from a caring soul ☹☹.

I have proposed to review all the files in the next code review exposing the changes that I made and discuss them. Otherwise, if someone volunteers, I will assign the ticket to him for feedback.

#### **#6 - 07/31/2019 10:10 AM - Lorena Lobato Pardavila**

- Target version changed from v3\_5\_1 to v3\_6\_1

After discussion, this will be reviewed in the next code review. It will be around mid- August

#### **#7 - 08/23/2019 04:38 PM - Lorena Lobato Pardavila**

- Start date changed from 04/29/2019 to 08/23/2019

- Due date set to 08/23/2019

due to changes in a related task: [#23166](#)

#### **#8 - 09/30/2019 10:42 AM - Lorena Lobato Pardavila**

- Target version changed from v3\_6\_1 to v3\_6\_2

#### **#9 - 10/14/2019 05:05 PM - Lorena Lobato Pardavila**

- Assignee deleted (Lorena Lobato Pardavila)

#### **#10 - 10/14/2019 05:05 PM - Lorena Lobato Pardavila**

- Target version changed from v3\_6\_2 to v3\_6\_1

- Assignee set to Lorena Lobato Pardavila