

---

Subject: Re: Passing Batch File Names

Posted by [Trae](#) on Wed, 03 Sep 2008 19:18:14 GMT

[View Forum Message](#) <> [Reply to Message](#)

---

> Your solution is very handy, but one should point out that blind execution of commands  
> read from a file is potentially dangerous. It's probably not that big of an issue for IDL,  
> but some of those batch file lines could be something like,

Oh yeah. I would never do this in perl or anything that had write permissions in critical areas. The batch files made would be few in number and only made a handful of people.

Still, it just became easier to change the structure of the previously written codes then to deal with all of the issues, this being a big one, that came about by trying to use batch files.

Good comment.

-Trae

On Sep 2, 2:16 pm, Paul van Delst <Paul.vanDe...@noaa.gov> wrote:

> Trae wrote:

>> Just in case another poor soul needs this and searches the group, here  
>> are the lines of code that made it work. NB: EXECUTE only accepts  
>> scalars as an input. Bogus.

>  
> Your solution is very handy, but one should point out that blind execution of commands  
> read from a file is potentially dangerous. It's probably not that big of an issue for IDL,  
> but some of those batch file lines could be something like,

>  
> stuff = file\_search('\*')  
> for i=0,n\_elements(stuff)-1 do file\_delete, stuff[i], /quiet,/recursive

>  
> Keep those batch files protected from disgruntled employees. :o)

>  
> Using exec capabilities blindly is generally discouraged in scripting languages.

>  
> cheers,

>  
> paulv

>  
>  
>  
>> First define a batch file named batch\_test.pro with the commands

>  
>> a=5

```
>> b=6
>> c=sin(!dpi/4.)
>
>> Now here are the commands to read and execute it.
>
>> file='batch_test.pro'
>
>> openr, lun, file, /GET_LUN
>> var_arr=""
>> ;while not eof do begin
>
>> WHILE ~ EOF(lun) DO BEGIN
>>   line=""
>>   READF,lun , line
>>   var_arr=[var_arr,line]
>> ENDWHILE
>> FREE_LUN, lun
>> var_arr=var_arr[1:~]
>
>> for i=0ul,n_elements( var_arr) -1ul do result=execute(var_arr[i])
>
>> END
>
>> So you can define the variable 'file' anyway you want and it can have
>> as many or as few commands as you need.
>
>> Thanks for the help everyone!
>
>> I'm off to be quite pleased with myself for awhile! :)
>
>> Cheers,
>> -Trae
```

---