NOTE: SCENARIO IS WHAT THE PUZZLE TAKER SEES.

SCENARIO:

Consider the Python function below that converts a user-specified video file into an AVI format. The `subprocess.call` function allows the user to run the command described by the arguments. When the `shell` argument is `False`, the program is started directly, and when it is `True`, the program is executed through the shell. Assume that the computer running this program has `ffmpeg` (a free software to convert videos to different formats used in line 7) installed. Considering the snippet of code below, answer the following questions, assuming that the code has all required permissions to execute.

```
01   import subprocess
02
03   def transcode_file():
04      filename = raw_input('File to transcode: ')
05
06      # file conversion
07      command = 'ffmpeg -i  %s output.avi' % (filename)
08      subprocess.call(command, shell=True)
```

Questions:
1. What will this function do when executed?

2. What will happen when the program executes if user enters the following as input:
`input.mp4 output.wmv;date;`
a. The Python program will crash.
b. If the `input.mp4` file exists, the program will produce a file named `output.avi`.
c. The program will produce same outputs, regardless of the `input.mp4` file's existence.
d. No conversion will take place because the program will not be able to interpret the input string as a valid file name.
e. None of the above.

[Other statistical questions will be imported here while creating the survey.]

NOTE: ANSWER IS TO BE SHOWN TO THE PUZZLE TAKER AT THE END OF THE SESSION.

ANSWERS:

1. The function converts a valid video file to an AVI file with the name `output.avi`. The inputs to the `subprocess.call` are the command string from the variable command and `shell=True`.

2. e
Since, the shell parameter was set to true, a user can enter commands as if they were interacting directly with the command line. If the `input.mp4` file exists, the `ffmpeg` tool will convert `input.mp4` to a WMV file and store it as `output.wmv`, then the `date` command will be executed, and at the end a message will be printed indicating "`output.avi` command not found".

NOTE: THE REST OF THIS DOCUMENT CONTAINS EXTRA INFORMATION FOR THE PROJECT RESEARCHERS. IT IS NOT TO BE SHOWN TO THE PUZZLE TAKERS.

TAGS:
python, shell, code-injection

CATEGORIES:
Blindspot - YES
Type - Injection
Number of distinct functions - 3
Number of total functions - 3
Blindspot function - `call()`
Function call omitted - NO
Blindspot type - Function misuse
Number of parameters in the blindspot function - 2 parameter
Cyclomatic complexity - 1

NAME:
subprocess.call(args, *, stdin=None, stdout=None, stderr=None, shell=False)

DESCRIPTION:
The command line arguments are passed as a list of strings, which avoids the need for escaping quotes or other special characters that might be interpreted by the shell. The return value from the `call` function is the exit code of the program. The caller is responsible for interpreting it to detect errors.

BLINDSPOT:
Executing programs through the shell means that all user input passed to the program is interpreted according to the syntax and semantic rules of the invoked shell. It can cause security leaks, because the user can execute arbitrary programs.

CORRECT USE EXAMPLE:
#N/A

MORE INFORMATION:
#N/A

REFERENCES:
1. http://kevinlondon.com/2015/07/26/dangerous-python-functions.html
2. https://docs.python.org/2/library/subprocess.html