NOTE: SCENARIO IS WHAT THE PUZZLE TAKER SEES.

SCENARIO:

Consider a central repository application that allows users to download files from a specific folder (`REPOSITORY_FOLDER`) on the server. To download a file, a user must give the name (not the path) of the file, and then the application will look for a file with the same name in `REPOSITORY_FOLDER`, and return the file content if it finds one. The following is the implementation of the download feature. In the snippet of code below the `os.path.join` function joins the filename to the repository folder. For example, if the repository folder is "`/home/foo`", then `os.path.join("/home/foo", "bar.txt")` produces "`/home/foo/bar.txt`". Considering the snippet of code below, answer the following questions, assuming that the code has all required permissions to execute.

```
01   import os
02
03   REPOSITORY_FOLDER = "/var/repository/"
04
05   def download(filename):
06     path = os.path.join(REPOSITORY_FOLDER, filename)
07
08     if os.path.exists(path):
09       return open(path).read()
10     else:
11       return None
```

Questions:
1. What will the `download` function do when executed?

2. Which of the following is correct if users are allowed to enter any string value as filename?
a. The function fails only when the given string value as filename is invalid.
b. The function fails only when the file does not exist in the repository folder.
c. The function is able to read any (readable) file on the system.
d. The function is able to read only (readable) files in the repository folder.
e. None of the above

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

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

ANSWERS:

1. The program will attempt to read and return the content of the file based on the given filename.

2. c
Considering the fact that the function does not normalize the given filename in any way, one can access to files outside of the repository folder using slash character or double dot notation. For example, to read the passwd file one must enter '`/etc/passwd`' as filename. The behavior of `os.path.join` is not simple concatenation. If the second argument is an absolute path, the function simply returns it as the result. Even if the given filename is not an absolute path, one can still read files outside of the repository folder using double dot notation.

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

TAGS:
python, io-operation, file-operation

NAME:
os.path.join(path, *paths)

CATEGORIES:
Blindspot - YES
Type - Injection
Number of distinct functions - 4
Number of total functions - 4
Blindspot function - `os.path.join`
Function call omitted - NO
Blindspot type - Validation missing
Number of parameters in the blindspot function - 2 parameters
Cyclomatic complexity - 3

DESCRIPTION:
Join one or more path component intelligently. The return value is the concatenation of path and any members of paths with exactly one directory separator (os.sep) following each non-empty part except the last, meaning that the result will only end in a separator if the last part is empty. If a component is an absolute path, all previous components are thrown away and joining continues from the absolute path component.

BLINDSPOT:
This function allows directory traversal using absolute path or double dot notation.

CORRECT USE EXAMPLE:
#N/A

MORE INFORMATION:
#N/A

REFERENCES:
1.http://lucumr.pocoo.org/2010/12/2http://lucumr.pocoo.org/2010/12/24/common-mistakes-as-web-developer/4/common-mistakes-as-web-developer/