

Shakespeare CTF – Official Writeup

Challenge Name: The False Soliloquy

Category: Misc / Esoteric Programming Language

Difficulty: Medium

Step 1: Executing the Provided Code

The given Shakespeare Programming Language (SPL) source code is executed using the online interpreter available at <https://esolangpark.vercel.app/ide/shakespeare>. Running the code initially produces theatrical dialogue but does not directly reveal the flag.

```
english-Vnglish.
Rehan, A true lover.
Ananya, A beautiful girl.
Dggy, a cartoon of cartoon network.
Harry, The one who always bark.

Act I: Loosing control over toungue.
Scene I: Harry started barking.

[Enter Harry and Rehan]

Harry:
You lying stupid fatherless big smelly disgusting coward! You are the product
of thyself and a pig! You are the product of yourself and a happy flower!
You are the sum of yourself and a big smelly disgusting rotten fatherless pig!
You are the sum of yourself and a happy peaceful little flower!

You are an amazing beautiful brave charming gentle healthy pony!
You are the sum of yourself and a beautiful little mighty gentle peaceful rose!
You are the sum of thyself and a plum!

You are a peaceful noble brave amazing gentle healthy king!
You are the sum of yourself and a beautiful little mighty gentle peaceful rose!
You are the sum of thyself and a plum! You are the sum of thyself and a cute flower!

You are a trustworthy handsome fair fine smooth gentle mighty lord!
You are the sum of thyself and a lying miserable stinking sorry bastard!
You are the sum of thyself and a stupid stinking wolf! You are the sum of yourself
and a pig!

You are a dirty misused miserable rotten snotty worried vile leech!
You are the product of thyself and a toad!
You are the sum of yourself and a stupid stinking snotty infected goat!
You are the sum of yourself and a sweet warm pony! You are the sum of thyself and a pig!

[Exit Rehan]

Scene II: Harry kept barking.
```

Step 2: Normalizing Character Names and Forcing Output

The source code contains unauthorized character names not supported by the Shakespeare interpreter. These names must be replaced with valid characters. Additionally, multiple arithmetic computations are performed internally without printing their results. By inserting the statement *Speak your mind!* multiple times, the hidden values are forced to print.

The screenshot shows the Esolang Park interface with a Shakespeare-themed program. The Code Editor on the left contains a script with various scenes and character lines. The Visualization panel on the right shows character counts for Romeo, Juliet, Ophelia, and Hamlet. The Execution Output panel shows the result of running the program: hacks{H3llo_c1ph3r}.

```
103
104     Scene VIII: Hamlet continues speaking.
105
106 Hamlet:
107 Thou art the product of the square of a beautiful rich king and the sum of a
108 beautiful rich king and the sum of a brave lord and a flower! Speak your mind!
109
110     Scene IX: Hamlet continues speaking.
111
112 Hamlet:
113 Thou art the difference between thyself and the cube of a brave lord! Speak your mind!
114
115     Scene X: Hamlet continues speaking.
116
117 Hamlet:
118 Thou art the sum of a flower and the product of a brave lord and the square of
119 the sum of a beautiful rich king and a flower! Speak your mind!
120
121     Scene XI: Hamlet continues speaking.
122
123 Hamlet:
124 Thou art the product of a brave lord and the sum of thyself and the sum of
125 a beautiful rich king and a brave lord! Speak your mind!
126
127 [Exit Romeo]
128
129
130     Act III: The Final words.
131
132     Scene I: Hamlet last words.
133
134 [Enter Romeo]
135
136 Hamlet:
137 You are the cube of the sum of a beautiful rich king and a flower! Speak your mind!
138
139 [Exeunt]
```

Visualization

The stage is empty

Romeo: 125
Juliet: 123
Ophelia: 0
Hamlet: 0

User Input

Enter program input here...

Execution Output

hacks{H3llo_c1ph3r}

Step 3: Retrieving the Flag

After correcting the character names and adding the required output statements, the program is executed again. The output now clearly reveals the flag.

Flag: hacks{H3llo_c1ph3r}