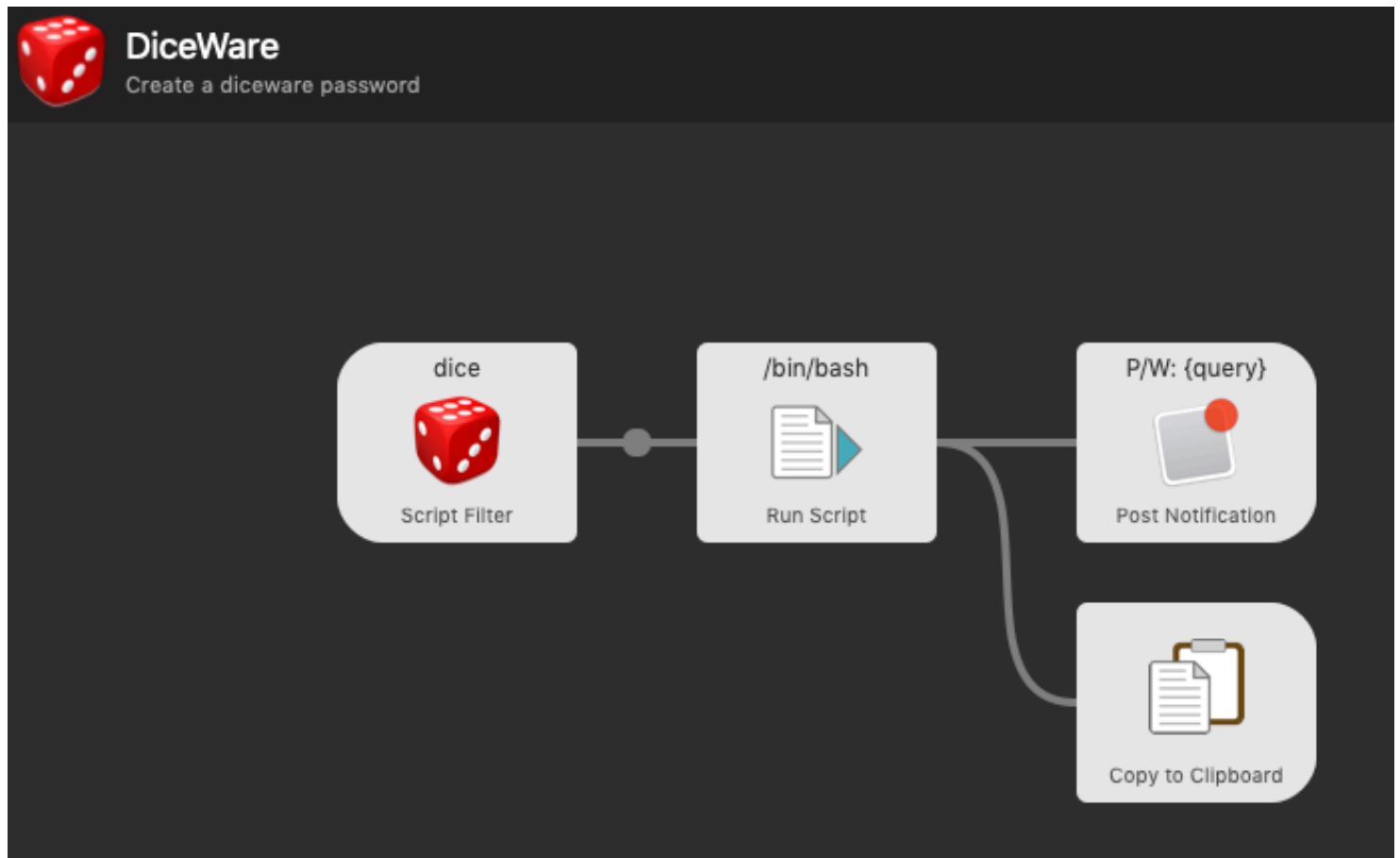


# Diceware password generator - an Alfred workflow

Based upon Alexey Shamrin's [Python diceware password generator](#). Baked into an Alfred workflow and using a script filter to select one of the three to use [EFF](#) lists.



## Script filter

The script filter allows list selection and feeds the argument to the main **Run Script** action.

Keyword:

Placeholder Title:

Placeholder Subtext:

"Please Wait" Subtext:



Drop an icon above.

The details of the placeholder result you will see when typing the keyword. The "Please Wait" subtext will be shown while waiting for the script's initial results.

Language:

☒ Alfred filters results

Script:

```
cat << EOB
{"items": [
  {
    "uid": "123",
    "title": "Diceware passwords - shorter words",
    "subtitle": "e.g., acid, acorn, acre, acts...",
    "arg": "short_word",
    "icon": {
      "path": "icon.png"
    }
  },
  {
    "uid": "456",
    "title": "Diceware passwords - medium words",
    "subtitle": "e.g., abacus, abdomen, abide...",
    "arg": "med_word",
    "icon": {
      "path": "icon.png"
    }
  },
  {
    "uid": "789",
    "title": "Diceware passwords - longer words",
    "arg": "long_word",
    "icon": {
      "path": "icon.png"
    }
  }
]}
```



```
cat << EOB
{"items": [

    {
        "uid": "123",
        "title": "Diceware passwords - shorter words",
        "subtitle": "e.g., acid, acorn, acre, acts...",
        "arg": "short_word",
        "icon": {
            "path": "icon.png"
        }
    },

    {
        "uid": "456",
        "title": "Diceware passwords - medium words",
        "subtitle": "e.g., abacus, abdomen, abide...",
        "arg": "med_word",
        "icon": {
            "path": "icon.png"
        }
    },

    {
        "uid": "789",
        "title": "Diceware passwords - longer words",
        "subtitle": "e.g., aardvark, abandoned, abbreviate...",
        "arg": "long_word",
        "icon": {
            "path": "icon.png"
        }
    }

]}
EOB
```

## Run Script action

---

Invokes the main Python script using the selected word file as the input.

Language:

running instances

Script:

## Python script - generate.py

---

Randomly select four words from the chosen word list.

```
#!/usr/bin/env python
import random, sys

fname = sys.argv[1]

w = [s.split()[1] for s in open(fname + '.txt')]

for i in range(4): print random.SystemRandom().choice(w),
```