

Random Passwords Aren't Good Enough

Rob Yoder // @rauxyo
Weaver of Webs at 1Password

Where we started

oBjyrXnzqqvDnxB

43

/}+

username
superadmin

password
oBj4y/rXn}zq+q3vDnxB

Regenerate Password

Characters

Words

length

20

digits

2

symbols

3

☒ Avoid ambiguous characters

secret

display
Suggest in browser

Where we started

oBj4y/rXn}zq+q3vDnxB

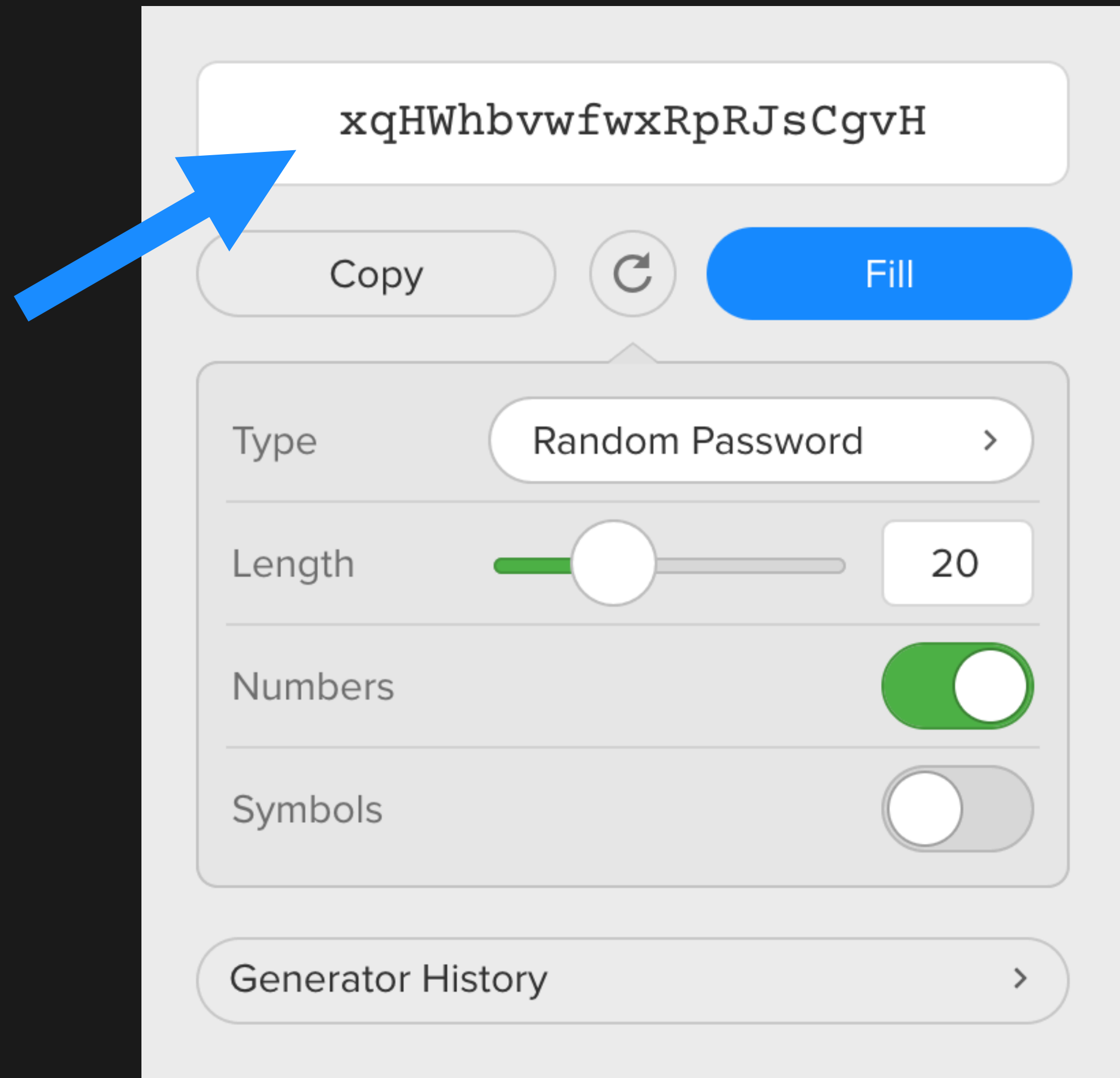
$$H(n) = l \lg |L| + d \lg |D| + s \lg |S| \\ + \lg \binom{l+d}{d} + \lg \binom{l+d+s}{s}$$

The screenshot shows a password generator interface. At the top, it displays the generated password: `username
superadmin
password
oBj4y/rXn}zq+q3vDnxB`. Below this is a green progress bar. A button labeled "Regenerate Password" is present. There are two tabs: "Characters" (selected) and "Words". Under the "Characters" tab, there are three sliders: "length" set to 20, "digits" set to 2, and "symbols" set to 3. A checkbox labeled "Avoid ambiguous characters" is checked. At the bottom, there is a button labeled "secret" and a section for "display" with the option "Suggest in browser" and a dropdown arrow.

Simplifying

- One slider
- Random mix
- Simpler entropy

$$H(n) = n \lg |A|$$



✓ 6 to 10 characters

✓ At least one lowercase letter (a-z)

✓ At least one number (0-9)

✓ No spaces, < or >

Password Must Contain

✗ at least 8 characters

✗ a number

✗ a lowercase letter

✗ an uppercase letter

✗ a special character (e.g. ,@\$%*#?&)

Change Password

Please provide a secure password for your account.

Password must be 7 - 10 characters in length, have at least one number, one special character and is case sensitive.

✗ Accepted symbols: _ ! . & @

✗ Password must be different from

ONE DOES NOT SIMPLY

PICK A RANDOM PASSWORD

There are rules!



16

✓ (e.g. 123456)

✓ (e.g. abcdefgh)

☒ (e.g. ABCDEFGH)

- ✔ (don't begin with a number or symbol)

✓ !"#\$%&'()*+,-./:;<=>?@[]^_`{|}~

- ☑ (don't use characters like i, l, 1, L, o, 0, O, etc.)

✅ (don't use the same character more than once)

- ✓ (don't use sequential characters, e.g. abc, 789)

☒ (generate passwords automatically when you open this page)

1

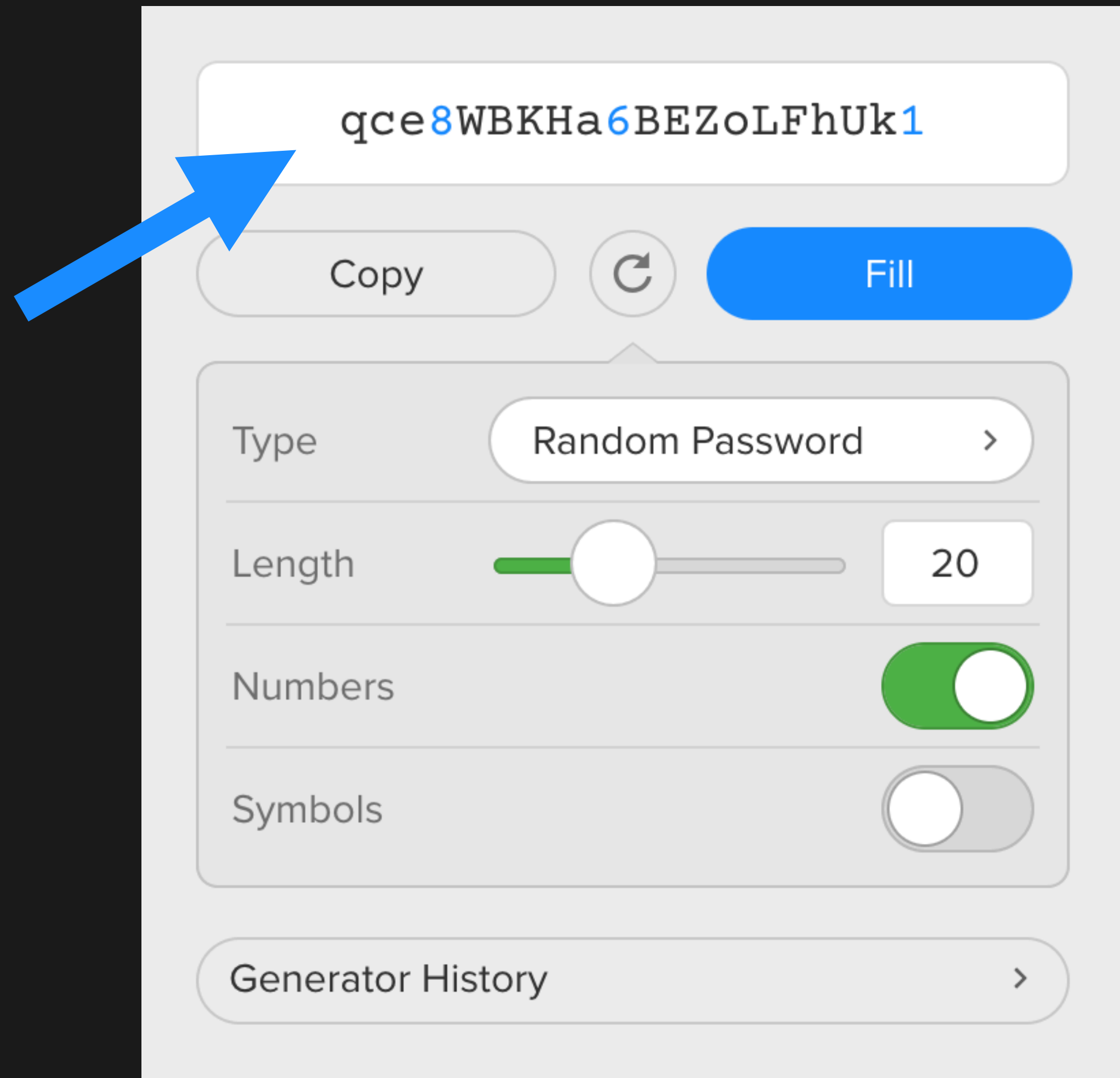
☐ (save all the settings above in cookies)

Generate Passwords

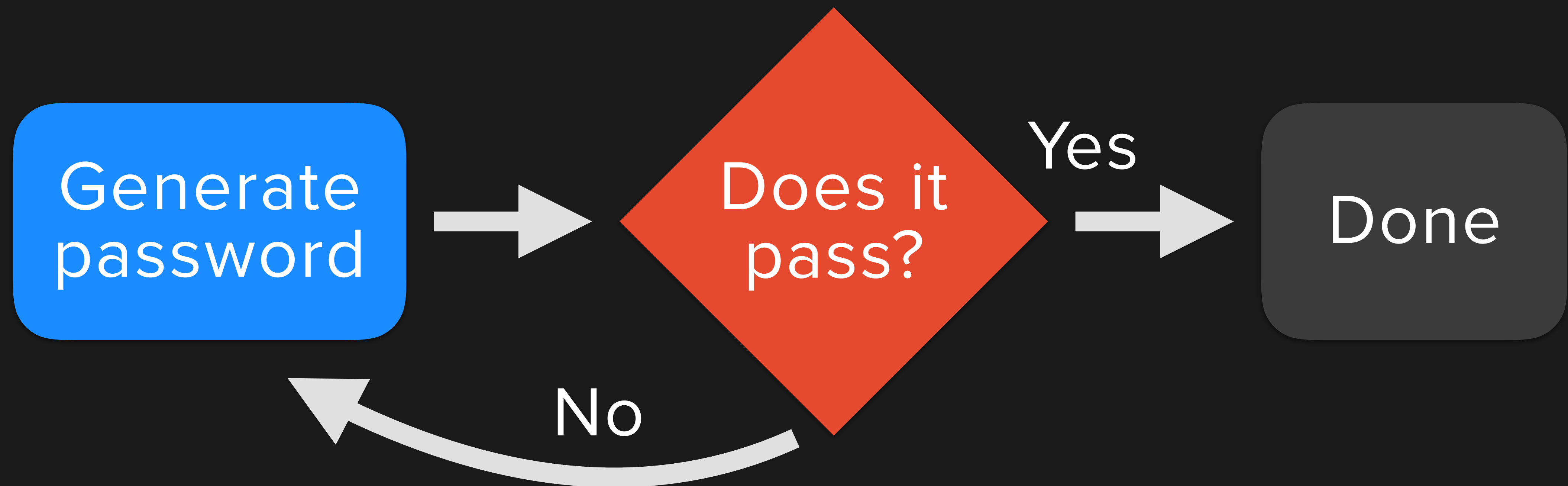
S:5gz>MZ4H&t%,VA

What we want

- Required characters
- Uniform distribution
- Accurate entropy
- Simple interface



The obvious dumb approach



The obvious dumb approach

An equation for entropy is easily derived:

$$H(n) = \lg(|L \cup D|^n - |L|^n - |D|^n)$$

Two drawbacks:

- Generator not guaranteed to terminate
- Entropy function operates on giant integers

I can do better.



Take 1

oBjy/rXnzq+q3vDnxB

4

}

Take 1

oBj4y/rXn}zq+q3vDnxB

Take 1

oBj4y/rXn}zq+q3vDnxB

a1

1

Take 1

oBj4y/rXn}zq+q3vDnxB

 a 1
1 ↑ ↑

Take 1

oBj4y/rXn}zq+q3vDnxB

a11

Take 1

oBj4y/rXn}zq+q3vDnxB

a11

✗ Uniform distribution

Take 2

oBjyr3X}nzqqvD+nxB

Take 2

oBjyrXnzqqvDnxB

3

}+

Take 2

oBjyrXnzqqvDnxB

43

/}+

Take 2

oBj4y/rXn}zq+q3vDnxB

Take 2

oBj4y/rXn}zq+q3vDnxB

a1

Take 2

oBj4y/rXn}zq+q3vDnxB

a

1

Take 2

oBj4y/rXn}zq+q3vDnxB

a

11

Take 2

oBj4y/rXn}zq+q3vDnxB

a11 1a1 11a

Take 2

oBj4y/rXn}zq+q3vDnxB

a11 1a1 11a

aa a1 1a 11

Take 2

aa

a1

1a

11

aa1

a11

a11

111

a1a

1a1

1a1

1aa

11a

11a

Take 2

aa

a1

1a

11

aa1

a11

111

a1a

1a1

1aa

11a

1/12

1/6

1/4

Take 2

oBj4y/rXn}zq+q3vDnxB

a11 1a1 11a

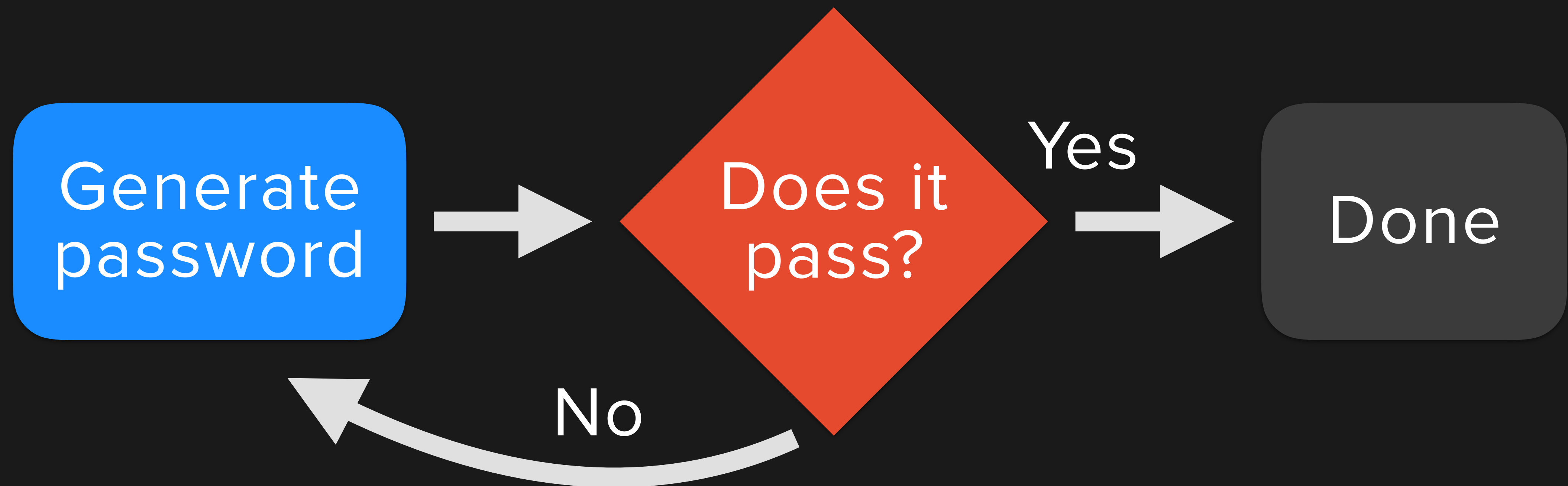
aa a1 1a 11

✗ Uniform distribution

Ok, I give up.



The obvious dumb approach



The obvious dumb approach

Recall our entropy equation:

$$H(n) = \lg(|L \cup D|^n - |L|^n - |D|^n)$$

And the drawbacks:

- Generator not guaranteed to terminate
- Entropy function operates on giant integers

Notation update

Entropy in bits

$$H(n) = \lg(|L \cup D|^n - |L|^n - |D|^n)$$

Number of possible results

$$N(n) = |L \cup D|^n - |L|^n - |D|^n$$

$$H(n) = \lg N(n)$$

What about symbols?

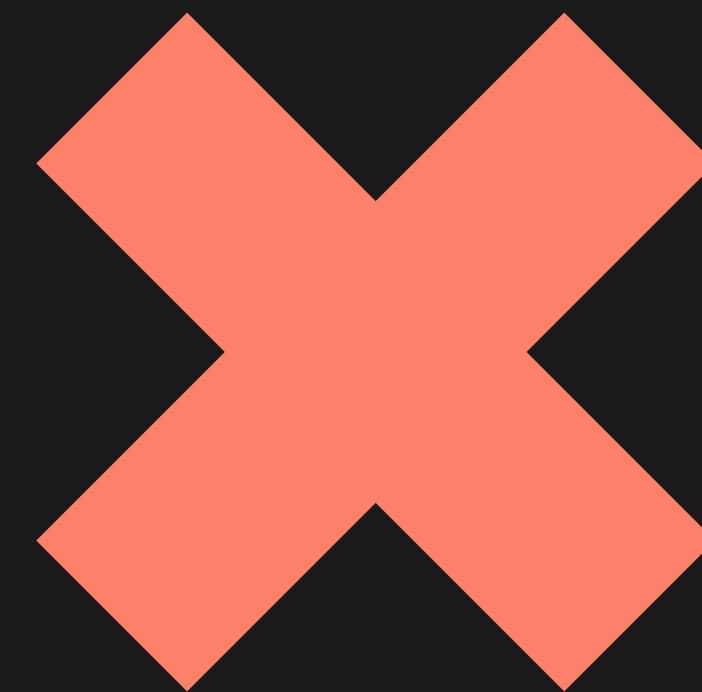
Two required sets:

$$N(n) = |L \cup D|^n - |L|^n - |D|^n$$



Three required sets:

$$N(n) = |L \cup D \cup S|^n - |L|^n - |D|^n - |S|^n \\ - |L \cup D|^n - |L \cup S|^n - |D \cup S|^n$$



What's wrong?

We said...
$$N(n) = |L \cup D \cup S|^n - |L|^n - |D|^n - |S|^n \\ - |L \cup D|^n - |L \cup S|^n - |D \cup S|^n$$

But...
$$L \subseteq L \quad L \subseteq (L \cup D) \quad L \subseteq (L \cup S)$$

We need...
$$N(n) = |L \cup D \cup S|^n - |L|^n - |D|^n - |S|^n \\ - N_{\{L,D\}}(n) - N_{\{L,S\}}(n) - N_{\{D,S\}}(n)$$

Notation update

The set of distinct required character sets

$$\mathcal{R} = \{R_1, R_2, R_3, \dots, R_k\}$$

e.g. $\mathcal{R} = \{\text{"abc"}, \text{"123"}, \text{"!@\#"}\}$

Possible passwords given \mathcal{R} and length n

$$N_{\mathcal{R}}(n)$$

Finding a pattern

$$N_{\emptyset}(n) = 0$$

$$N_{\{R_1\}}(n) = \left| R_1 \right|^n - \left(N_{\emptyset}(n) \right)$$

$$N_{\{R_1, R_2\}}(n) = \left| R_1 \cup R_2 \right|^n - \left(N_{\emptyset}(n) + N_{\{R_1\}}(n) + N_{\{R_2\}}(n) \right)$$

$$\begin{aligned} N_{\{R_1, R_2, R_3\}}(n) = & \left| R_1 \cup R_2 \cup R_3 \right|^n - \left(N_{\emptyset}(n) + N_{\{R_1\}}(n) + N_{\{R_2\}}(n) \right. \\ & \left. + N_{\{R_3\}}(n) + N_{\{R_1, R_2\}}(n) + N_{\{R_1, R_3\}}(n) + N_{\{R_2, R_3\}}(n) \right) \end{aligned}$$

The power set

The set of all subsets of a set

$$\mathcal{P}(\mathcal{R}) = \{x : x \subseteq \mathcal{R}\}$$

$$\mathcal{P}(\{R_1, R_2, R_3\}) = \{\emptyset, \{R_1\}, \{R_2\}, \{R_3\}, \{R_1, R_2\}, \{R_1, R_3\}, \\ \{R_2, R_3\}, \{R_1, R_2, R_3\}\}$$

The set of all proper subsets of a set

$$\{x : x \subset \mathcal{R}\} \rightarrow \mathcal{P}(\mathcal{R}) \setminus \{\mathcal{R}\}$$

To recurse, divine

$$N_{\{R_1, R_2, R_3\}}(n) = \left| R_1 \cup R_2 \cup R_3 \right|^n - \left(N_{\emptyset}(n) + N_{\{R_1\}}(n) + N_{\{R_2\}}(n) \right. \\ \left. + N_{\{R_3\}}(n) + N_{\{R_1, R_2\}}(n) + N_{\{R_1, R_3\}}(n) + N_{\{R_2, R_3\}}(n) \right)$$

$$N_{\mathcal{R}}(n) = \left| \bigcup_{X \in \mathcal{R}} X \right|^n - \sum_{Y \in \mathcal{P}(\mathcal{R}) \setminus \{\mathcal{R}\}} N_Y(n)$$

A final touch

Let A be the set of other allowed characters

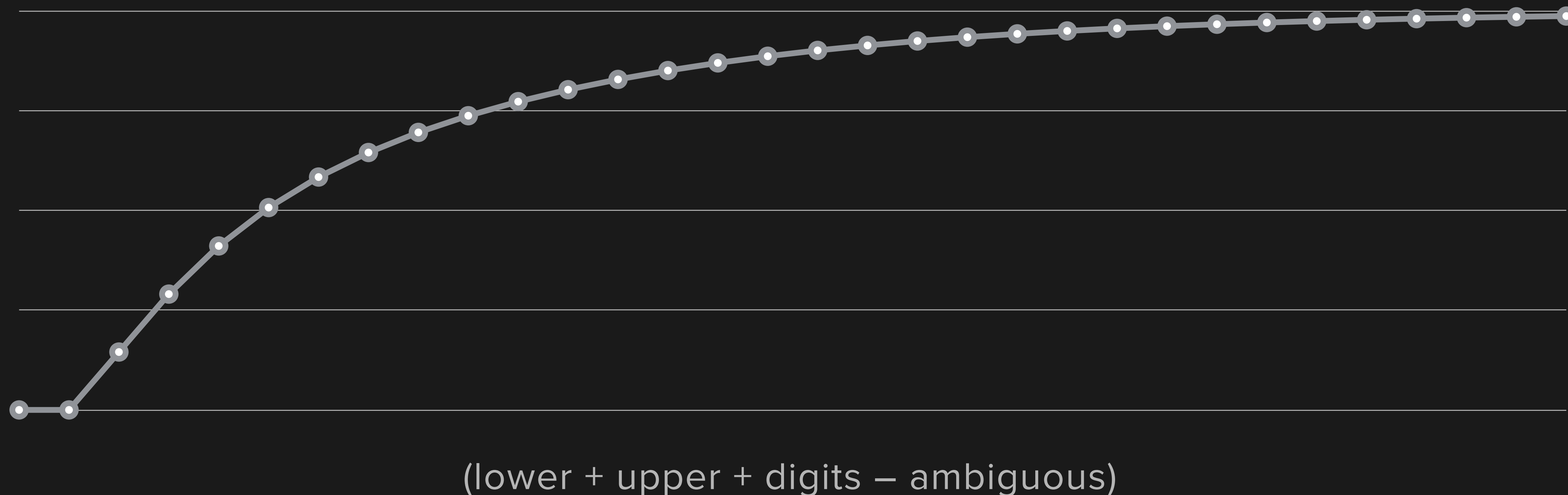
$$N_{\mathcal{R}}^A(n) = \left| \bigcup_{X \in \mathcal{R} \cup \{A\}} X \right|^n - \sum_{Y \in \mathcal{P}(\mathcal{R}) \setminus \{\mathcal{R}\}} N_Y^A(n)$$

$$N_{\emptyset}^A(n) = |A|^n$$

$$H_{\mathcal{R}}^A(n) = \lg N_{\mathcal{R}}^A(n)$$

Considering drawbacks

Probability of keeping a candidate



Considering drawbacks

Probability of keeping a candidate

Size of values in memory (golang)

`big.Int`

`float64`

`math.MaxFloat64` (1024 bits)

What we want

Required characters ✓

Uniform distribution ✓

Accurate entropy ✓

Simple interface ✓

The image shows a user interface for a password generator. At the top, a white text box displays the generated password: "qce8WBKHa6BEZoLFhUk1". Below this, there are three buttons: "Copy", a circular refresh icon, and a blue "Fill" button. A settings panel is open below the buttons, containing four options: "Type" set to "Random Password", "Length" set to 20 with a slider, "Numbers" enabled with a green toggle, and "Symbols" disabled with a grey toggle. At the bottom of the settings panel is a "Generator History" link with a right arrow.

Good enough.



Get it

Code, slides, and demo CLI:

github.com/1password/spg

```
import "go.1password.io/spg"
```

In action:

“1Password X” on the Chrome App Store

1password.com/password-generator