# THE STATE OF CREDENTIAL STUFFING.

Bad bots and the automation war

### JARROD OVERSON

Perpetually frustrated maker of things.



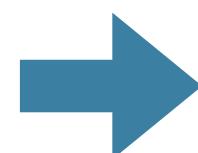




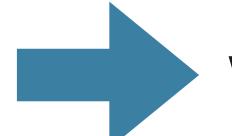


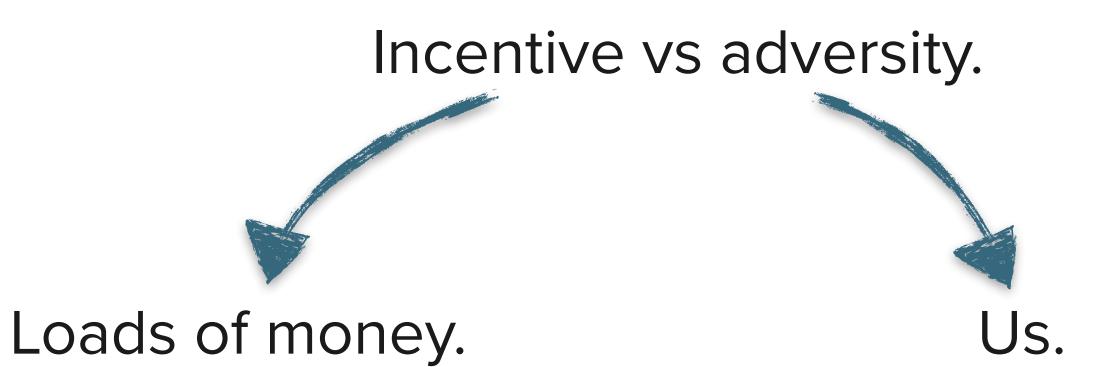
How has automation evolved?

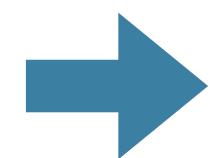
Where do we go from here?



Incentive vs adversity.







Incentive vs adversity.



What does it take?

1. Credentials

Total: \$0

What does it take?

- 1. Credentials
- 2. An account checker

Total: \$0 + \$0

What does it take?

- 1. Credentials
- 2. An account checker
- 3. Proxies

Total: \$0 + \$0 + \$.0005

What does it take?

- 1. Credentials
- 2. An account checker
- 3. Proxies
- 4. CAPTCHA bypasses

Total: \$0 + \$0 + \$.0005 + \$.001

What does it take?

Total: less than  $^2/_{10}$  of one US penny per tested credential.

The break-even rate for accounts worth \$1 is 0.2%

Accounts are regularly worth \$3 - >\$100 At Shape, we saw success rates around 2%

How has automation evolved?

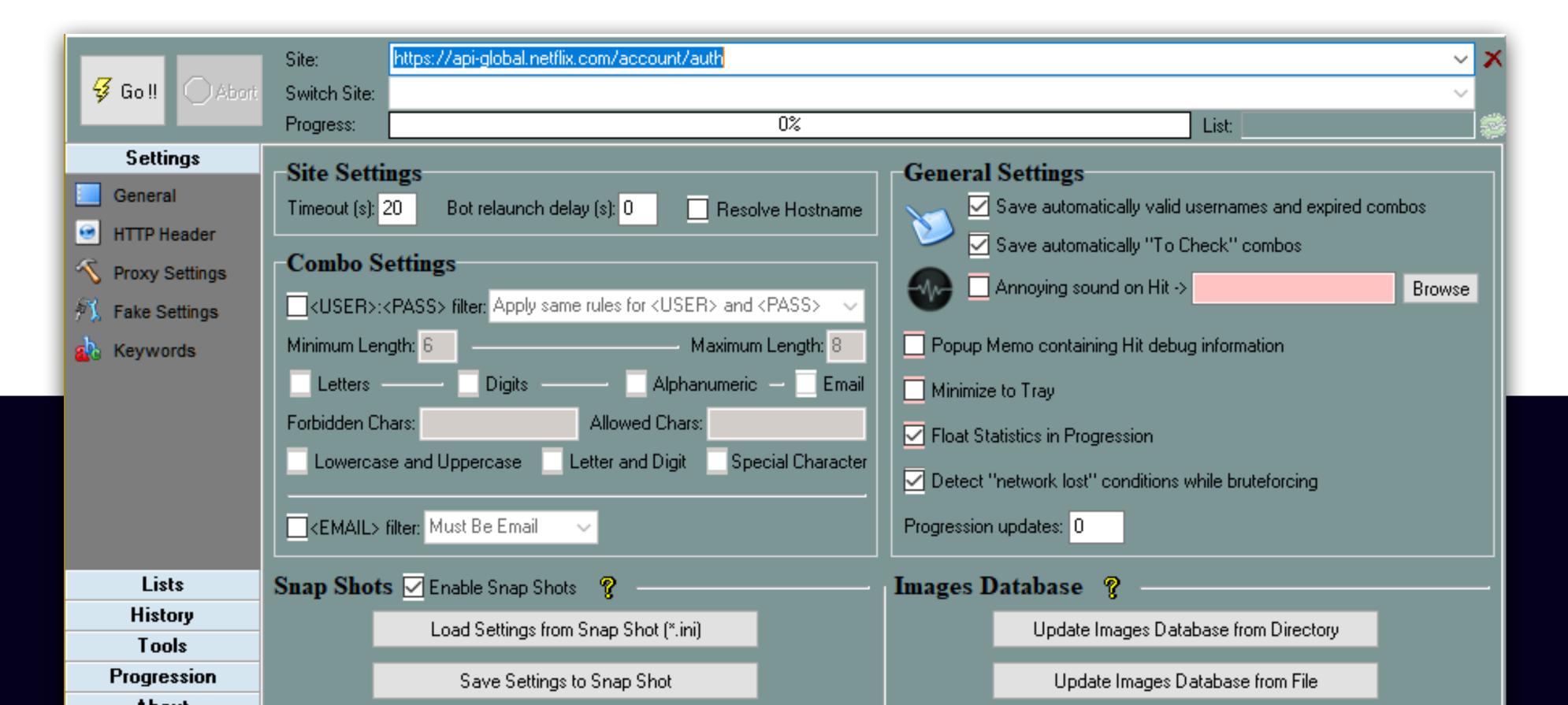
Where do we go from here?

# Generation 0: Basic HTTP requests with common tools

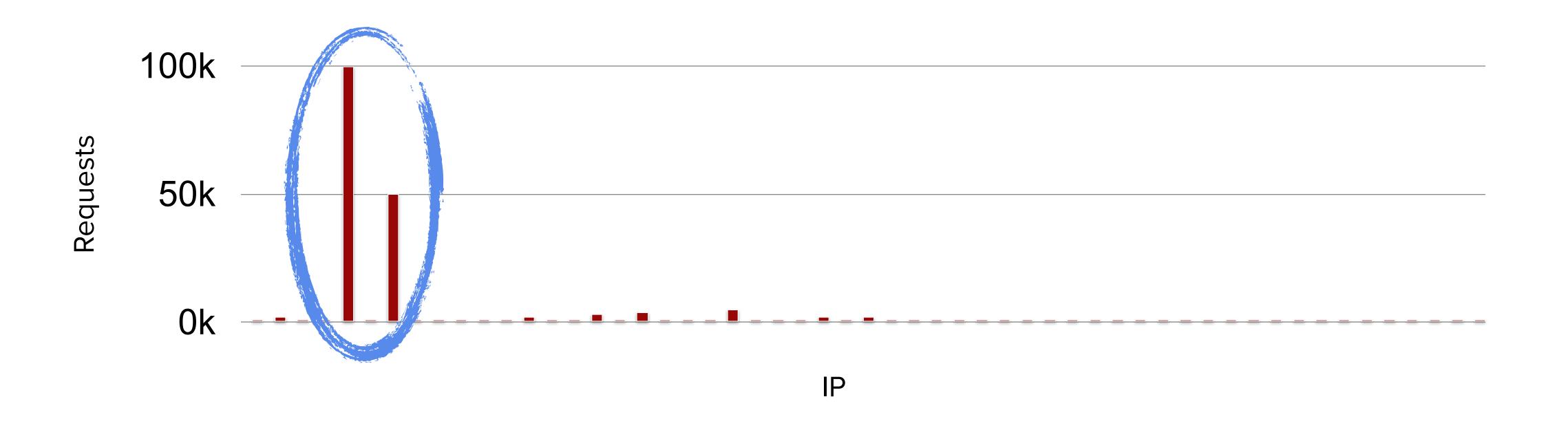


#### SentryMBA

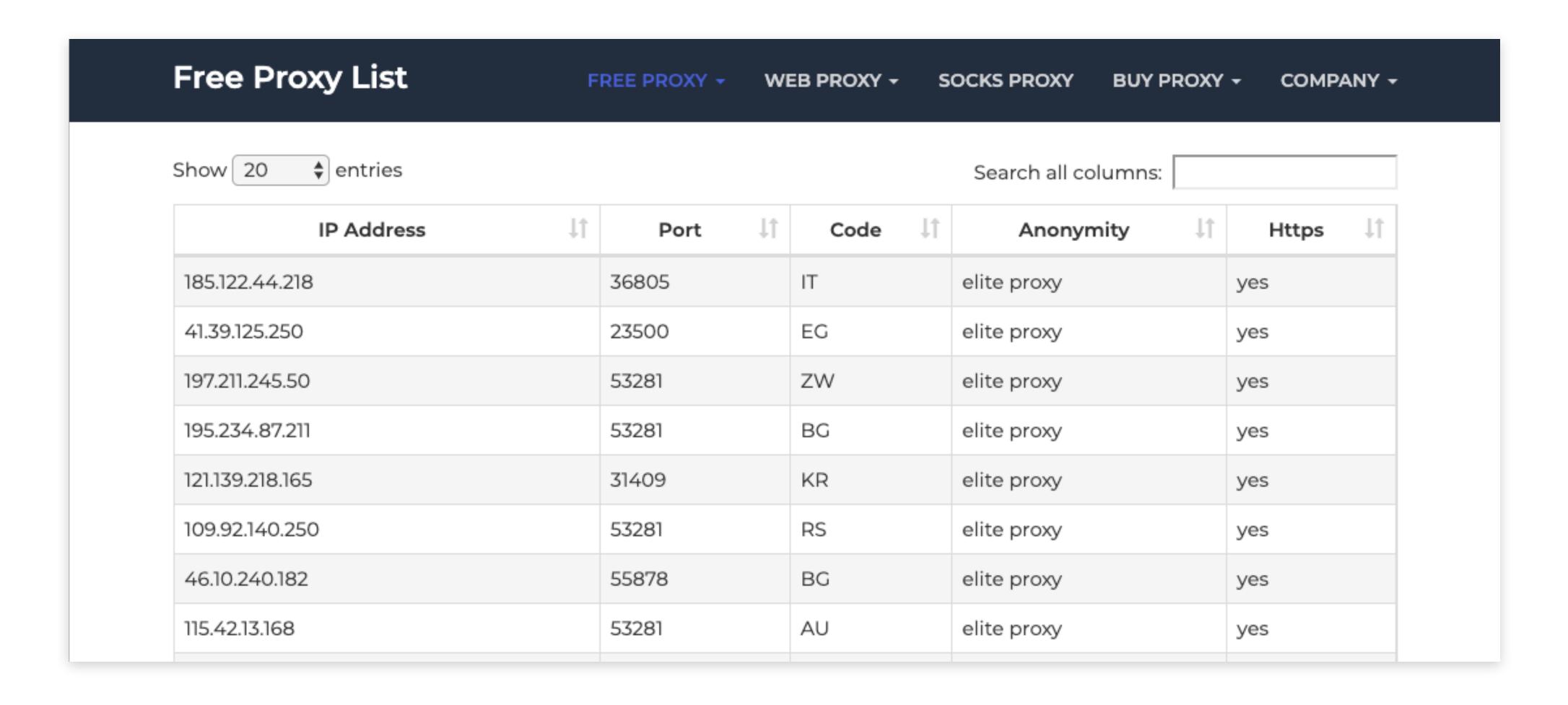
- Performs basic HTTP requests.
- Extensible and highly configurable.
- Tailored towards specific attack use cases.



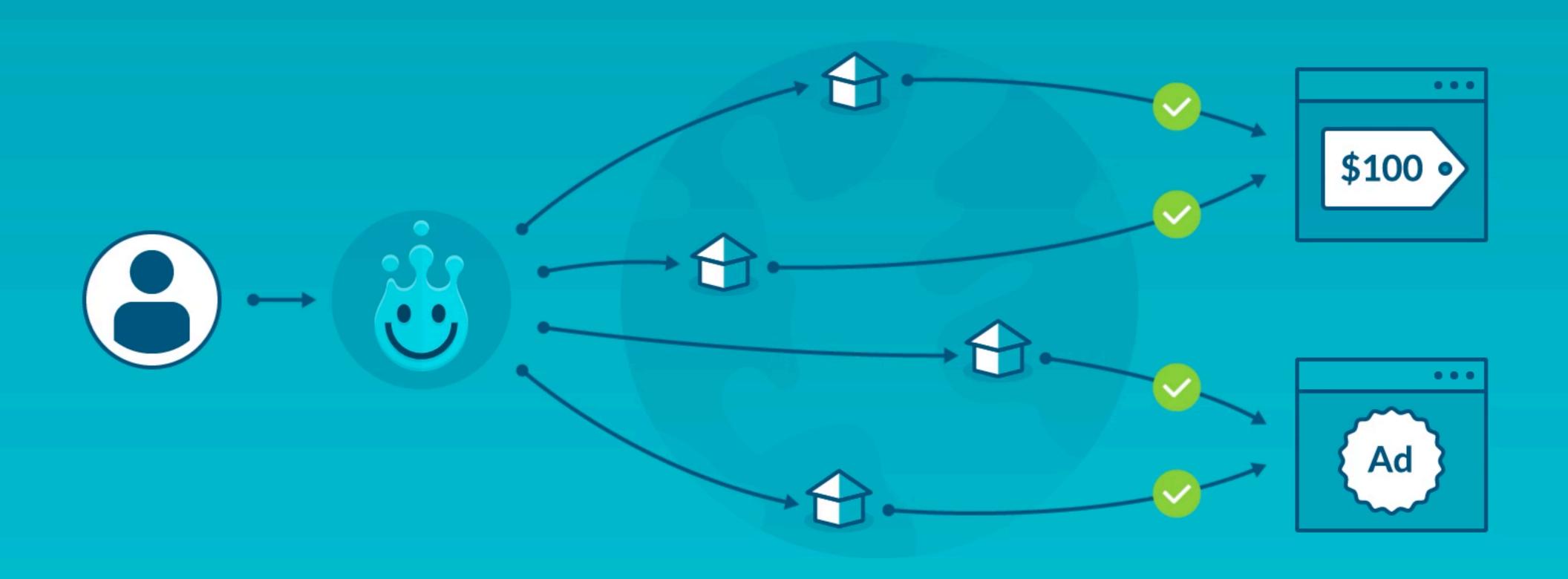
# Early defense: IP Rate limiting.



### Iteration 1: Rotate through proxies

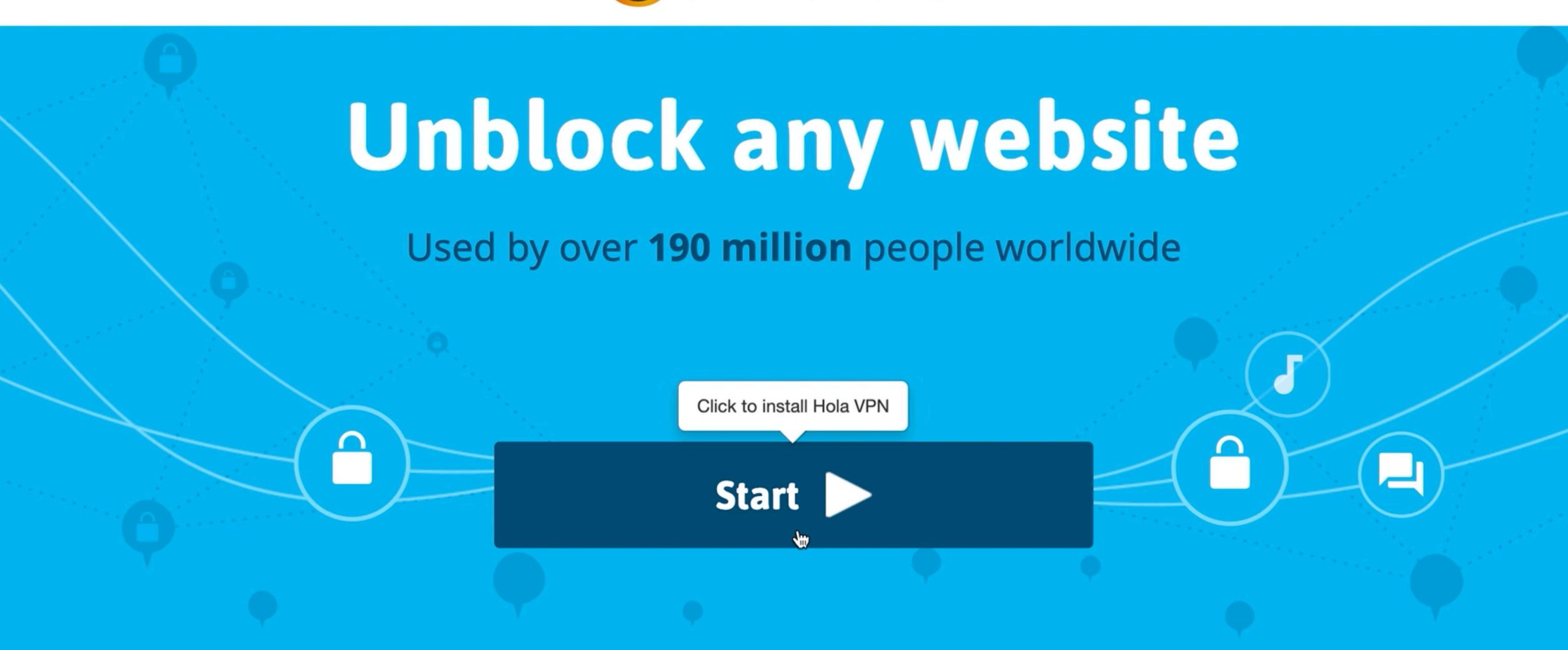


#### Luminati — Residential proxy network



Every country and every city in the world





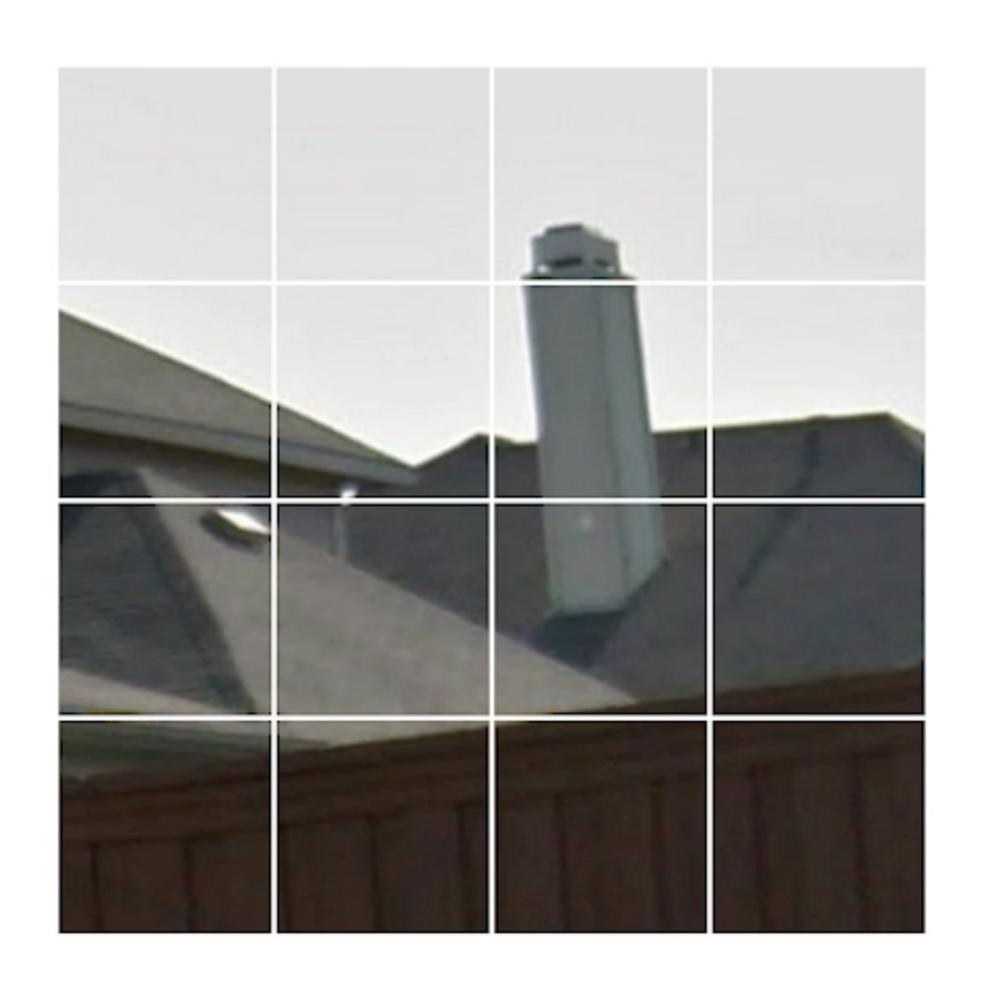
#### Defense: Text-based CAPTCHAs



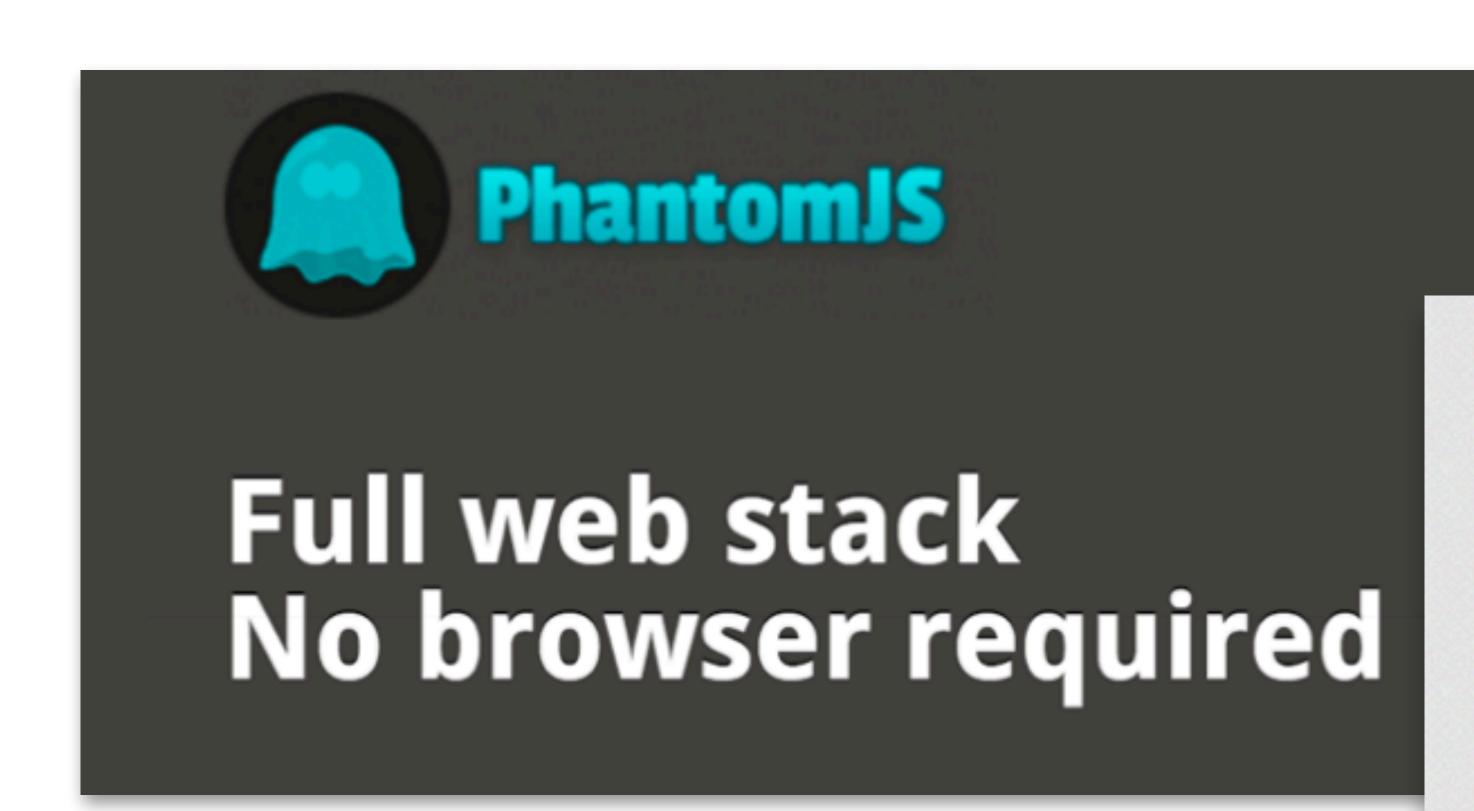
#### **Iteration 2: CAPTCHA Solvers**



# Defense: Dynamic sites and JavaScript heavy defenses.



### Iteration 3: Scriptable WebViews





# Defense: Header Fingerprinting & Environment Checks

```
Host: Dealhost:1337

Correction: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/
*;q=0.8
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5) AppleWebKit/537.
(KHTML, like Gecko) Chrome/39.0.2171.95 Safari/537.36
Accept-Encoding: gzip, deflate, sdch
```



GET / HTTP/1.1

```
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X) AppleWebKit/534.34 (KHT like Gecko) PhantomJS/1.9.8 Safari/534.34
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 Connection: Keep-Alive Accept-Encoding: gzip Accept-Language: en-US,*
Host: Dcalhost:1337
```

### Modern Era

# Iteration 4: Scriptable Consumer Browsers



#### Playwright & Puppeteer

Playwright and puppeteer are automation drivers for modern browsers.

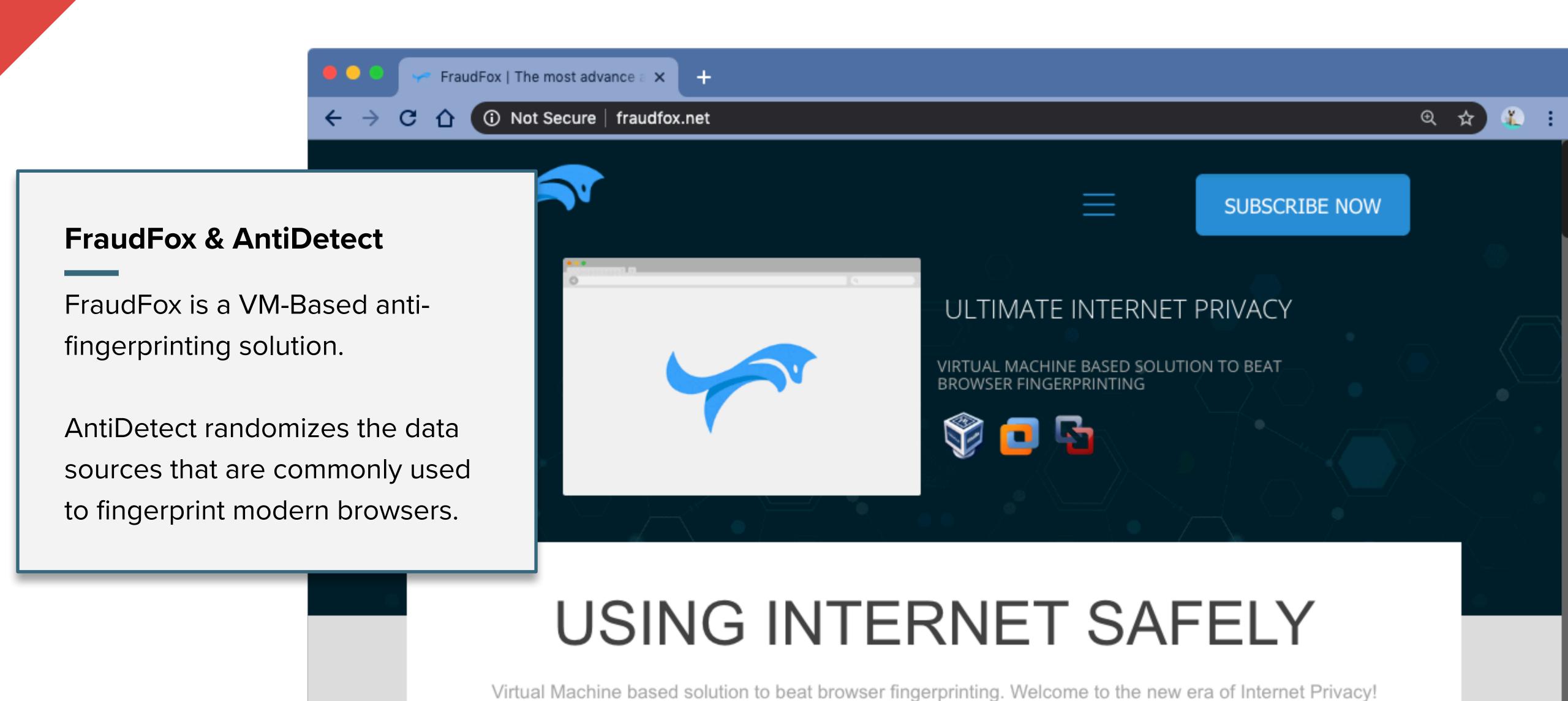
### Defense: Browser Fingerprinting



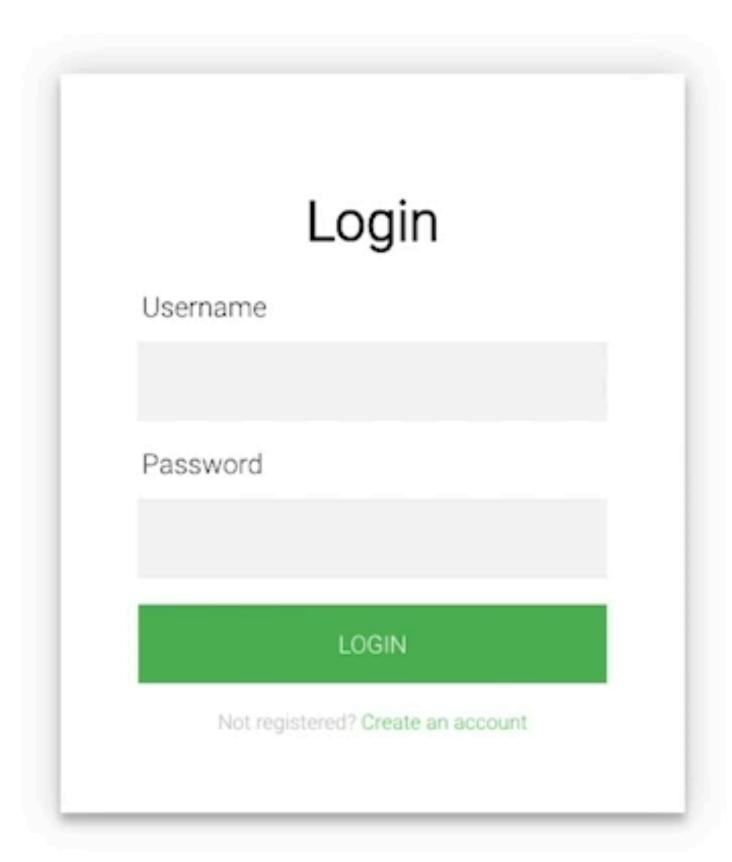
#### **Browser Fingerprinting**

Collecting data points like screen size, fonts, and plugins can produce an effective fingerprint.

### Iteration 5: Randomizing Fingerprint Data Sources



# Defense: Behavior Analysis for Negative Traits



#### **Behavior Analysis**

Bots always click in the same spot, humans don't.

Bots type with lightning speed, humans don't.

Basic behavior analysis catches all simple bots.

#### Iteration 6: Human Behavior Emulation

Store

Gmail Images



#### **Browser Automation Studio**

BAS is an automation tool that combines CAPTCHA solving, proxy rotation, and emulated human behavior.

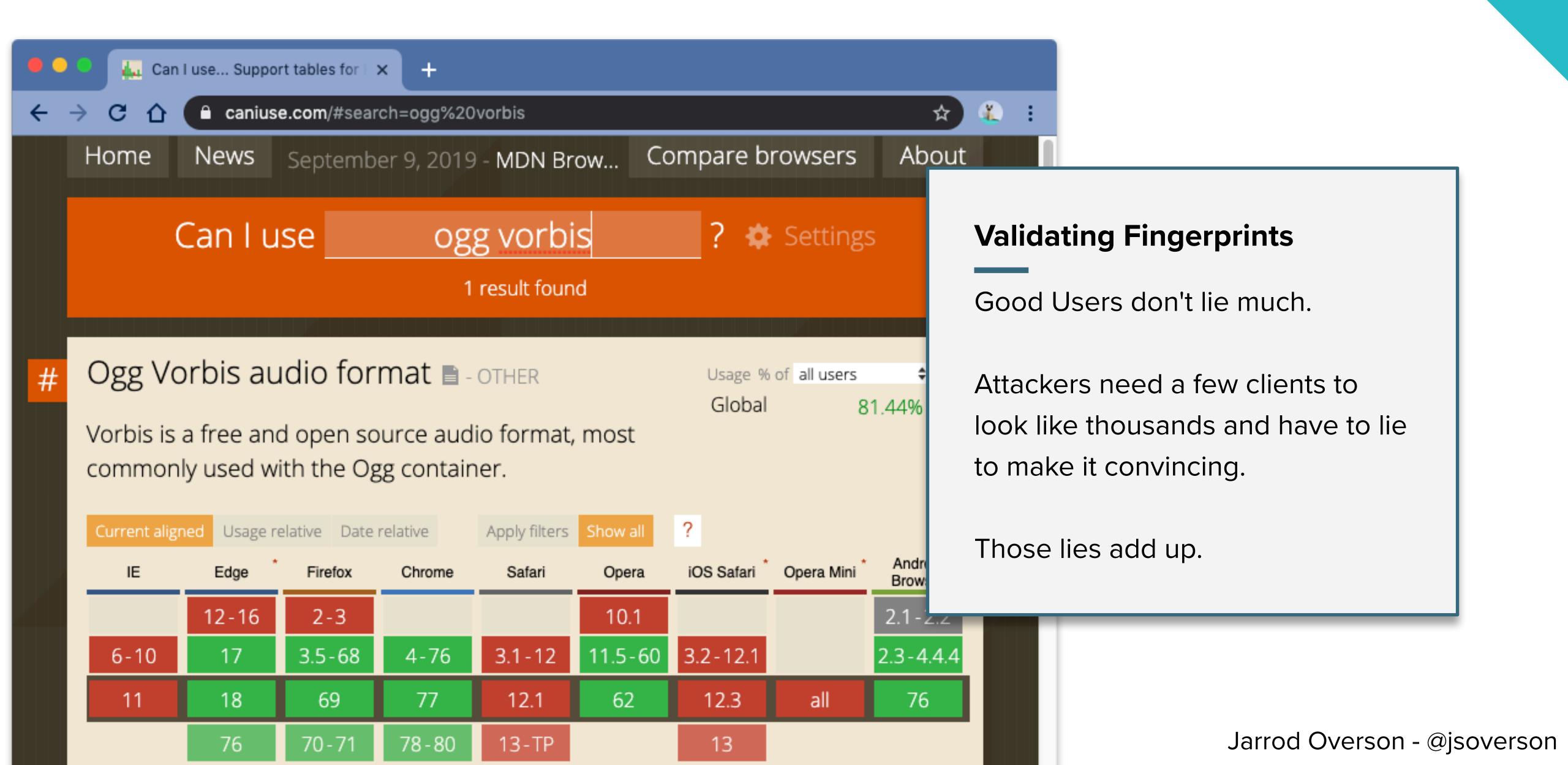




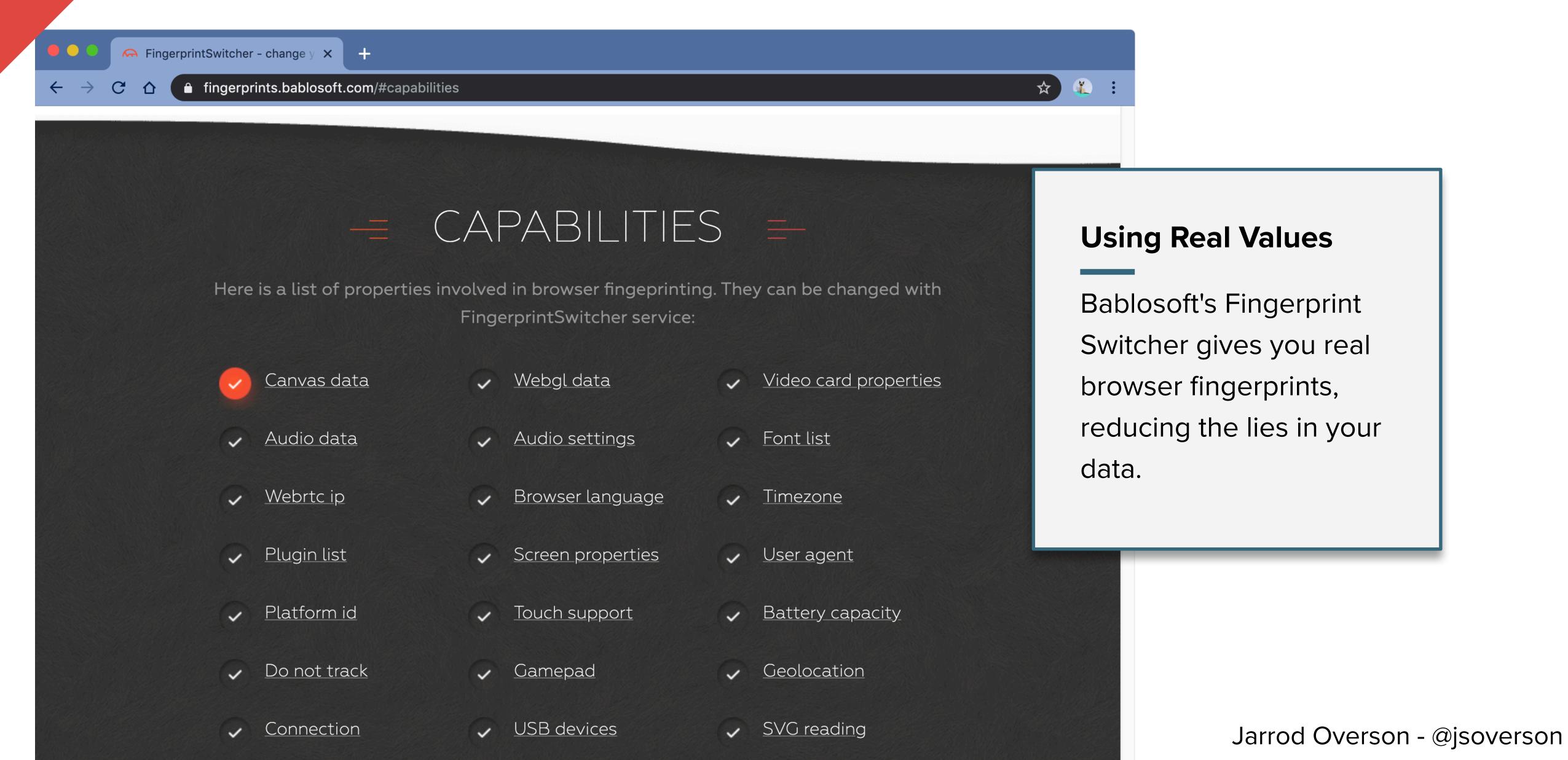




### Defense: Browser Consistency Checks



#### Iteration 7: Use real device data



### Defense: MFA and risk scoring



#### Enter code

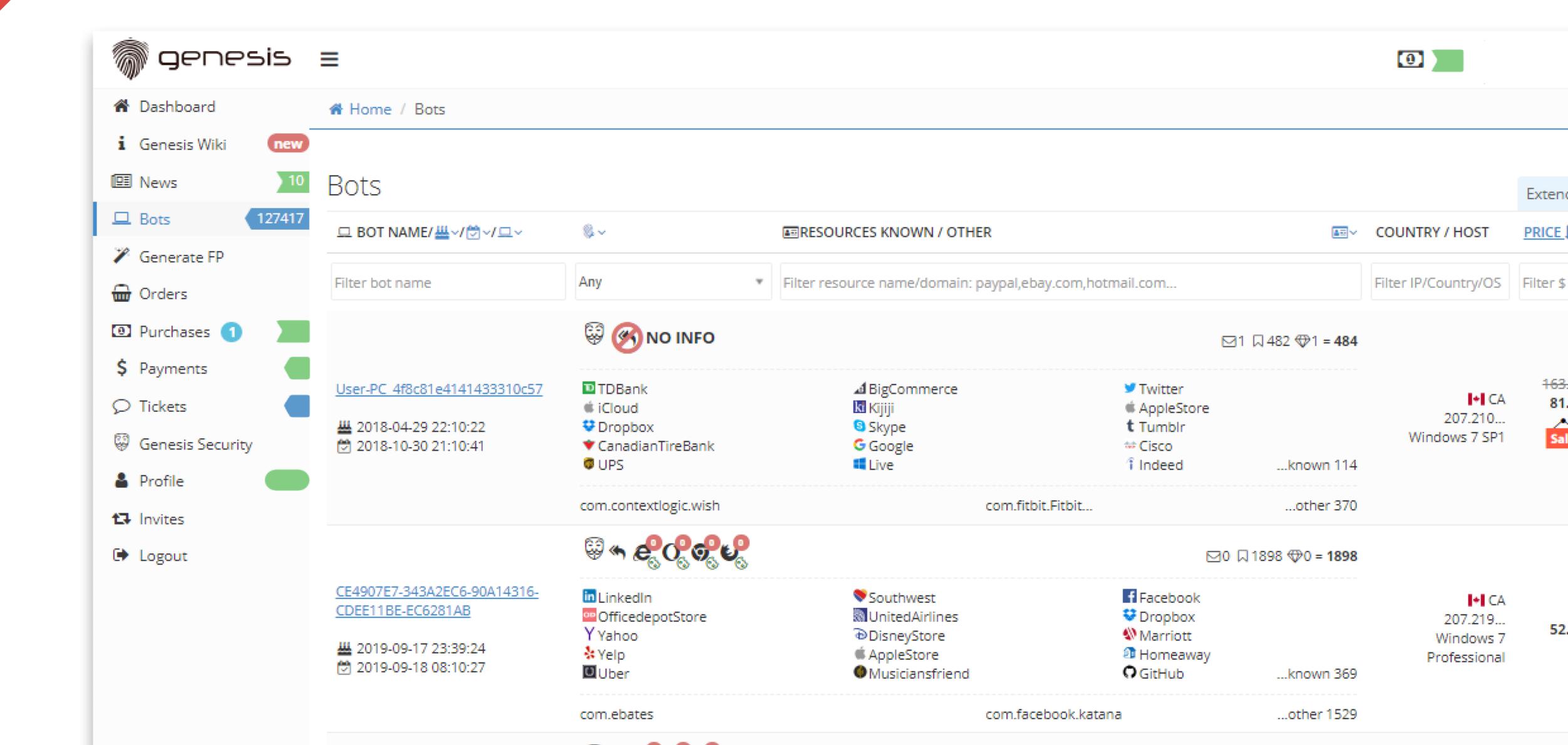
Please type in the code displayed on your authenticator app from your device

Code

More information

Verify

#### Iteration 8: Hijack sessions and user computers.



How has automation evolved?

Where do we go from here?

#### As a company: bring security & fraud into UX decisions

The best defenses are internal.

#### Limit, delay, and vary feedback in the UX.

Advanced adversaries are developers. Don't aid them with helpful messages.

#### Reduce your surface area

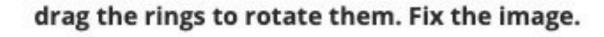
Funnel risk into bottlenecks to simplify modeling and response.

#### Build a system of dials and levers

Adversaries iterate quickly and defenses need to change rapidly. Build for speed.

# Stop using CAPTCHAs

They are trivially bypassed and are more frustrating to users than adversaries.







#### THANK YOU!

@jsoverson on twitter, medium

Visit vino.dev to learn about what I'm up to next.