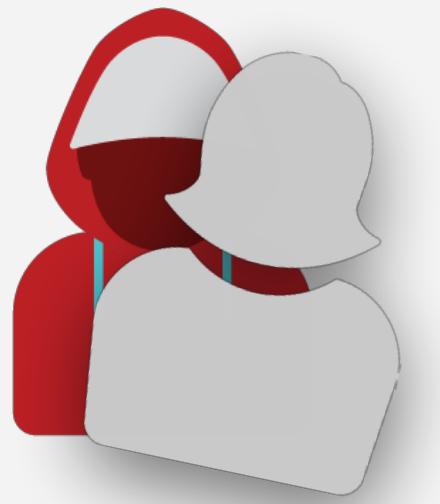


Imitation Attacks

How account takeovers are evolving.

Two themes to keep in mind



Imitation

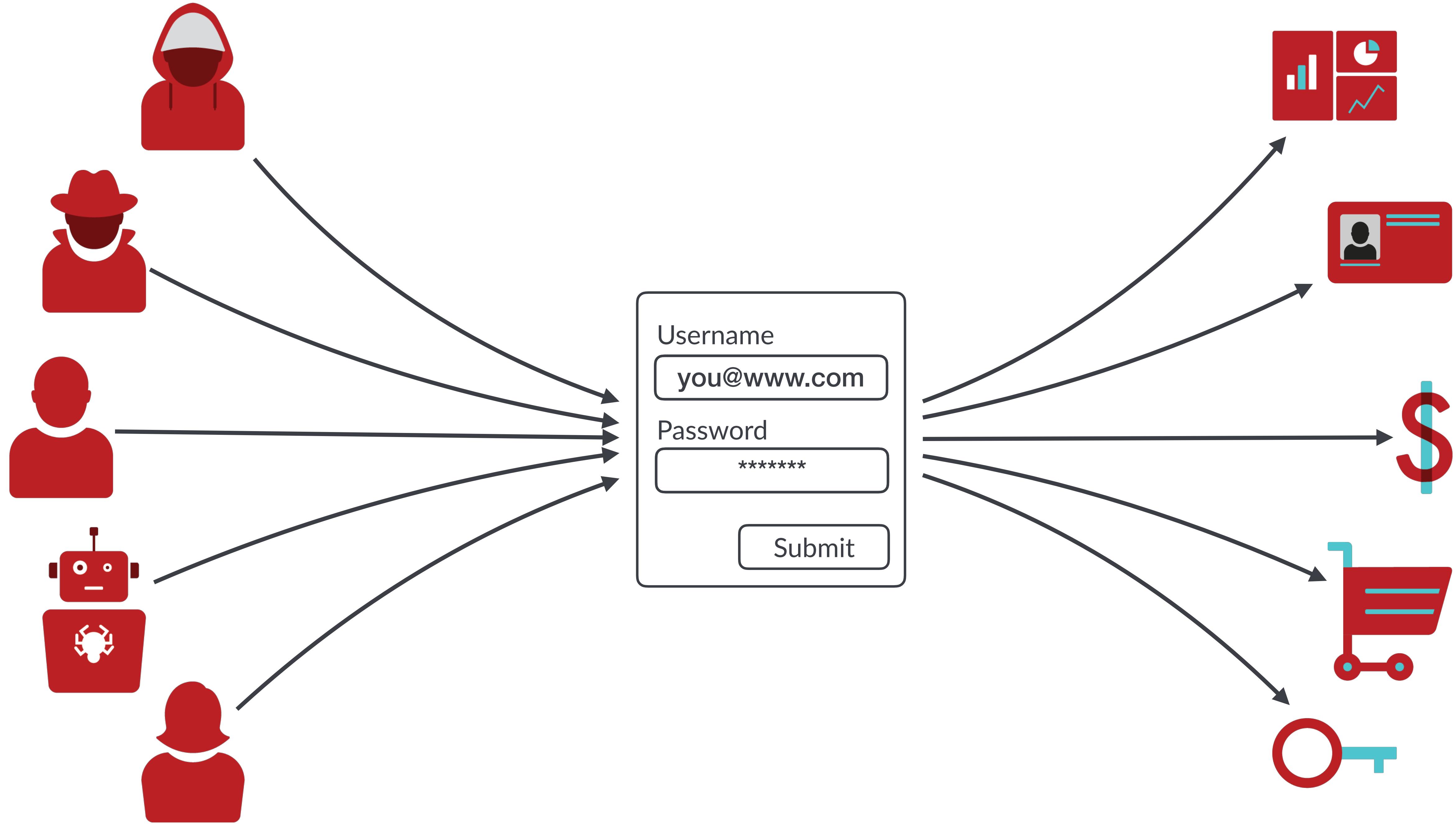


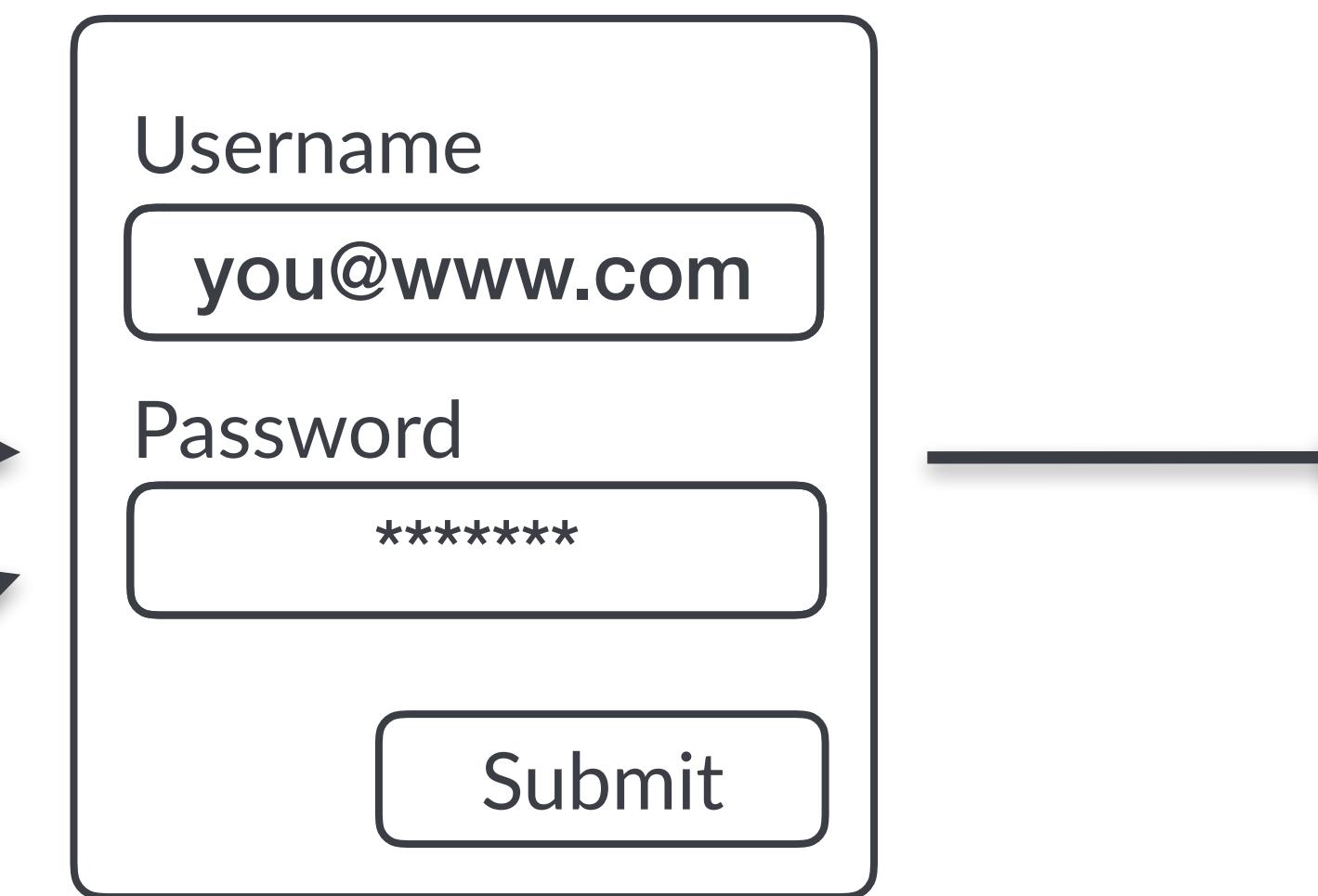
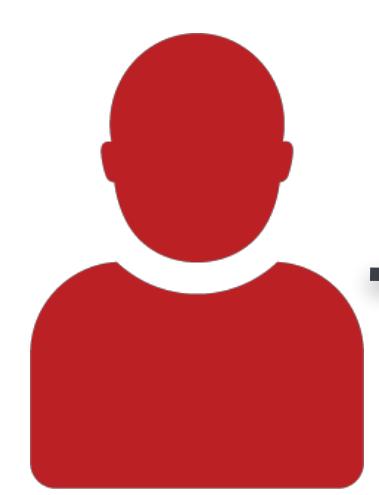
Cost vs Value

Username

Password

Submit

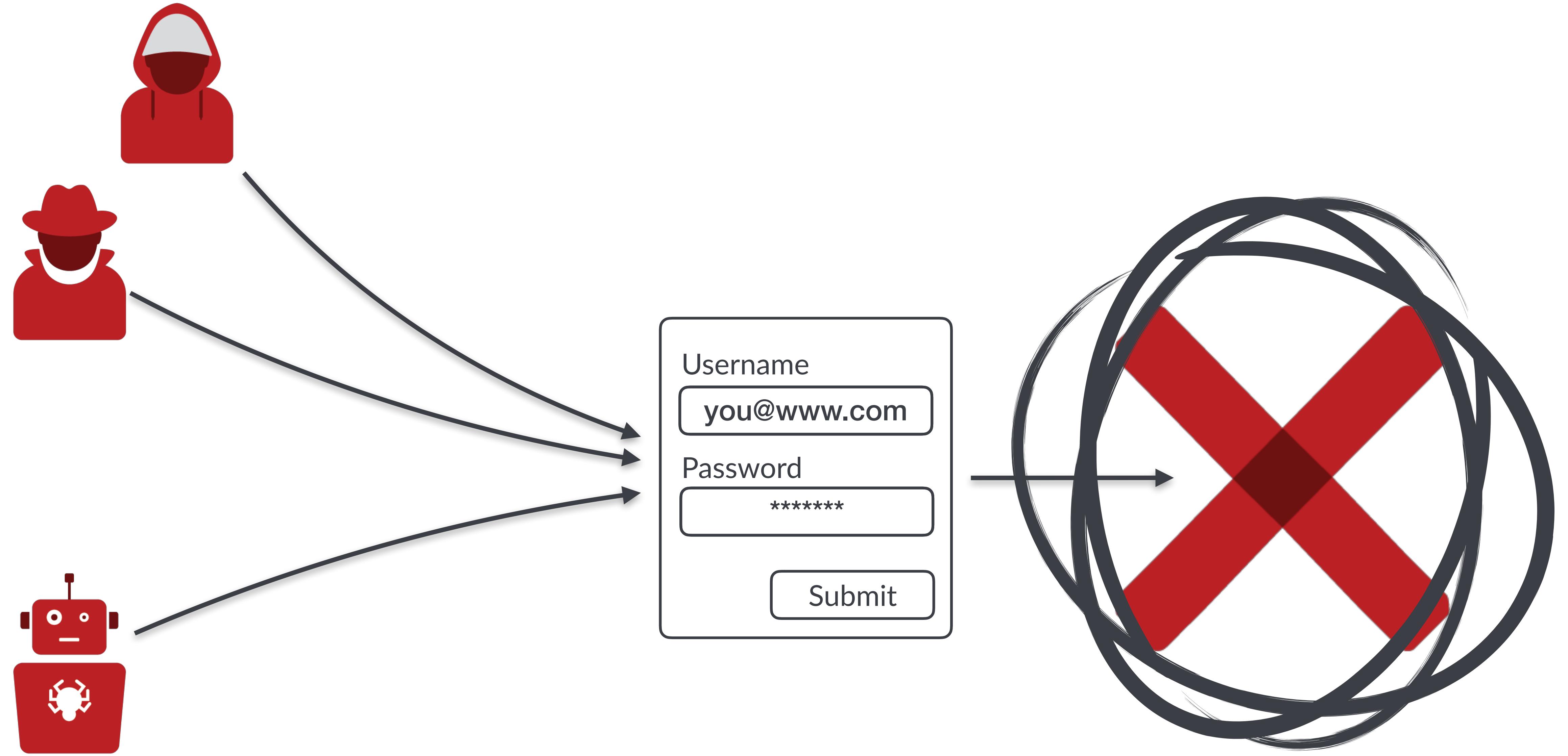


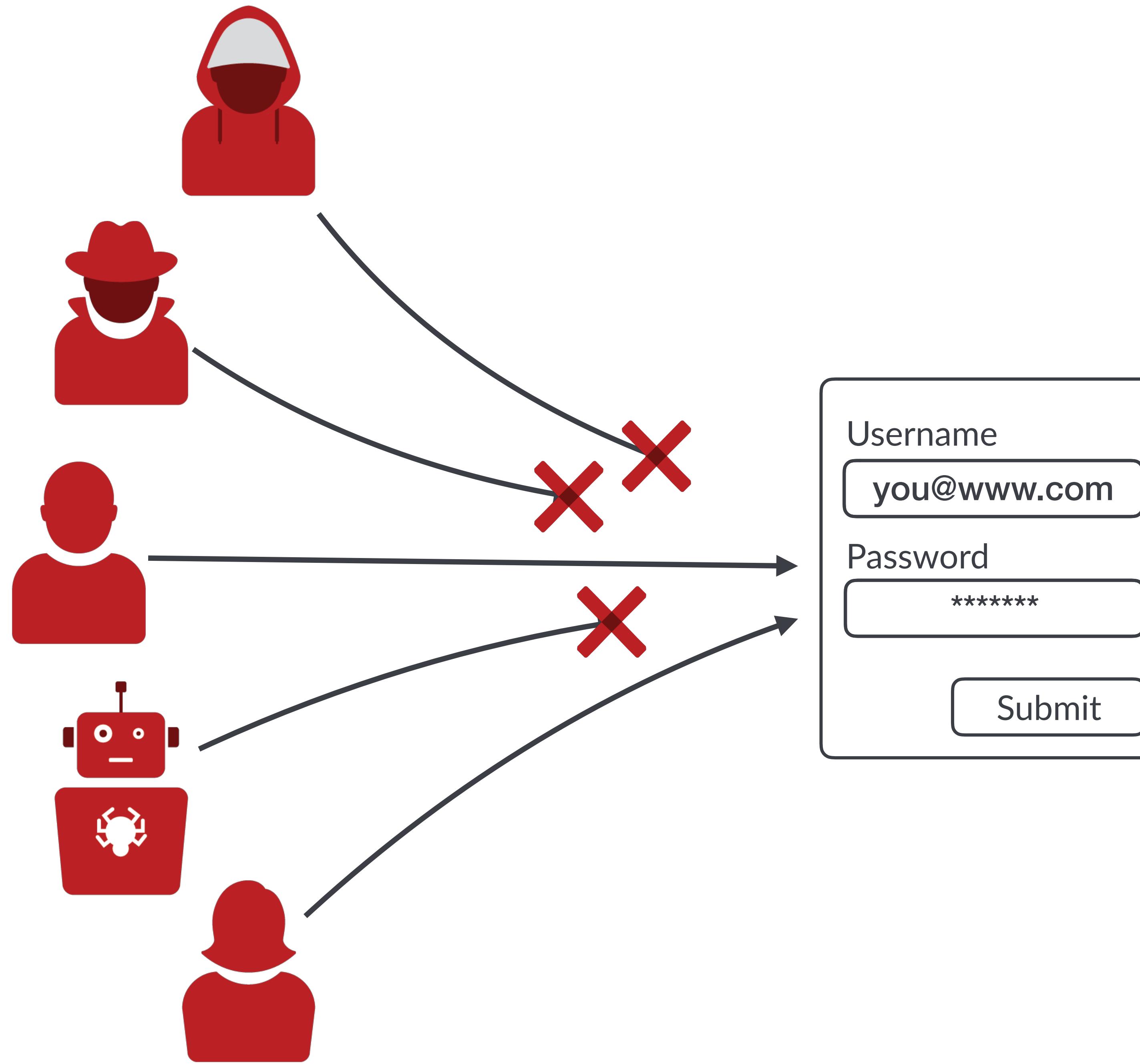


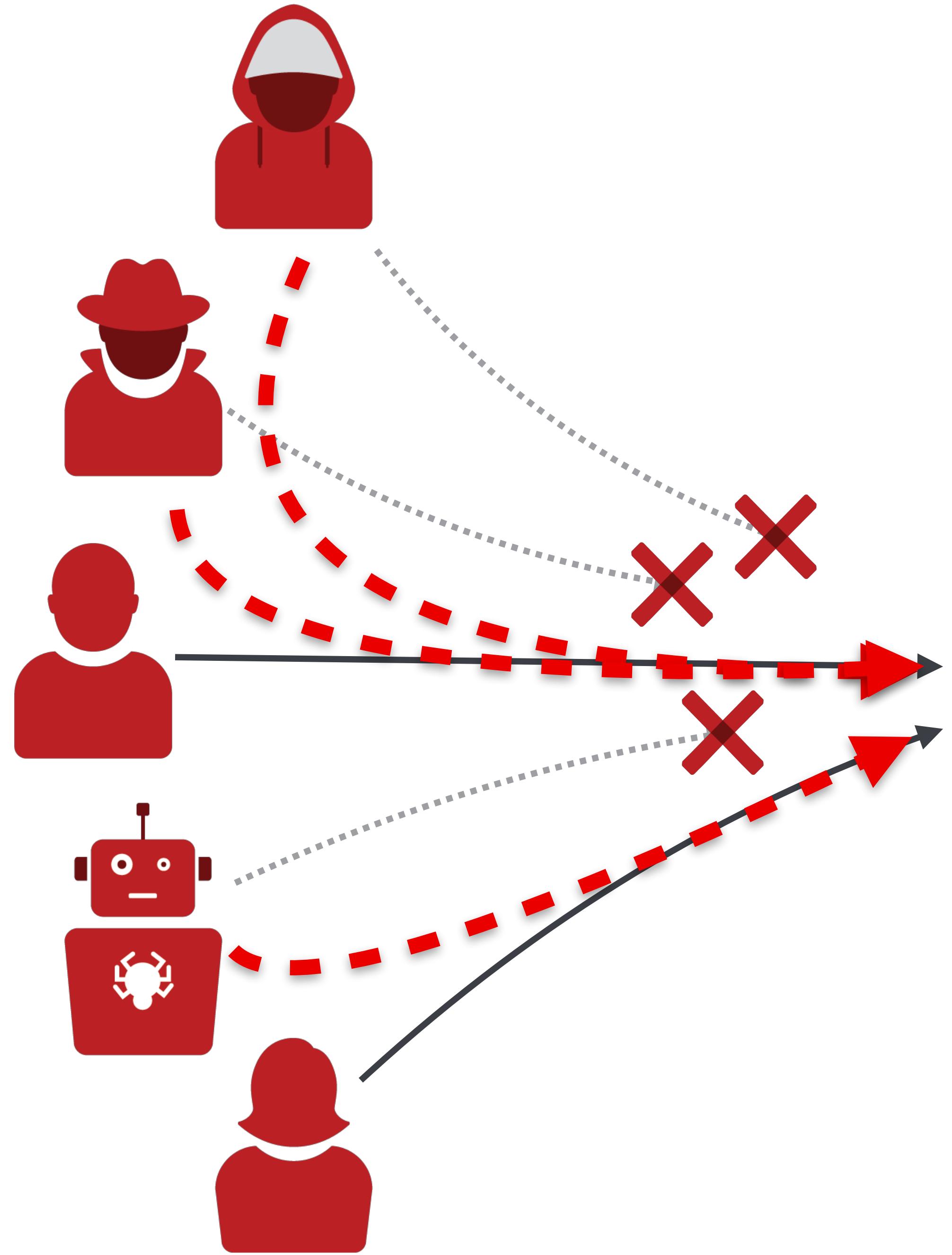
A diagram illustrating a login process. On the left, two user icons point to a central rectangular form. The top user has a horizontal arrow pointing right to the form. The bottom user has a curved arrow pointing right to the form. The form contains the following fields:

- Username: `you@www.com`
- Password: `*****`
- Submit button

```
graph LR; User1((User)) --> Form[Login Form]; User2((User)) --> Form; subgraph Form [ ]; direction TB; Username["Username  
you@www.com"]; Password["Password  
*****"]; Submit["Submit"]; end;
```







Imitation Attacks

Attacks through publicly accessible APIs or websites that can only be successful if they mimic legitimate user behavior.

User behavior may mean demographics, traffic origin, User-Agent, mouse and keyboard input, challenge responses, sensor data, et al.

Username

Password

 I'm not a robot

 reCAPTCHA

Privacy - Terms

Submit

Username

Password

I'm not a robot

 reCAPTCHA
Privacy - Terms

Submit



Enter your one-time password

Submit

Username

Password

I'm not a robot

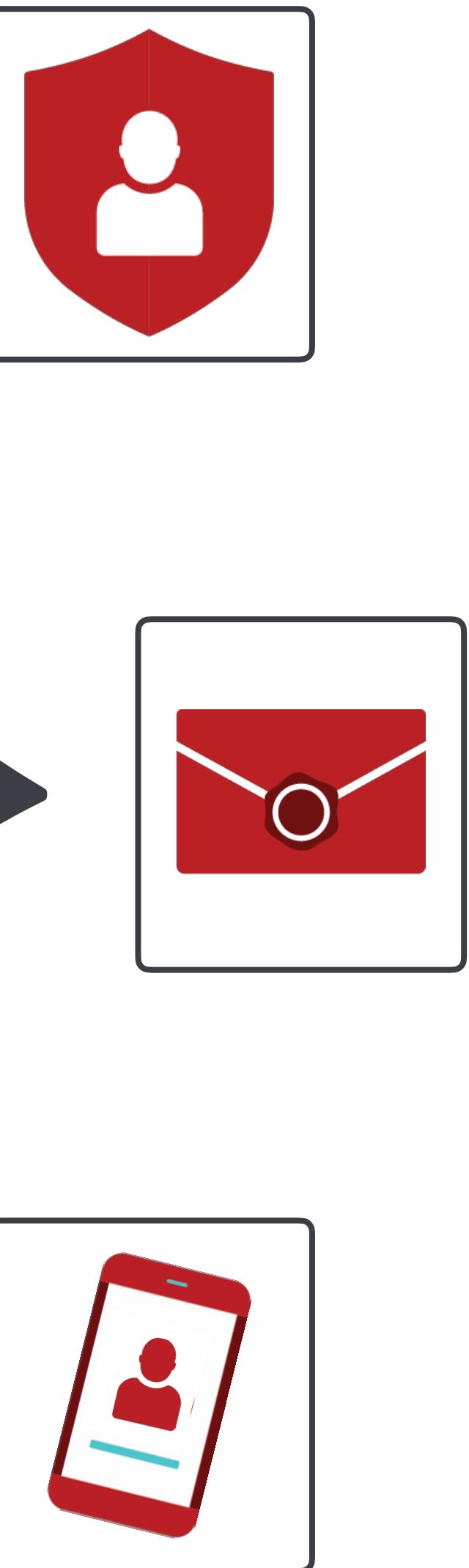
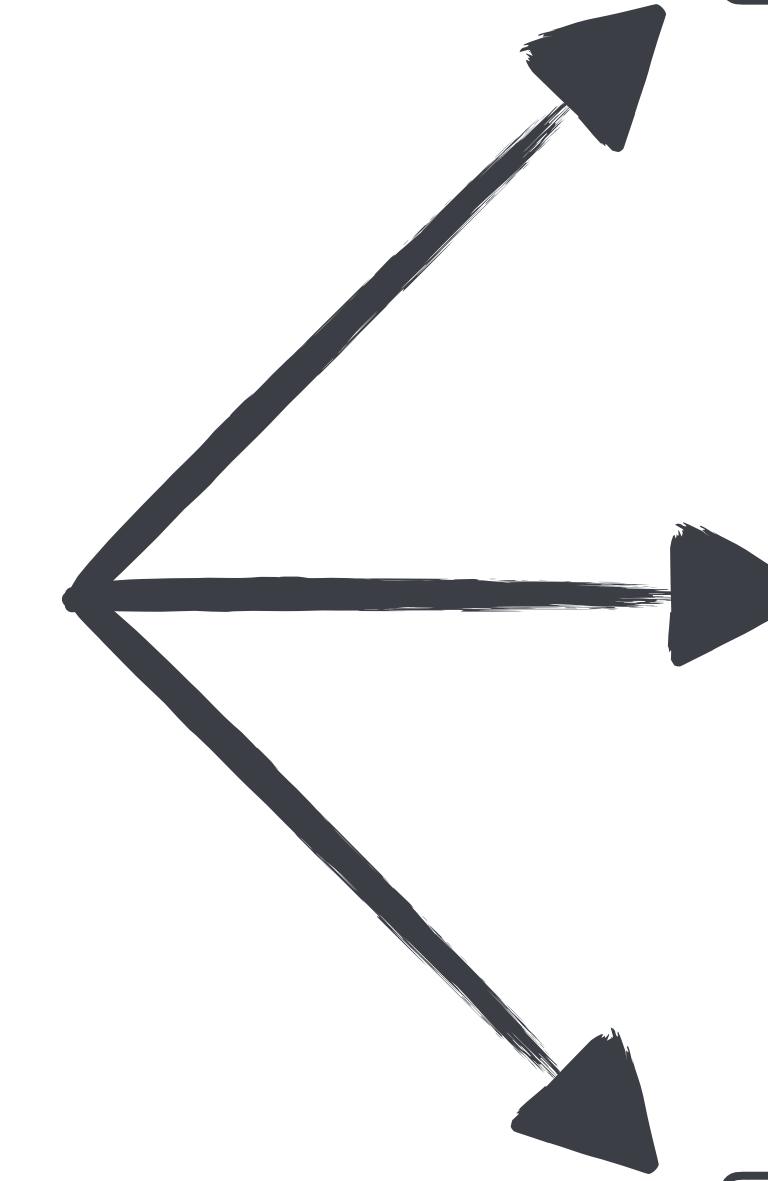
 reCAPTCHA
Privacy - Terms

Submit

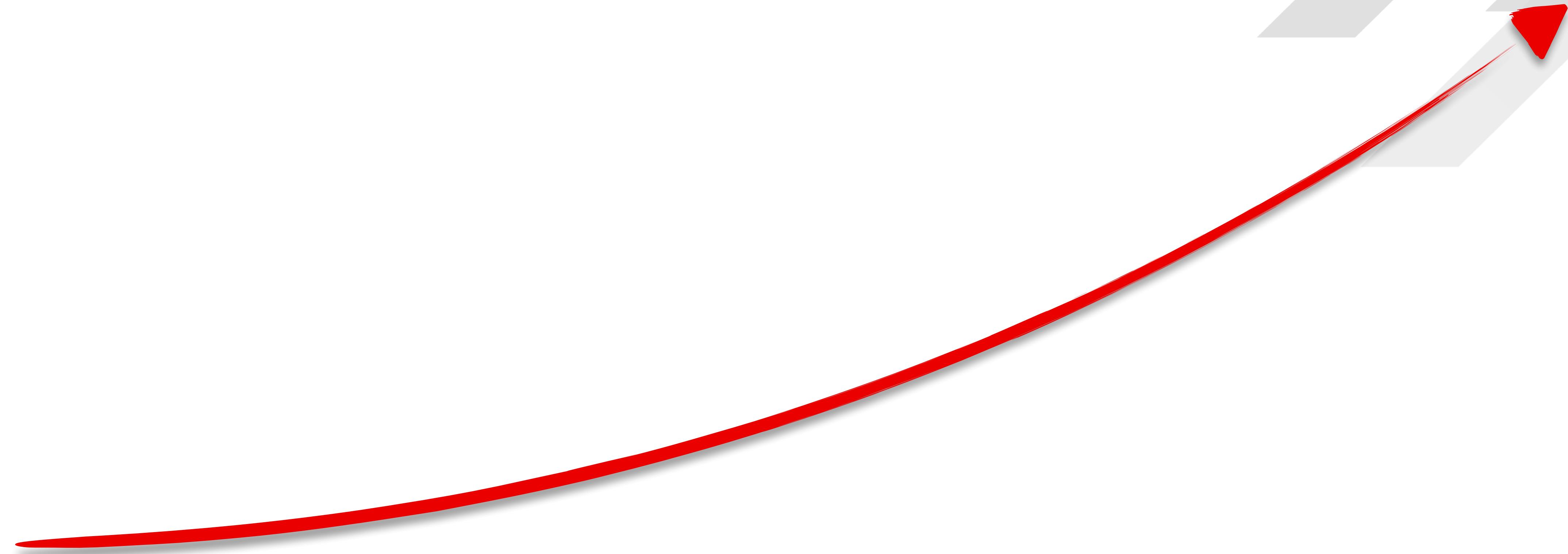


Enter your one-time password

Submit



Even with all these defenses, attacks are increasing,



and they are only getting easier to perform.



IN THE FIRST 6 WEEKS OF 2019

IN THE FIRST 6 WEEKS OF 2019

2.3 BILLION

credentials were leaked to the open web.

Shape recorded its largest ever attack

2.93 BILLION TRANSACTIONS

For **1 CUSTOMER** against **1 FLOW** in **1 WEEK**

Agenda

1

Current attack landscape

2

Attacks in detail

3

The arms race

4

How do we adapt?

MANUAL ATTACKS

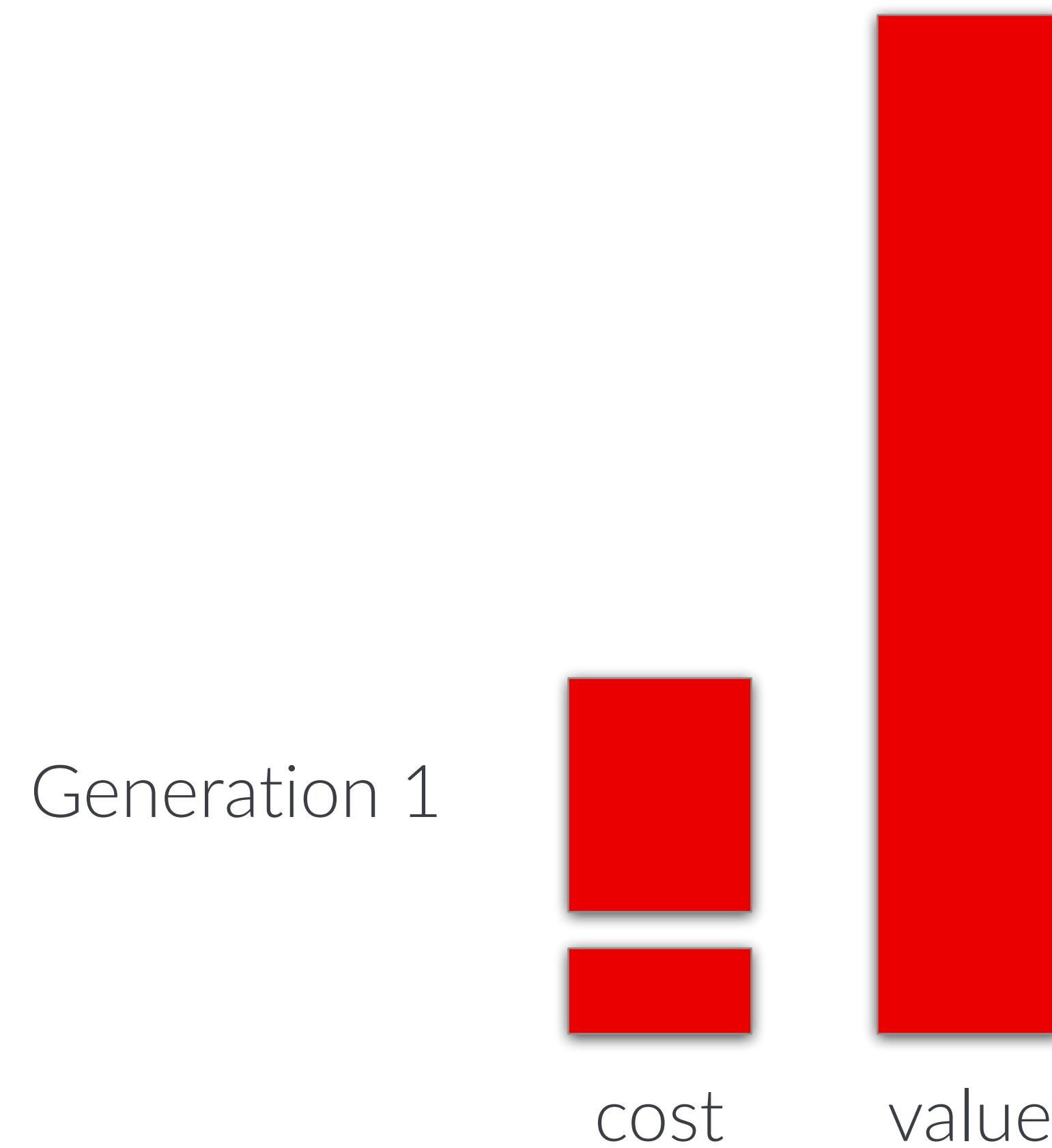
AUTOMATED ATTACKS



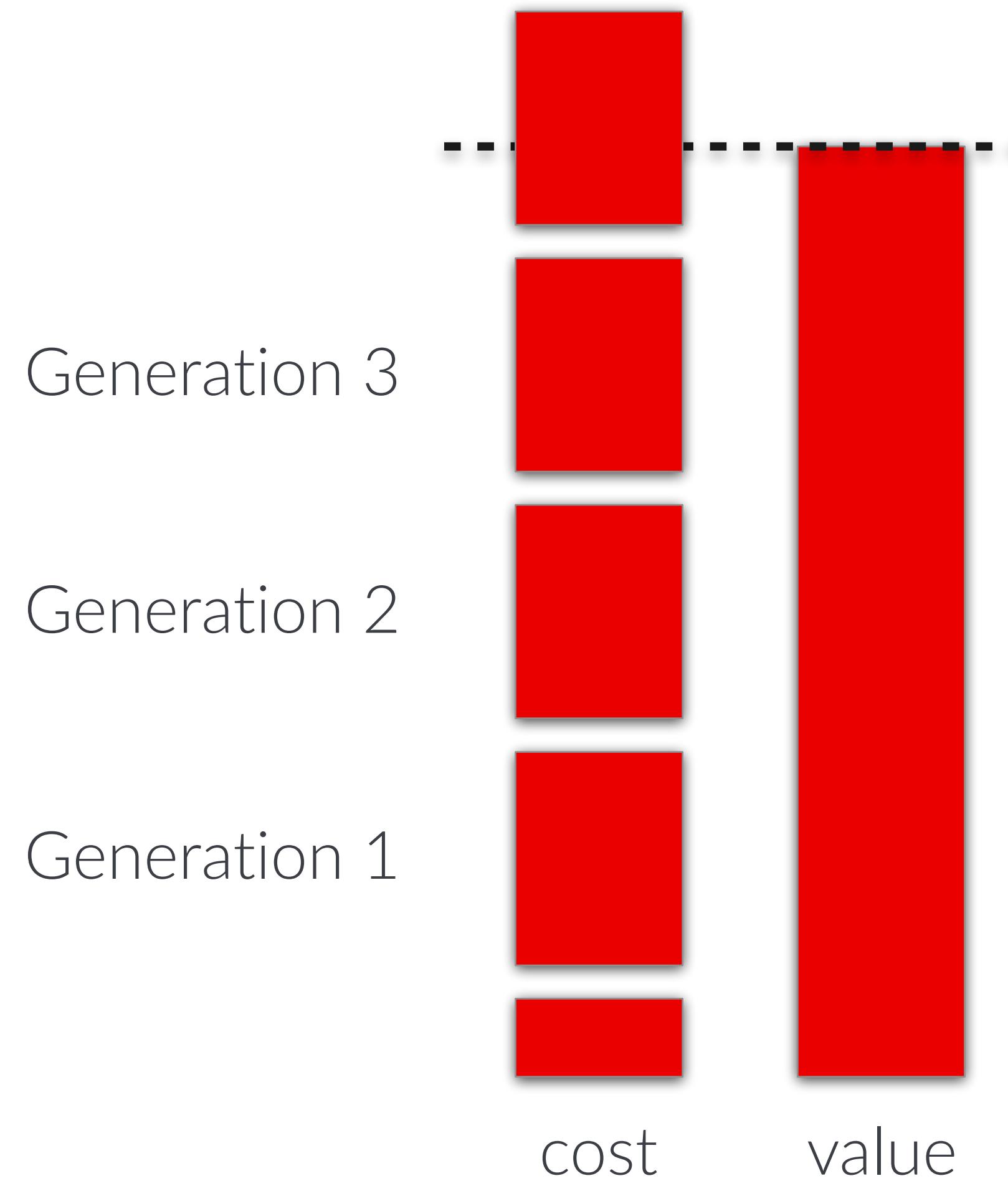
If there are no defenses in place, costs are negligible.



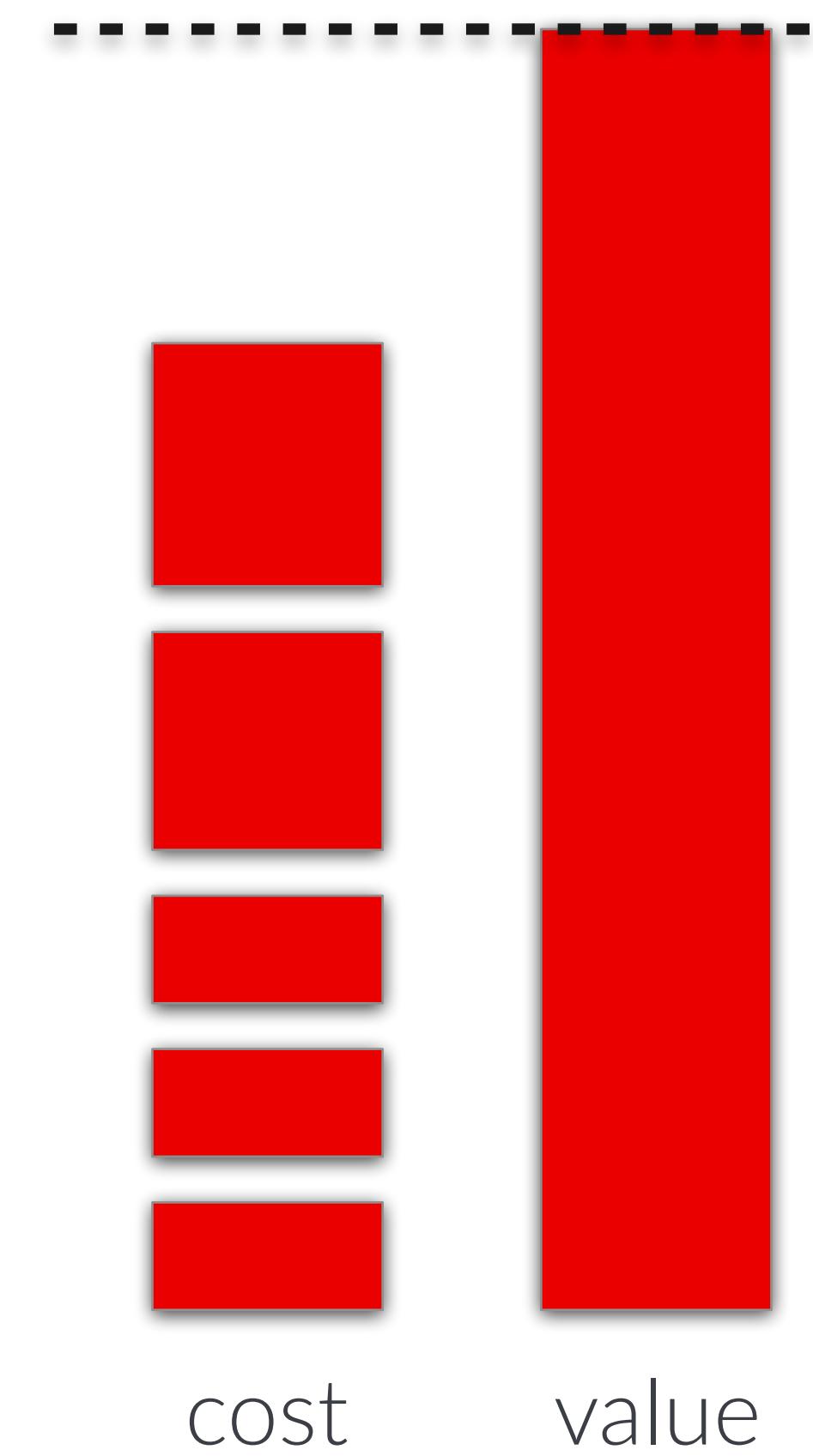
Any defense increases the cost by forcing a generational shift.



The sophistication increases until it is unsustainable.



The cost of entry for each generation decreases over time.

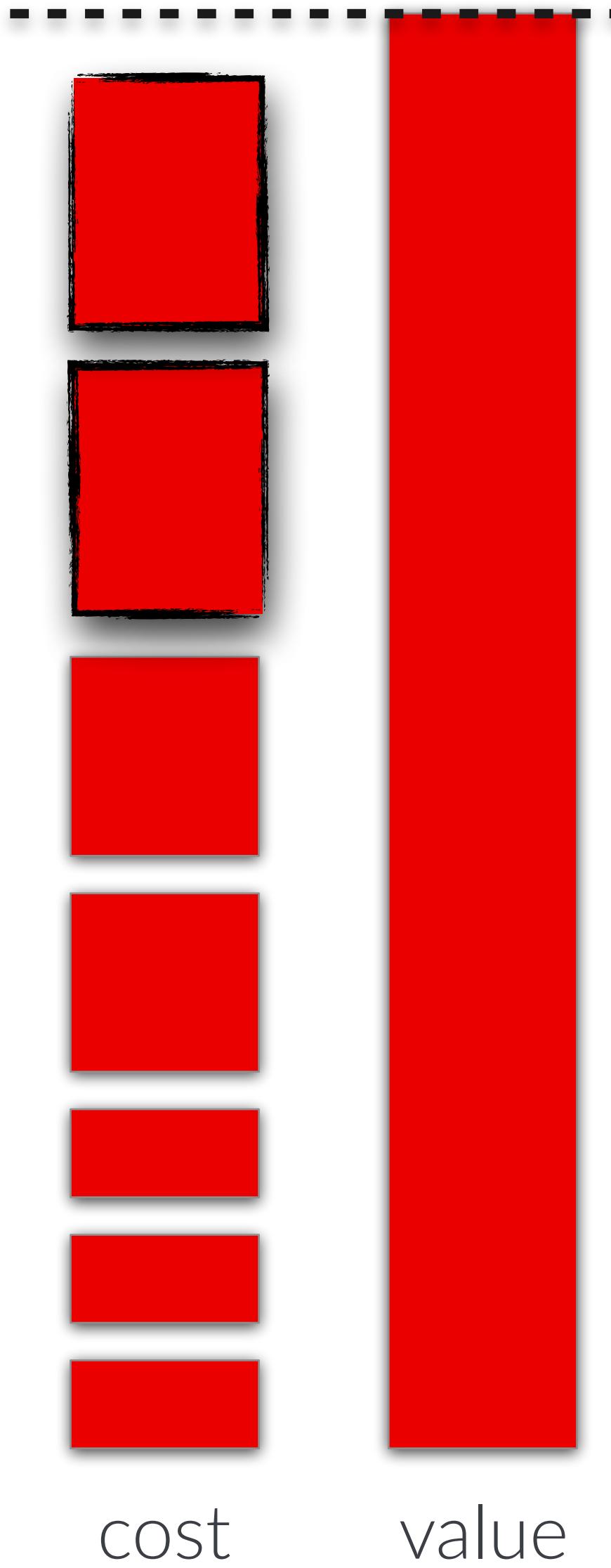


Not only that, the value of these accounts only goes up.

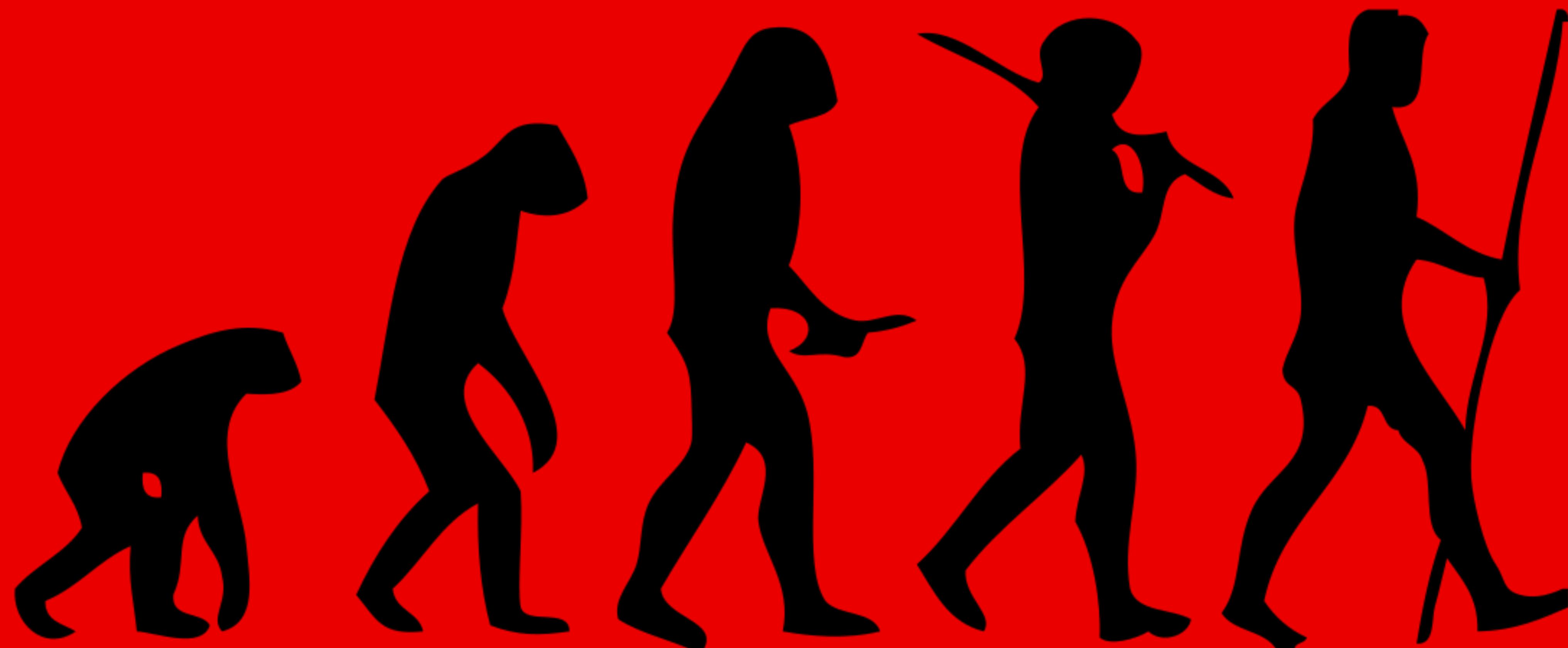


And every generation necessitates new defenses.

and sophistication is
growing **rapidly**



THE EVOLUTION OF ATTACK TOOLS



curl / wget

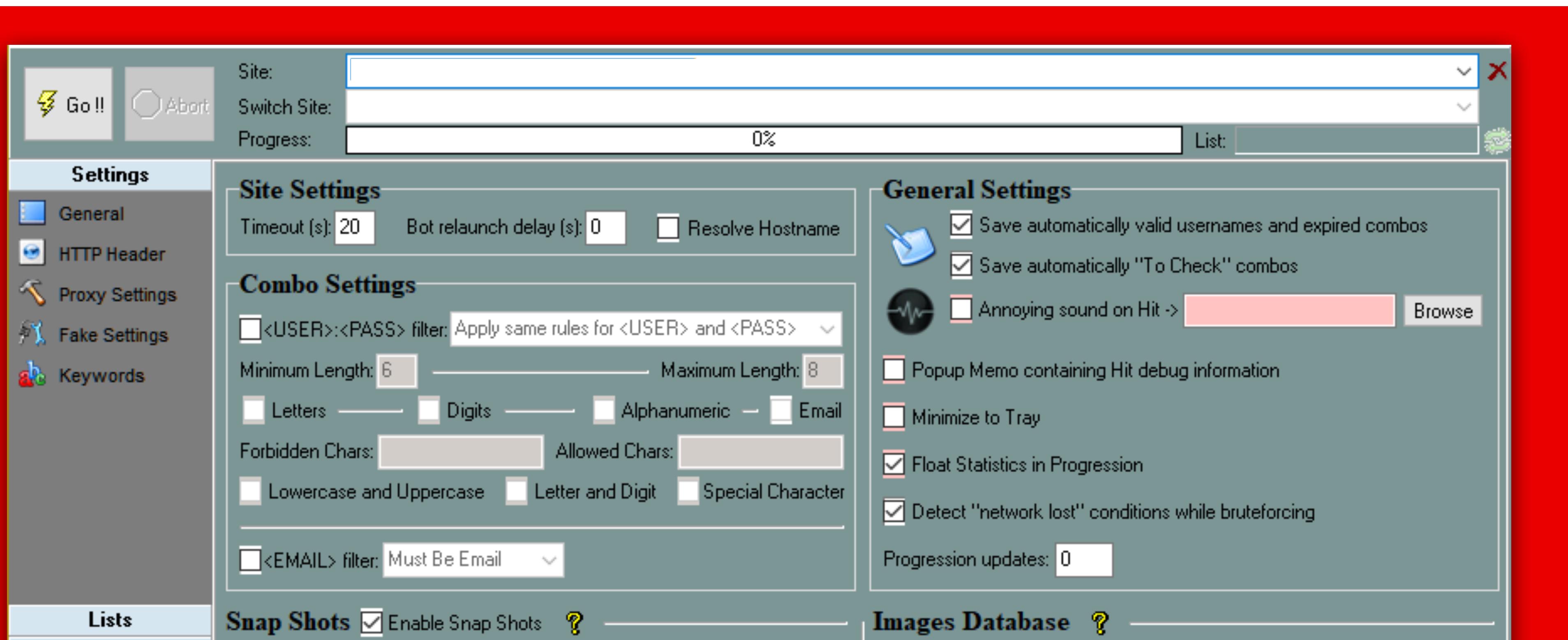
- Basic HTTP requests.
- Replaying tokens or reusing cookies.
- Changing form data for each transaction.



```
* About to connect() to google.com port 443 (#0)
*   Trying 172.217.10.14... connected
* successfully set certificate verify locations:
*   CAfile: none
*   CApth: /etc/ssl/certs
* SSLv3, TLS handshake, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using ECDHE-ECDSA-AES128-SHA
* Server certificate:
*       subject: C=US; ST=California; L=Mountain View; O=Google Inc; CN=*.google.com
*       start date: 2017-05-16 14:17:12 GMT
*       expire date: 2017-08-08 13:41:00 GMT
*       subjectAltName: google.com matched
*       issuer: C=US; O=Google Inc; CN=Google Internet Authority G2
*       SSL certificate verify ok.
> GET / HTTP/1.1
```

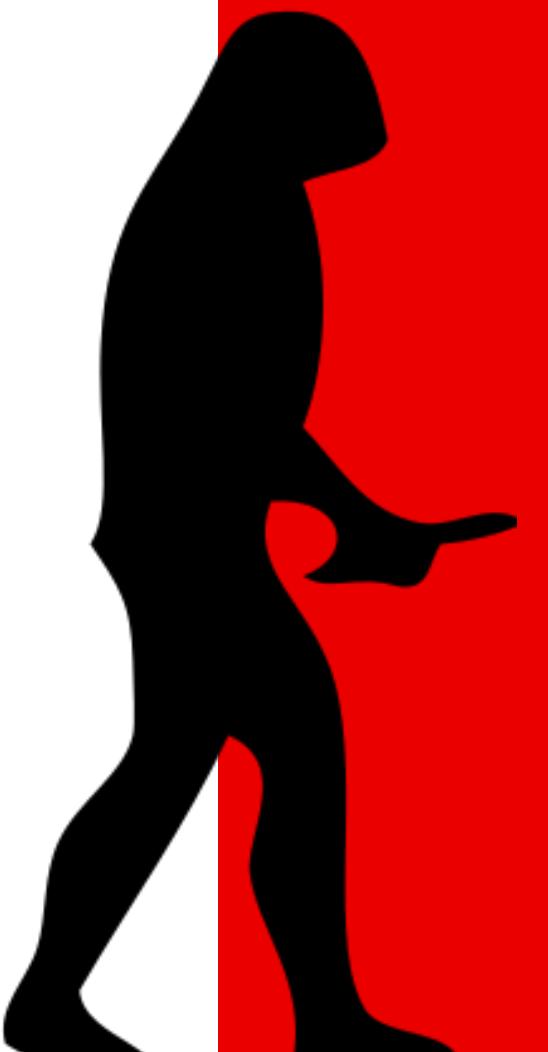
SentryMBA

- Still Basic HTTP requests.
- Extensible and highly configurable.
- Tailored towards specific attack use cases.



PhantomJS

- Full browser environment for developers.
- Scriptable and headless.
- Bypasses trivial protection with no effort.



The image shows a black silhouette of a person standing on the left, facing right and pointing their right hand towards the center of the screen where the PhantomJS website is displayed.

Fork me on GitHub

PhantomJS

SOURCE CODE **DOCUMENTATION** **API** **EXAMPLES** **FAQ**

Full web stack No browser required

PhantomJS is a headless WebKit scriptable with a JavaScript API. It has **fast** and **native** support for various web standards: DOM handling, CSS selector, JSON, Canvas, and SVG.

[Download v2.1](#) [Get started](#)

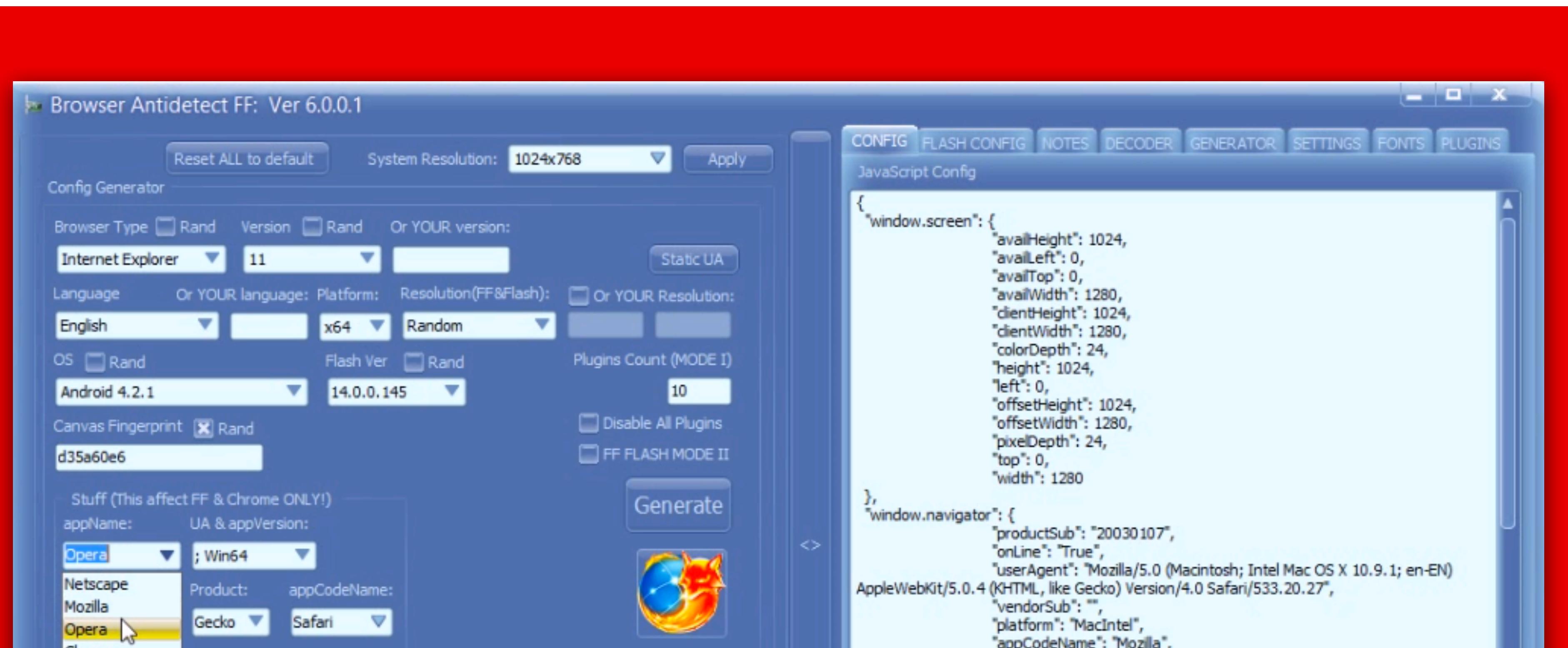
```
// Simple Javascript example

console.log('Loading a web page');
var page = require('webpage').create();
var url = 'http://phantomjs.org/';
page.open(url, function (status) {
    //Page is loaded!
    phantom.exit();
});
```

Community: [Read the release notes](#) [Join the mailing list](#) [Report bugs](#)

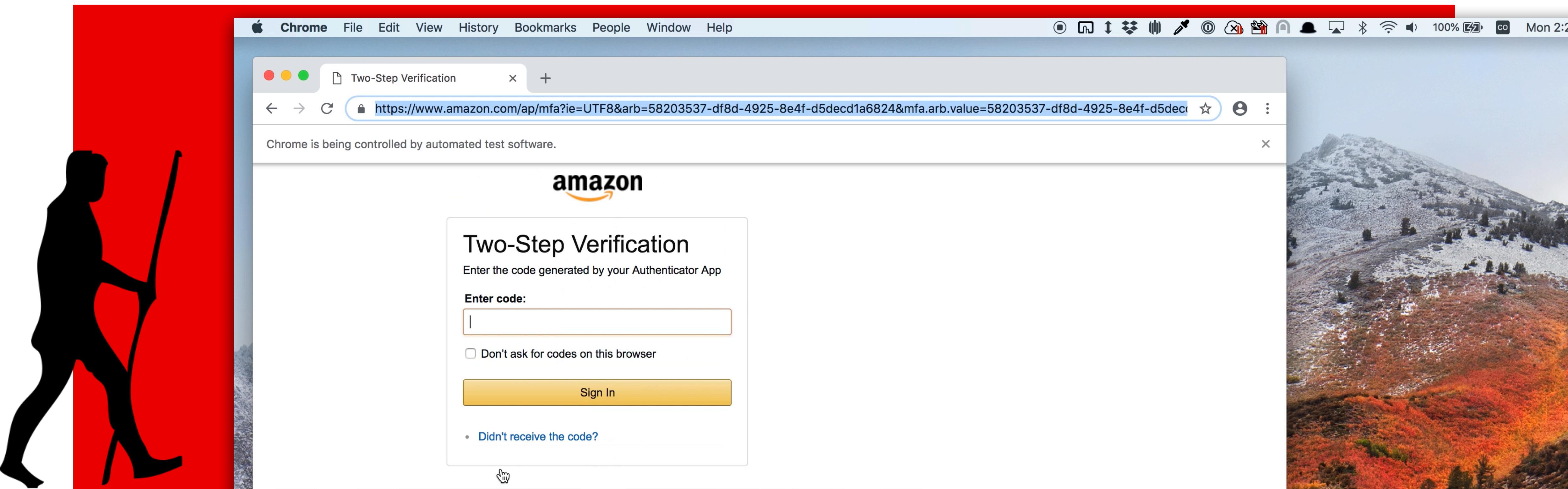
Browser AntiDetect

- Extension for FireFox and Chrome.
- Randomizes fingerprintable data points.
- Designed specifically to blend in.



Headless Chrome & Puppeteer

- Drives Production Chrome, the world's most-used browser.
- Scriptable with low level features that benefit attackers.
- Powerful foundation for imitation attacks.



Agenda

1

Current attack landscape

2

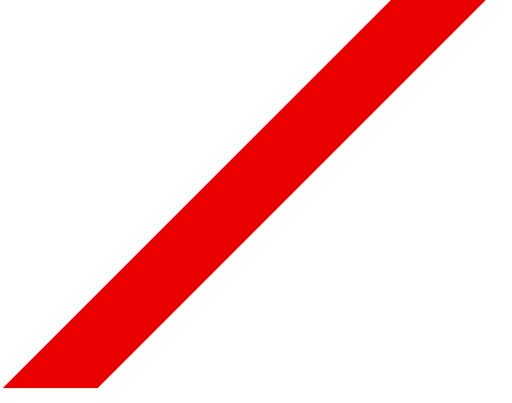
Attacks in detail

3

The arms race

4

How do we adapt?



Attack Detail: Credential Stuffing

CREDENTIAL STUFFING

cre·den·tial stuff·ing

/krə'den(t)SHəl 'stəfiNG/

The automated replay of breached username/password pairs across many sites in order to take over accounts where passwords have been reused.

A STEP BY STEP GUIDE

1

Get Credentials

2

Automate Login

3

Defeat Automation Defenses

4

Distribute Globally

CREDENTIAL STUFFING

1

1. Get Credentials

Bookmarks People Window Help

RF Collection #1-5 & Zabagur & A... X +

https://raidforums.com/Thread-Collection-1-5-Zabagur-AntiPublic-Latest-120GB-1TB-TOTAL-Leaked-Download

f g+ YouTube p

Need proof? The layout is same as troy's, size is same, + here's original sales thread from owner:

Folders & Size

| Collection | Size |
|-------------------|-----------|
| Collection #1 | 87.18 GB |
| Collection #2 | 526.11 GB |
| Collection #3 | 37.18 GB |
| Collection #4 | 178.58 GB |
| Collection #5 | 42.79 GB |
| AP MYR&ZABUGOR #2 | 24.53 GB |
| ANTIPUBLIC #1 | 102.04 GB |

(Blurred as the owner is under a lot of heat right now due to the exposure of this, so done out of respect, not that I care or anything just don't want drama).

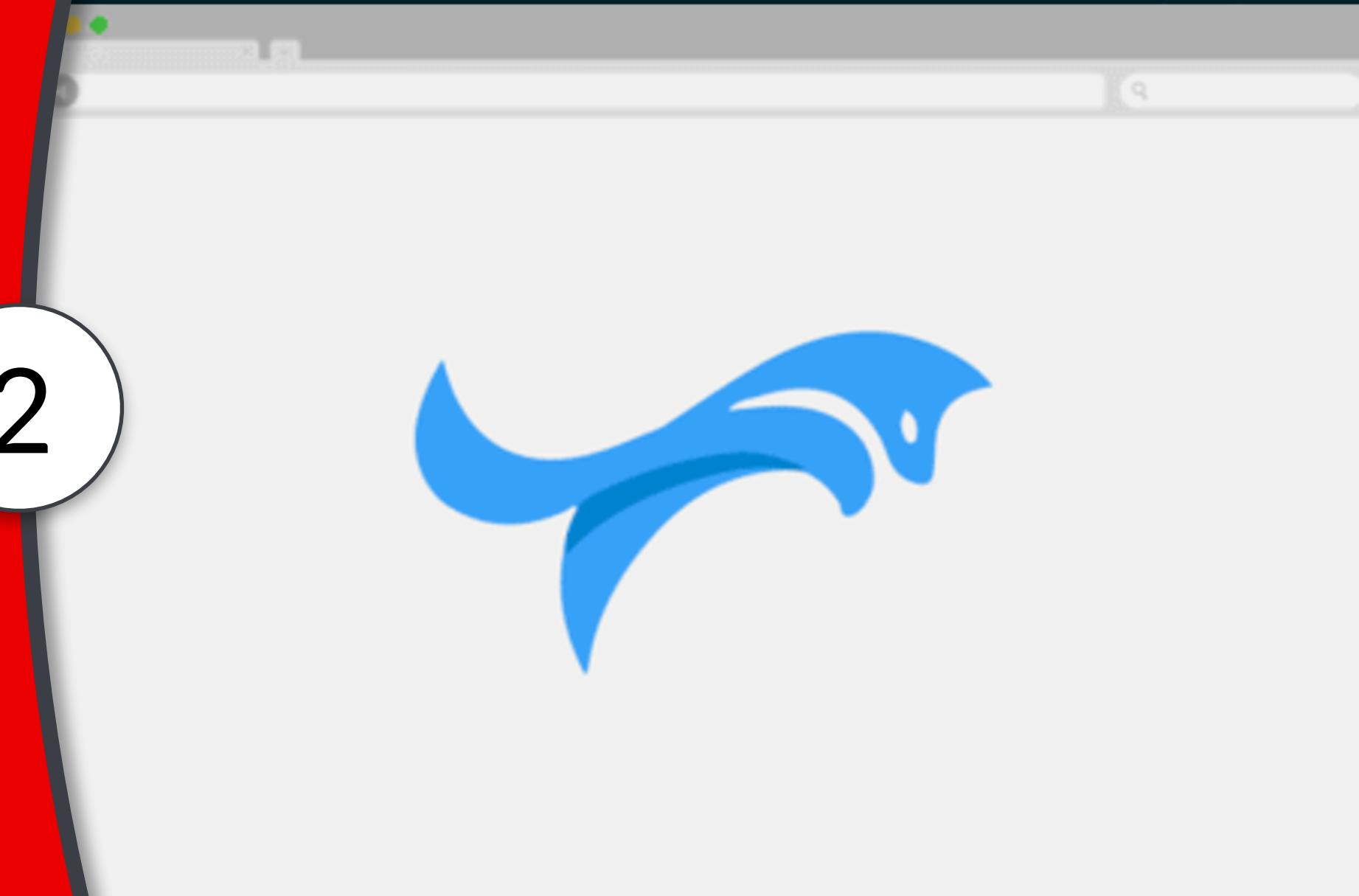
Collection #1 to #5 in .torrent form thanks to user @neob and every seeder.

Hidden Content:
Unlock for 8 credits.

CREDENTIAL STUFFING

1. Get Credentials
2. Automate Login

2



ULTIMATE INTERNET PRIVACY

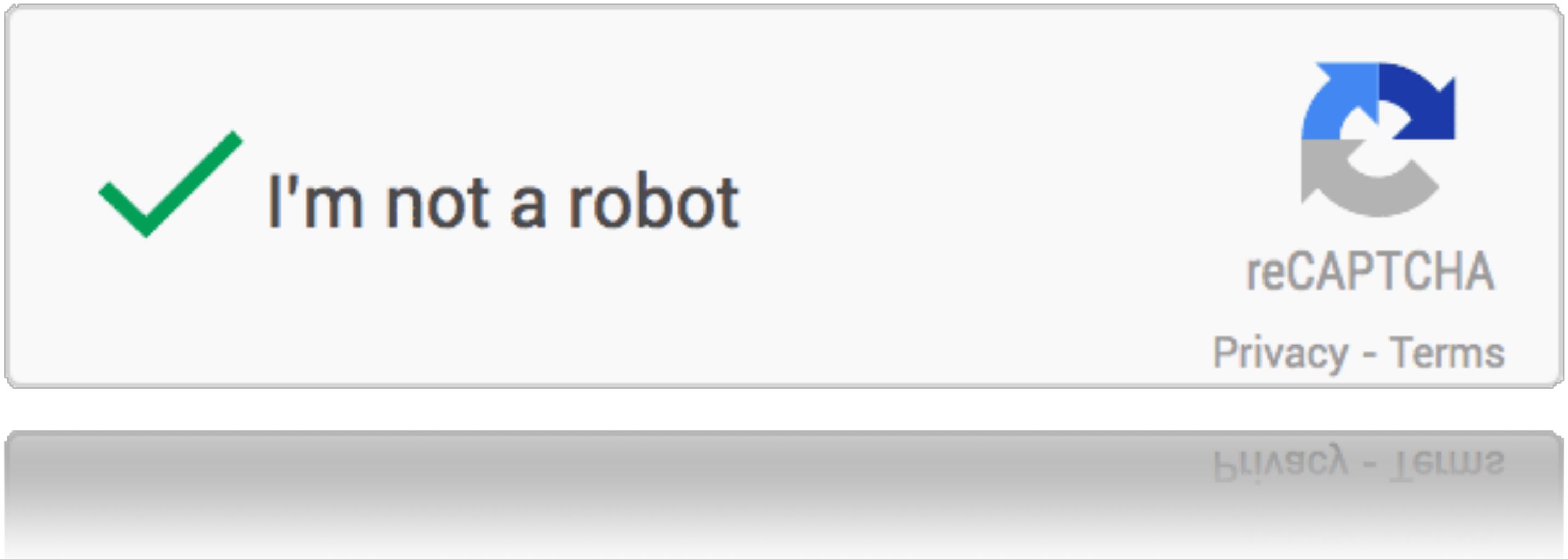
VIRTUAL MACHINE BASED SOLUTION TO BEAT
BROWSER FINGERPRINTING



CREDENTIAL STUFFING

1. Get Credentials
2. Automate Login
3. Defeat Defenses

3



CREDENTIAL STUFFING

3

1. Get Credentials
2. Automate Login
3. Defeat Defenses



Home

F.A.Q.

API

Order CAPTCHAs

DBC Points

Testimonials

Contact Us

English Русский

STATUS: OK

Average solving time 1 minute ago: 10 s
5 minutes ago: 11 sec
15 minutes ago: 11 sec
Today's average accuracy rate: 90.5 %
(updated every minute)

Create a FREE account

Log In

Best CAPTCHA Solver Bypass Service

With Death by Captcha you can solve any CAPTCHA. All you need to do is implement our API, pass us your CAPTCHAs and we'll return the text. It's that easy!

Please note that our services should be used only for research projects and any illegal use of our services is strictly prohibited. Any bypass and CAPTCHA violations should be reported to help@deathbycaptcha.com

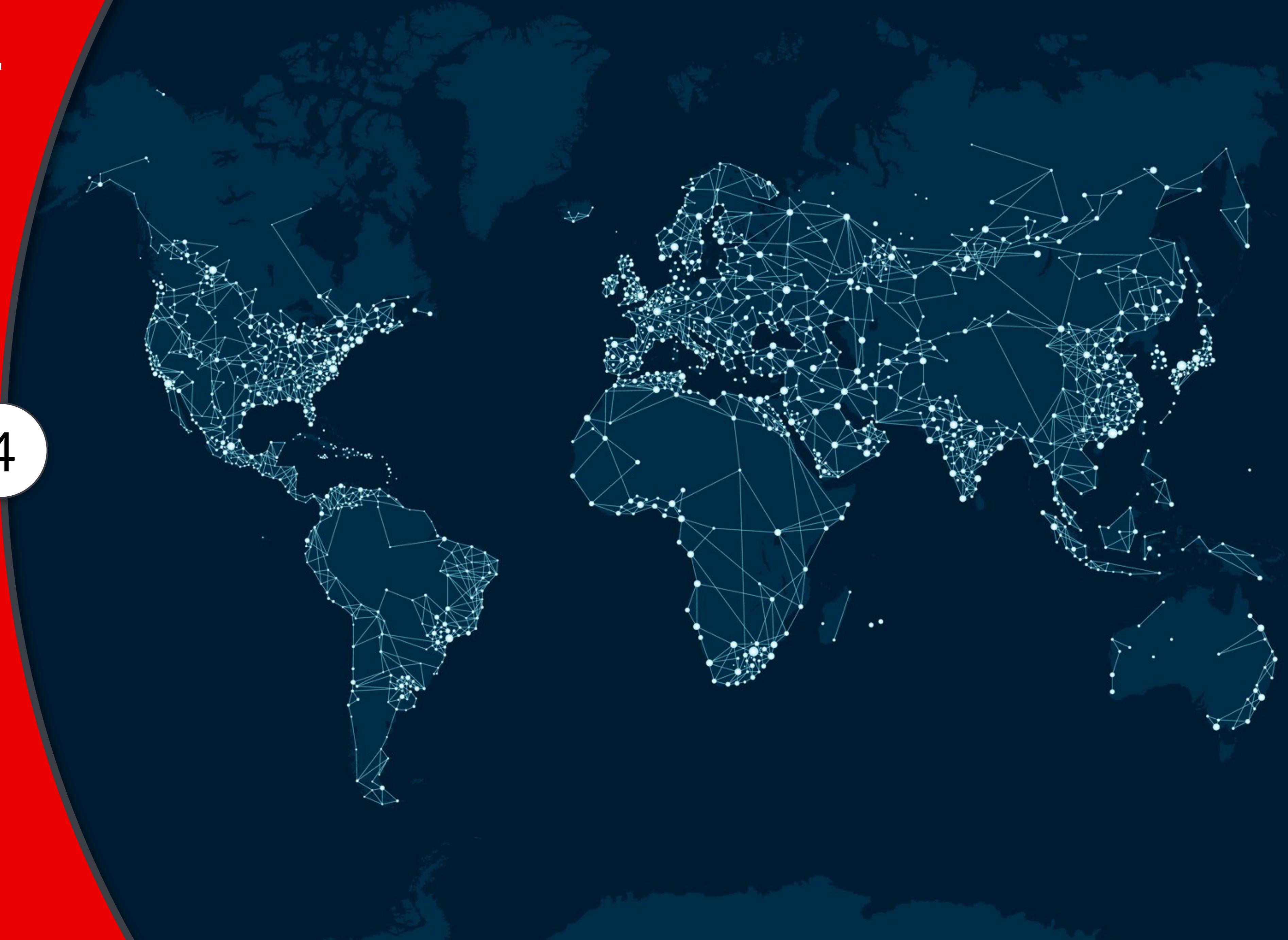
Death By Captcha Offers:

- Starting from an incredible low price of \$1.39 (\$0.99 for **Gold Members !**) for 1000 solved CAPTCHAs.
- A hybrid system composed of the most advanced OCR system on the market, along with a 24/7 team of CAPTCHA solvers.

CREDENTIAL STUFFING

4

1. Get Credentials
2. Automate Login
3. Defeat Defenses
4. Distribute



CREDENTIAL STUFFING

4

1. Get Credentials
2. Automate Login
3. Defeat Defenses
4. Distribute

CREDENTIAL STUFFING

1. Get Credentials
2. Automate Login
3. Defeat Defenses
4. Distribute



Unblock any website

Used by over **190 million** people worldwide

4

Click to install Hola VPN

Start 

\$0 : 2.3 billion credentials

\$50 : per site configuration

\$139 : for 100,000 CAPTCHAs

\$2 : for 1000 global IPs

Less than \$200
for 100,000 ATO
attempts

The screenshot shows a web browser window with three tabs. The active tab is titled "Search PP" and has the URL <https://slilppnyhik6febe.onion/searchpp.php?submitted=1>. The page content includes a sidebar with various buttons like "News", "Add funds", "Support", "Profile", and "PayPal". Below this is a message about PPs in stock (478264) and an "UPDATE" section. On the right side, there's a sidebar with a shopping cart icon and a welcome message. A large red arrow originates from the search bar at the top of the page and points to a vertical list of over 400 brand names on the right.

• apple
• airbnb
• bergfreunde
• britishairways
• capitalone
• deliveroo
• discovercard
• epicgames
• facebook
• groupon
• ikea
• marriot
• netflix
• +400 more

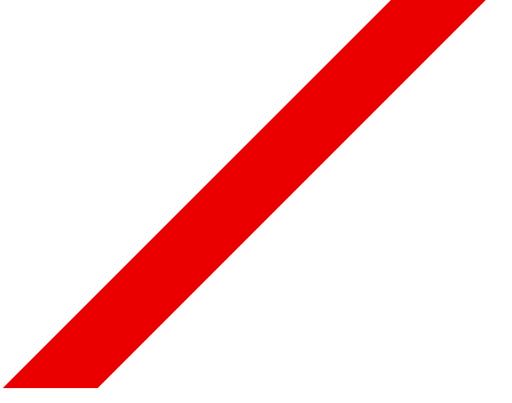
Pages: -1- 2 >
To page: 1 Go

| Shop | Balance | Points | Name | Type | Country State Zip | CC | Bank | Info | Last order | Mail domain | Uploaded | Seller | Price (\$): |
|------------|---------|--------|------|----------|----------------------|-----|------|---|---------------|----------------|-------------|--------|----------------|
| amazon.com | 795.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @aim.com | 14 Mar 2019 | sec | 15 |
| amazon.com | 757.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @cox.net | 14 Mar 2019 | sec | 15 |
| amazon.com | 613.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @verizon.net | 14 Mar 2019 | sec | 15 |
| amazon.com | 613.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @verizon.net | 14 Mar 2019 | sec | \$15 |
| | | | | | | | | UPDATED 2FA BYPASS METHOD! | | | | | |
| amazon.com | 238.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @verizon.net | 14 Mar 2019 | sec | 5 |
| amazon.com | 224.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @verizon.net | 14 Mar 2019 | sec | 5 |
| amazon.com | 223.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @optonline.net | 14 Mar 2019 | sec | 5 |
| amazon.com | 215.00 | N/A | | Personal | Usa | N/A | | E-MAIL access only! Do a password reset to enter Amazon account. Balance = last order price. WITH UPDATED 2FA BYPASS METHOD! | HQ | @optonline.net | 14 Mar 2019 | sec | 5 |

Search X Search X +

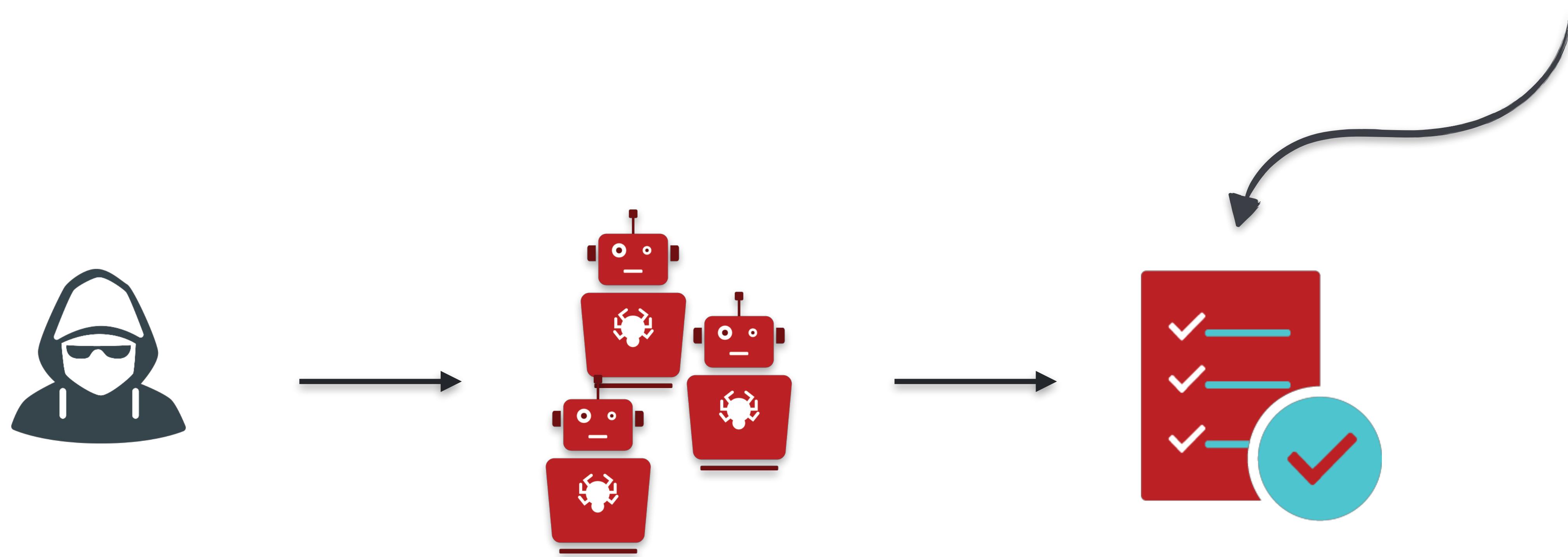
To page: 1 Go

| Shop | Balance | Points | Name | Type | Country | State | Zip | CC | Bank | Info | Last order | Mail domain | Uploaded | Seller | Price (\$): |
|--------------------|-------------|----------------|--------------|------------|-----------|-------|-----|------------|------------|------------|------------|-----------------|--------------------|----------------|---------------|
| sephora.com | 0.00 | 0.00 | kim | N/A | N/A | | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | karen | N/A | N/A | 07031 | | N/A | N/A | N/A | N/A | @aol.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | sandra | N/A | N/A | | | N/A | N/A | N/A | N/A | @cox.net | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | Christina | N/A | N/A | 27609 | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 235.00 | patty | N/A | N/A | 77043 | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 3.17 |
| sephora.com | 0.00 | 4121.00 | Janet | N/A | Us | | | N/A | N/A | N/A | N/A | @aol.com | 27 Feb 2019 | Mrtikov | \$22.6 |
| sephora.com | 0.00 | 0.00 | shelley | N/A | Us | | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 37.00 | Sophia | N/A | N/A | | | N/A | N/A | N/A | N/A | @aol.com | 27 Feb 2019 | Mrtikov | 2.18 |
| sephora.com | 0.00 | 0.00 | tiffany | N/A | N/A | | | N/A | N/A | N/A | N/A | @aol.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | sharon | N/A | N/A | | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | joseph | N/A | N/A | 33018 | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 0.00 | shery | N/A | N/A | | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |
| sephora.com | 0.00 | 4121.00 | Janet | N/A | Us | | | N/A | N/A | N/A | N/A | @aol.com | 27 Feb 2019 | Mrtikov | 22.6 |
| sephora.com | 0.00 | 20.00 | Page | N/A | N/A | | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2.1 |
| sephora.com | 0.00 | 0.00 | Kelly | N/A | N/A | 21227 | | N/A | N/A | N/A | N/A | @yahoo.com | 27 Feb 2019 | Mrtikov | 2 |

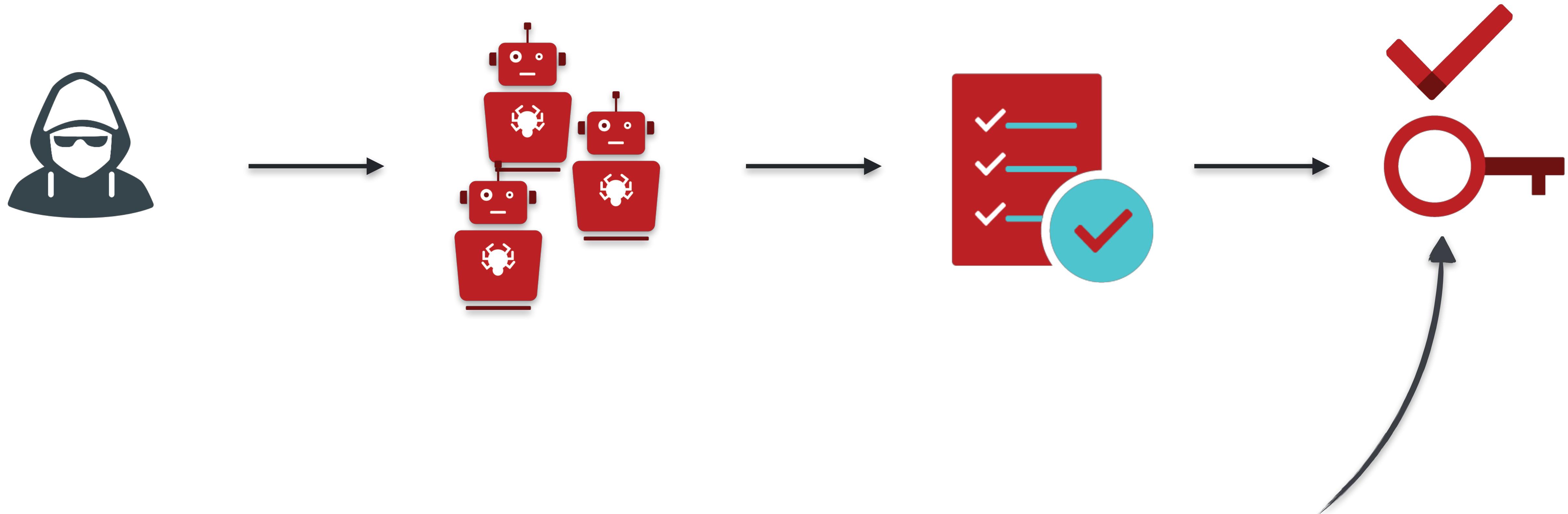


But Multi-Factor Authentication solves this, right?

The purpose of credential stuffing is to find valid accounts.

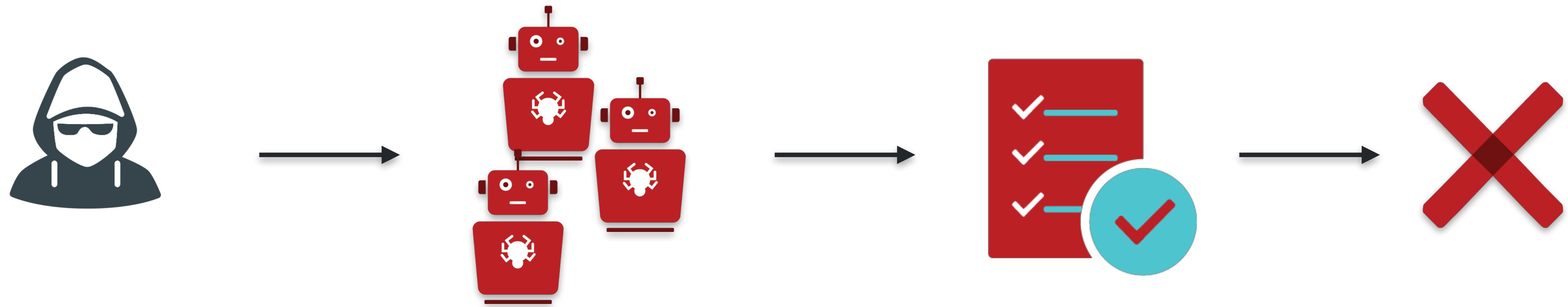


The purpose of credential stuffing is to find valid accounts.

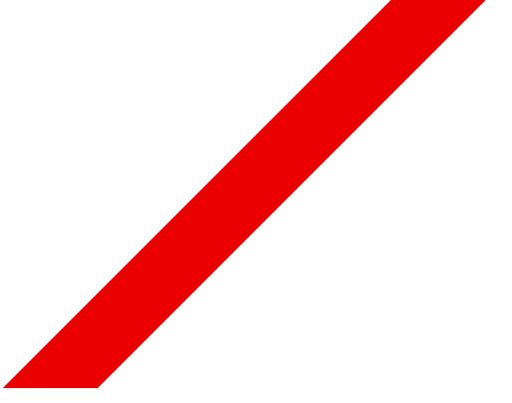


Valid accounts without MFA lead directly to account takeovers.

Valid accounts **MFA** do not lead directly to ATOs.



But the attacker still has valid accounts.



MFA is a hurdle that adds cost to the attack.

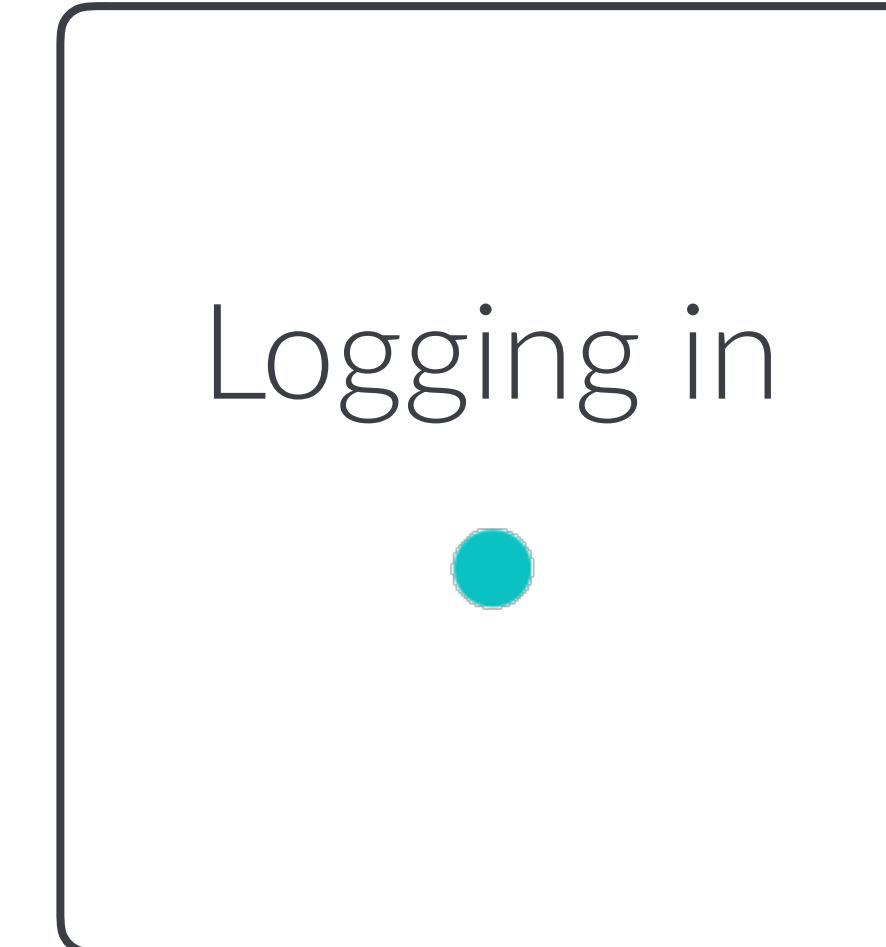
It is good but it is not a silver bullet.

All hurdles will be crossed if the value is there.

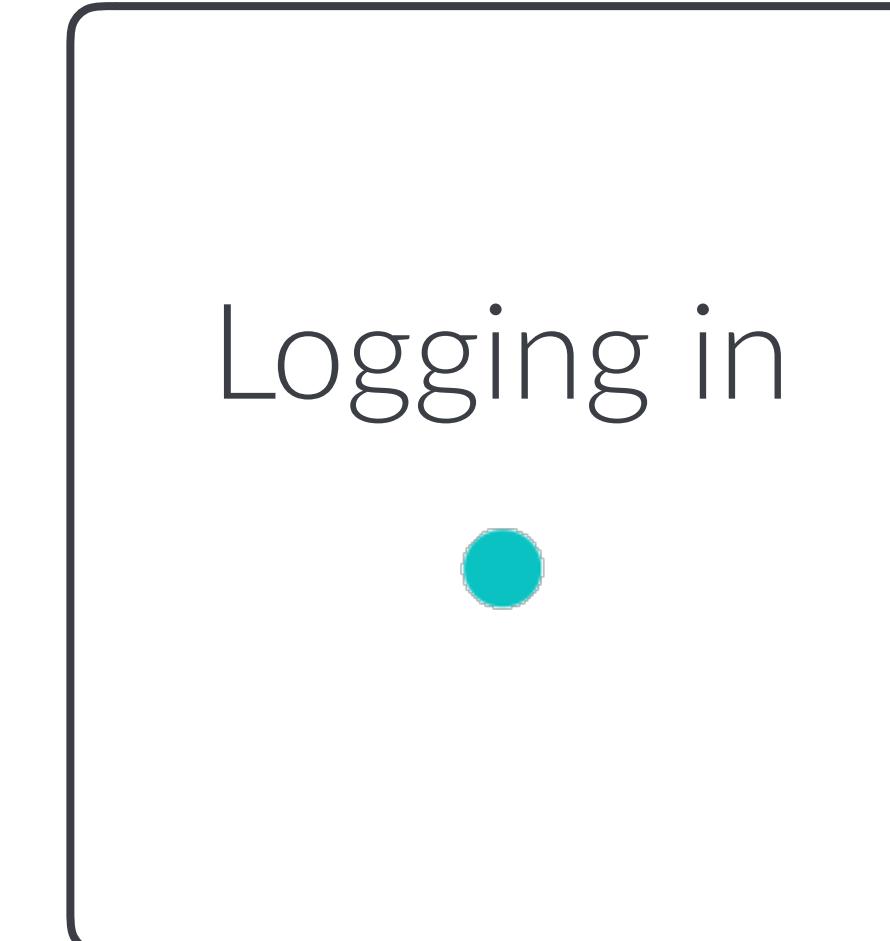


A wireframe-style diagram of a login form. At the top left is a placeholder icon for a user profile picture. To its right is a rectangular input field labeled "Username" containing the text "barry@gmail.com". Below this is another rectangular input field labeled "Password" containing the text "*****". At the bottom right of the form is a blue rectangular button labeled "Submit". A mouse cursor arrow is positioned directly over the "Submit" button, indicating it is the target of the click.

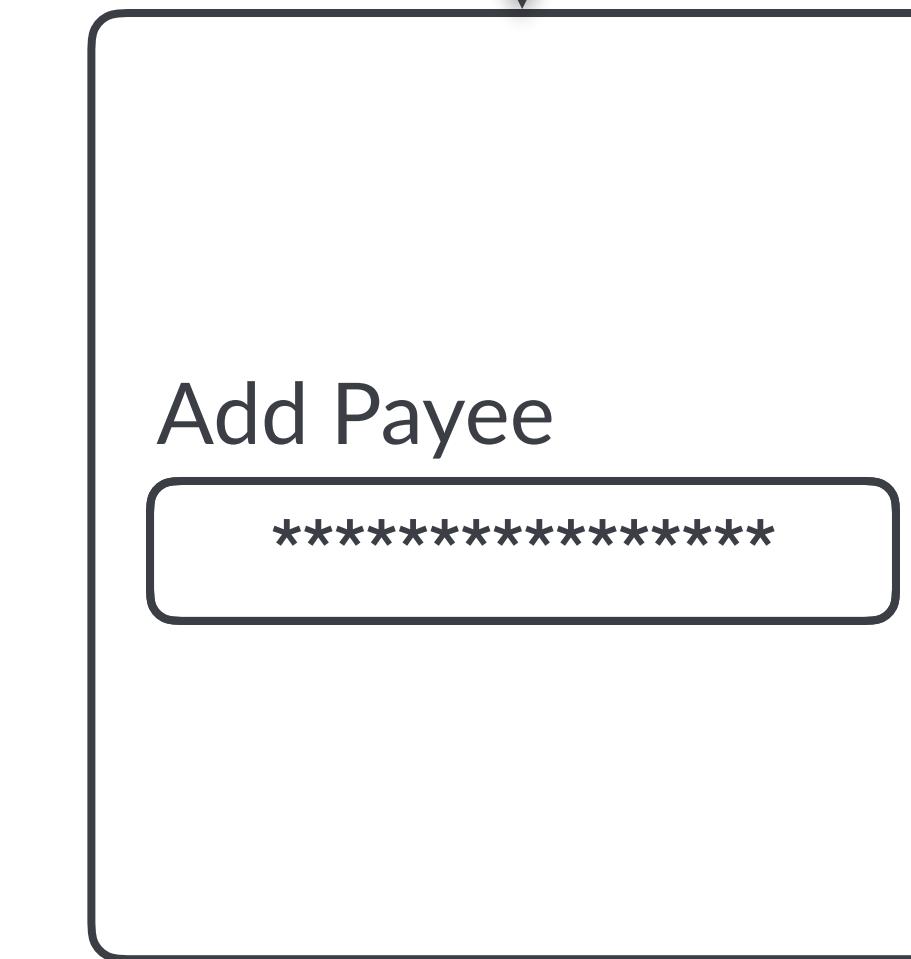
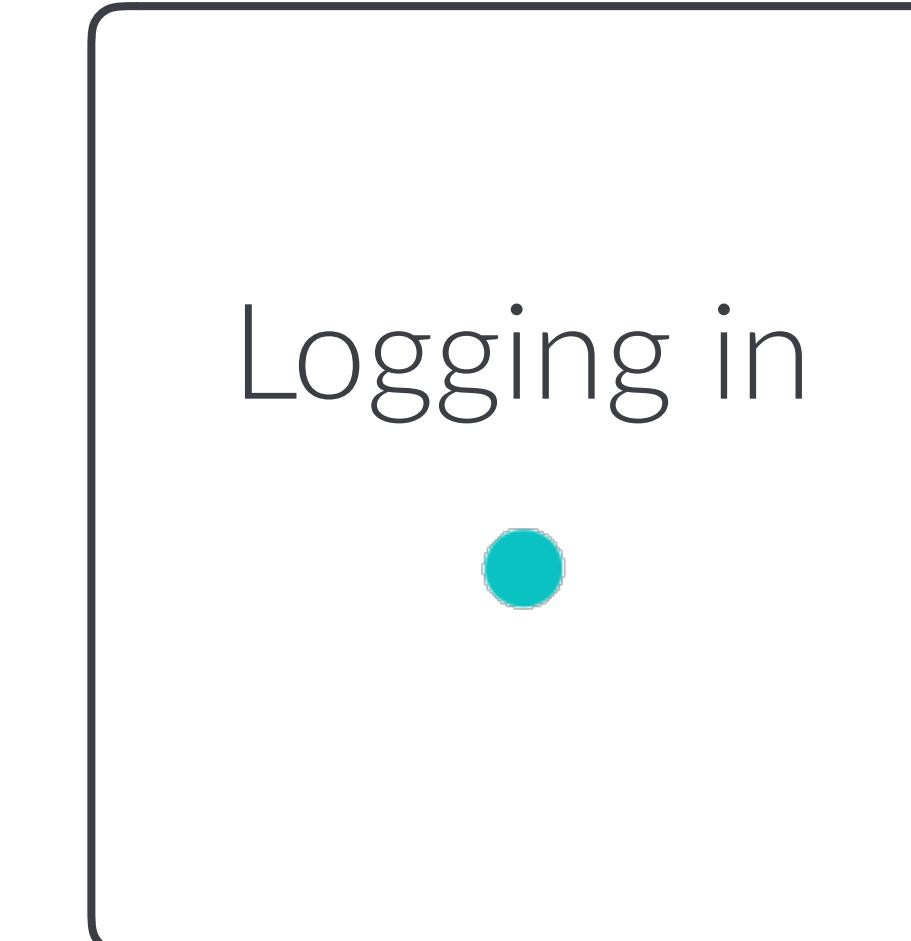
Take Barry, for example. An everyday user logging in as normal.



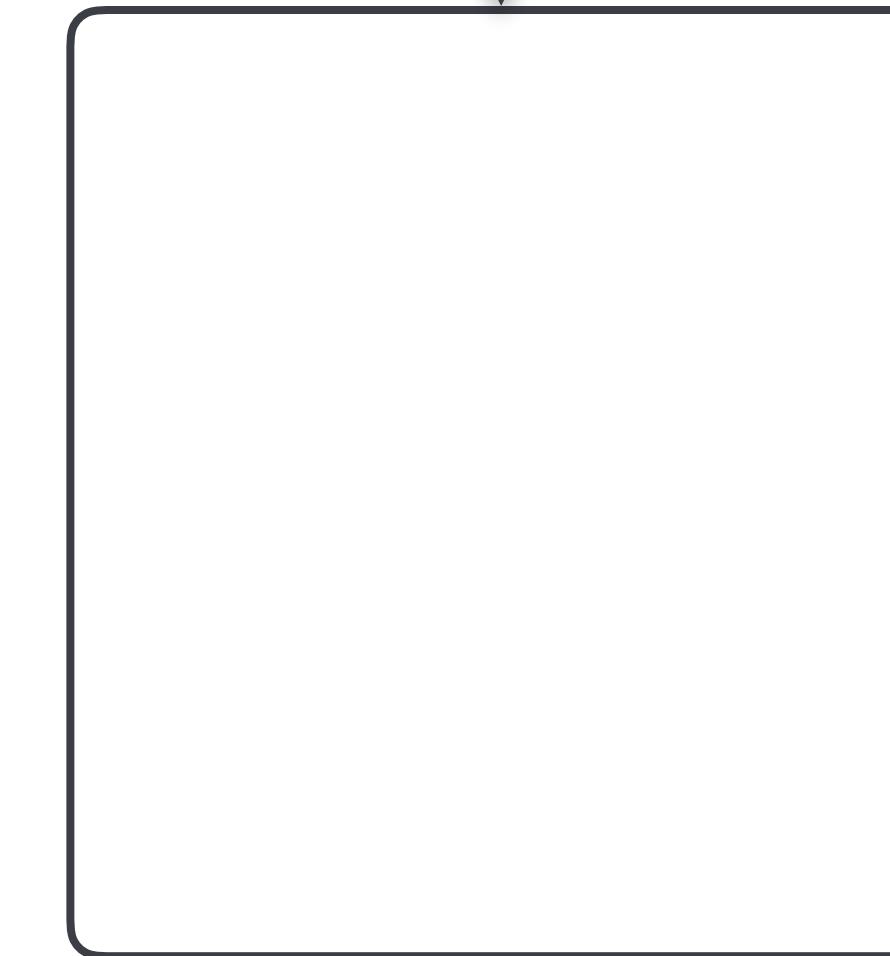
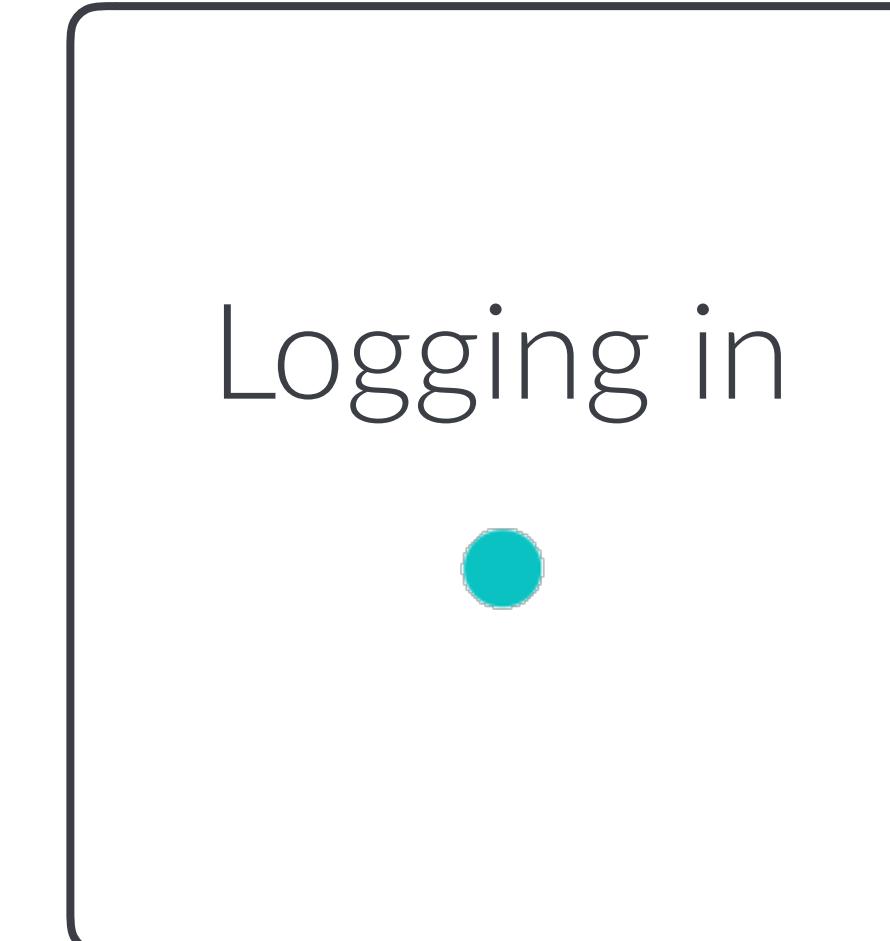
Barry experiences a login delay but he is used to that.



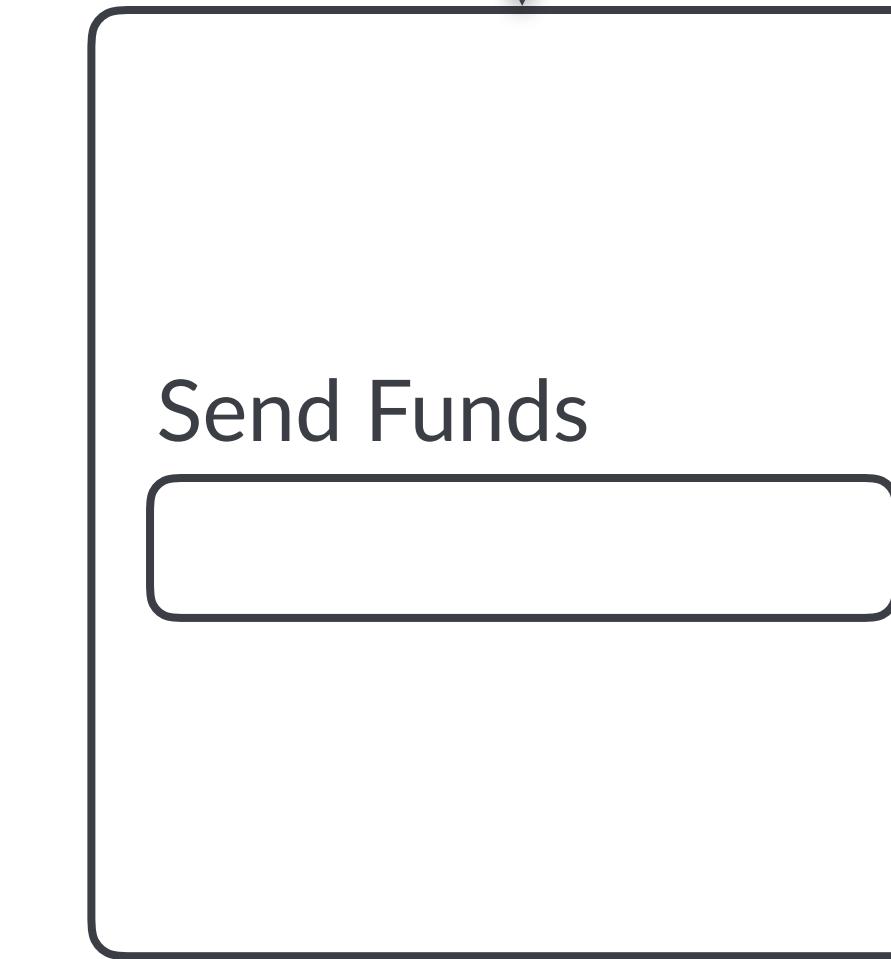
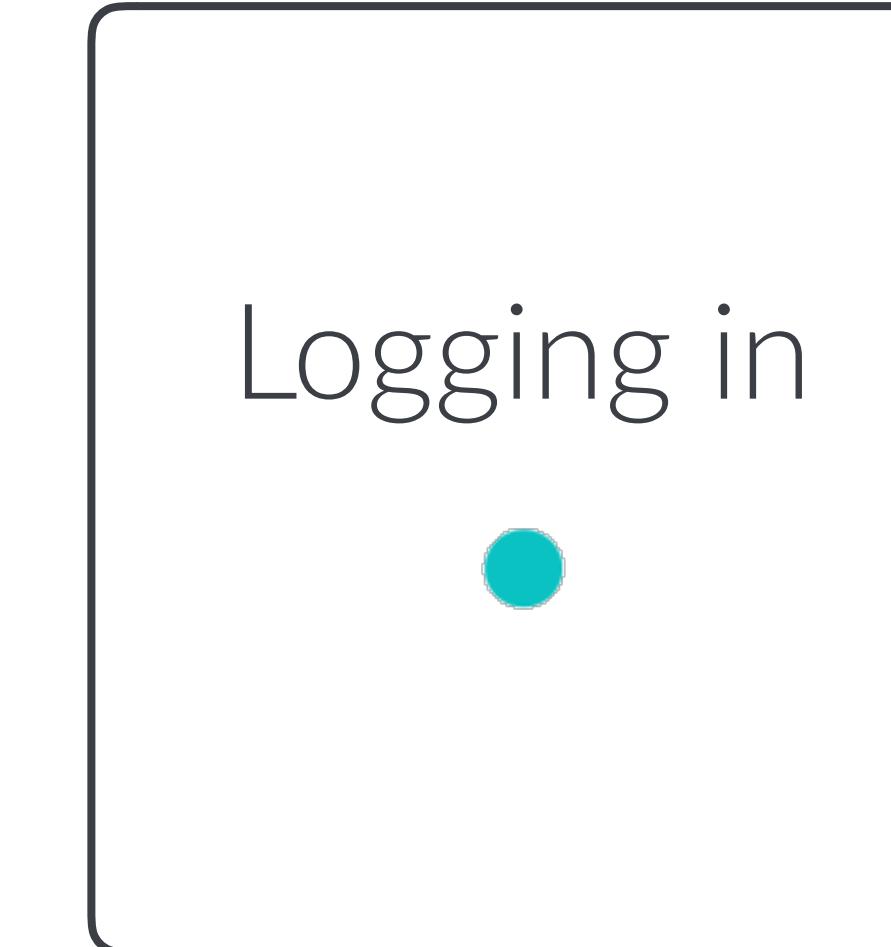
Uh oh. An injected script or malicious extension kicks in.



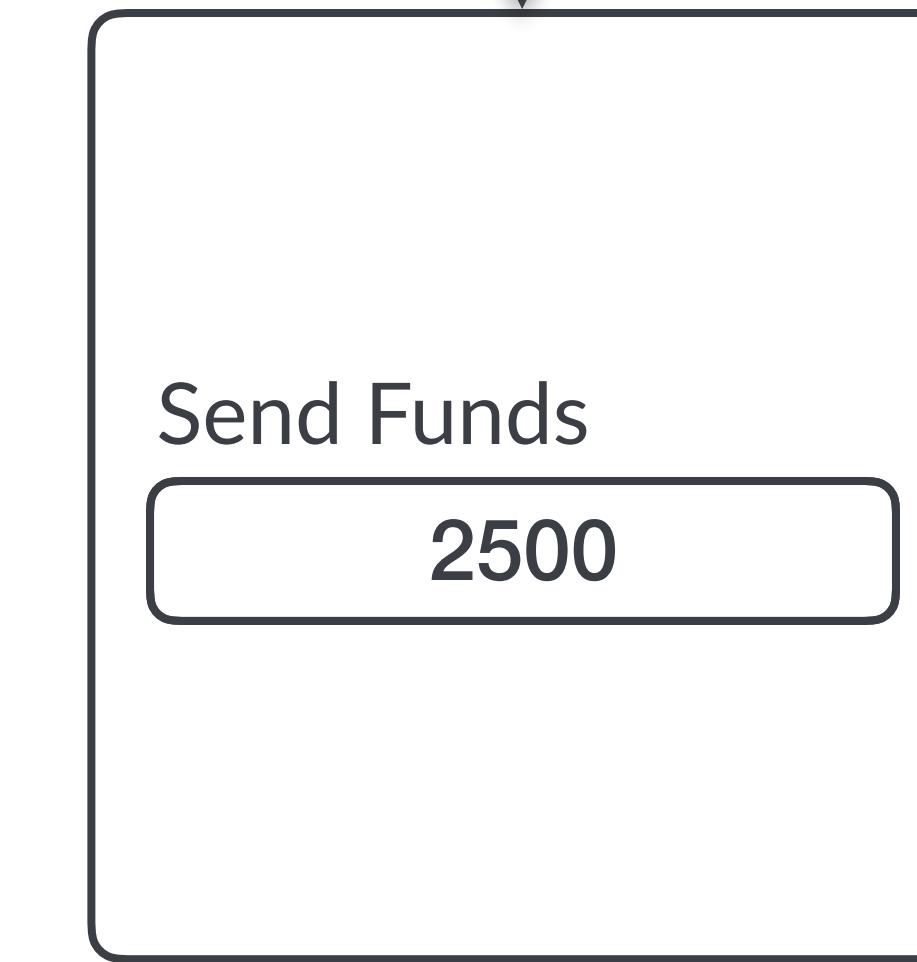
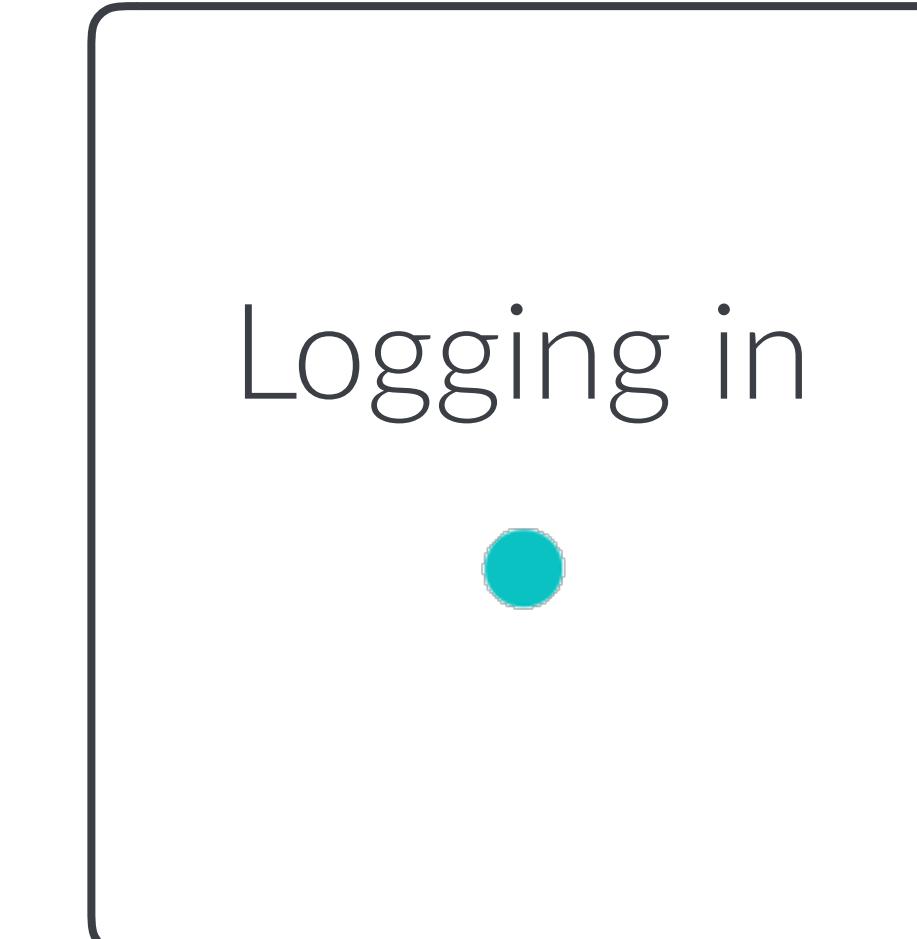
This script tries to add a new payee...



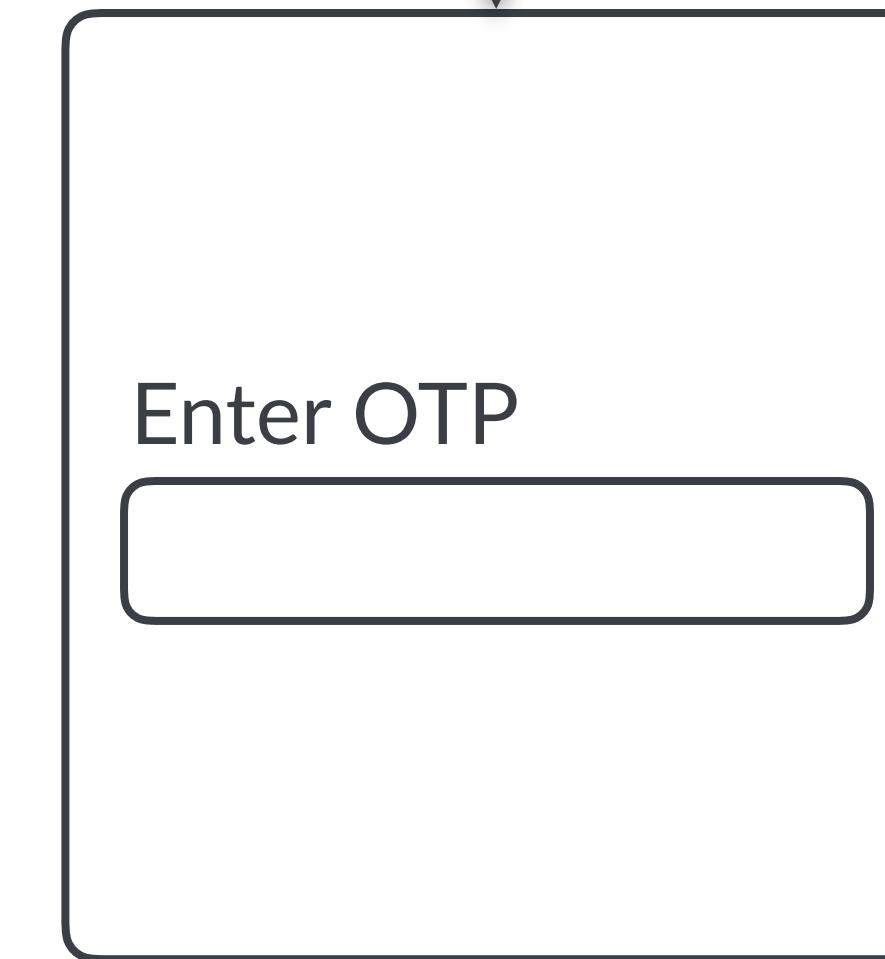
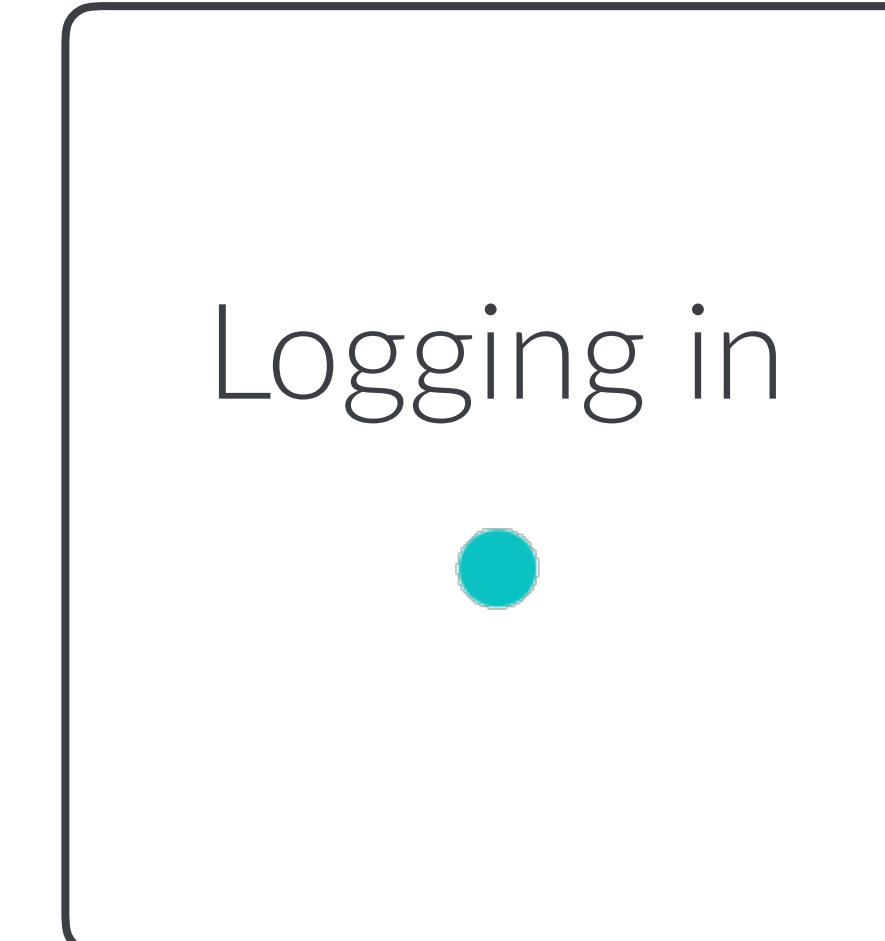
...which is successful because why wouldn't it be?



The script then attempts to transfer funds.



Usually a flat number or percentage, whichever is lower.



The risk score is too high, time to ask for a one-time password.



Enter OTP

A rectangular input field containing the text "072344".

Enter OTP

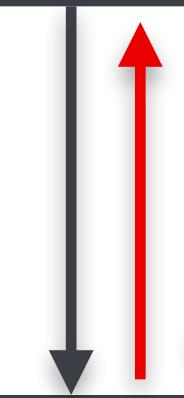
An empty rectangular input field.

But wait, Barry's used to this flow and doesn't see a problem.



Enter OTP

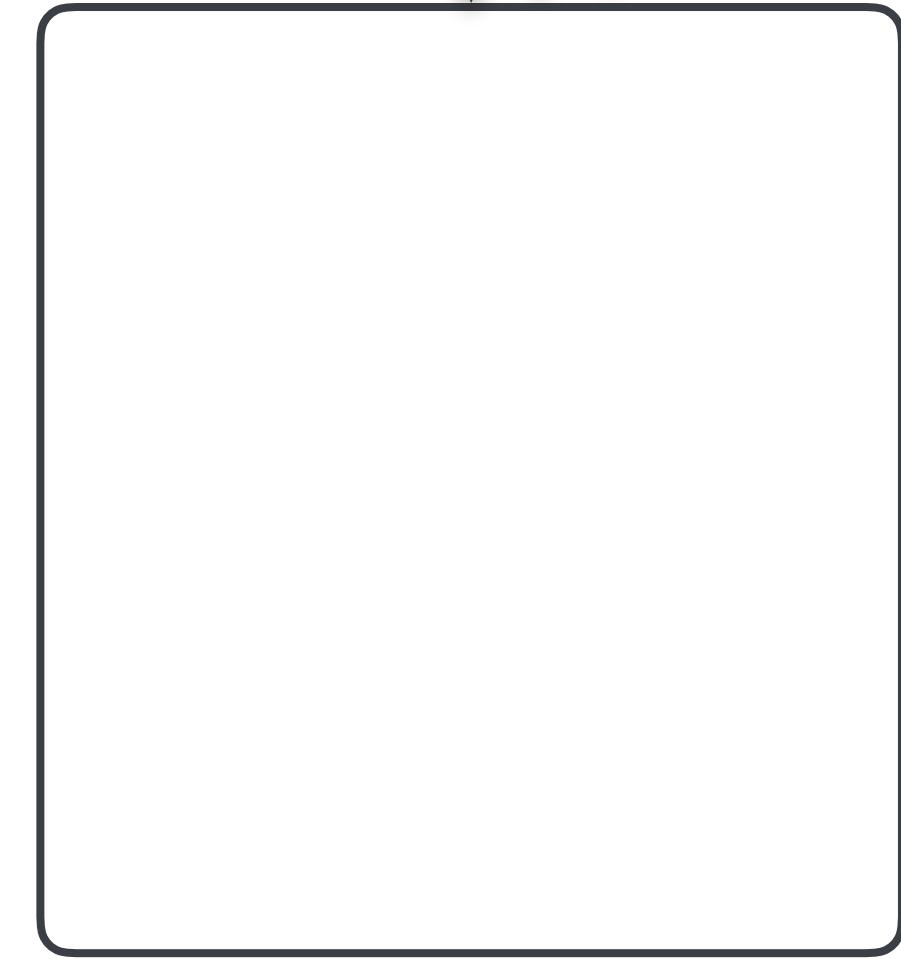
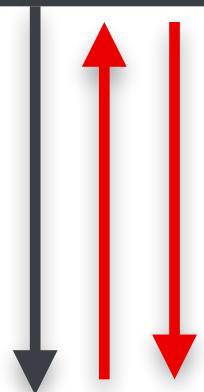
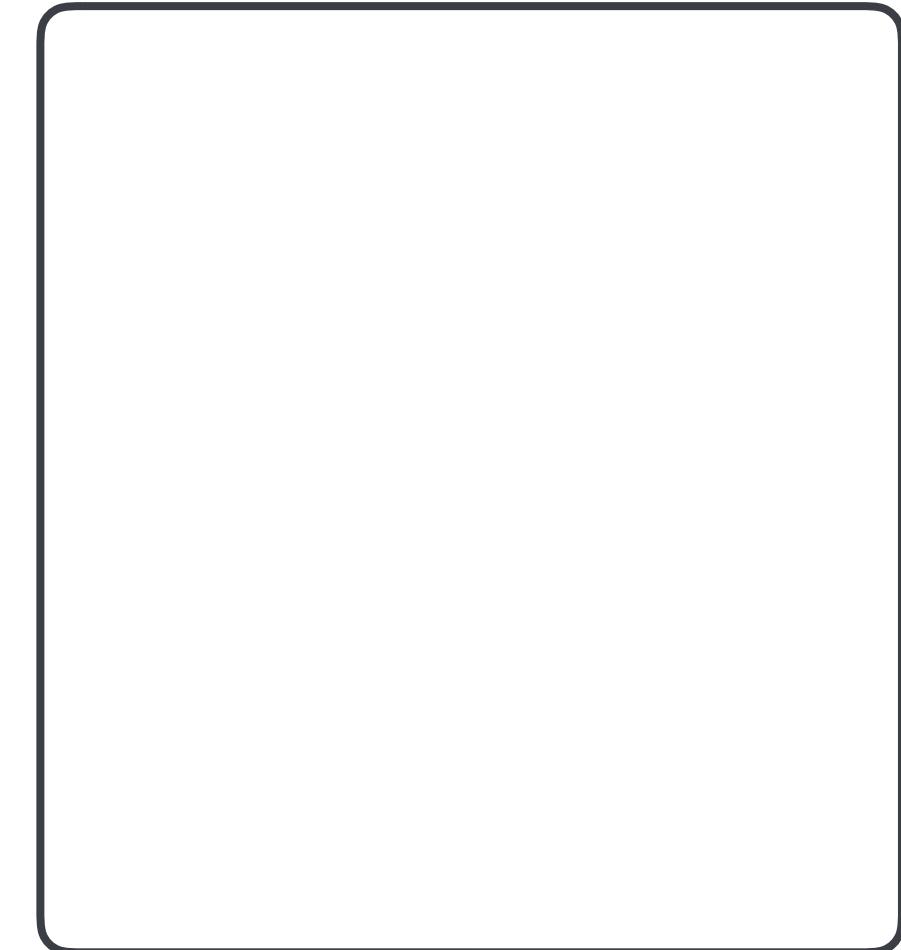
072344



Enter OTP

072344

The script grabs the token and funds are transferred.



CW Spammers buy Chrome extensi X +

← → C https://www.computerworld.com/article/2486674/spammers-buy-chrome-extensions-and-turn-them... ☆ Incognito 🕵️ :

DON'T MISS: Review: Samsung's Galaxy S10+ · How AI is changing office suites · IDG TECH(talk) ·

COMPUTERWORLD
FROM IDG

INSIDER ➔ Sign In | Register

Home > Internet

NEWS

Spammers buy Chrome extensions and turn them into adware

Two developers who sold their popular Chrome extensions saw them misused for aggressive advertising



By **Lucian Constantin**

Romania Correspondent, IDG News Service | JANUARY 20, 2014 08:03 AM PT

Malicious code found in npm package event-stream

https://snyk.io/blog/malicious-code-found-in-npm-package-event-stream/ Incognito

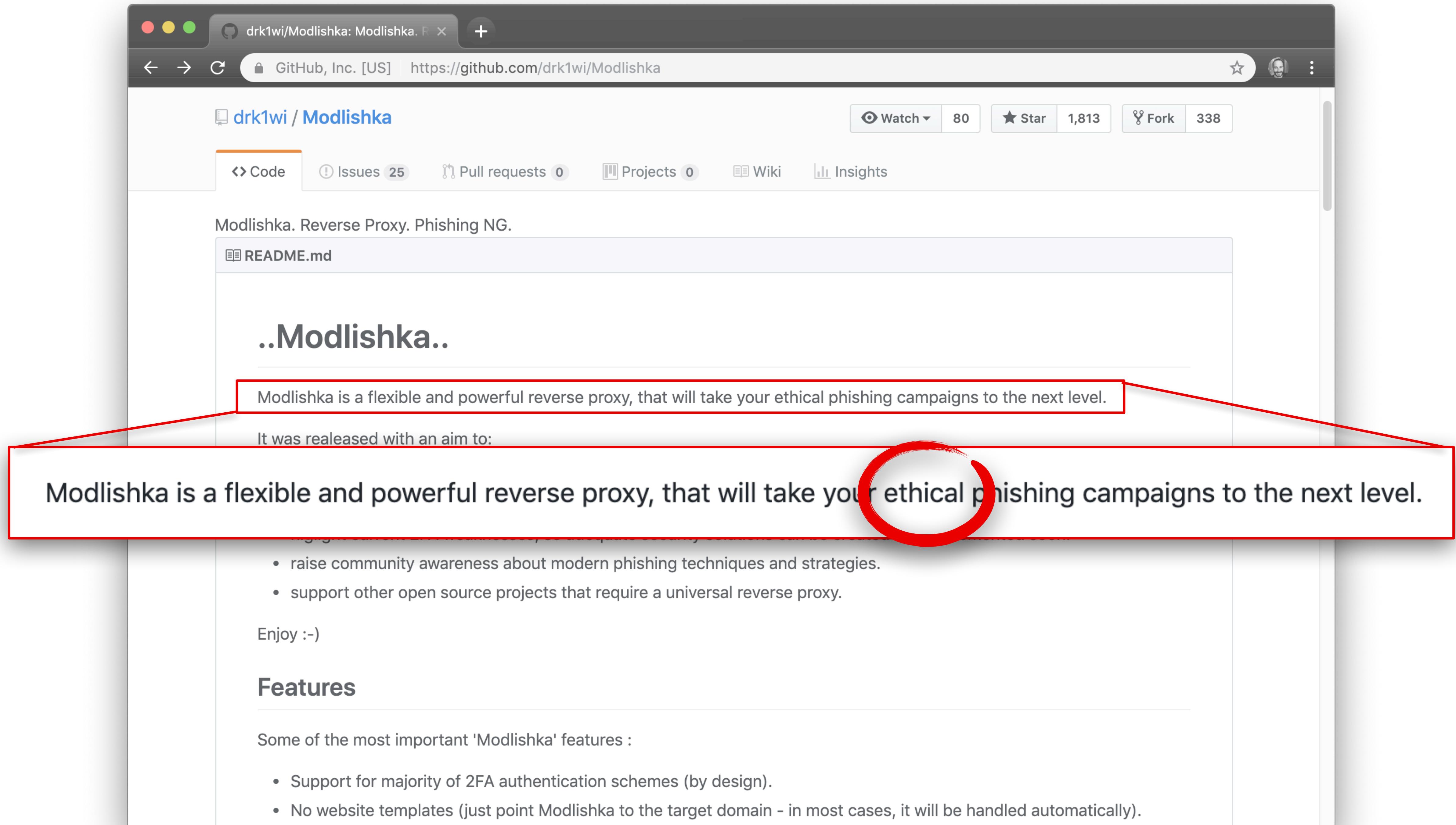
snyk

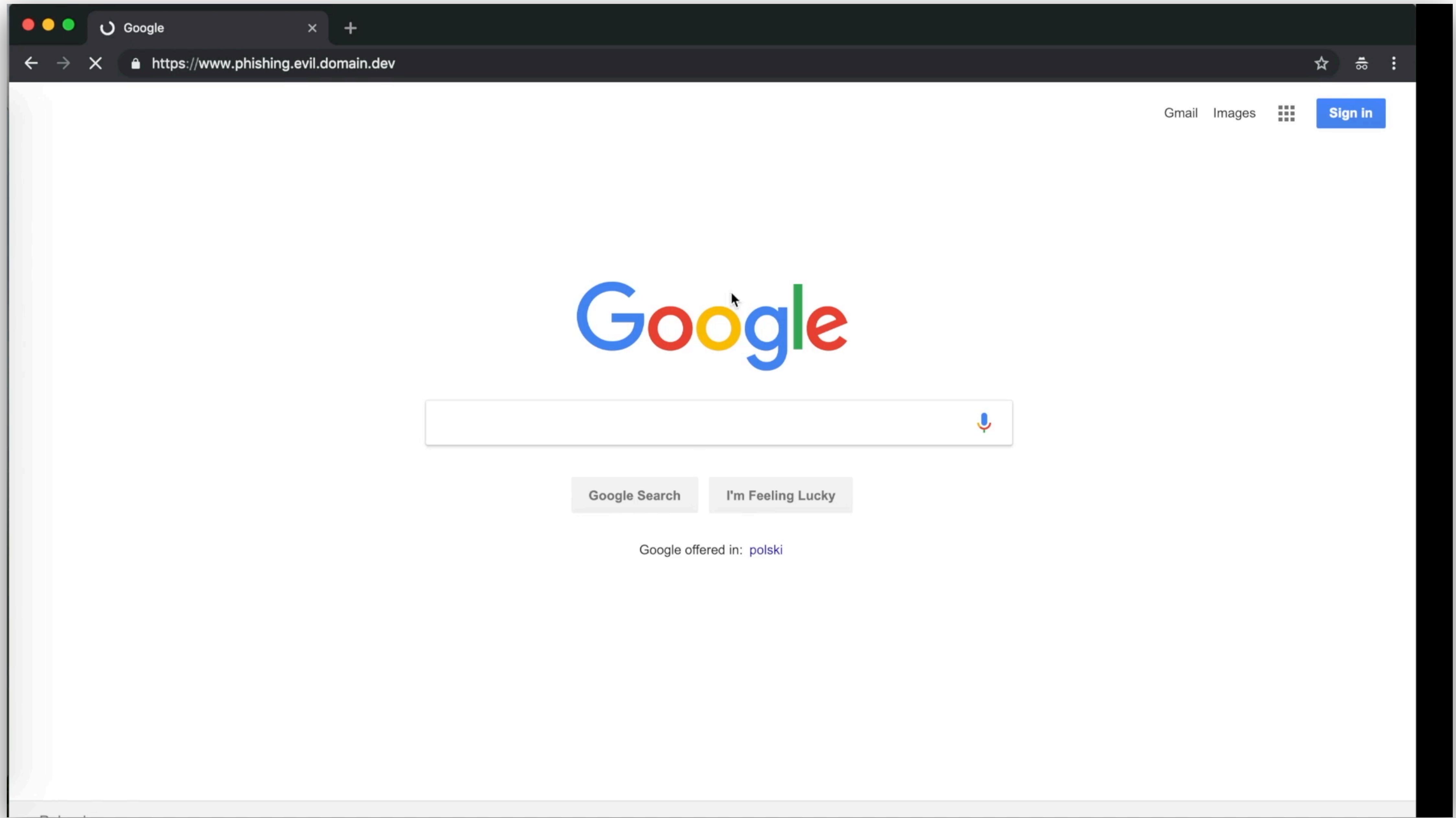
```
.4': '2016-07-17T07:24:09.783Z',
.5': '2018-09-05T05:27:47.219Z',
.6': '2018-09-16T11:20:20.931Z',
.0': '2018-09-29T06:55:49.102Z'
```

Malicious code found in npm package event-stream downloaded 8 million times in the past 2.5 months

NOVEMBER 26, 2018 | IN VULNERABILITIES | BY DANNY GRANDER

"I'm mostly concerned about phishing"





Agenda

1

Current attack landscape

2

Attacks in detail

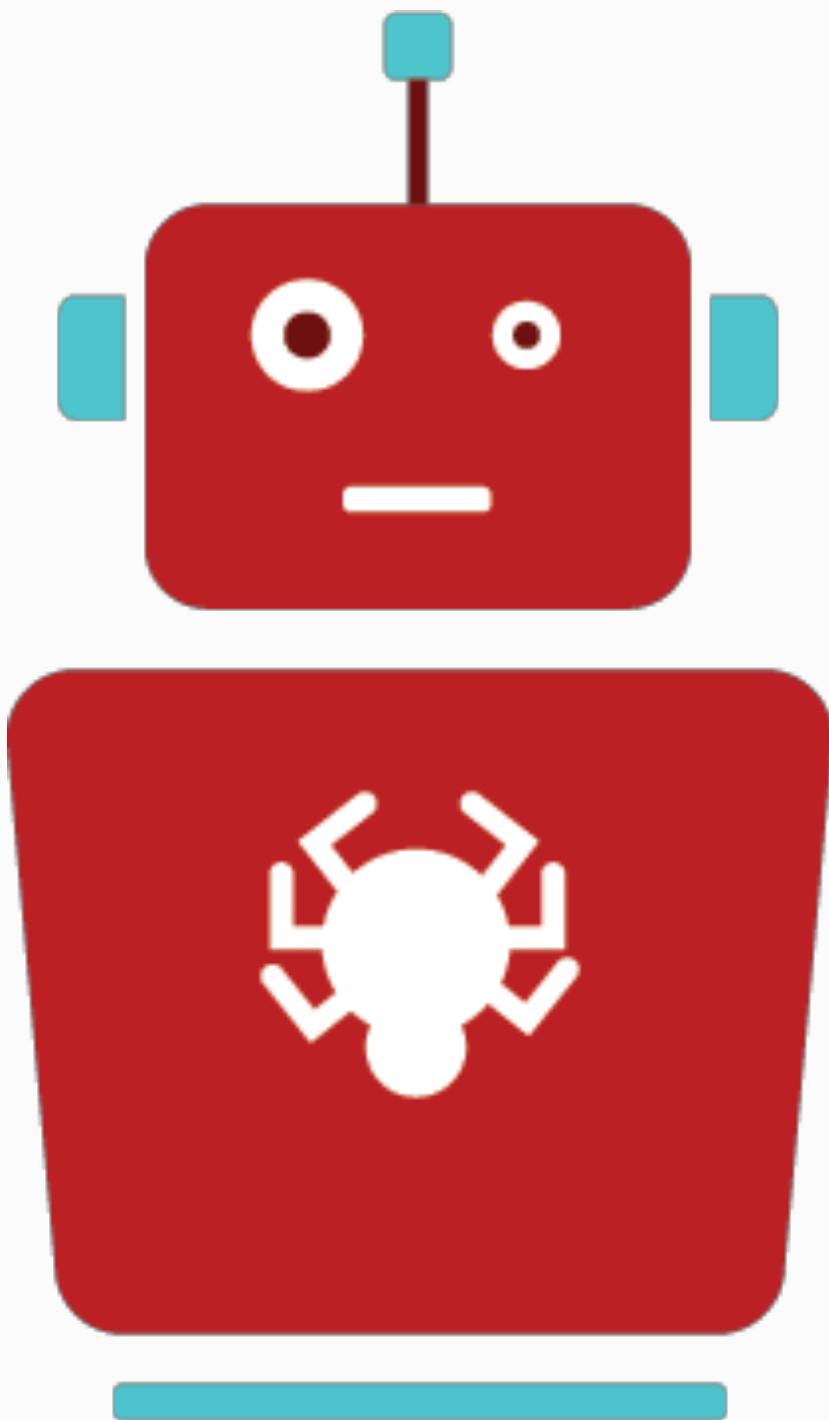
3

The arms race

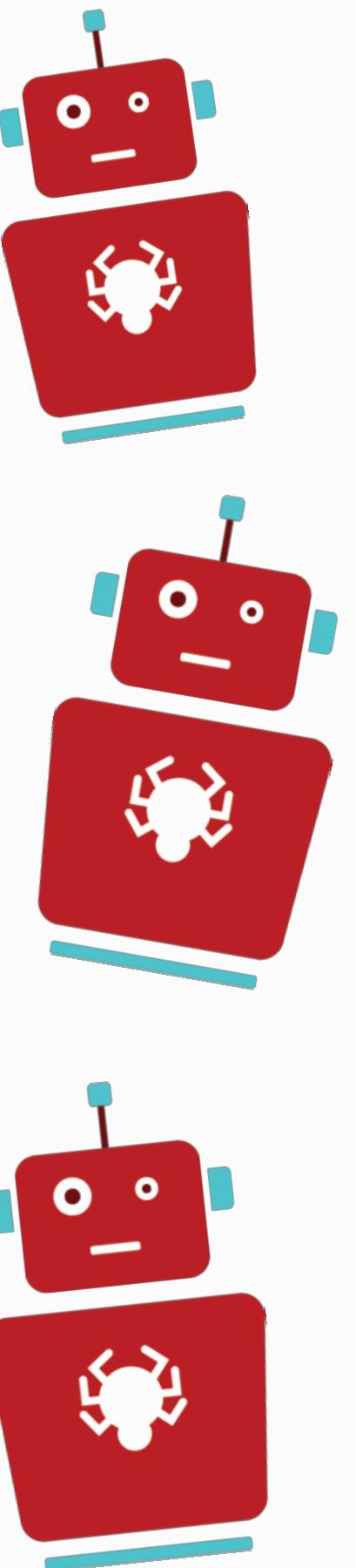
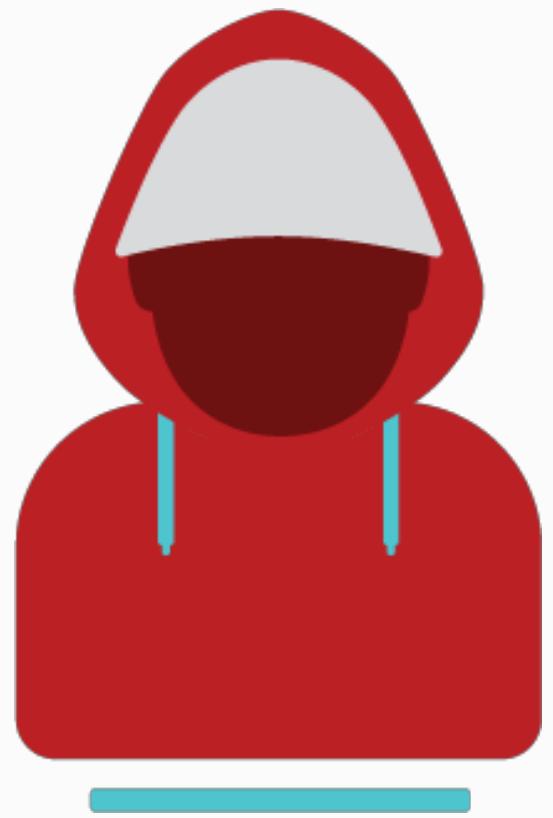
4

How do we adapt?

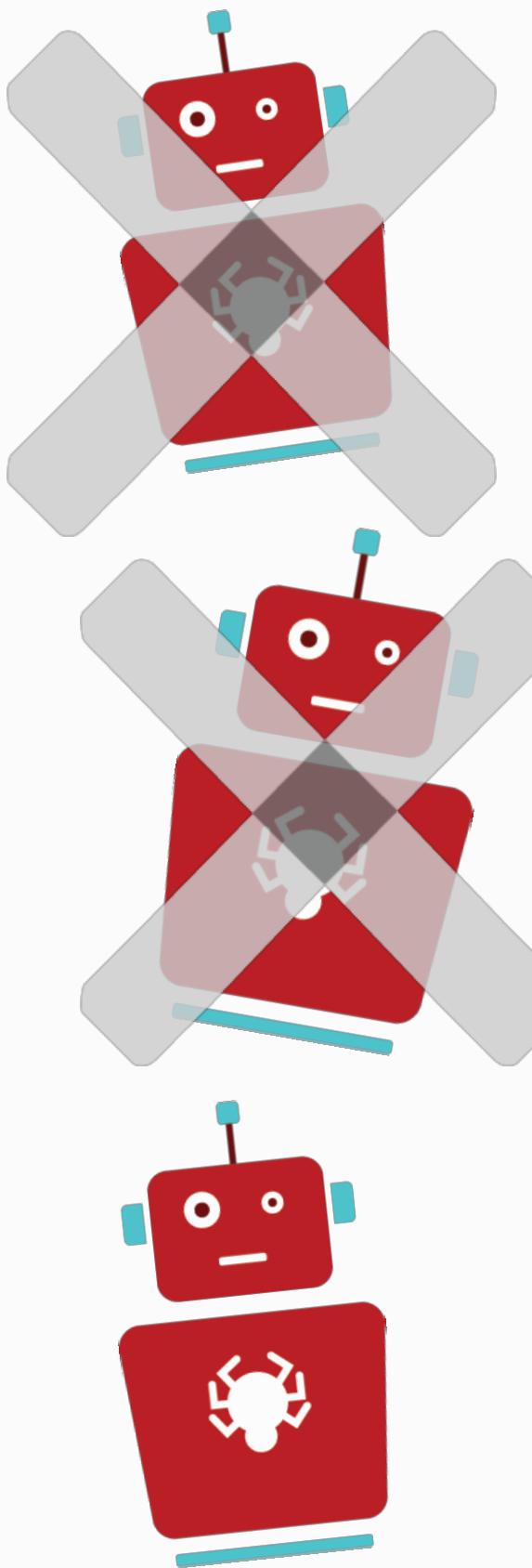
Millions of marketing dollars talk about bots...

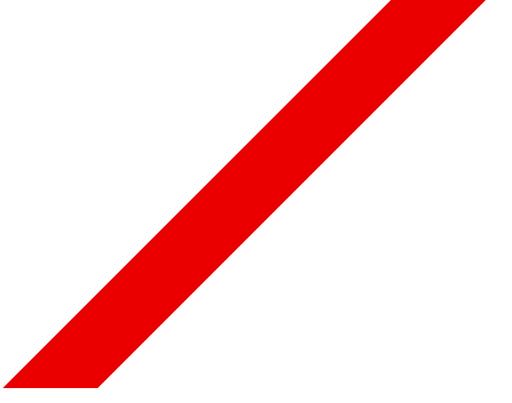


Bots are just one symptom, not the cause



Treating the symptom won't fix the problem





There are no silver bullets

There will always be a back and forth

Start attack with curl

Block on missing fields

Add fields until it the attack passes

Block on header order

Retool with SentryMBA

Require JavaScript execution

Retool with headless chrome

Block on browser fingerprint

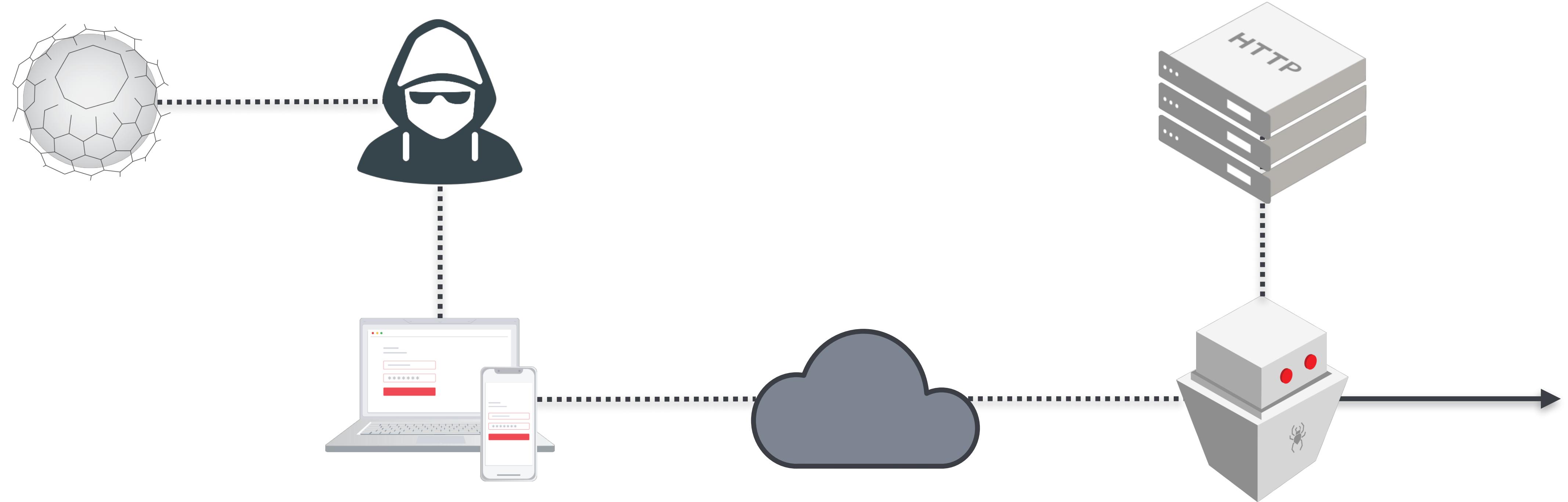
Randomize fingerprint

Block on framework signature

Modify framework signature

Implement AI based mitigation

Implement AI based evasion

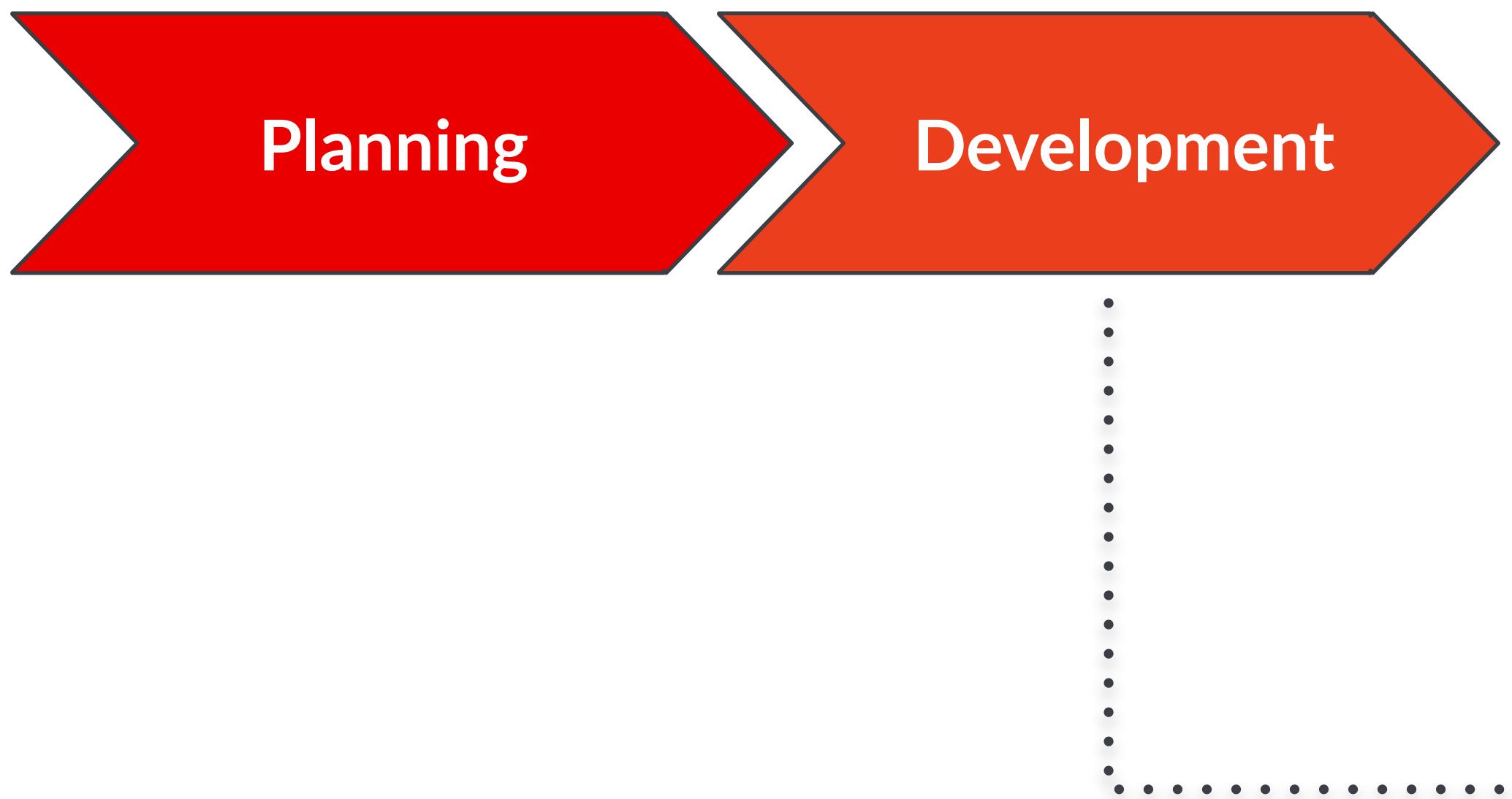


It's not as simple as blocking a bot.
It's targeting what will cost the attacker the most. Over and over again.

The Software Development Lifecycle

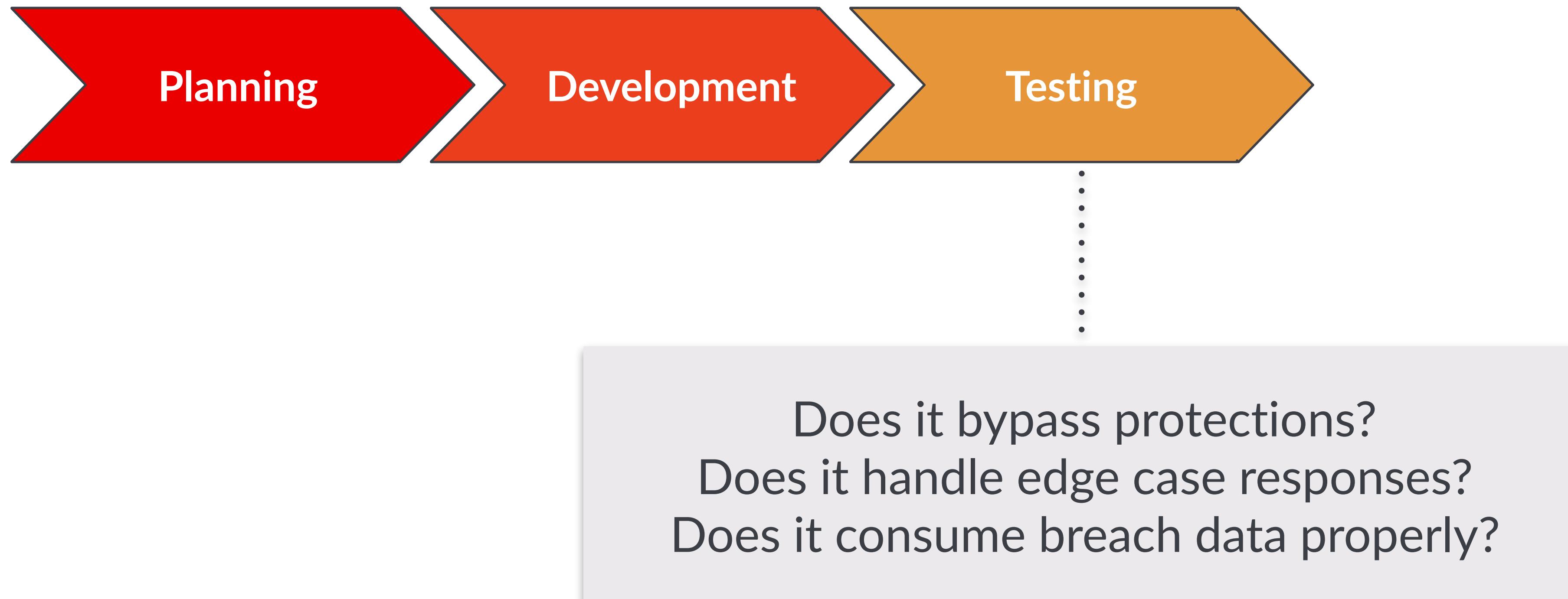


The Software Development Lifecycle

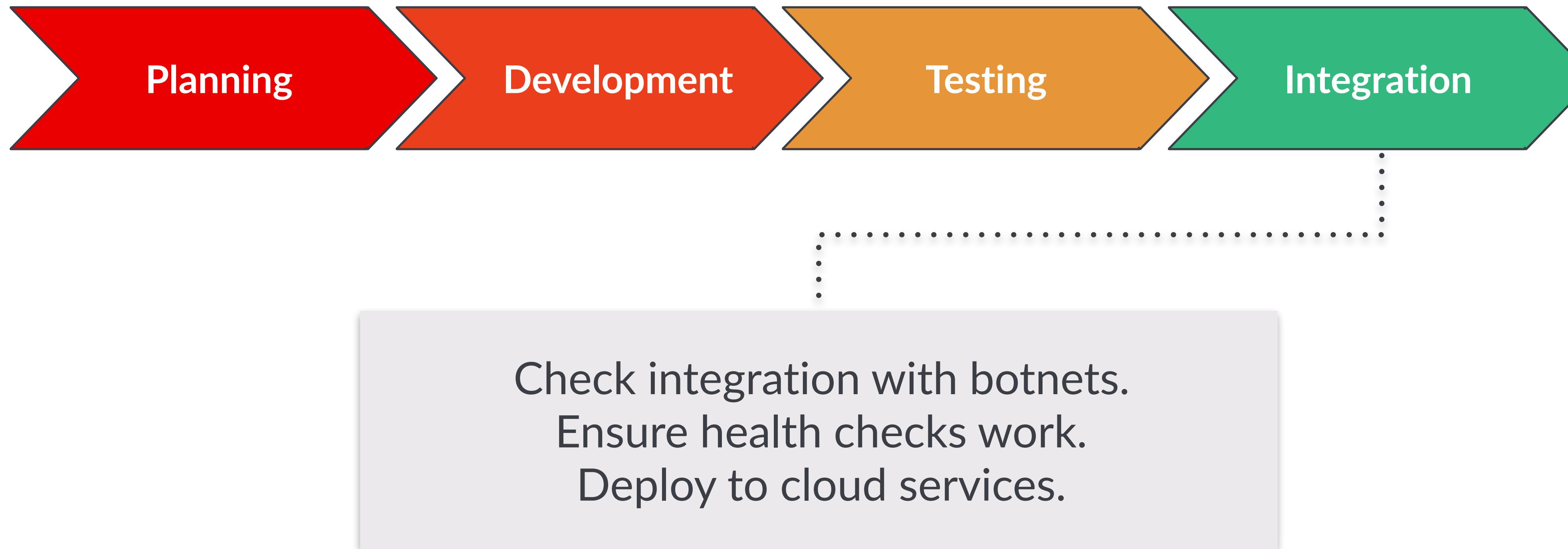


Investment in a framework of choice.
Custom development against a site.
Building in proxy/botnet hooks.

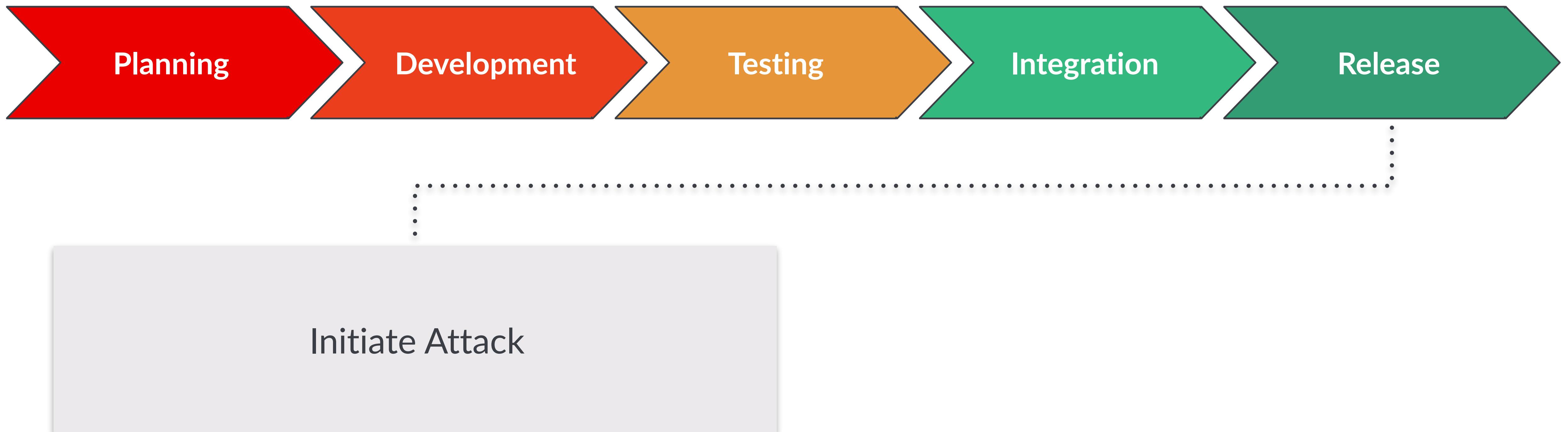
The Software Development Lifecycle



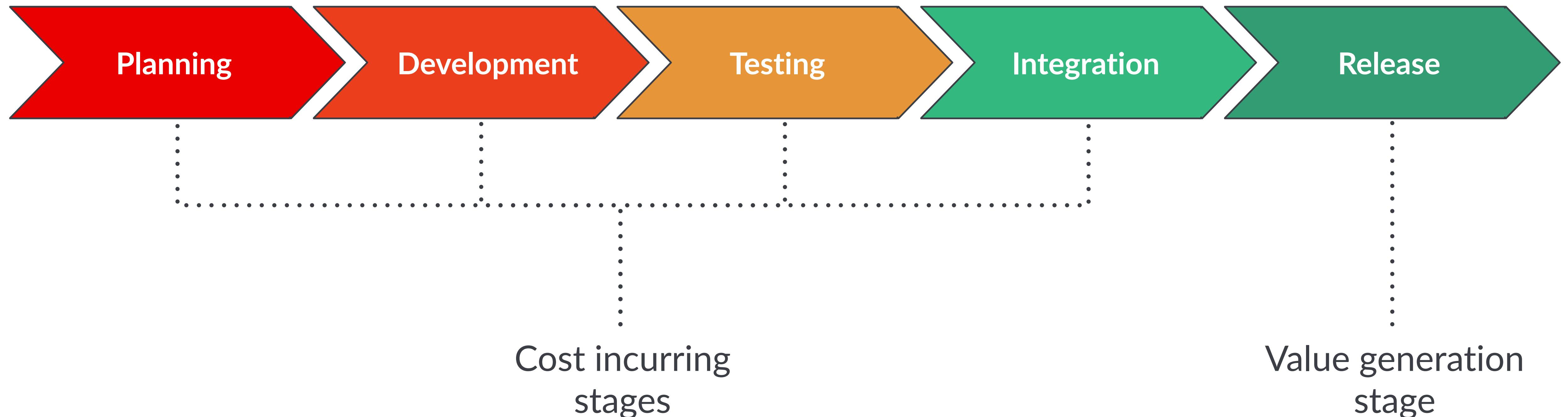
The Software Development Lifecycle



The Software Development Lifecycle



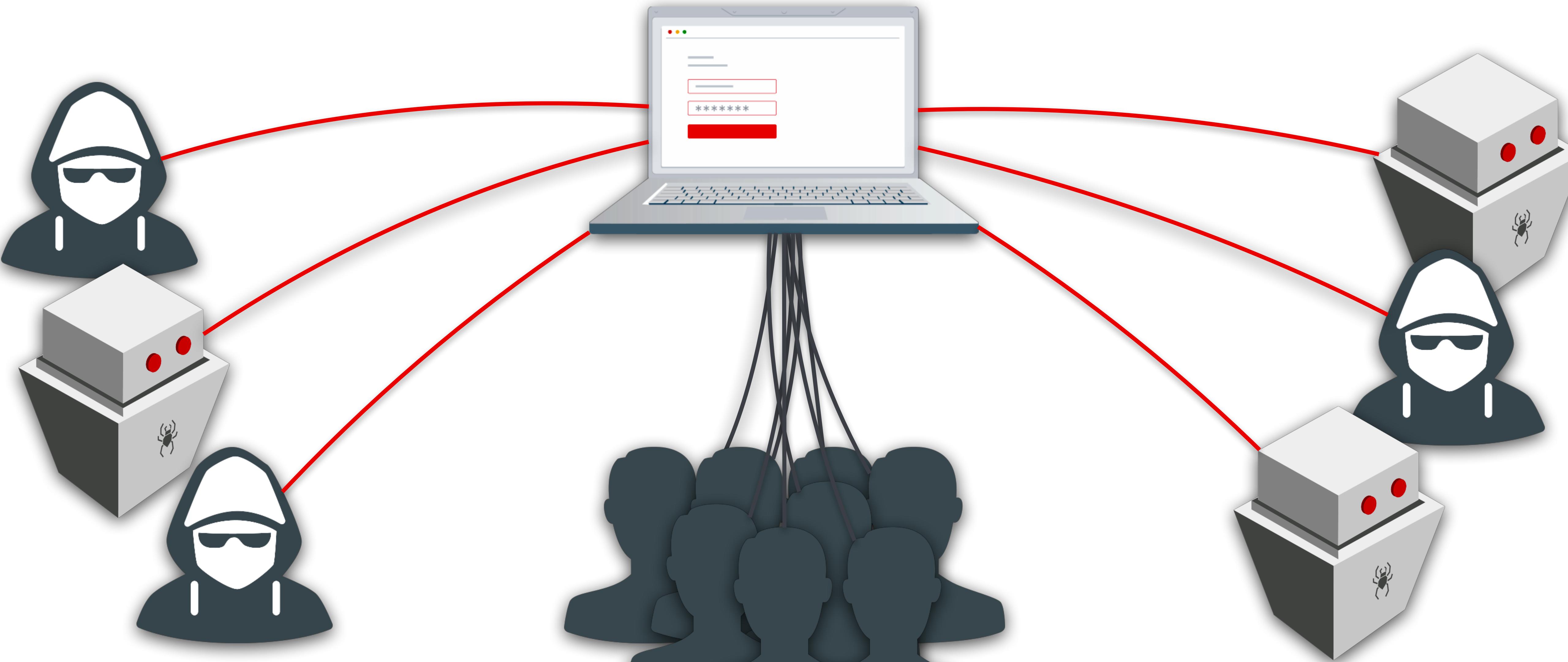
The Software Development Lifecycle



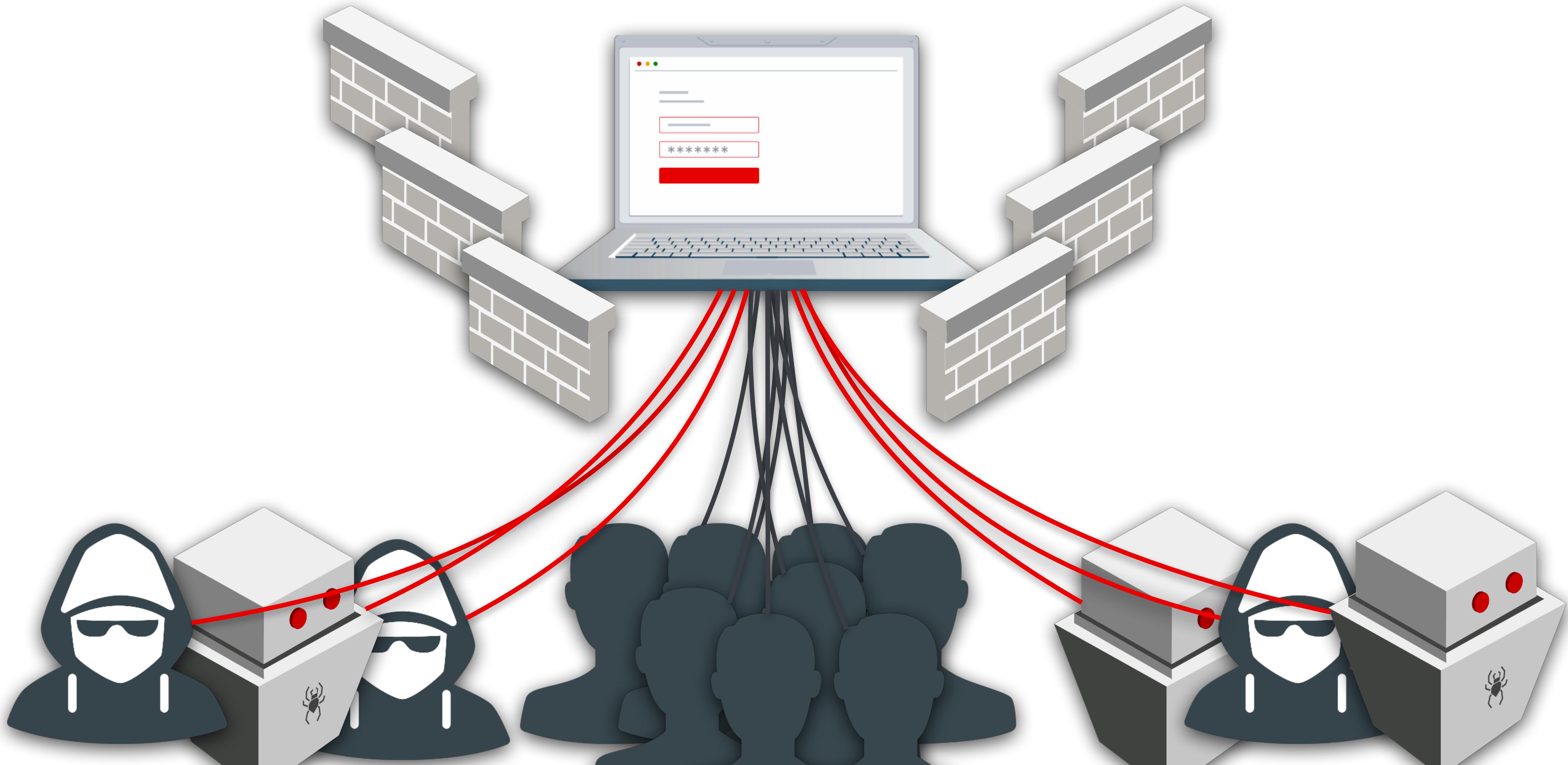
REMEMBER THE
WALLS WE PUT UP?

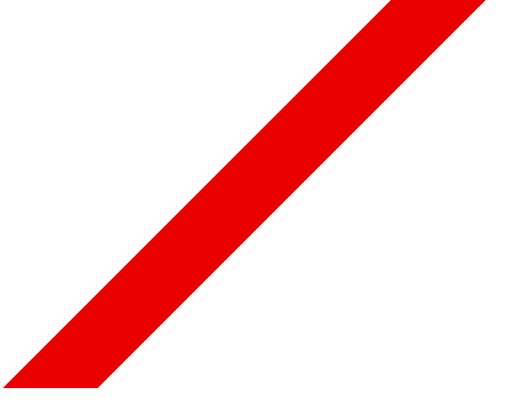


Without feedback, attacks don't evolve.



With feedback the attackers learn how to evolve the tools





This is a problem well suited for AI on the attacker's side.

Getting better is a matter of getting the right data.



Apps Installed On Millions Of Android Phones Track...



Apps Installed On Millions Of Android Phones Tracked User Behavior To Execute A Multimillion-Dollar Ad Fraud Scheme

A BuzzFeed News investigation uncovered a sophisticated ad fraud scheme involving more than 125 Android apps and websites, some of which were targeted at kids.



Craig Silverman

BuzzFeed News Reporter

Posted on October 23, 2018, at 1:07 p.m. ET



Tweet



Share



Copy



BuzzFeed News

Apps Installed On Millions Of Android Phones Track...



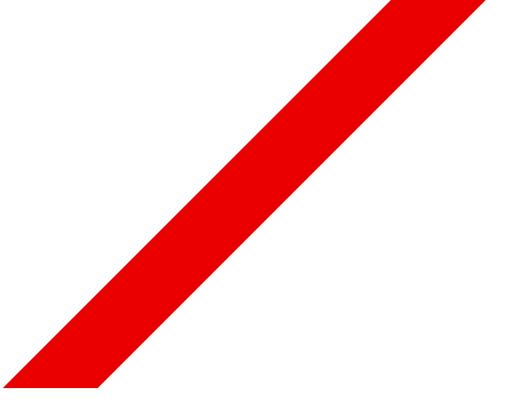
Involvement in the scheme estimates it has stolen hundreds of millions of dollars from brands whose ads were shown to bots instead of actual humans. (A full list of the apps, the websites, and their associated companies connected to the scheme can be found in [this spreadsheet](#).)

One way the fraudsters find apps for their scheme is to acquire legitimate apps through We

One way the fraudsters find apps for their scheme is to acquire legitimate apps through We Purchase Apps and transfer them to shell companies. They then capture the behavior of the app's human users and program a vast network of bots to mimic it, according to analysis from Protected Media, a cybersecurity and fraud detection firm that analyzed the apps and websites at BuzzFeed News' request.

these apps were secretly tracked as they scrolled and clicked inside the application. By copying actual user behavior in the apps, the fraudsters were able to generate fake traffic that bypassed major fraud detection systems.

"This is not your run-of-the-mill fraud scheme," said Asaf Greiner, the CEO of Protected Media.



Any score based solution can be gamed.

M Researchers Created Fake 'Mas X

https://motherboard.vice.com/en_us/article/bjenyd/researchers-created-fake-master-fingerprints-to-unlock-smartphones

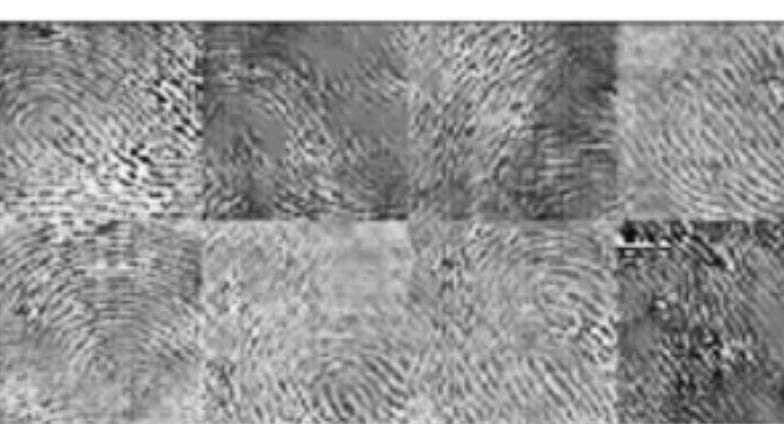
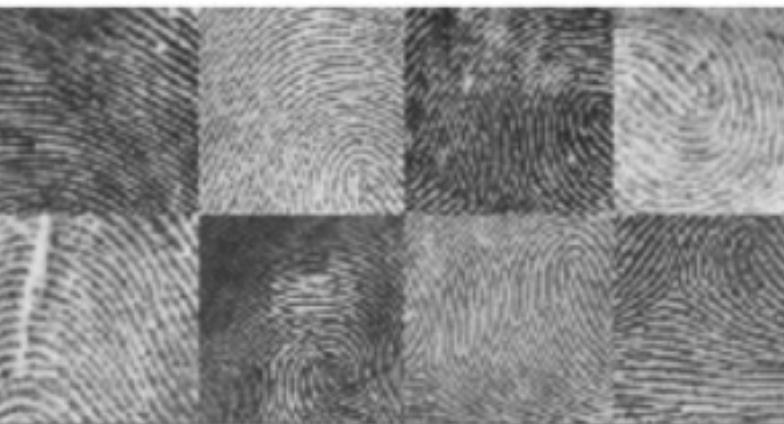
MOTHERBOARD Moveable Hacking Environment Space Gaming Health

VICE

Researchers Created Fake 'Master' Fingerprints to Unlock Smartphones

It's the same principle as a master key, but applied to biometric identification with a high rate of success.

SHARE  TWEET 



Agenda

1

Current attack landscape

2

Attacks in detail

3

The arms race

4

How do we adapt?

How do we adapt?

Optimize for speed to response.

- Security & Fraud teams need security features built into products.
- Defenses need dials and knobs that can be adjusted in real time.
- Embrace collaboration and trust across industry and with vendors.



How do we adapt?

Optimize for speed to response.

- Security & Fraud teams need security features built into products.
- Defenses need dials and knobs that can be adjusted in real time.
- Embrace collaboration and trust across industry and with vendors.



Develop hurdles that slow down attackers.

- Vary responses to attackers, limit direct feedback.
- Only enable countermeasures when they're necessary.
- Invest in client side protection.



How do we adapt?

Optimize for speed to response.

- Security & Fraud teams need security features built into products.
- Defenses need dials and knobs that can be adjusted in real time.
- Embrace collaboration and trust across industry and with vendors.



Develop hurdles that slow down attackers.

- Vary responses to attackers, limit direct feedback.
- Only enable countermeasures when they're necessary.
- Invest in client side protection.



Invest in continuous improvement.

- No silver bullet exists, don't be fooled into resting.
- Security & Fraud teams need to be proactive.
- Red teams are worth the cost.



How we do it at **SH=PE**

Optimize for speed to response.

- Fully scriptable platform. New defenses can be deployed in a day.
- Client-side code generated and composed on the fly.
- Shape collaborates with teams to demotivate sophisticated attackers.



Develop hurdles that slow down attackers.

- JavaScript is compiled to bytecode and randomized automatically.
- Tamper evidence checks alert Shape when a retool is occurring.
- Payloads can be delivered to an attacker to sabotage the development.

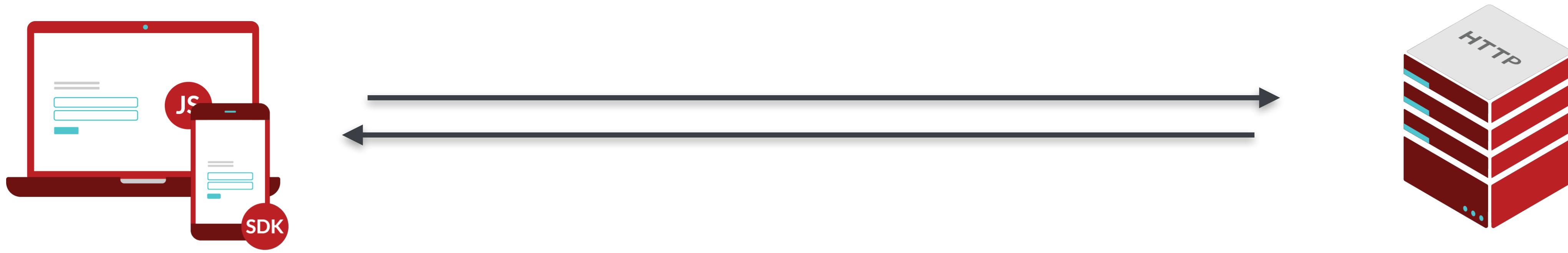


Invest in continuous improvement.

- Shape has been a member of TC-39 (the JS committee) for 4+ years.
- Shape works with browser vendors on specs like CSP, SRI, WebDriver.
- Shape's open source tools are being used to improve the entire internet.

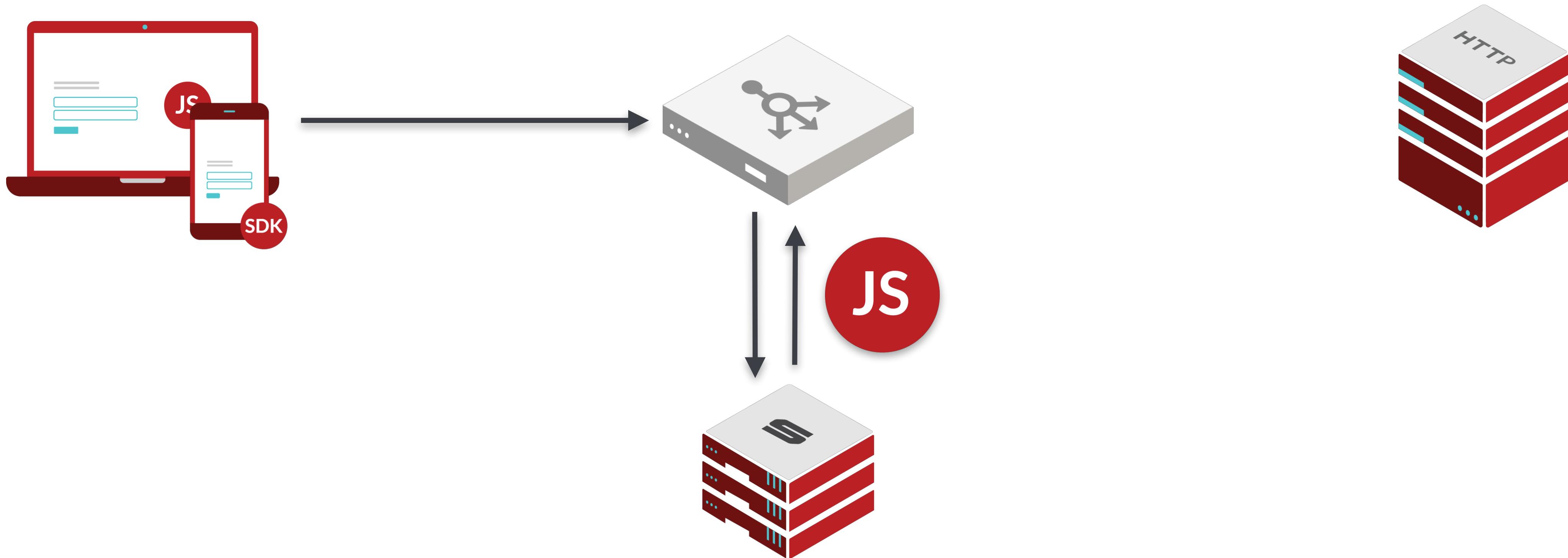


How we do it at Shape



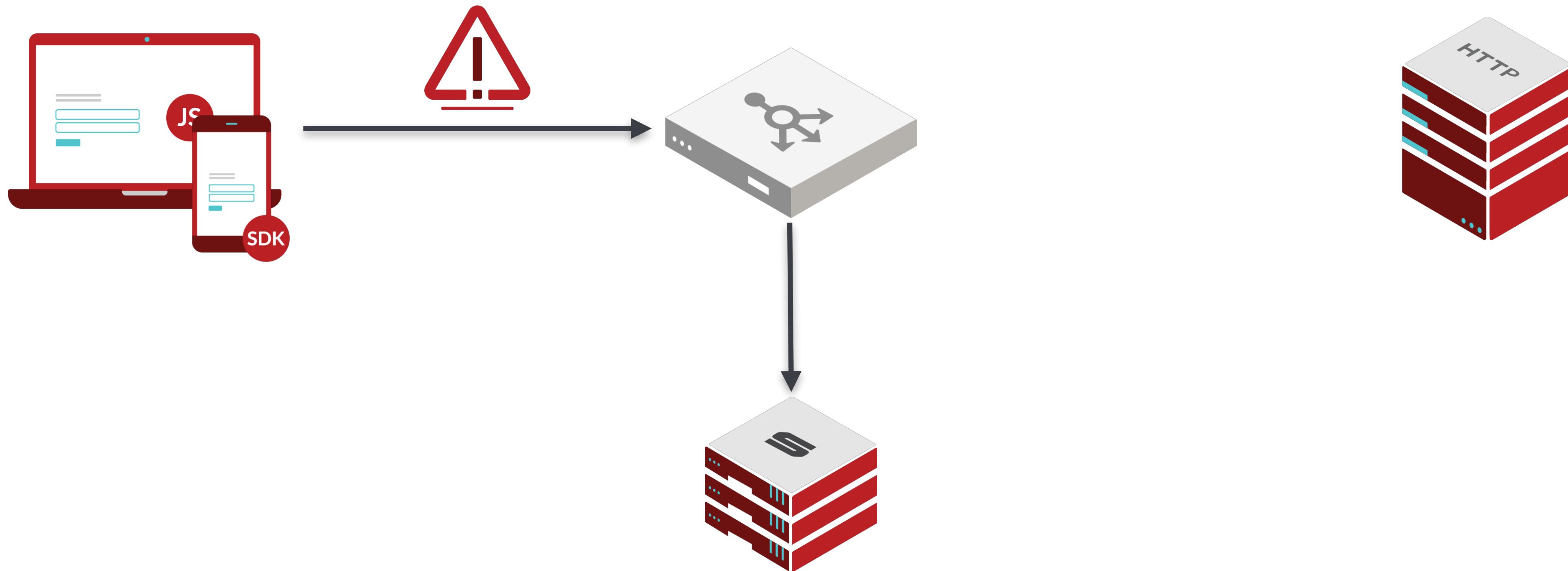
A client makes a request as normal

How we do it at Shape



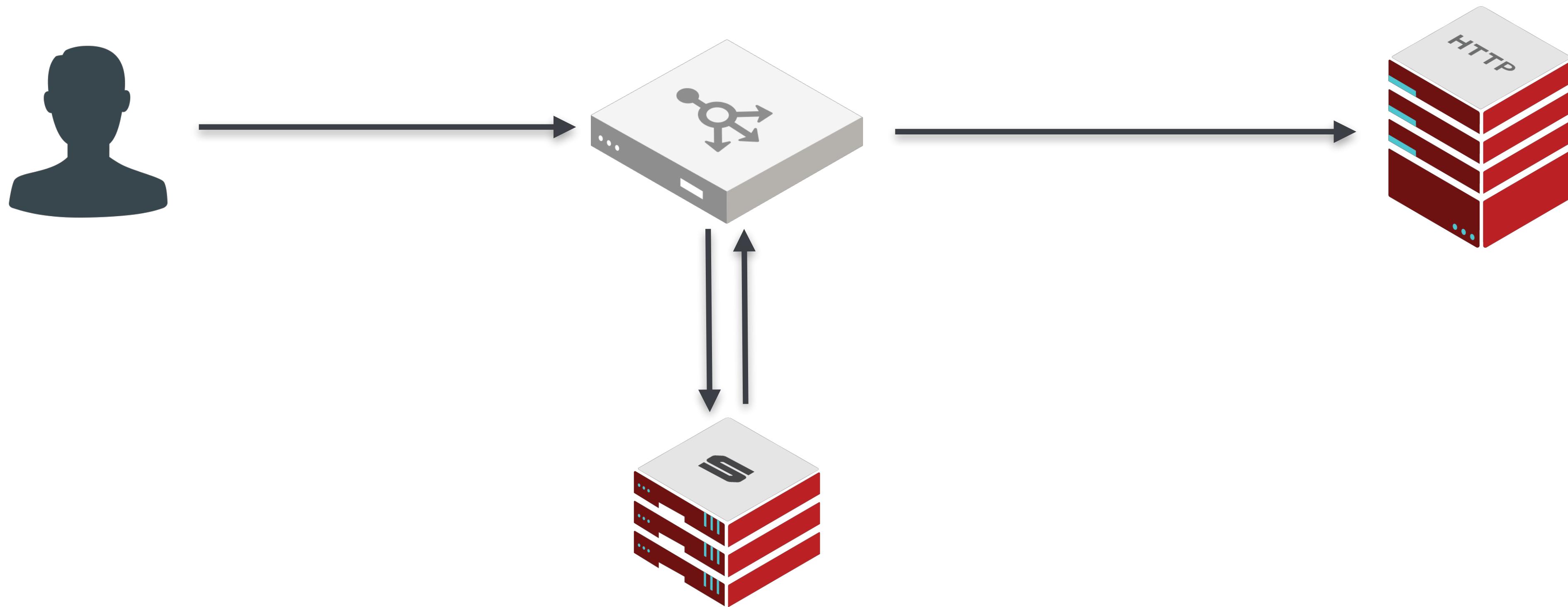
The client then makes a request for a Shape's JavaScript

How we do it at Shape



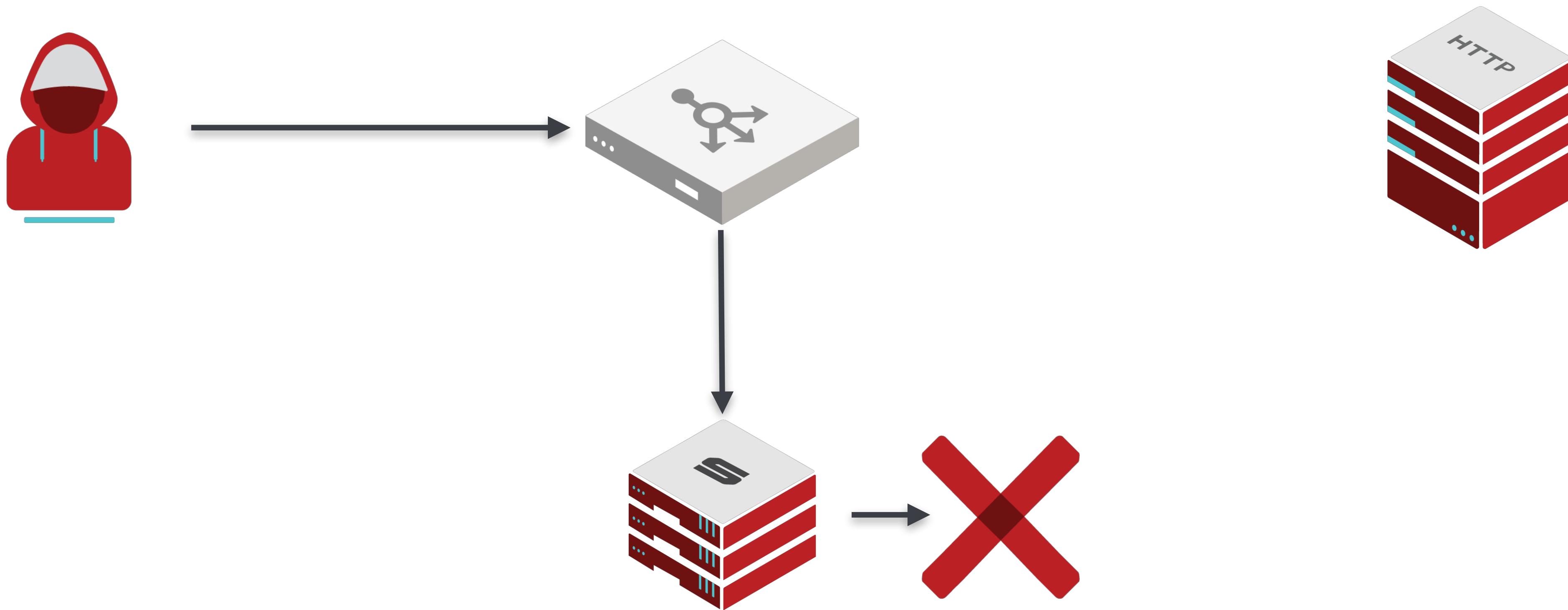
When the client makes a request to a protected URL,
the request is decorated with Shape execution results and telemetry

How we do it at Shape



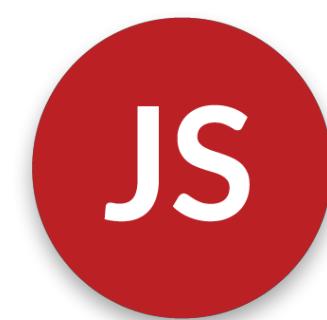
Legitimate traffic passes through to the originating servers.

How we do it at Shape



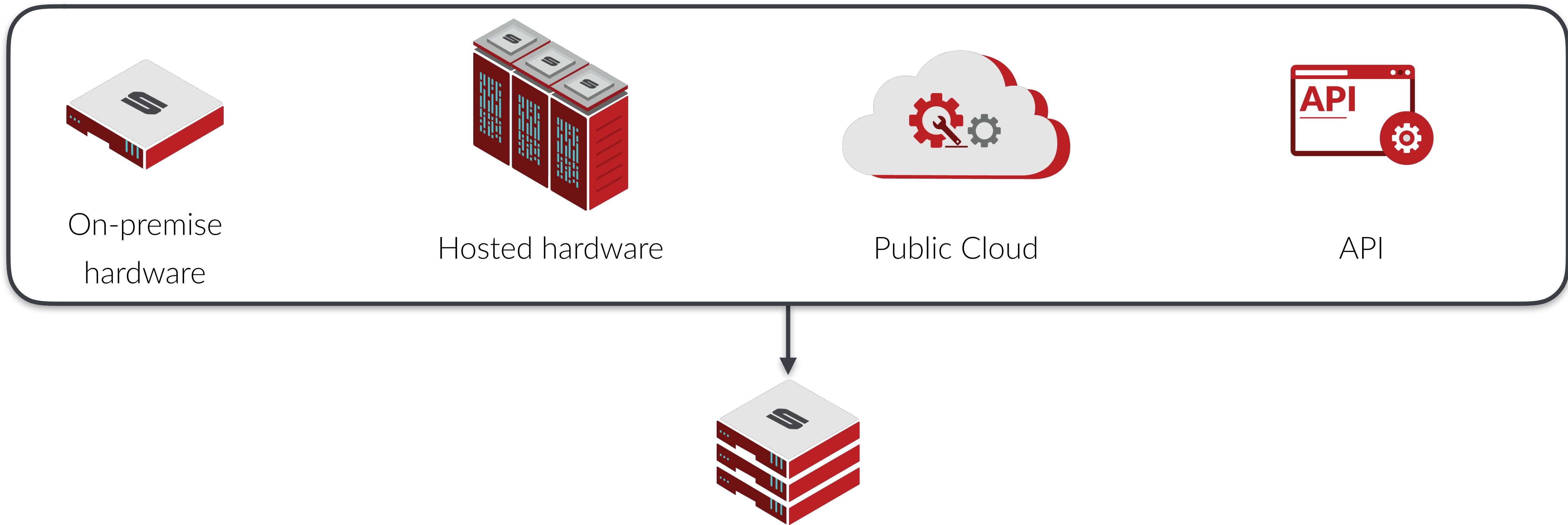
When Shape detects an illegitimate transaction it's stopped.

How we do it at Shape



- JavaScript is compiled to bytecode on the fly
- All JavaScript is generated per-request
- Bytecode + Virtual Machine is regenerated every 5 minutes
- Dozens of tamper evidence checks
- 120+ configurable countermeasure modules
- Returns...
 - User, Environment, Browser Telemetry
 - Execution characteristics
 - Proofs of hardware
 - Proofs of work
 - and much more...

How we do it at Shape



The service is available in many different configurations

How we do it at Shape



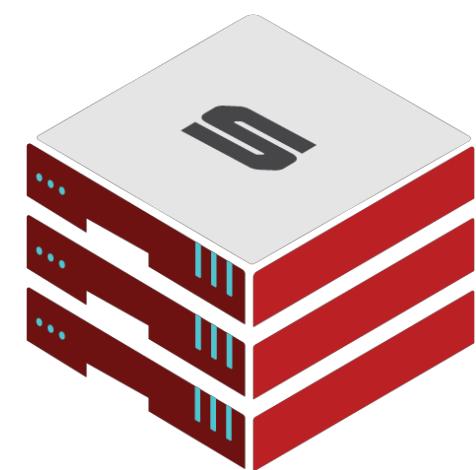
The service asynchronously sends the data to Shape's cloud

How we do it at Shape



Where 3 layers of analysis find anomalies and update the service

How we do it at Shape



Service updates regularly to run...

- Generated machine learning models per attacker.
- Custom rules written by security analysts.
- Variable responses designed for advanced attackers.

Thank You!

Jarrod Overson
@jsoverson

